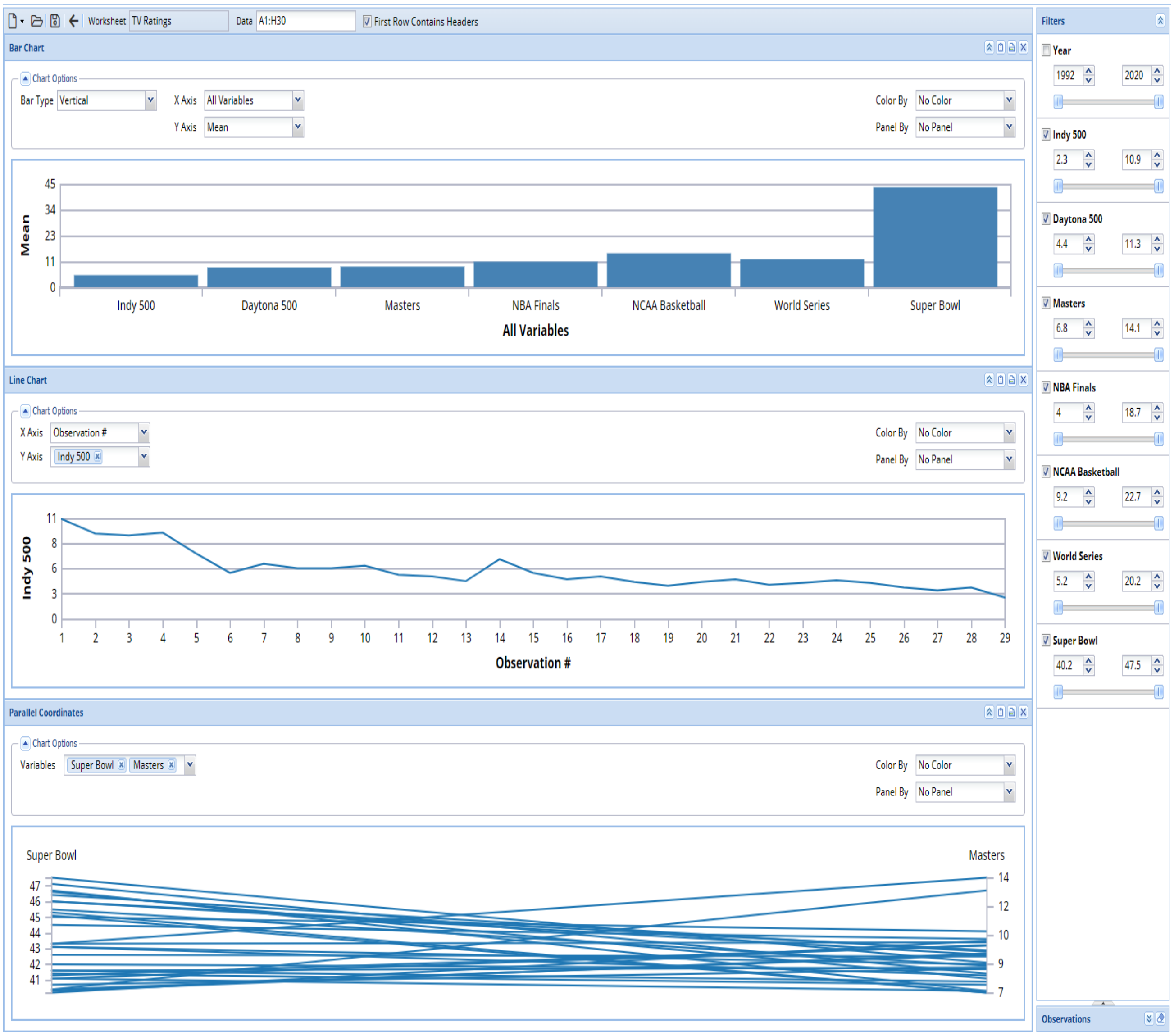
Version 2025 Q1

Analytic Solver

Data Science Reference Guide



Worksheet TV Ratings   Data A1:H30   ☑ First Row Contains Headers

Bar Chart

Chart Options

Bar Type Vertical   X Axis All Variables   Color By No Color

Y Axis Mean   Panel By No Panel

Line Chart

Chart Options

X Axis Observation #   Color By No Color

Y Axis Indy 500   Panel By No Panel

Parallel Coordinates

Chart Options

Variables Super Bowl Masters   Color By No Color

Panel By No Panel

Filters

□ Year 1992 2020

☑ Indy 500 2.3 10.9

☑ Daytona 500 4.4 11.3

☑ Masters 6.8 14.1

☑ NBA Finals 4 18.7

☑ NCAA Basketball 9.2 22.7

☑ World Series 5.2 20.2

☑ Super Bowl 40.2 47.5

Observations

Copyright

Trademarks

Patent Pending

Systems and Methods for Automated Risk Analysis of Machine Learning Models.

Acknowledgements

Thanks to Dan Fylstra and the Frontline Systems development team for a 25-year cumulative effort to build the best possible optimization and simulation software for Microsoft Excel. Thanks to Frontline's customers who have built many thousands of successful applications, and have given us many suggestions for improvements.

Risk Solver Pro and Risk Solver Platform have benefited from reviews, critiques, and suggestions from several risk analysis experts:

- Sam Savage (Stanford Univ. and AnalyCorp Inc.) for Probability Management concepts including SIPs, SLURPs, DISTs, and Certified Distributions.
- Sam Sugiyama (EC Risk USA & Europe LLC) for evaluation of advanced distributions, correlations, and alternate parameters for continuous distributions.
- Savvakis C. Savvides for global bounds, censor bounds, base case values, the Normal Skewed distribution and new risk measures.

How to Order

Contact Frontline Systems, Inc., P.O. Box 4288, Incline Village, NV 89450.
Tel (775) 831-0300 Fax (775) 831-0314 Email info@solver.com Web http://www.solver.com

Table of Contents

Generate Data 45

Exploring Data 69

Exploring a Time Series Dataset 246

Smoothing Techniques 260

Discriminant Analysis Classification Method 311

Logistic Regression 338

k – Nearest Neighbors Classification Method 367

Classification Tree Classification Method 384

Naïve Bayes Classification Method 413

Neural Network Classification Method 431

Ensemble Methods for Classification 470

Introduction to Analytic Solver Data Science

Introduction

Analytic Solver Data Science V2025 Q1 comes in two versions: **Analytic Solver Desktop** – a traditional "COM add-in" that works only in Microsoft Excel for Windows PCs (desktops and laptops), and **Analytic Solver Cloud** – a modern "JavaScript add-in" that works in Excel for Windows and Excel for Macintosh (desktops and laptops), and also in Excel for the Web (formerly Excel Online) using Web browsers such as Chrome, FireFox and Safari. Your license gives you access to both versions, and your Excel workbooks and optimization, simulation and data science models work in both versions, no matter where you save them (though OneDrive is most convenient).

This Reference Guide gives step-by-step instructions on how to utilize the data science and predictive methods and algorithms included in both the Desktop and Cloud applications. The overwhelming majority of features in Analytic Solver Desktop are also included in the Cloud app. However, there are a few variations between the two products. This guide documents any key differences between the two products.

Ribbon Overview

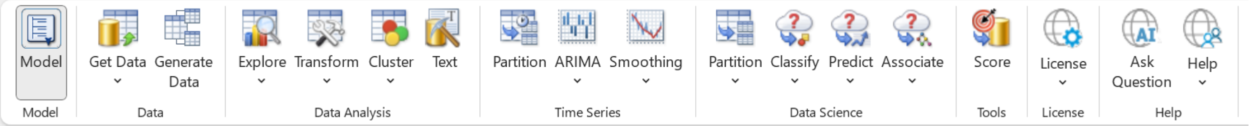
Analytic Solver Data Science, previously referred to as XLMiner^{TM} and more recently Analytic Solver Data Science, is a comprehensive data science software package for use in the Cloud or as an add-in to Excel. Data science is a discovery-driven data analysis technology used for identifying patterns and relationships in data sets. With overwhelming amounts of data now available from transaction systems and external data sources, organizations are presented with increasing opportunities to understand their data and gain insights into it. Data science is still an emerging field, and is a convergence of fields like statistics, machine learning, and artificial intelligence.

Often, there may be more than one approach to a problem. Analytic Solver Data Science is a tool belt to help you get started quickly offering a variety of methods to analyze your data. It has extensive coverage of statistical and machine learning techniques for classification, prediction, affinity analysis and data exploration and reduction.

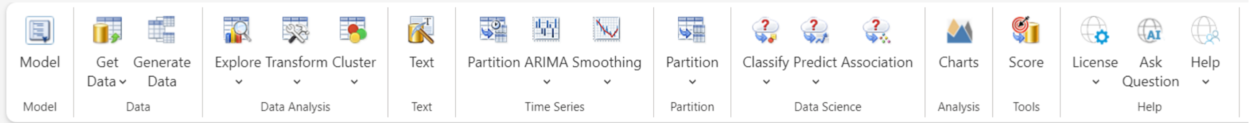
Data Science Ribbon

The Data Science ribbon is divided into 5 sections: Get Data, Data Analysis, Time Series, Data Science and Tools. A ribbon from each application (Desktop and Cloud) is shown below and on the next page. Notice that these Ribbons are *almost* identical. This is "by design" to ensure an almost seamless integration for our users.

Desktop Analytic Solver Data Science



Data Science Cloud



- Click the Model button to display the Solver Task Pane. This feature allows you to quickly navigate through datasets and worksheets containing Analytic Solver Data Science results.
- Click the **Get Data** button to draw a random sample of data, or summarize data from a (i) an Excel worksheet, (ii) the PowerPivot "spreadsheet data model" which can hold 10 to 100 million rows of data in Excel, (iii) an external SQL database such as Oracle, DB2 or SQL Server, or (iv) a dataset with up to billions of rows, stored across many hard disks in an external **Big Data** compute cluster running Apache Spark (https://spark.apache.org/).
- You can use the **Data Analysis** group of buttons to explore your data, both visually and through methods like cluster analysis, transform your data with methods like Principal Components, Missing Value imputation, Binning continuous data, and Transforming categorical data, or use the Text Mining feature to extract information from text documents.
- Use the **Time Series** group of buttons for time series forecasting, using both Exponential Smoothing (including Holt-Winters) and ARIMA (Auto-Regressive Integrated Moving Average) models, the two most popular time series forecasting methods from classical statistics. These methods forecast a single data series forward in time.
- The **Data Science** group of buttons give you access to a broad range of methods for prediction, classification and affinity analysis, from both classical statistics and data science. These methods use multiple input variables to predict an outcome variable or classify the outcome into one of several categories. Ensemble Methods are available for use with all data science and regression learners. The Find Best Model feature which allows you to input your data once and run all classification or regression learners at one time.
- Use the **Predict** button to build prediction models using Multiple Linear Regression (with variable subset selection and diagnostics), k-Nearest Neighbors, Regression Trees, and Neural Networks. Use Ensemble Methods with Regression Trees and Neural Networks to create more accurate prediction models. Use Find Best Model to run all 4 regression methods and 3 ensemble methods at once, and select the best fit model.
- Use the **Classify** button to build classification models with Discriminant Analysis, Logistic Regression, k-Nearest Neighbors, Classification Trees, Naïve Bayes, and Neural Networks. Use Ensemble Methods with Classification Trees and Neural Networks to create more accurate classification models. Use Find Best Model to run all 6 classification methods and 3 ensemble methods at once, and select the best fit model.
- Use the **Associate** button to perform affinity analysis ("what goes with what" or market basket analysis) using Association Rules.

- If forecasting and data science are new for you, don't worry – you can learn a lot about them by consulting our **AI Agent**, Frontline's artificial intelligence technical support assistant. AI Agent is designed to provide assistance and support for users of Frontline Solvers' Analytic Solver and Analytic Solver Data Science software. The AI Agent is knowledgeable about the functionality and features of the software, as well as the concepts and processes involved in optimization, simulation and data science/forecasting. Just enter a topic or question such as "What classification algorithms are supported in Analytic Solver Data Science?" and click Submit Query to get started.
- Use the **License** button to manage your account and licenses.
- Use the **Help** button to open example models, open the Help Center, where you can find pre-recorded webinars or access our Knowledge Base or explore our User Guides.

If you'd like to learn more and get started as a 'data scientist,' consult the excellent book *Data Mining for Business Intelligence*, which was written by the original Data Science (formally known as XLMiner and most recently Analytic Solver Data Mining) designers and early academic users. You'll be able to run all the Data Science examples and exercises in Analytic Solver.

Analytic Solver Data Science, along with the Data Science Cloud(formerly Data Science Cloud) app, can be purchased as a stand-alone product. A stand-alone license for Analytic Solver Data Science includes all of the data analysis, time series data capabilities, classification and prediction features available in Analytic Solver Comprehensive but does not support optimization or simulation. See Data Specifications in the Analytic Solver Data Science User Guide for each product.

Differences Between Desktop and Cloud Versions

Analytic Solver Data Science was constructed form the "ground up" to be as similar to Desktop Analytic Solver Data Science as possible. Ultimately, however, a few differences have arisen. They are noted here and also throughout this guide and the Data Science User Guide where applicable.

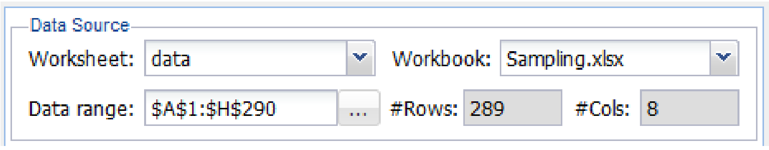
Differences

- Minor Ribbon Differences:
    - The Text Mining icon is located in the Text section of the Ribbon
    - Standard Partitioning is located in the Partition section of the Ribbon
    - The Tools section of the Ribbon includes Score
- A new icon, License, has been added. Click this icon to manage your Analytic Solver licenses. See the section "License in the Cloud Apps" within this guide for more information.
- The Options button has been removed. All menu items previously appearing on this menu, now appear on the License or Help menus.
- Workflows created in Data Science Cloud are not supported in Analytic Solver Desktop.

- A Help Center has been added to the Help menu. Click Help – Help Center to find example models, start a live chat, open user guides, listen to recorded and live webinars, etc. In the Cloud app, items such as Welcome Screen, Help Text and About Data Science are not applicable and do not appear on the menu, while User Guide and Reference Guide have been combined to simply User Guides. See the section "Help in the Cloud Apps" within this guide for more information.
- Sampling from a File Folder or Database is not supported.
- Scoring to a database is also not supported.
- To view output charts, click the Charts icon on the ribbon, select the desired output sheet for *Worksheet* and the desired chart for *Chart*. Viewing output charts is documented in each of the chapters included in this guide.
- The Model tab on the Solver Task Pane does not exist in the Cloud app, only the Workflow tab. See the "Model" section within this guide for more information.
- Dynamic Arrays for the Psi Data Science Functions (PsiForecast, PsiPredict, PsiPosteriors, and PsiTransform) are supported in the Data Science Cloud app. To use a Dynamic Array in place of an Excel Control array, simply enter the Psi Data Science Function into one cell. The Dynamic Array will "spill" down. See the section **Using Data Science Psi Functions in Excel** within the Data Science User Guide for more information.
- When creating Neural Networks using Automatic Architecture (under Classify or Predict), links in the Architecture Search Error Log will not open a new Neural Network dialog.
- Text files, used in the Text Mining example that occurs later in this guide, are not available in Analytic Solver Cloud. This is because Analytic Solver Cloud does not support Sampling from a File Folder.

Common Dialog Options

These options, fields and command buttons appear on most Analytic Solver Data Science dialogs.



Data Source

Worksheet: data

Workbook: Sampling.xlsx

Data range: $A$1:$H$290

#Rows: 289

#Cols: 8

Worksheet

The active worksheet appears in this field.

Workbook

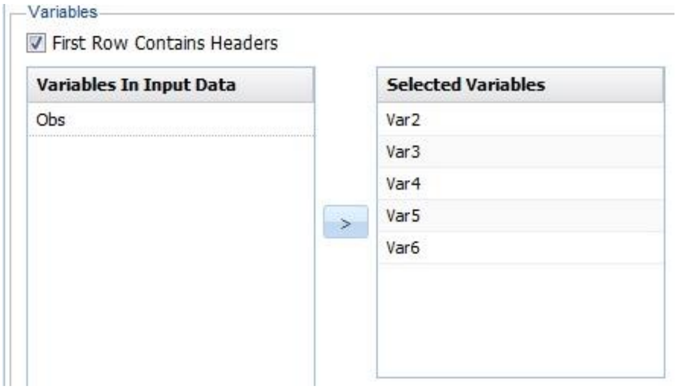
The active workbook appears in this field.

Data Range

The range of the dataset appears in this field.

# Rows # Cols

The number of rows and columns in the dataset appear in these two fields, respectively.



☑ First Row Contains Headers

Variables In Input Data	Selected Variables
Obs	Var2
	Var3
	Var4
	Var5
	Var6

First row contains headers

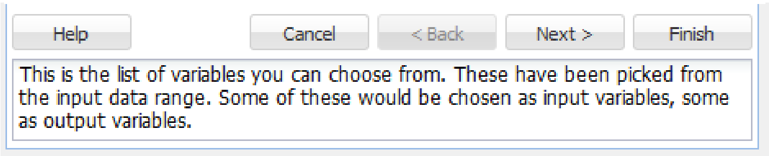
If this option is selected, variables will be listed according to the first row in the dataset.

Variables in the Input Data

All variables contained in the dataset will be listed in this field.

Selected Variables

Variables listed in this field will be included in the output. Select the desired variables listed in the *Variables In Input Data* listbox, then click the > button to shift variables to the *Selected Variables* field.



This is the list of variables you can choose from. These have been picked from the input data range. Some of these would be chosen as input variables, some as output variables.

Help

Click this command button to open the Analytic Solver Data Science Help text file.

Next

Click this command button to progress to the next tab of the dialog.

Reset

Click this command button to reset the options for the selected method.

OK/Finish

Click this command button to initiate the desired method and produce the output report.

Cancel

Click this command button to close the open dialog without saving any options or creating an output report.

References

See below for a list of references sited when compiling this guide.

Websites

1. The Data & Analysis Center for Software. <https://www.thecsiac.com>
2. NEC Research Institute Research Index: The NECI Scientific Literature Digital Library. <http://www.iicm.tugraz.at/thesis/cguetl_diss/literatur/Kapitel02/URL/NEC/cs.html>.
3. Thearling, Kurt. Data Mining and Analytic Technologies. <http://www.thearling.com>

Books

1. Anderberg, Michael R. Cluster Analysis for Applications. Academic Press (1973).
2. Berry, Michael J. A., Gordon S. Linoff. Mastering Data Mining. Wiley (2000).
3. Breiman, Leo Jerome H. Friedman, Richard A. Olshen, Charles J. Stone. Classification and Regression Trees. Chapman & Hall/CRC (1998).
4. Han, Jiawei, Micheline Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers (2000).
5. Hand, David, Heikki Mannila, Padhraic Smyth. Principles of Data Mining. MIT Press, Cambridge" (2001).
6. Hastie, Trevor, Robert Tibshirani, Jerome Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York (2001).
7. Shmueli, Galit, Nitin R. Patel, Peter C. Bruce. Data Mining for Business Intelligence. Wiley, New Jersey (2010).

Big Data Options

Introduction

Large amounts of data are being generated and collected continuously from a multitude of sources every minute of every day. From your toothbrush to your vehicle GPS to Twitter/Facebook/Google/Yahoo, data is everywhere. Being able to make decisions based on this information requires the ability to extract trends and patterns that can be buried deeply within the numbers.

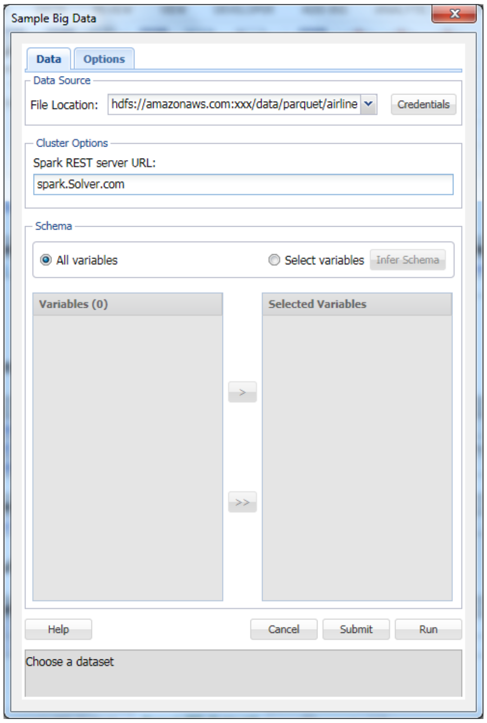
Generally these large datasets contain millions of records (rows) requiring multiple gigabytes or terabytes of storage space across multiple hard drives in an external compute cluster. Analytic Solver Platform V2016-R3 enables users, for the first time, to 'pull' sampled and summarized data into Excel from compute clusters running Apache Spark, the open-source software widely embraced by Big Data vendors and users.

See the Analytic Solver Data Science User Guide for a complete step by step example illustrating how to sample and summarize big data using Analytic Solver. See the Big Data Options section below for a comprehensive explanation of each option that appears on the Big Data dialog tabs.

Big Data Options

The following options are included in each of the five Big Data tabs.

Sample Big Data, Data tab



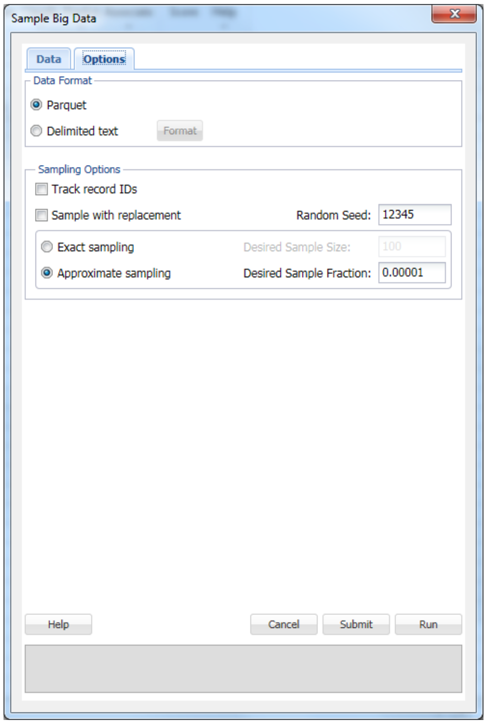
File Location: hdfs://amazonaws.com:xxx/data/parquet/airline

Spark REST server URL: spark.Solver.com

● All variables

Choose a dataset

Option Name	Sample Big Data Dialog Option Description – Data tab
File Location	Enter the location of the file here.
Credentials	If your dataset is located on Amazon S3, click Credentials to enter your Access and Secret Keys.
Schema	When *All Variables* is selected for this option, all columns (features) in the dataset would be selected for the analysis without the need for the user to select the variables. When *Select variables* is selected for this option, the command button *Infer Schema* is enabled. Once *Infer Schema* is clicked, schema (variables) will be inferred from the dataset on the cluster and listed in the *Variables* grid. Users may use the > and < buttons to select variables for inclusion in the sample.
Variables	Variables available for inclusion in the sample will appear here. Use the > button to select variables to be included in the sample.
Select Variables	Variables transferred here will be included in the sample. Use the < button to remove variables from the sample.
Submit	Clicking *Submit* sends a request for sampling to the compute cluster but does not wait for completion. The result is output containing the Job ID and basic information about the submitted job so different submissions may be identified. This information can be used at any time later for querying the status of the job and generating reports based on the results of the completed job.
Run	Sends a request for sampling to the Apache Spark compute cluster where the Frontline Systems access server is installed and waits for the results. Once the job is completed and results are returned to the Analytic Solver Data Science client, a report is inserted into the Model tab of the Analytic Solver Task Pane under Data Science – Results - Sampling.
Cancel	Click this command button to close the open dialog without saving any options or creating an output report.



Sample Big Data

Data | Options

Data Format

● Parquet

○ Delimited text Format

Sampling Options

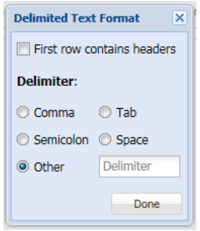
□ Track record IDs

□ Sample with replacement Random Seed: 12345

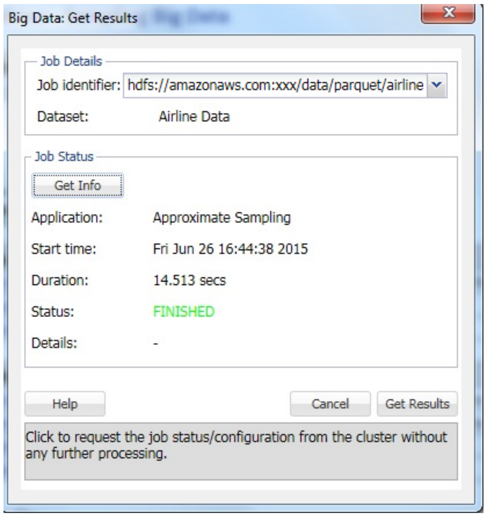
○ Exact sampling Desired Sample Size: 100

● Approximate sampling Desired Sample Fraction: 0.00001

Help Cancel Submit Run

Option Name	Sample Big Data Option Description – Options dialog
Data Format	If your data is in Apache Parquet format, select Parquet for this option. If your data is in Delimited Text format, select Delimited text. When Delimited Text is selected, the Format button will be enabled. Click, to open the Delimited Text Format dialog. Here you can specify if your *First row contains headers* along with the delimiter used in your data.  Delimited Text Format □ First row contains headers Delimiter: ○ Comma ○ Tab ○ Semicolon ○ Space ● Other Delimiter Done Analytic Solver Data Science can process data from Hadoop Distributed File System (HDFS), local file systems that are visible to Spark cluster and Amazon S3. Performance is best with HDFS, and it is recommended that you load data from a local file system or Amazon S3 into HDFS. If the local file system is used, the data must be accessible at the same path on all Spark workers, either via a network path, or because it was copied to the same location on all workers. At present, Analytic Solver Data Science can process data in Apache Parquet and CSV (delimited text) formats. Performance is far better with Parquet, which stores data in a compressed, columnar representation; it is highly recommended that you convert CSV data to Parquet before you seek to sample or summarize the data.
Track Record IDs	If this option is selected, data records in the resulting sample will carry the correct ordinal IDs that correspond to the original data records, so that records can be matched. Note: Selecting this option may significantly increase running time so it should be applied only when necessary.
Sample with Replacement	When selected, records in the dataset may be chosen for inclusion in the sample multiple times.
Random Seed	If an integer value appears for *Random seed*, Analytic Solver Data Science will use this value to set the feature selection random number seed. Setting the random number seed to a nonzero value ensures that the same sequence of random numbers is used each time the dataset is chosen for sampling. The default value is "12345". If left blank, the random number generator is initialized from the system clock, so the random sample would be represented by different records from run to run. If you need the results from successive samples to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts positive integers with up to 7 digits.
Exact Sampling	When this option is selected, Analytic Solver Data Science will return a fixed – size sampled subset of data according to the setting for *Desired Sample Size*.
Desired Sample Size	Enter the number of records to be included in the sample.
Approximate Sampling	When this option is selected, the size of the resultant sample will be determined by the value entered for Desired Sample Fraction. Approximate sampling is much faster than Exact Sampling. Usually, the resultant fraction is very close to the Desired Sample Fraction so this option should be preferred over exact sampling as often as possible. Even if the resultant sample slightly deviates from the desired size, this would be easy to correct in Excel.
Desired Sample Fraction	This is the expected size of the sample as a fraction of the dataset's size. If *Sampling with Replacement* is selected, the value for *Desired Sample Fraction* must be greater than 0. If Sampling without replacement (i.e. Sampling with Replacement is not selected), the Desired Sample Fraction becomes the probability that each element is chosen and, as a result, *Desired Sample Fraction* must be between 0 and 1.

Big Data Get Results dialog



Job identifier: hdfs://amazonaws.com:xxx/data/parquet/airline

Dataset: Airline Data

Application: Approximate Sampling

Start time: Fri Jun 26 16:44:38 2015

Duration: 14.513 secs

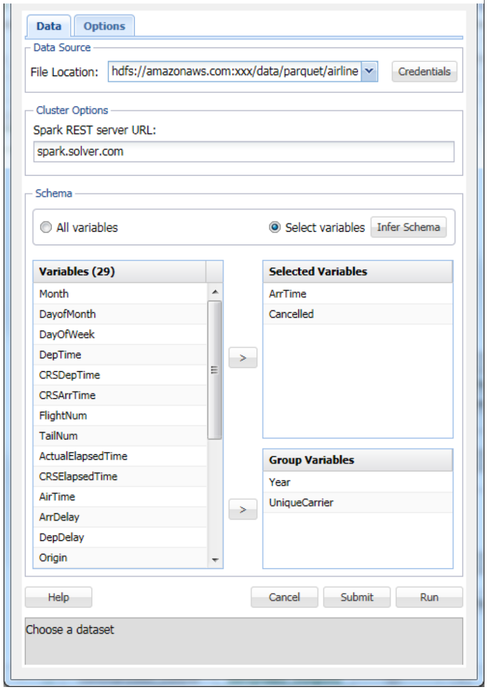
Status: FINISHED

Details: -

Click to request the job status/configuration from the cluster without any further processing.

Option Name	Big Data Get Results Option Description
Job Identifier	Click the down arrow to the right of this option to obtain results from the previously submitted (using Get Data – Big Data – Sample/Summarize) job.
Get Info	Click this command button to check the status of the previously submitted job. The following information will be returned. *Application* is the type of the submitted job. *Start Time* displays the date and time when the job was submitted. Start Time will always be displayed in the user's Local Time. *Duration* shows the elapsed time since job submission if the job is still RUNNING and total compute time if the job is FINISHED. *Status* is the current state of the job: FINISHED, FAILED, ERRORED or RUNNING. FINISHED indicates that the job has been completed and results are available for retrieval. FAILED or ERRORED indicates that the job has not completed due to an internal cluster failure. When this occurs, *Details* will contain a message indicating the reason. In our example, this dialog displays the status as FINISHED.
Get Results	If Status is FINISHED, you may click the *Get Results* button to obtain the results from the cluster and populate the report as shown below. Note: It is not required to click *Get Info* before *Get Results*. If *Get Results* is clicked, the status of the job will be checked and if the status is FINISHED, the results will be pulled from the cluster and the report will be created. Otherwise, Status will be updated with the appropriate message to reflect the state of the submitted job: FAILED, ERRORED, or RUNNING.
Cancel	Click this command button to close the open dialog without saving any options or creating an output report.
Help	Click this command button to open the Analytic Solver Data Science Help Text.

Summarize Big Data, Data tab



File Location: hdfs://amazonaws.com:xxx/data/parquet/airline

Spark REST server URL: spark.solver.com

○ All variables ● Select variables

Variables (29): Month, DayofMonth, DayOfWeek, DepTime, CRSDepTime, CRSArrTime, FlightNum, TailNum, ActualElapsedTime, CRSElapsedTime, AirTime, ArrDelay, DepDelay, Origin

Selected Variables: ArrTime, Cancelled

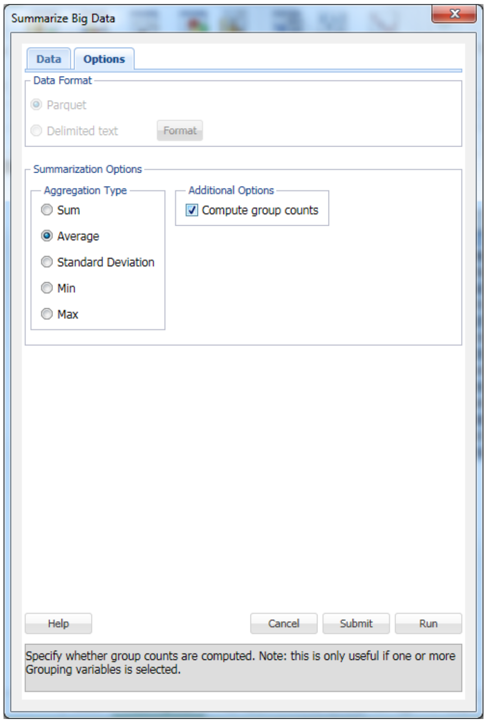
Group Variables: Year, UniqueCarrier

Choose a dataset

See the *Sample Big Data, Data* tab above for all option explanations except *Group Variables*.

Option Name	Summarize Big Data Options Dialog Description – Data tab
Group Variables	Group Variables are variables from the dataset that are treated as key variables for aggregation. In the screenshot above, two variables have been selected as Group Variables: Year and UniqueCarrier. The variables will be grouped so that all records with the same Year and UniqueCarrier are included in the same group, and then all aggregate functions for each group will be calculated.

Summarize Big Data, Options dialog



Data Format

● Parquet

○ Delimited text

Summarization Options

Aggregation Type

○ Sum

● Average

○ Standard Deviation

○ Min

○ Max

Additional Options

☑ Compute group counts

Specify whether group counts are computed. Note: this is only useful if one or more Grouping variables is selected.

See the *Sample Big Data, Data* tab above for all option explanations except *Data Format, Aggregation Type and Compute Group Counts*.

Option Name	**Summarize Big Data Options Dialog Description – Options dialog**
Data Format	See option description above.
Aggregation Type	*Aggregation Type* provides 5 statistics that can be inferred from the dataset: sum, average, standard deviation, minimum and maximum.
Compute Group Counts	This option is enabled when 1 or more Grouping Variables is selected. When this option is selected, the number of records belonging to each group is computed and reported.

Sampling or Importing from a Database, Worksheet or File Folder

Introduction

Sampling

A statistician often comes across huge volumes of information from which he or she wants to draw inferences. Since time and cost limitations make it impossible to go through every entry in these enormous datasets, statisticians must resort to sampling techniques. These sampling techniques choose a reduced sample or subset from the complete dataset. The statistician can then perform his or her statistical procedures on this reduced dataset saving much time and money.

Let’s review a few statistical terms. The entire dataset is called the **population**. A **sample** is the portion of the population that is actually examined. A good sample should be a true representation of the population to avoid forming misleading conclusions. Various methods and techniques have been developed to ensure a representative sample is chosen from the population. A few are discussed here.

- **Simple Random Sampling** This is probably the simplest method for obtaining a good sample. A simple random sample of say, size *n*, is chosen from the population in such a way that every random set of *n* items from the population has an equal chance of being chosen to be included in the sample. Thus simple random sampling not only avoids bias in the choice of individual items but also gives every possible sample an equal chance.

  The Data Sampling utility in Analytic Solver Data Science offers the user the freedom to choose sample size, seed for randomization, and sampling with or without replacement.

- **Stratified Random Sampling** In this technique, the population is first divided into groups of similar items. These groups are called strata. Each stratum, in turn, is sampled using simple random sampling. These samples are then combined to form a stratified random sample.

  The Data Sampling utility in Analytic Solver Data Science offers the user the freedom to choose a sorting seed for randomization and sampling with or without replacement. The desired sample size can be prefixed by the user depending on which method is being chosen for stratified random sampling.

Analytic Solver Data Science *Desktop* allows sampling either from a worksheet, database or file folder. *Analytic Solver Data Science Cloud, does not support sampling from a database or file folder.*

Importing from a File System in Desktop Addin

In order to run desktop Analytic Solver Data Science's Text Miner tool, we must first import the text. Text may be present in a **worksheet**, as a column (variable) where the cell in each row contains a comment, paragraph or other free-form text or a text field in a **database**. In both of these cases, each text 'document' is naturally associated on each row / observation with other structured input fields, and for supervised learning, with an outcome variable.

Note: If using Analytic Solver Data Science within desktop Excel, we recommend the use of Microsoft's free Power Query add-in, or the facilities built into the free Power Pivot add-in, for importing data from a wide range of online and on premise databases into Excel's "spreadsheet data model", which has no limit other than memory on the number of rows. Analytic Solver Data Science may be used to draw a random sample from this data, to be brought onto a worksheet for analysis and model-building.

Text may also be present in a series of **document** files in a disk or network folder (where each document represents an observation). In this case, the menu option **Get Data – File Folder** may be used to read either all of the documents in a folder, or a representative sample of these documents.

If the documents are relatively small (each one no more than 32,767 characters, the limit on the length of a string in a single worksheet cell), the document *contents* may be brought into the output. Otherwise, Analytic Solver Data Science can import only the document *filenames/paths* into the output; the document contents are read only during the text mining operation, and document size is not limited except by memory and time.

The output produced by **Get Data – File Folder** (see the example below) may be used directly as input to the text mining operation, described later on in this guide. But if there are other structured fields, or an outcome variable in the dataset, it is up to the user to assemble a worksheet that associates each document with the correct observation for the other fields. Excel's many tools for editing rows, columns and sheets can aid considerably in this process, but it is not automatic. *Note*: AnalyticSolver.com does not support data manipulation of this type. If you are using AnalyticSolver.com or the Data Science Cloud app, you can perform these edits in Excel, then upload the revised worksheet to AnalyticSolver.com.

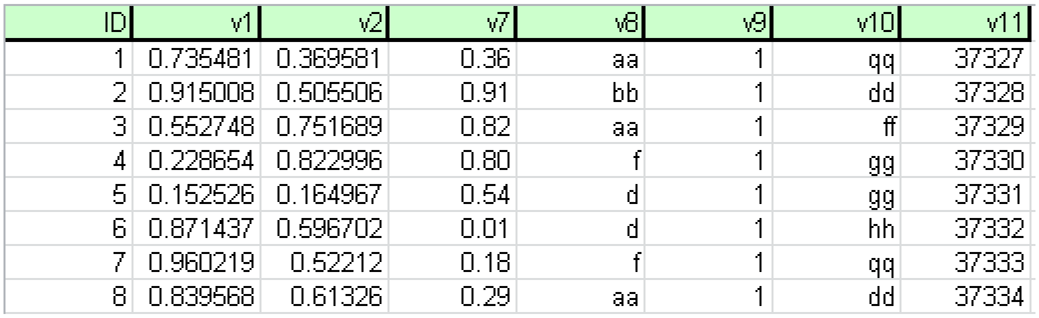
Sampling from a Worksheet

Below are three examples that illustrate how to perform Simple Random Sampling with and without replacement and Stratified Random Sampling from a worksheet. Each example uses the sample dataset in the Sampling.xlsx example workbook.

To open this workbook, click **Help – Example Models** on the Data Science Desktop Ribbon, then **Forecasting/Data Science Examples – Sampling.**

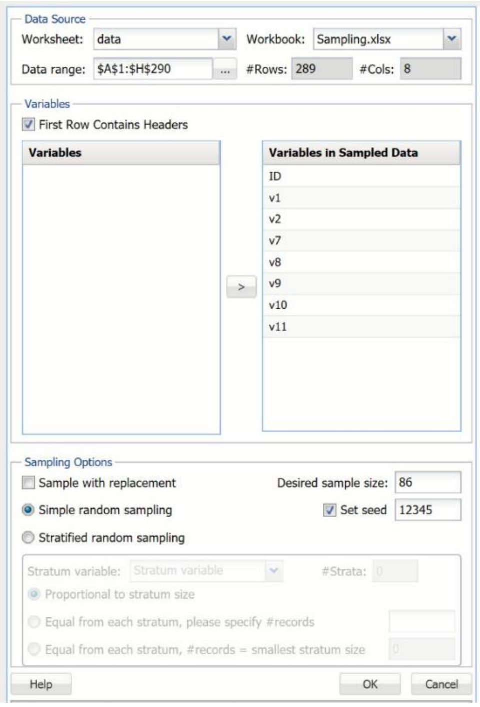
Example: Sampling from a Worksheet using Simple Random Sampling

The Sampling.xlsx dataset contains a variable ID for the record identification and seven variables, v1, v2, v7, v8, v9, v10, v11.



ID	v1	v2	v7	v8	v9	v10	v11
1	0.735481	0.369581	0.36	aa	1	qq	37327
2	0.915008	0.505506	0.91	bb	1	dd	37328
3	0.552748	0.751689	0.82	aa	1	ff	37329
4	0.228654	0.822996	0.80	f	1	gg	37330
5	0.152526	0.164967	0.54	d	1	gg	37331
6	0.871437	0.596702	0.01	d	1	hh	37332
7	0.960219	0.52212	0.18	f	1	qq	37333
8	0.839568	0.61326	0.29	aa	1	dd	37334

To start, click **Get Data – Worksheet**. In this example, the default option, *Simple Random Sampling*, is used. Select all variables under *Variables*, click > to include them in the sample data, then click **OK**.



Data Source

Worksheet: data

Workbook: Sampling.xlsx

Data range: $A$1:$H$290

#Rows: 289

#Cols: 8

Variables

☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options

☐ Sample with replacement

Desired sample size: 86

● Simple random sampling

☑ Set seed 12345

○ Stratified random sampling

Stratum variable: Stratum variable

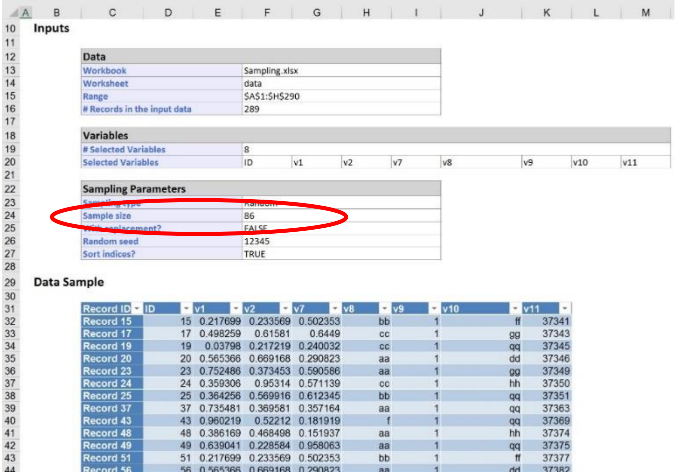
#Strata: 0

● Proportional to stratum size

○ Equal from each stratum, please specify #records

○ Equal from each stratum, #records = smallest stratum size 0

The result *Sampling* will be inserted in the Model tab of the Analytic Solver task pane under Transformations – Sample From Worksheet. A portion of the output is shown below.



Inputs

Data	
Workbook	Sampling.xlsx
Worksheet	data
Range	$A$1:$H$290
# Records in the input data	289

Variables								
# Selected Variables	8							
Selected Variables	ID	v1	v2	v7	v8	v9	v10	v11

Sampling Parameters	
Sampling type	Random
Sample size	86
With replacement?	FALSE
Random seed	12345
Sort indices?	TRUE

Data Sample

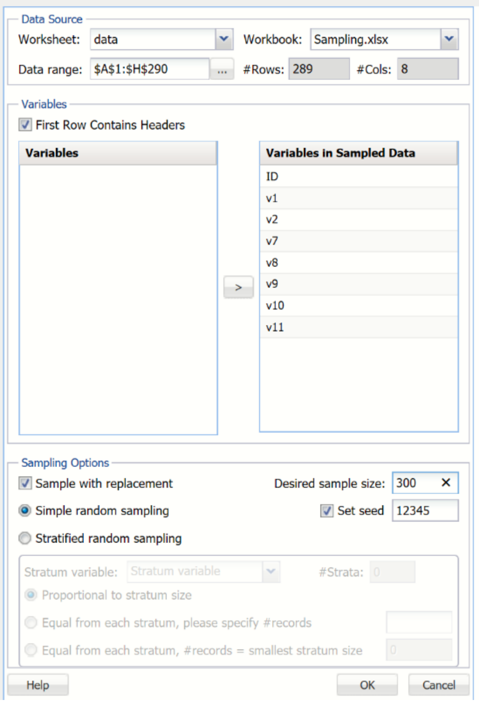
Record ID	ID	v1	v2	v7	v8	v9	v10	v11
Record 15	15	0.217699	0.233569	0.502353	bb	1	ff	37341
Record 17	17	0.498259	0.61581	0.6449	cc	1	gg	37343
Record 19	19	0.03798	0.217219	0.240032	cc	1	qq	37345
Record 20	20	0.565366	0.669168	0.290823	aa	1	dd	37346
Record 23	23	0.752486	0.373453	0.590586	aa	1	gg	37349
Record 24	24	0.359306	0.95314	0.571139	cc	1	hh	37350
Record 25	25	0.364256	0.569916	0.612345	bb	1	qq	37351
Record 37	37	0.735481	0.369581	0.357164	aa	1	qq	37363
Record 43	43	0.960219	0.52212	0.181919	f	1	qq	37369
Record 48	48	0.386169	0.468498	0.151937	aa	1	hh	37374
Record 49	49	0.639041	0.228584	0.958063	aa	1	qq	37375
Record 51	51	0.217699	0.233569	0.502353	bb	1	ff	37377
Record 56	56	0.565366	0.669168	0.290823	aa	1	dd	37382

The output is a simple random sample without replacement, with a default random seed setting of 12345. The desired sample size is 86 records as shown in the Sample Size field.

Example: Sampling from a Worksheet using Sampling with Replacement

This second example illustrates how to sample from a worksheet using sampling with replacement.

Click **Get Data – Worksheet** to bring up the *Sample From Worksheet* dialog. Again, select all variables in the *Variables* section and click > to include each in the sample data. Check *Sample with replacement* and enter **300** for *Desired sample size*. Since we are choosing sampling with replacement, Analytic Solver Data Science will generate a sample with a larger number of records than the dataset. Click **OK**.



Data Source

Worksheet: data

Workbook: Sampling.xlsx

Data range: $A$1:$H$290

#Rows: 289

#Cols: 8

Variables

☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options

☑ Sample with replacement

Desired sample size: 300

● Simple random sampling

☑ Set seed 12345

○ Stratified random sampling

Stratum variable: Stratum variable

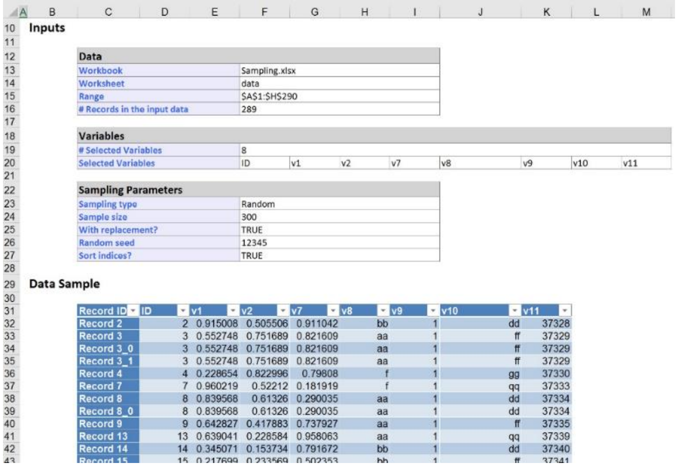
#Strata: 0

● Proportional to stratum size

○ Equal from each stratum, please specify #records

○ Equal from each stratum, #records = smallest stratum size 0

The result, Sampling1, will be inserted into the Analytic Solver task pane under Transformations - Sample From Worksheet. A portion of the output is shown below.



Inputs

Data	
Workbook	Sampling.xlsx
Worksheet	data
Range	$A$1:$H$290
# Records in the input data	289

Variables								
# Selected Variables	8							
Selected Variables	ID	v1	v2	v7	v8	v9	v10	v11

Sampling Parameters	
Sampling type	Random
Sample size	300
With replacement?	TRUE
Random seed	12345
Sort indices?	TRUE

Data Sample

Record ID	ID	v1	v2	v7	v8	v9	v10	v11
Record 2	2	0.915008	0.505506	0.911042	bb	1	dd	37328
Record 3	3	0.552748	0.751689	0.821609	aa	1	ff	37329
Record 3_0	3	0.552748	0.751689	0.821609	aa	1	ff	37329
Record 3_1	3	0.552748	0.751689	0.821609	aa	1	ff	37329
Record 4	4	0.228654	0.822996	0.79808	f	1	gg	37330
Record 7	7	0.960219	0.52212	0.181919	f	1	qq	37333
Record 8	8	0.839568	0.61326	0.290035	aa	1	dd	37334
Record 8_0	8	0.839568	0.61326	0.290035	aa	1	dd	37334
Record 9	9	0.642827	0.417883	0.737927	aa	1	ff	37335
Record 13	13	0.639041	0.228584	0.958063	aa	1	qq	37339
Record 14	14	0.345071	0.153734	0.791672	bb	1	dd	37340
Record 15	15	0.217699	0.233569	0.502353	bb	1	ff	37341

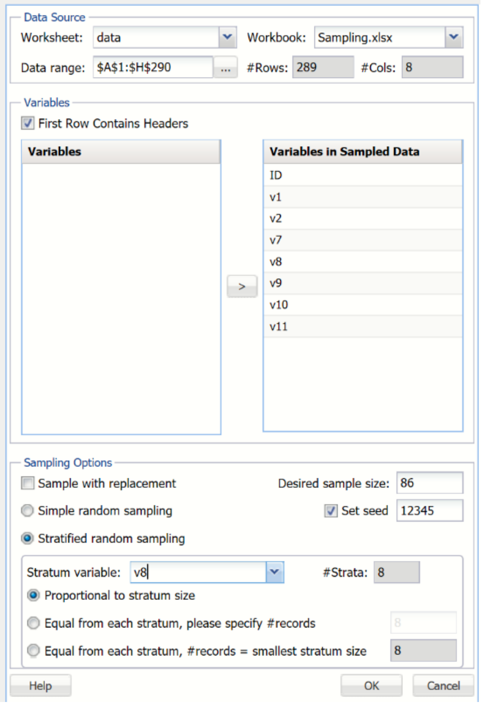
The output indicates "True" for Sampling with Replacement. As a result, the desired sample size is greater than the number of records in the input data (289 records vs a sample size of 300). Looking closely at the ID column, you'll see that multiple records have been sampled more than once.

Example: Sampling from a Worksheet using Stratified Random Sampling

This example illustrates how to sample from a worksheet using stratified random sampling.

Click **Get Data – Worksheet** and select all variables under *Variables*, click **>** to include them in the sample data. Select **Stratified random sampling**.

Click the down arrow next to *Stratum Variable* and select v8. The strata number is automatically displayed once you select v8. Keep the default setting selected, **Proportional to stratum size**. Then click **OK**.



Data Source

Worksheet: data

Workbook: Sampling.xlsx

Data range: $A$1:$H$290

#Rows: 289

#Cols: 8

Variables

☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options

□ Sample with replacement

Desired sample size: 86

○ Simple random sampling

☑ Set seed 12345

● Stratified random sampling

Stratum variable: v8

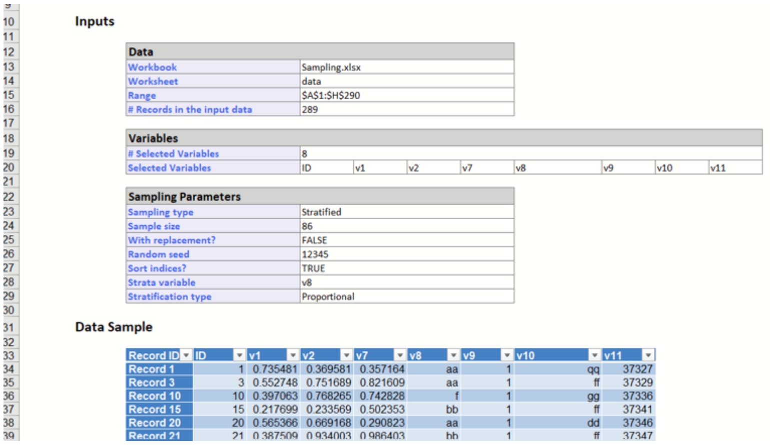
#Strata: 8

● Proportional to stratum size

○ Equal from each stratum, please specify #records 8

○ Equal from each stratum, #records = smallest stratum size 8

The results, contained within Sampling2, is inserted into Transformations – Sample From Worksheet. A portion of the output is shown below.



Inputs

Data	
Workbook	Sampling.xlsx
Worksheet	data
Range	$A$1:$H$290
# Records in the input data	289

Variables								
# Selected Variables	8							
Selected Variables	ID	v1	v2	v7	v8	v9	v10	v11

Sampling Parameters	
Sampling type	Stratified
Sample size	86
With replacement?	FALSE
Random seed	12345
Sort indices?	TRUE
Strata variable	v8
Stratification type	Proportional

Data Sample

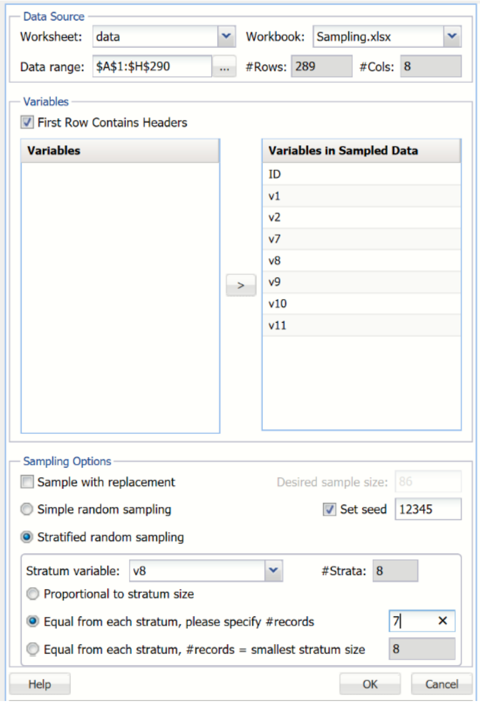
Record ID	ID	v1	v2	v7	v8	v9	v10	v11
Record 1	1	0.735481	0.369581	0.357164	aa	1	qq	37327
Record 3	3	0.552748	0.751689	0.821609	aa	1	ff	37329
Record 10	10	0.397063	0.768265	0.742828	f	1	gg	37336
Record 15	15	0.217699	0.233569	0.502353	bb	1	ff	37341
Record 20	20	0.565366	0.669168	0.290823	aa	1	dd	37346
Record 21	21	0.387509	0.934003	0.986403	bb	1	ff	37347

Analytic Solver Data Science calculated the percentage representation of V8 in the dataset and maintained that percentage in the sample.

Let’s see what happens to our output when we select a different option for Stratified Sampling.

Click **Get Data -- Worksheet**. Select all variables under *Variables*, click > to include them in the sample data. Select **Stratified random sampling**. Choose **v8** as the *Stratum variable*. The *#strata* is displayed automatically. Select **Equal from each stratum**, **please specify #records**.

Enter the #records. Remember, this number should not be greater than the smallest stratum size. In this case the smallest stratum size is 8. (Note: The smallest stratum size appears automatically in a box next to the option, *Equal from each stratum, please specify #records.*) Enter **7**, which is less than the limit of 8, and then click OK.



Data Source

Worksheet: data Workbook: Sampling.xlsx

Data range: $A$1:$H$290 #Rows: 289 #Cols: 8

Variables

☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options

☐ Sample with replacement Desired sample size: 86

○ Simple random sampling ☑ Set seed 12345

● Stratified random sampling

Stratum variable: v8 #Strata: 8

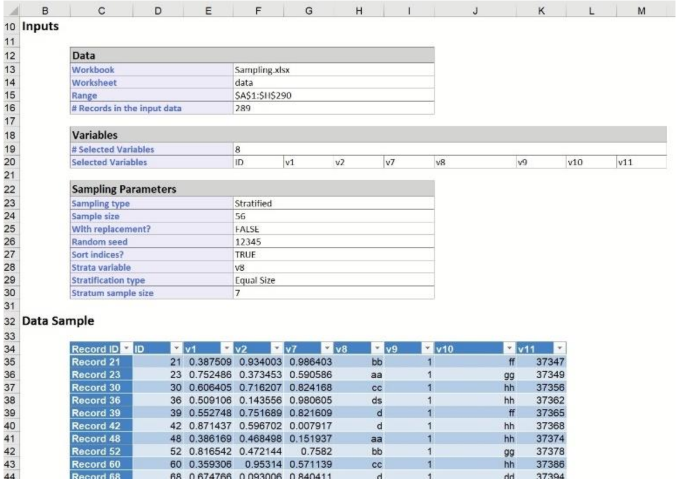
○ Proportional to stratum size

● Equal from each stratum, please specify #records 7

○ Equal from each stratum, #records = smallest stratum size 8

Sampling3 will be inserted into the Analytic Solver task pane under Transformations -- Sample From Worksheet. A portion of the output is shown below.

As you can see in the output, the number of records in the sampled data is 56 or 7 records per stratum for 8 strata (7 * 8 = 56).



Inputs

Data	
Workbook	Sampling.xlsx
Worksheet	data
Range	$A$1:$H$290
# Records in the input data	289

Variables								
# Selected Variables	8							
Selected Variables	ID	v1	v2	v7	v8	v9	v10	v11

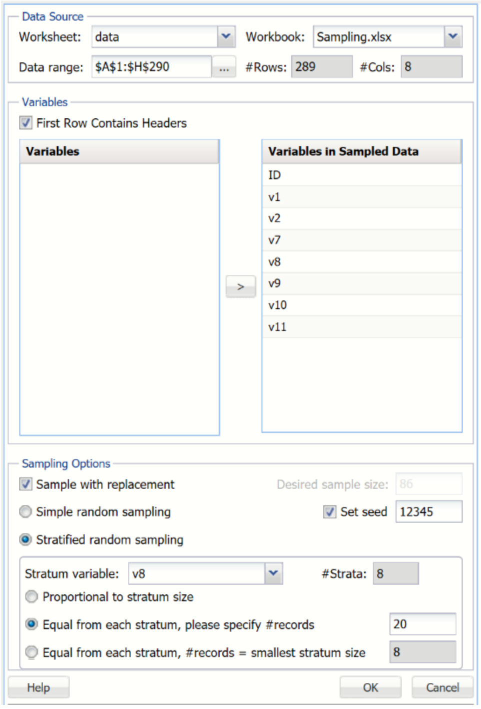
Sampling Parameters	
Sampling type	Stratified
Sample size	56
With replacement?	FALSE
Random seed	12345
Sort indices?	TRUE
Strata variable	v8
Stratification type	Equal Size
Stratum sample size	7

Data Sample

Record ID	ID	v1	v2	v7	v8	v9	v10	v11
Record 21	21	0.387509	0.934003	0.986403	bb	1	ff	37347
Record 23	23	0.752486	0.373453	0.590586	aa	1	gg	37349
Record 30	30	0.606405	0.716207	0.824168	cc	1	hh	37356
Record 36	36	0.509106	0.143556	0.980605	ds	1	hh	37362
Record 39	39	0.552748	0.751689	0.821609	d	1	ff	37365
Record 42	42	0.871437	0.596702	0.007917	d	1	hh	37368
Record 48	48	0.386169	0.468498	0.151937	aa	1	hh	37374
Record 52	52	0.816542	0.472144	0.7582	bb	1	gg	37378
Record 60	60	0.359306	0.95314	0.571139	cc	1	hh	37386
Record 68	68	0.674766	0.093006	0.840411	d	1	dd	37394

If a sample with an equal number of records for each stratum but of bigger size is desired, use the same options above with *Sampling with Replacement* selected.

Click **Get Data -- Worksheet** once again. Select all variables under *Variables*, click > to include them in the sample data. Select **Sample with replacement** and **Stratified random sampling**. Select **V8** for the *Stratum variable*. Select **Equal from each stratum, please specify #records** and enter **20**. Though the smallest stratum size is 8 in this dataset, we can acquire more records for our sample since we are *Sampling with replacement*. Keeping all other options the same, the output is as follows. Click **OK**.



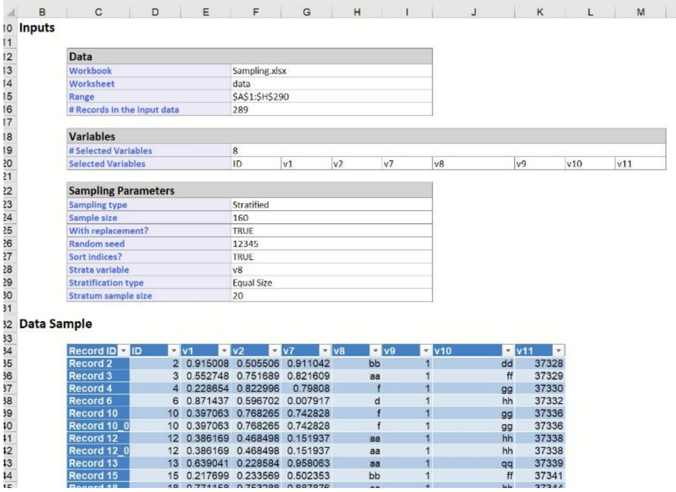
Data Source
Worksheet: data Workbook: Sampling.xlsx
Data range: $A$1:$H$290 #Rows: 289 #Cols: 8

Variables
☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options
☑ Sample with replacement Desired sample size: 86
○ Simple random sampling ☑ Set seed 12345
● Stratified random sampling
Stratum variable: v8 #Strata: 8
○ Proportional to stratum size
● Equal from each stratum, please specify #records 20
○ Equal from each stratum, #records = smallest stratum size 8

Click Transformations -- Sample From Worksheet in the Model tab of the Analytic Solver Task Pane for the result, Sampling4. A portion of the output is shown below.



Inputs

Data	
Workbook	Sampling.xlsx
Worksheet	data
Range	$A$1:$H$290
# Records in the input data	289

Variables								
# Selected Variables	8							
Selected Variables	ID	v1	v2	v7	v8	v9	v10	v11

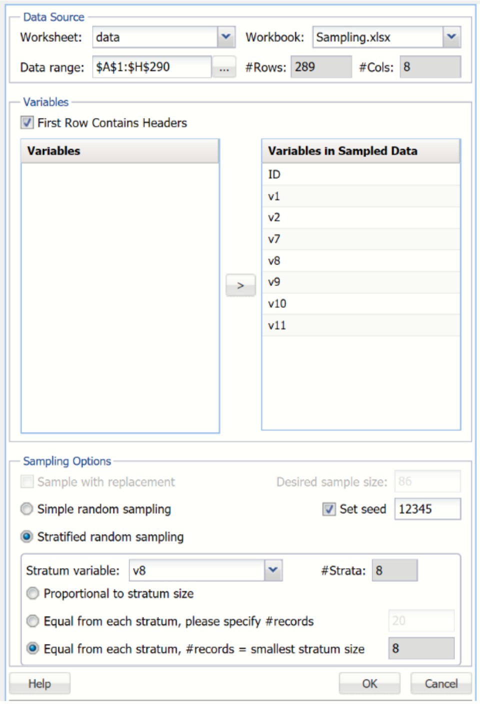
Sampling Parameters	
Sampling type	Stratified
Sample size	160
With replacement?	TRUE
Random seed	12345
Sort indices?	TRUE
Strata variable	v8
Stratification type	Equal Size
Stratum sample size	20

Data Sample

Record ID	ID	v1	v2	v7	v8	v9	v10	v11
Record 2	2	0.915008	0.505506	0.911042	bb	1	dd	37328
Record 3	3	0.552748	0.751689	0.821609	aa	1	ff	37329
Record 4	4	0.228654	0.822996	0.79808	f	1	gg	37330
Record 6	6	0.871437	0.596702	0.007917	d	1	hh	37332
Record 10	10	0.397063	0.768265	0.742828	f	1	gg	37336
Record 10_0	10	0.397063	0.768265	0.742828	f	1	gg	37336
Record 12	12	0.386169	0.468498	0.151937	aa	1	hh	37338
Record 12_0	12	0.386169	0.468498	0.151937	aa	1	hh	37338
Record 13	13	0.639041	0.228584	0.958063	aa	1	qq	37339
Record 15	15	0.217699	0.233569	0.502353	bb	1	ff	37341

Since the output sample has 20 records per stratum, the *#records in sampled data* is 160 (20 records per stratum for 8 strata).

Click **Get Data -- Worksheet** one last time. Select all variables under *Variables*, click > to include them in the sample data. Select **Stratified random sampling**. Select **V8** for the *Stratum variable* and select **Equal from each stratum, # records = smallest stratum size**. The edit box to the right of the option is prefilled with the number 8. This is the smallest stratum size in the dataset.



Data Source

Worksheet: data Workbook: Sampling.xlsx

Data range: $A$1:$H$290 #Rows: 289 #Cols: 8

Variables

☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options

□ Sample with replacement Desired sample size: 86

○ Simple random sampling ☑ Set seed 12345

● Stratified random sampling

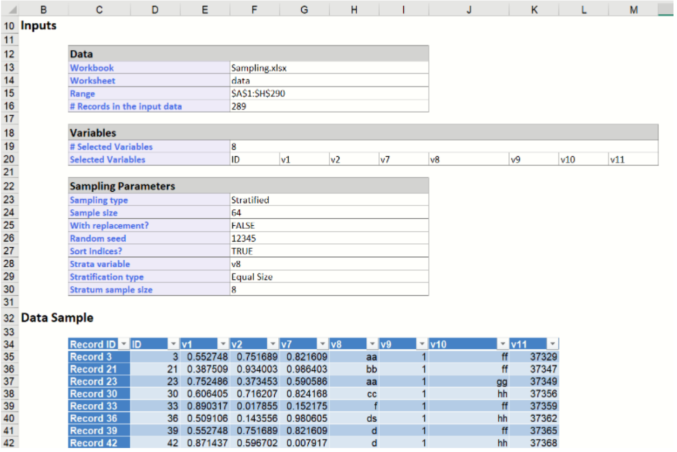
Stratum variable: v8 #Strata: 8

○ Proportional to stratum size

○ Equal from each stratum, please specify #records 20

● Equal from each stratum, #records = smallest stratum size 8

Keeping all other options the same, click **OK**. The output, found under Transformations -- Sample From Worksheet in the Analytic Solver task pane, is below.



Inputs

Data	
Workbook	Sampling.xlsx
Worksheet	data
Range	$A$1:$H$290
# Records in the input data	289

Variables								
# Selected Variables	8							
Selected Variables	ID	v1	v2	v7	v8	v9	v10	v11

Sampling Parameters	
Sampling type	Stratified
Sample size	64
With replacement?	FALSE
Random seed	12345
Sort Indices?	TRUE
Strata variable	v8
Stratification type	Equal Size
Stratum sample size	8

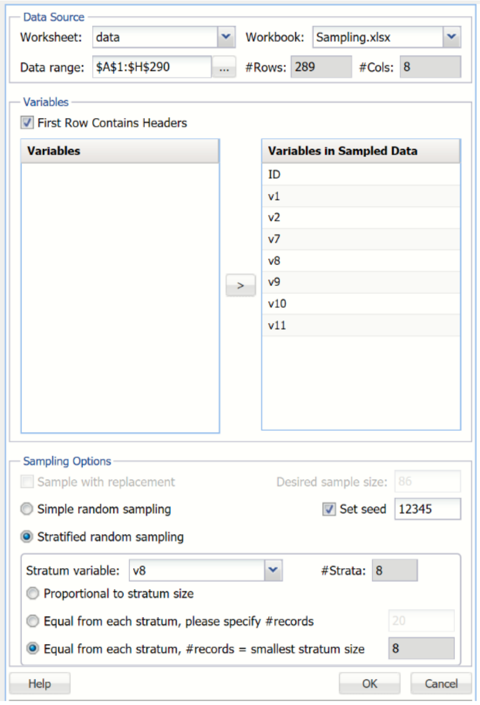
Data Sample

Record ID	ID	v1	v2	v7	v8	v9	v10	v11
Record 3	3	0.552748	0.751689	0.821609	aa	1	ff	37329
Record 21	21	0.387509	0.934003	0.986403	bb	1	ff	37347
Record 23	23	0.752486	0.373453	0.590586	aa	1	gg	37349
Record 30	30	0.606405	0.716207	0.824168	cc	1	hh	37356
Record 33	33	0.890317	0.017855	0.152175	f	1	ff	37359
Record 36	36	0.509106	0.143556	0.980605	ds	1	hh	37362
Record 39	39	0.552748	0.751689	0.821609	d	1	ff	37365
Record 42	42	0.871437	0.596702	0.007917	d	1	hh	37368

Since the output sample has 8 records per stratum, the *Sample Size* is 64 (8 records per stratum for 8 strata).

Sample from Worksheet Options

Please see below for a complete list of each option contained on the *Sample from Worksheet* and *Sample from Database* dialogs.



Data Source

Worksheet: data

Workbook: Sampling.xlsx

Data range: $A$1:$H$290

#Rows: 289

#Cols: 8

Variables

☑ First Row Contains Headers

Variables	Variables in Sampled Data
	ID
	v1
	v2
	v7
	v8
	v9
	v10
	v11

Sampling Options

□ Sample with replacement

Desired sample size: 86

○ Simple random sampling

☑ Set seed 12345

● Stratified random sampling

Stratum variable: v8

#Strata: 8

○ Proportional to stratum size

○ Equal from each stratum, please specify #records 20

● Equal from each stratum, #records = smallest stratum size 8

Data Range

Either type the address directly into this field, or use the reference button, to enter the data range from the worksheet or data set. If the cell pointer (active cell) is already somewhere in the data range, Analytic Solver Data Science automatically picks up the contiguous data range surrounding the active cell. After the data range is selected, Analytic Solver Data Science displays the number of records in the selected range.

First Row Contains Headers

When this box is checked, Analytic Solver Data Science picks up the headings from the first row of the selected data range. When the box is unchecked, Analytic Solver Data Science follows the default naming convention, i.e., the variable in the first column of the selected range will be called "Var1", the second column "Var2," etc.

Variables

This list box contains the names of the variables in the selected data range. If the first row of the range contains the variable names, then these names appear in this list box. If the first row of the dataset does not contain the headers, then Analytic Solver Data Science lists the variable names using its default naming convention. In this case the first column is named *Var1*; the second column is named *Var2* and so on. To select a variable for sampling, select the variable, then click the ">" button. Use the CTRL key to select multiple variables.

Sample With replacement

If this option is checked the data will be sampled with replacement. The default is Sampling without Replacement.

Set Seed

Enter the desired sorting seed here. The default seed is 12345.

Desired sample size

Enter the desired sample size here. (Note that the actual sample size in the output may vary a little, depending on additional options selected.)

Simple random sampling

The data is sorted using the simple random sampling technique, taking into account the additional parameter settings.

Stratified random sampling

If selected, Analytic Solver Data Science enables the Stratum Variable options.

Stratum Variable

Select the variable to be used for stratified random sampling by clicking the down arrow and selecting the desired variable. As the user selects the variable name, Analytic Solver Data Science displays the *#Strata* that variable contains in a box to the left and the smallest stratum size in a field beside the option *Equal from each stratum, #records = smallest stratum size*. (Note: Analytic Solver Comprehensive and Data Science support an unlimited number of variables each having an unlimited number of distinct values. Versions of Analytic Solver with basic limits support variables with 2 to 30 distinct values.)

Proportionate to stratum size

Analytic Solver Data Science detects the proportion of each stratum in the dataset and maintains the same in sampling. Due to this, Analytic Solver Data Science sometimes must increase the sample size in order to maintain the proportionate stratum size.

Equal from each stratum, please specify #records

On specifying the number of records, Analytic Solver Data Science generates a sample which has the same number of records from each stratum. In this case the number chosen automatically decides the desired sample size. As a result, the option to enter the desired sample size is disabled.

Equal from each stratum, #records = smallest stratum size

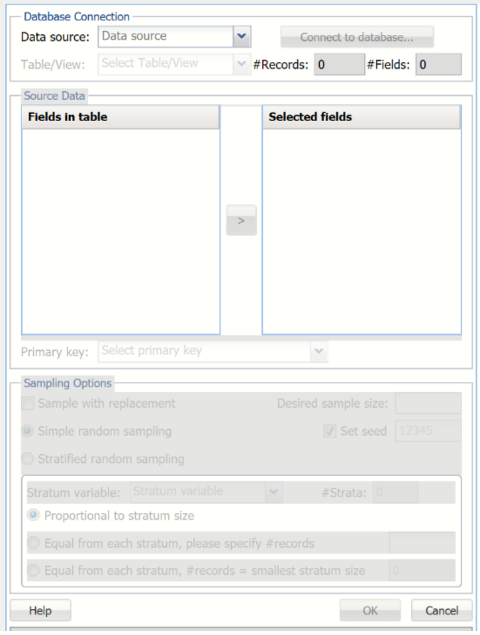
Analytic Solver Data Science detects the smallest stratum size and generates a sample wherein every stratum has a representation of that size. If this option is selected, *Sample with replacement* and *Desired sample size* are both disabled.

Analytic Solver Data Science performs the stratified random sampling with or without replacement. If Sample with replacement is not selected, the desired sample size must be less than the number of records in the dataset.

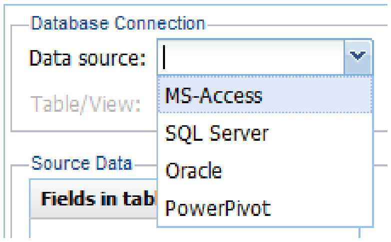
If Sample with Replacement is selected, Analytic Solver Data Science is limited to 1,000,000 records in the sample output.

Sampling from a Database

Click **Get Data – Database** on the Data Science Desktop ribbon to display the following dialog. Note: Sampling from a Database is currently not supported in the Data Science Cloud app.

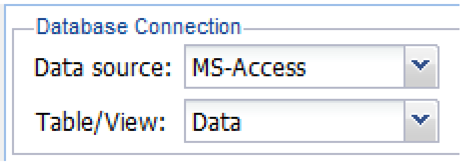

Database Connection
Data source: Data source
Connect to database...
Table/View: Select Table/View
#Records: 0
#Fields: 0
Source Data
Fields in table
>
Selected fields
Primary key: Select primary key
Sampling Options
Sample with replacement
Desired sample size:
Simple random sampling
☑ Set seed 12345
Stratified random sampling
Stratum variable: Stratum variable
#Strata: 0
● Proportional to stratum size
○ Equal from each stratum, please specify #records
○ Equal from each stratum, #records = smallest stratum size
Help
OK
Cancel

Click the down arrow next to *Data Source* and select **MS-Access**, then click **Connect to database**.



An Open file dialog opens, browse to C:\ProgramData\Frontline Systems\ Datasets. Select the **Demo.mdb** Microsoft Access database, and then click **Open**.

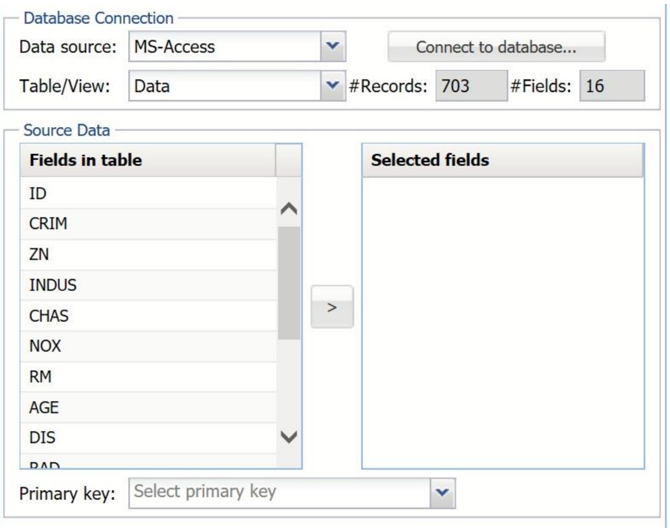
Data is auto populated for *Table/View*.



Data source: MS-Access

Table/View: Data

The *Fields in table* listbox is populated as shown in the screenshot below.



Database Connection

Data source: MS-Access

Table/View: Data #Records: 703 #Fields: 16

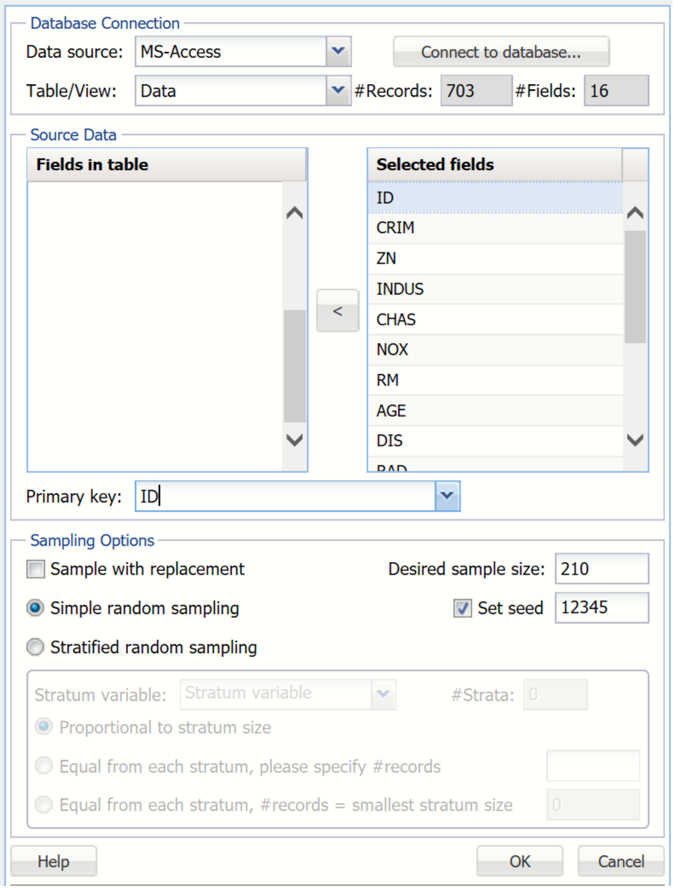
Source Data

Fields in table: ID, CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD

Selected fields:

Primary key: Select primary key

Select all fields from *Fields in table* and click > to move all fields to *Selected fields*. Select **ID** as the *Primary key*. A *Primary key* must contain non-null and unique values across all rows in the table.



Database Connection

Data source: MS-Access

Table/View: Data #Records: 703 #Fields: 16

Source Data

Fields in table:

Selected fields: ID, CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD

Primary key: ID

Sampling Options

□ Sample with replacement Desired sample size: 210

● Simple random sampling ☑ Set seed 12345

○ Stratified random sampling

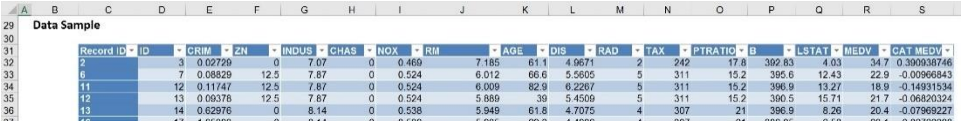
Stratum variable: Stratum variable #Strata: 0

○ Proportional to stratum size

○ Equal from each stratum, please specify #records

○ Equal from each stratum, #records = smallest stratum size 0

Click **OK**. A portion of the output is below.



Data Sample

Record ID	ID	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT MEDV
2	3	0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	0.390938746
6	7	0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	-0.00966843
11	12	0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	-0.14931534
12	13	0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	-0.06820324
13	14	0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	-0.07969227

For more information on the Sampling Options, refer to the examples above for Sampling from a Worksheet. You can sample from a database using all the methods described in this section.

Importing from a File Folder

The following example is used to illustrate how to import 1,000 text files saved in the same file folder. Click Help – Example Models on the Data Science Desktop ribbon, then click Forecasting/Data Science. *Note: This functionality is not supported in Data Science Cloud app.*

Browse to C:\ProgramData\Frontline Systems\Datasets and open the Text Mining Example Documents.zip archive file. Unzip the contents of this file to a location of your choice. Four folders will be created beneath Text Mining Example Documents: Autos, Electronics, Additional Autos and Additional Electronics. One thousand, two hundred short text files will be extracted to the location chosen. This example is based on the text dataset at http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/news20.html, which consists of 20,000 messages, collected from 20 different netnews newsgroups. We selected about 1,200 of these messages that were posted to two interest groups, for Autos and Electronics (about 50% in each).

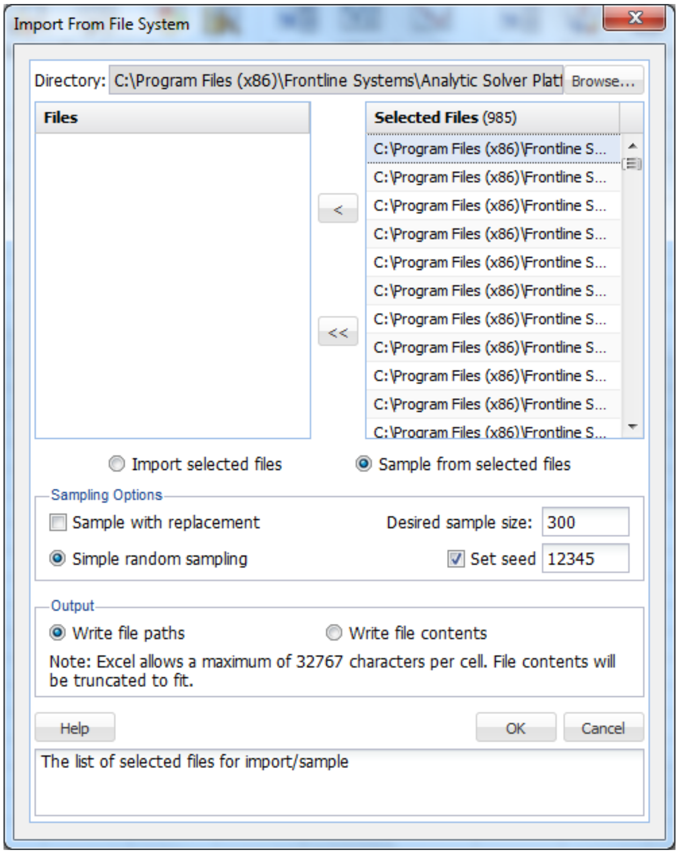
Select **Get Data – File Folder** to open the *Import From File System* dialog. At the top of the dialog, click **Browse…** to navigate to the *Autos* subfolder (C:\ProgramData\Frontline Systems\Datasets\Text Mining Example Documents\Autos). Set the *File Type* to All Files (*.*), then select all files in the folder and click the **Open** button. The files will appear in the left list box under *Files*. Click the >> button to move the files from the *Files* list box to the *Selected Files* list box. Now repeat these steps for the Electronics subfolder. When these steps are completed, 985 files will appear under *Selected Files*.

Select **Sample from selected files** to enable the *Sampling Options*. Analytic Solver Data Science will perform sampling from the files in the *Selected Files* field. Enter **300** for *Desired sample size* while leaving the default settings for *Simple random sampling* and *Set Seed.*

Note: If you are using the educational version of Analytic Solver Data Science, enter "100" for Desired Sample Size. This is the upper limit for the number of files supported when sampling from a file system when using Analytic Solver Data Science. For a complete list of the capabilities of Analytic Solver Data Science and Analytic Solver Data Science for Education, click here.

Analytic Solver Data Science will select 300 files using Simple random sampling with a seed value of 12345. Under *Output*, leave the default setting of *Write file paths.* Rather than writing out the file contents into the report, Analytic Solver Data Science will include the file paths.

Note: Currently, Analytic Solver Data Science only supports the import of delimited text files. A delimited text file is one in which data values are separated by a character such as quotation marks, commas or tabs. These characters define a beginning and end of a string of text.



Import From File System

Directory: C:\Program Files (x86)\Frontline Systems\Analytic Solver Platf

Files	Selected Files (985)
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...
	C:\Program Files (x86)\Frontline S...

○ Import selected files ● Sample from selected files

Sampling Options

□ Sample with replacement Desired sample size: 300

● Simple random sampling ☑ Set seed 12345

Output

● Write file paths ○ Write file contents

Note: Excel allows a maximum of 32767 characters per cell. File contents will be truncated to fit.

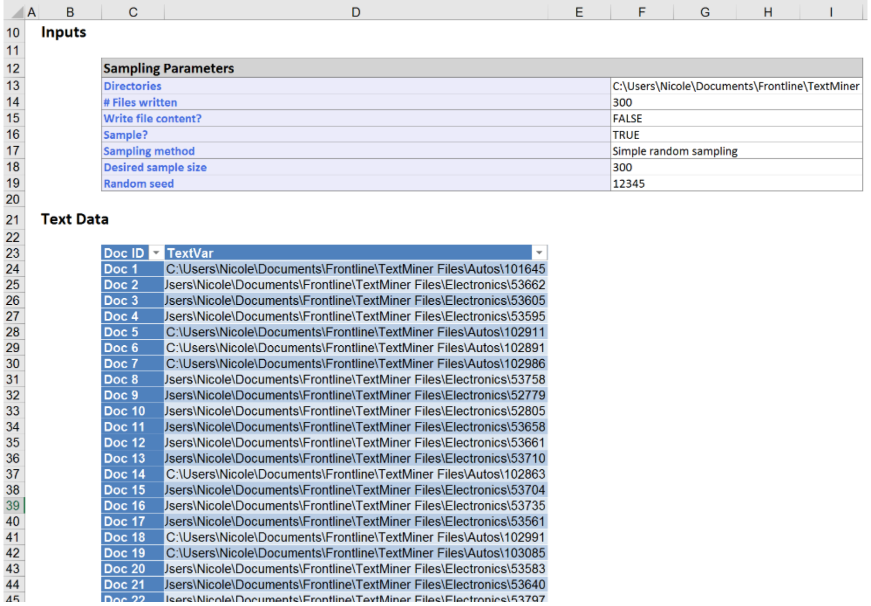
The list of selected files for import/sample

Click **OK**. The *FileSampling* worksheet will be inserted into the Analytic Solver task pane under Data Science – Transformations – Sample From File Folder with contents similar to that shown on the next page.

The *Data* portion of the report displays the selections we made on the *Import From File System* dialog. Here we see the path of the directories, the number of files written, our choice to write the paths or contents (*File Paths*), the sampling method, the desired sample size and the seed value (*12345*).

Underneath the *Data* portion are paths to the 300 text files in random order that were sampled by Analytic Solver Data Science. If *Write file contents* had been selected, rather than *Write file paths*, the report would contain the RowID, File Path, and the first 32,767 characters present in the document.

From here, one could use Excel's sort features to categorize the paths by "Autos" and "Electronics" for use with the Text Mining tool. See subsequent Text Mining chapter for an example on how to use this feature.



Inputs

Sampling Parameters	
Directories	C:\Users\Nicole\Documents\Frontline\TextMiner
# Files written	300
Write file content?	FALSE
Sample?	TRUE
Sampling method	Simple random sampling
Desired sample size	300
Random seed	12345

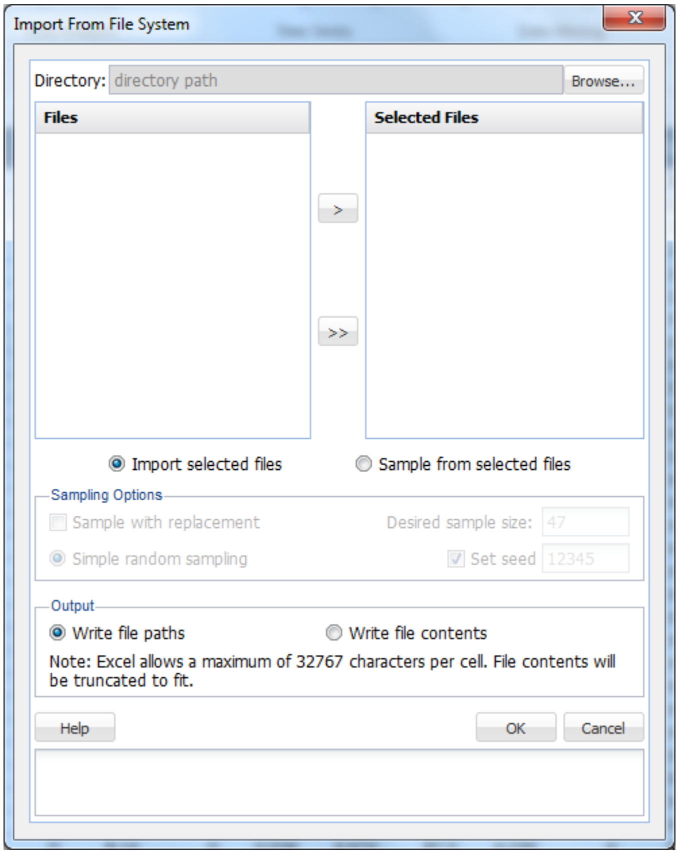
Text Data

Doc ID	TextVar
Doc 1	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101645
Doc 2	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53662
Doc 3	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53605
Doc 4	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53595
Doc 5	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102911
Doc 6	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102891
Doc 7	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102986
Doc 8	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53758
Doc 9	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\52779
Doc 10	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\52805
Doc 11	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53658
Doc 12	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53661
Doc 13	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53710
Doc 14	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102863
Doc 15	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53704
Doc 16	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53735
Doc 17	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53561
Doc 18	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102991
Doc 19	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\103085
Doc 20	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53583
Doc 21	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53640
Doc 22	Jsers\Nicole\Documents\Frontline\TextMiner Files\Electronics\53797

Importing from File Folder Options

See below for an explanation of each option as displayed on the Import from File System dialog in Data Science Desktop.

Note: Analytic Solver Data Science only supports the import of delimited text files. A delimited text file is one in which data values are separated by a character such as quotation marks, commas or tabs. These characters define a beginning and end of a string of text. This functionality if not supported in Data Science Cloud.



● Import selected files ○ Sample from selected files

□ Sample with replacement Desired sample size: 47

● Simple random sampling ☑ Set seed 12345

● Write file paths ○ Write file contents

Note: Excel allows a maximum of 32767 characters per cell. File contents will be truncated to fit.

Directory

Click *Browse* to navigate to the directory that contains the collection of text documents.

Files

The files contained within the file folder as selected for *Directory* will appear here. Click the > command button to move individual files or the >> button to move the entire collection to the *Selected Files* listbox.

Selected Files

The text files listed here have been selected for import or sampling.

Import selected files

Select this option to import the selected text files.

Sample from selected files

Select this option to choose a randomly selected sample from the collection of text documents according to the options selected within the Sampling Options section.

Sample With replacement

If this option is checked, the text files will be sampled with replacement. The default is Sampling without Replacement. When Sampling with replacement, text documents chosen during sampling will not be removed from the collection.

Desired sample size

Enter a value for the desired sample size. This value determines the number of text documents to be included in the sample. The default value is half of the number of documents listed in the *Selected Files* list box.

Simple random sampling

If this option is selected, a simple random sample of say, size *n*, is chosen from the documents in the Selected Files list box in such a way that every random set of *n* items from the population has an equal chance of being chosen to be included in the sample. Thus simple random sampling not only avoids bias in the choice of individual items but also gives every possible document an equal chance of being selected. This option is selected by default when *Sample from selected files* is enabled.

Set Seed

This option initializes the random number generator. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the sample of documents is selected. The default value is "12345". When the seed is zero, the random number generator is initialized from the system clock, so the sequence of documents selected will be different each time a sample is taken. If you need the results from successive runs to be strictly comparable, you should set the seed. To do this, select the checkbox next to the Set Seed edit box, or type the number you want into the box. This option is selected by default when *Sample from selected files* is enabled. This option accepts both positive and negative integers with up to 9 digits.

Output

If *Write file paths* is selected, pointers to the file locations are stored on the *FileSampling* output sheet. If *Write file contents* is selected, the content of each text document will be written to a cell on the *FileSampling* output, up to a maximum of 32,767 characters.

Generate Data

Introduction

The newly released Synthetic Data Generation feature included in the latest version of Analytic Solver Data Science allows users to generate synthetic data by automated Metalog probability distribution selection and parameter fitting, Rank Correlation or Copula fitting, and random sampling. This can be beneficial for several reasons such as when the actual training data is limited or when the data owner is unwilling to release the actual, full dataset but agrees to supply a limited copy or a synthetic version that statistically resembles the properties of the actual dataset.

This process consists of three main steps.

1. Fit and select a marginal probability distribution to each feature – by automated and semi-automated search within the family of bounded, semi-bounded or unbounded Metalog distributions.
2. Identify correlations among features, by using Rank Correlation or one of available Copulas – Clayton, Gumbel, Frank, Student or Gauss.
3. Generate the random sample consistent with the best-fit probability distributions and correlations.

Additional to the generated synthetic data, Analytic Solver Data Science can optionally provide the details of the fitting process – fitted coefficients and goodness-of-fit metrics for all fitted candidate Metalog distributions, selected distribution for each feature and fitted correlation matrix.

To further explore the original/synthetic data and compare them, one may easily compute basic and advanced statistics for original and/or synthetic data, including but not limited to percentiles and Six Sigma metrics.

Notes: Supported only in Analytic Solver Data Science and Analytic Solver Comprehensive. This button will be disabled in all other product licenses.

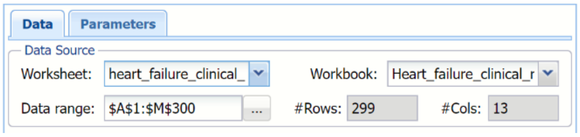
Generate Data Example

This example utilizes the Heart Failure example dataset (Chicco & Jurman, 2020) to illustrate how to generate synthetic data using Analytic Solver Data Science.

1. Open the Heart Failure dataset by clicking Help - Example Models - Forecasting/Data Science Examples – Heart Failure. This example dataset contains information pertaining to patients presenting with heart failure as seen in a cardiology clinic.
2. Confirm that the heart_failure_clinical_records tab is selected, and click Generate Data on the Data Science ribbon to bring up the Generate Synthetic Data tab.
3. The top of the dialog displays the information for the Data Source: Worksheet name, heart_failure_clinical_records, Workbook name,

Heart_failure_clinical_records, the data range, A1:M300 and the number of rows and columns in the dataset.

Data Source section of Generate Synthetic Data tab



Data Source

Worksheet: heart_failure_clinical_

Workbook: Heart_failure_clinical_r

Data range: $A$1:$M$300

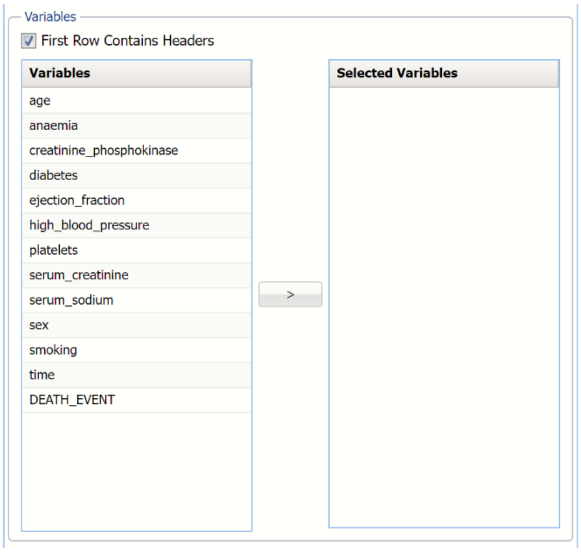
#Rows: 299

#Cols: 13

4. Select all continuous variables under Variables and then use the > to move them to Selected Variables.

   Recall that categorical variables are not supported. Anaemia, diabetes, high_blood_pressue, sex, smoking and death_event are all categorical variables that will not be included in the example.

   Variables section of Generate Synthetic Data tab



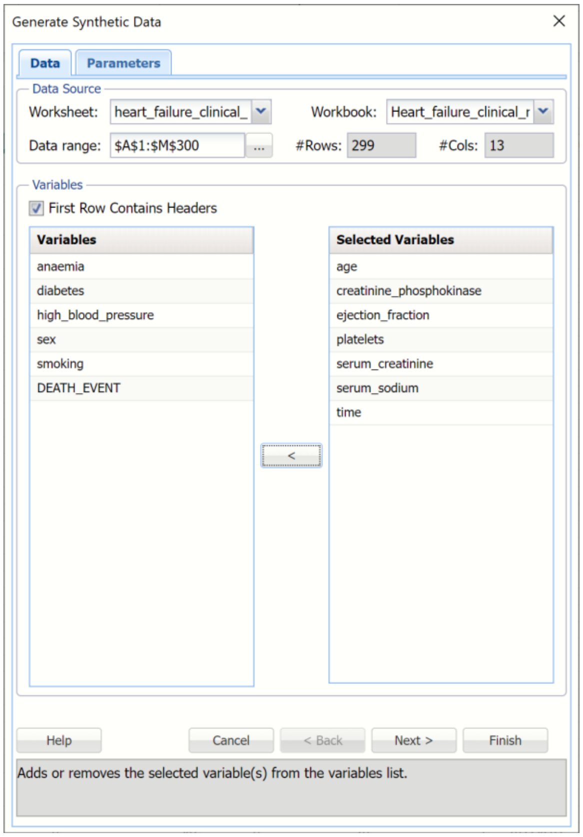
Variables

☑ First Row Contains Headers

Variables	Selected Variables
age	
anaemia	
creatinine_phosphokinase	
diabetes	
ejection_fraction	
high_blood_pressure	
platelets	
serum_creatinine	
serum_sodium	
sex	
smoking	
time	
DEATH_EVENT	

5. Click Next to move to the Parameters tab.

Populated Generate Synthetic Data, Data tab



Generate Synthetic Data

Data Source

Worksheet: heart_failure_clinical_

Workbook: Heart_failure_clinical_r

Data range: $A$1:$M$300

#Rows: 299

#Cols: 13

Variables

☑ First Row Contains Headers

Variables	Selected Variables
anaemia	age
diabetes	creatinine_phosphokinase
high_blood_pressure	ejection_fraction
sex	platelets
smoking	serum_creatinine
DEATH_EVENT	serum_sodium
	time

Adds or removes the selected variable(s) from the variables list.

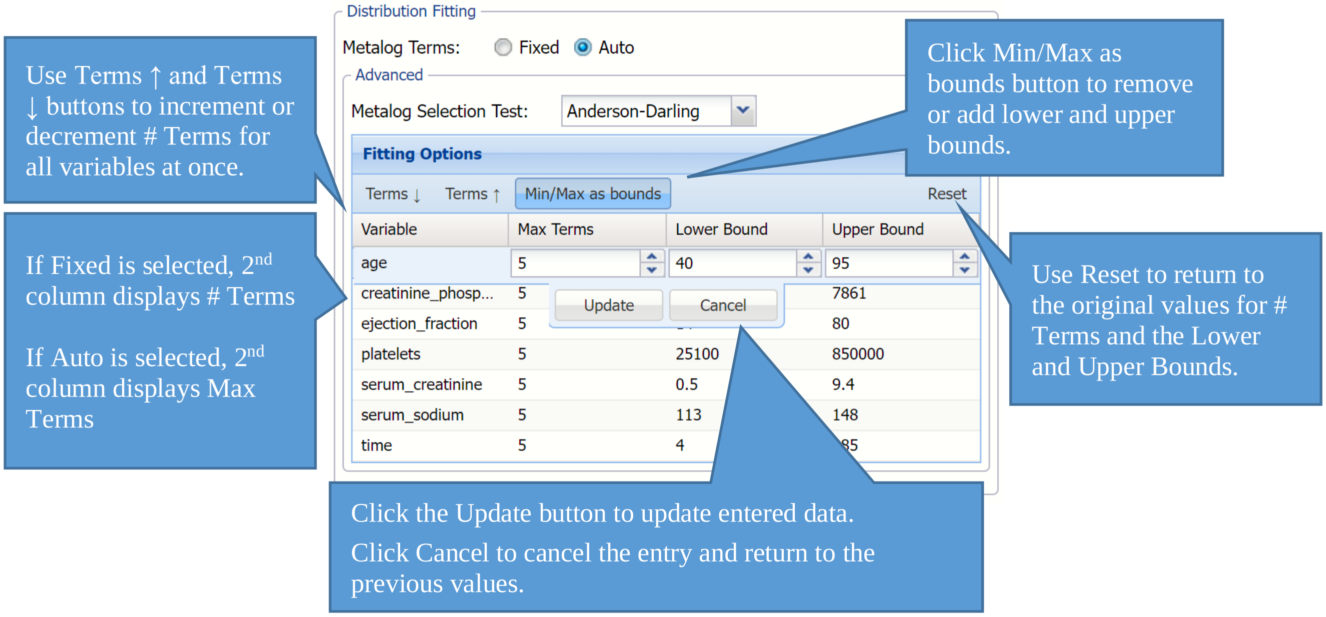
6. For this example, select Auto and leave Metalog Selection Test at the default setting of Anderson-Darling.

Metalog Terms:

- If Fixed is selected, Analytic Solver will attempt to fit and use the Metalog distribution with the specified number of terms entered into the # Terms column. (Only 1 distribution will be fit.) If Fixed is selected, Metalog Selection Test is disabled.
- If Auto is selected, Analytic Solver will attempt to fit *all possible* Metalog distributions, up to the entered value for Max Terms, and select and utilize the best Metalog distribution according to the goodness-of-fit test selected in the Metalog Selection Test menu.

Click the down arrow on the right of Fitting Options to enter either the maximum number of terms (if Auto is selected) or the exact number of terms (if Fixed is selected) for each variable as well as a lower and/or upper bound. By default the lower and upper bounds are set to the variable's minimum and maximum values, respectively. If no lower or upper bound is entered, Analytic Solver will fit a semi- (with one bound present) or unbounded (with no bounds present) Metalog function.

Distribution Fitting section of the Generate Data tab



Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5	40	95
creatinine_phosp...	5		7861
ejection_fraction	5		80
platelets	5	25100	850000
serum_creatinine	5	0.5	9.4
serum_sodium	5	113	148
time	5	4	85

Use Terms ↑ and Terms ↓ buttons to increment or decrement # Terms for all variables at once.

If Fixed is selected, 2nd column displays # Terms

If Auto is selected, 2nd column displays Max Terms

Click Min/Max as bounds button to remove or add lower and upper bounds.

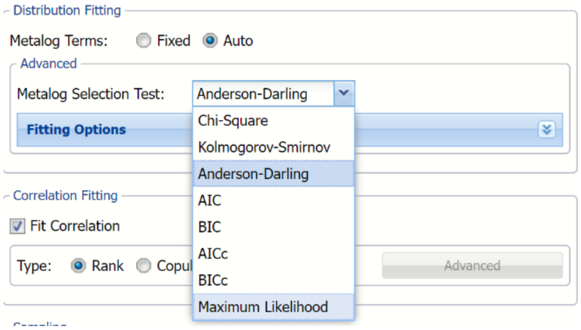
Use Reset to return to the original values for # Terms and the Lower and Upper Bounds.

Click the Update button to update entered data.

Click Cancel to cancel the entry and return to the previous values.

Metalog Selection Test: Click the down arrow to select the desired Goodness-of-Fit test used by Analytic Solver. The Goodness of Fit test is used to select the best Metalog form for each data variable among the candidate distributions containing a different number of terms, from 2 to the value entered for Max Terms. The default Goodness-of-Fit test is Anderson-Darling.

Metalog Selection Test menu



Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copul

Goodness of Fit Tests:

- Chi Square – Uses the chi-square statistic and distribution to rank the distributions. Sample data is first divided into intervals using either equal probability, then the number of points that fall into each interval are compared with the expected number of points in each interval. The null hypotheses is rejected using a 90% significance level, if the chi-squared test statistic is greater than the critical value statistic.

  Note: The Chi Square test is used indirectly in continuous fitting as a support in the AIC test. The AIC test must succeed in fitting as this is a necessary condition as well as the fitting of at least one of the tests, Chi Squared, Kolmogorov-Smirnoff, or Anderson-Darling.

- Kolmogorov-Smirnoff –This test computes the difference (D) between the continuous distribution function (CDF) and the empirical cumulative distribution function (ECDF). The null hypothesis is rejected if, at the 90% significance level, D is larger than the critical value statistic.

- Anderson (Default) -Darling –Ranks the fitted distribution using the Anderson Darling statistic, A2 . The null hypothesis is rejected using a 90% significance level, if A2 is larger than the critical value statistic. This test awards more weight to the distribution tails then the Kolmogorov-Smirnoff test.

- AIC – The AIC test is a Chi Squared test corrected for the number of distribution parameters and sample size. AIC = 2 * p – 2 + ln(L) where p is the number of distribution parameters, n is the fitted sample size (number of data points) and ln(L) is the log-likelihood function computed on the fitted data.

- AICc –When the sample size is small, there is a significant chance that the AIC test will select a model with a large number of parameters. In other words, AIC will overfit the data. AICc was developed to reduce the possibility of overfitting by applying a penalty to the number of parameters. Assuming that the model is univariate, is linear in the parameters and has normally-distributed residuals, the formula for AICc is: AICc = AIC + 2 * p *(p + 1) / (n – p − 1) where n = sample size, p = # of parameters. As the sample size approaches infinity, the penalty on the number of parameters converges to 0 resulting in AICc converging to AIC.

- BIC – The Bayesian information criterion (BIC) is defined as: BIC = ln(n) * p -2 * ln(L) where p is the number of distribution parameters, n is the fitted sample size (number of data points) and ln(L) is the log-likelihood function computed on the fitted data.

- BICc – The BICc is the alternative version of BIC, corrected for the sample size BICc = BIC + 2 * p * (p + 1) / (n – p - 1).

- Maximum Likelihood – The (negated) raw value of the estimated maximum log likelihood utilized in tests described above.

7. Select **Fit Correlation** to fit a correlation between the variables. If this option is left unchecked, correlation fitting will not be performed. Leave Rank, the default setting, selected for Type.

   If *Rank* is selected Analytic Solver will use the *Spearman rank order correlation coefficient* to compute a correlation matrix that includes all included variables.

   Selecting *Copula* opens the Copula Options dialog where you can select and drag five types of copulas into a desired order of priority.

   Correlation Fitting section of the Generate Data tab

The Spearman rank order correlation coefficient is a *nonparametric* measure of correlation that is computed from a rank ordering of the trial values drawn for all variables. It can be used to induce correlations between *any* two uncertain variables, whether they have the same or different analytic distributions, or even custom distributions. This correlation coefficient ranges in value from -1 to +1.

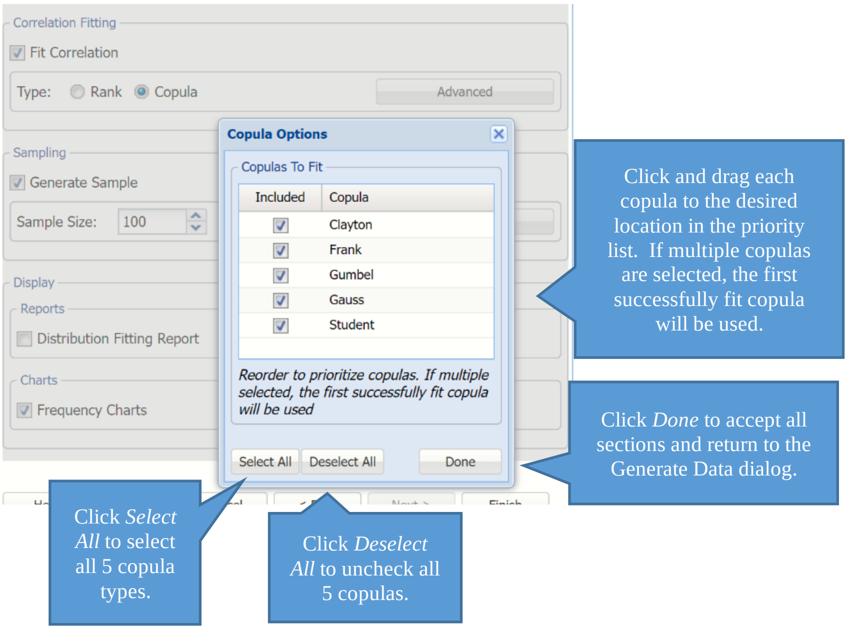
Analytic Solver Data Science includes copulas to improve the method of defining the correlation or dependence between two or more variables. Copulas offer more flexibility over the rank order correlation method, and are able to capture complex correlations.

An n – dimensional copula C is a multi-variate probability distribution where the marginal probability distribution of each variable follows the Uniform(0,1) distribution. A major benefit of copulas is that they allow two or more uncertain variables to be correlated without changing the shape of the original uncertain variable distributions.

For some copula C, a multi-variate distribution F with distributions of F_1, F_2, … F_n can be written as:

$$\mathbf{F(x_1, \ldots, x_n) = C(F_1(x_1), F_2(x_2), \ldots, F_n(x_n))}$$

Analytic Solver supports five types of copulas: three Archimedean copulas (clayton, frank, and gumbel) and two elliptical copulas (Gauss and Student).



Correlation Fitting

☑ Fit Correlation

Type: ○ Rank ● Copula

Sampling

☑ Generate Sample

Sample Size: 100

Display

Reports

□ Distribution Fitting Report

Charts

☑ Frequency Charts

Copula Options

Copulas To Fit

Included	Copula
☑	Clayton
☑	Frank
☑	Gumbel
☑	Gauss
☑	Student

Reorder to prioritize copulas. If multiple selected, the first successfully fit copula will be used

Click and drag each copula to the desired location in the priority list. If multiple copulas are selected, the first successfully fit copula will be used.

Click *Done* to accept all sections and return to the Generate Data dialog.

Click *Select All* to select all 5 copula types.

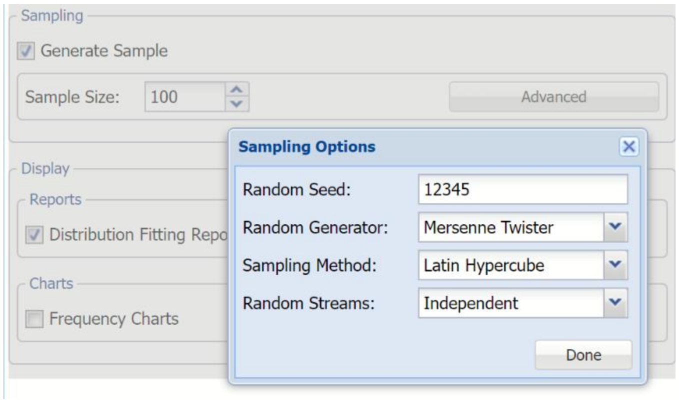
Click *Deselect All* to uncheck all 5 copulas.

8. Select **Generate Sample** to generate synthetic data for each selected variable. Use the Sample Size field to increase the size of the sample generated. Keep the default of 100.

   If this option is left unchecked, variable data will be fitted to a Metalog distribution and also correlations, if Fit Correlation is selected, but no synthetic data will be generated.

   Click Advanced to open the Sampling Options dialog.

   Sampling Options dialog



Sampling

☑ Generate Sample

Sample Size: 100

Display

Reports

☑ Distribution Fitting Repo

Charts

□ Frequency Charts

Sampling Options

Random Seed: 12345

Random Generator: Mersenne Twister

Sampling Method: Latin Hypercube

Random Streams: Independent

From this dialog, users can set the Random Seed, Random Generator, Sampling Method and Random Streams.

Random Seed: Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the *same* sequence of random numbers is used for each simulation. When the seed is zero or the field is left empty, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each simulation. If you need the results from one simulation to another to be strictly comparable, you should set the seed. To do this, simply type the desired number into the box. (Default Value = 12345)

Random Generator: Use this menu to select a random number generation algorithm. Analytic Solver Data Science includes an advanced set of random number generation capabilities.

Computer-generated numbers are never truly “random,” since they are always computed by an algorithm – they are called *pseudorandom* numbers. A random number generator is designed to quickly generate sequences of numbers that are as close to statistically independent as possible. Eventually, an algorithm will generate the same number seen sometime earlier in the sequence, and at this point the sequence will begin to repeat. The *period* of the random number generator is the number of values it can generate before repeating.

A long period is desirable, but there is a tradeoff between the length of the period and the degree of statistical independence achieved within the period. Hence, Analytic Solver Data Science offers a choice of four random number generators:

- *Park-Miller* “Minimal” Generator with Bayes-Durham shuffle and safeguards. This generator has a period of 2^{31}-2. Its properties are good, but the following choices are usually better.
- Combined Multiple Recursive Generator of L’Ecuyer (*L’Ecuyer-CMRG*). This generator has a period of 2^{191}, and excellent statistical independence of samples within the period.
- Well Equidistributed Long-period Linear (*WELL*) generator of Panneton, L’Ecuyer and Matsumoto. This generator combines a long period of 2^{1024} with very good statistical independence.
- *Mersenne Twister* (default setting) generator of Matsumoto and Nishimura. This generator has the longest period of 2^{19937}-1, but the samples are not as “equidistributed” as for the WELL and L-Ecuyer-CMRG generators.
- *HDR* Random Number Generator, designed by Doug Hubbard. Permits data generation running on various computer platforms to generate identical or independent streams of random numbers.

Samplng Method: Use this option group to select *Monte Carlo, Latin Hypercube*, or *Sobol RQMC* sampling.

- *Monte Carlo:* In standard Monte Carlo sampling, numbers generated by the chosen random number generator are used directly to obtain sample values. With this method, the variance or estimation error in computed samples is inversely proportional to the square root of the number of trials (controlled by the Sample Size); hence to cut the error in half, four times as many trials are required.

Analytic Solver Data Science provides two other sampling methods than can significantly improve the ‘coverage’ of the sample space, and thus reduce the variance in computed samples. This means that you can achieve a given level of accuracy (low variance or error) with fewer trials.

- *Latin Hypercube (default):* Latin Hypercube sampling begins with a stratified sample in each dimension (one for each selected variable), which constrains the random numbers drawn to lie in a set of subintervals from 0 to 1. Then these one-dimensional samples are combined and randomly permuted so that they ‘cover’ a unit hypercube in a stratified manner.

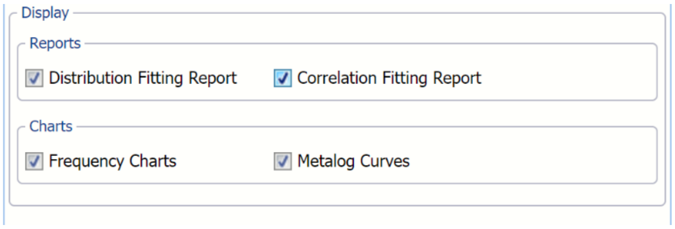
- *Sobol RQMC (Randomized QMC)***.** Sobol numbers are an example of so-called "Quasi Monte Carlo" or "low-discrepancy numbers," which are generated with a goal of coverage of the sample space rather than "randomness" and statistical independence. Analytic Solver Data Science adds a "random shift" to Sobol numbers, which improves their statistical independence.

Random Streams: Use this option group to select a *Single Stream* for each variable or an *Independent Stream (the default)* for each variable.

If *Single Stream* is selected, a single sequence of random numbers is generated. Values are taken consecutively from this sequence to obtain samples for each selected variable. This introduces a subtle dependence between the samples for all distributions in one trial. In many applications, the effect is too small to make a difference – but in some cases, better results are obtained if *independent* random number sequences (streams) are used for each distribution in the model. Analytic Solver Data Science offers this capability for Monte Carlo sampling and Latin Hypercube sampling; it does not apply to Sobol numbers.

9. In the Display section, keep Distribution Fitting Report and Correlation Fitting Report selected. Select Frequency Charts and Metalog Curves.

Generate Synthetic Data Tab, Display section of Parameters tab



Display

Reports

☑ Distribution Fitting Report ☑ Correlation Fitting Report

Charts

☑ Frequency Charts ☑ Metalog Curves

Distribution Fitting Report: Report included on the SyntheticData_Output worksheet includes the number of terms, the coefficients for each term, the lower and upper bounds and the goodnesss of fit statistics used when fitting each Metalog distribution.

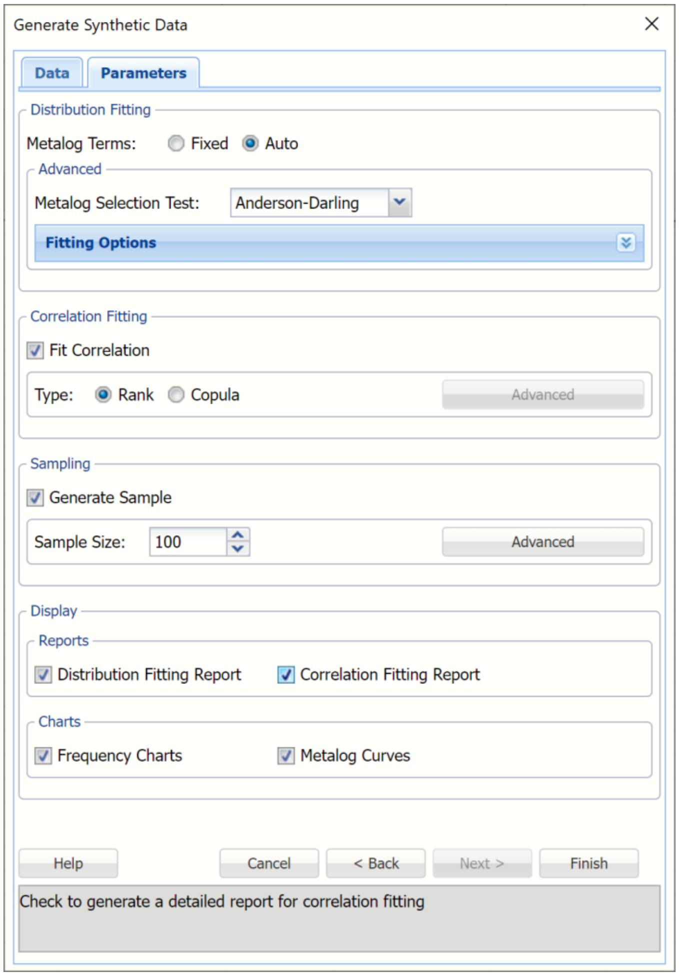
Correlation Fitting Report: Displays the correlation matrix SyntheticData_Output worksheet

Frequency Charts: Displays the multivariate chart produced by the Analyze Data feature. Double click each chart to view an interactive chart and detailed data (statistics, percentiles and six sigma indices) about each variable included in the analysis. For more information on the Analyze Data feature included in the latest version of Analytic Solver Data Science, see the Exploring Data chapter that appears later in this guide.

Metalog Curves: Select this Chart option to add Metalog distribution curves to each variable displayed in the multivariate chart and interactive charts described above for Frequency Charts.

10. **Click Finish.**

Generate Synthetic Data, Parameters tab



Generate Synthetic Data

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

☑ Generate Sample

Sample Size: 100

Display

Reports

☑ Distribution Fitting Report ☑ Correlation Fitting Report

Charts

☑ Frequency Charts ☑ Metalog Curves

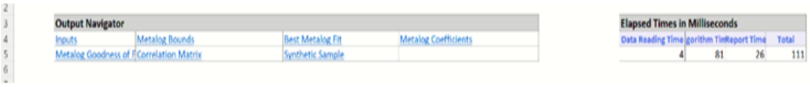
Check to generate a detailed report for correlation fitting

Results

With the selected options shown in the screenshot above, two workbooks will be inserted into the workbook, SyntheticData_Output and SyntheticData_Sample.

At the top of both worksheets is the Output Navigator. Click any link to easily navigate to that section of the worksheet.

SyntheticData_Output: Output Navigator



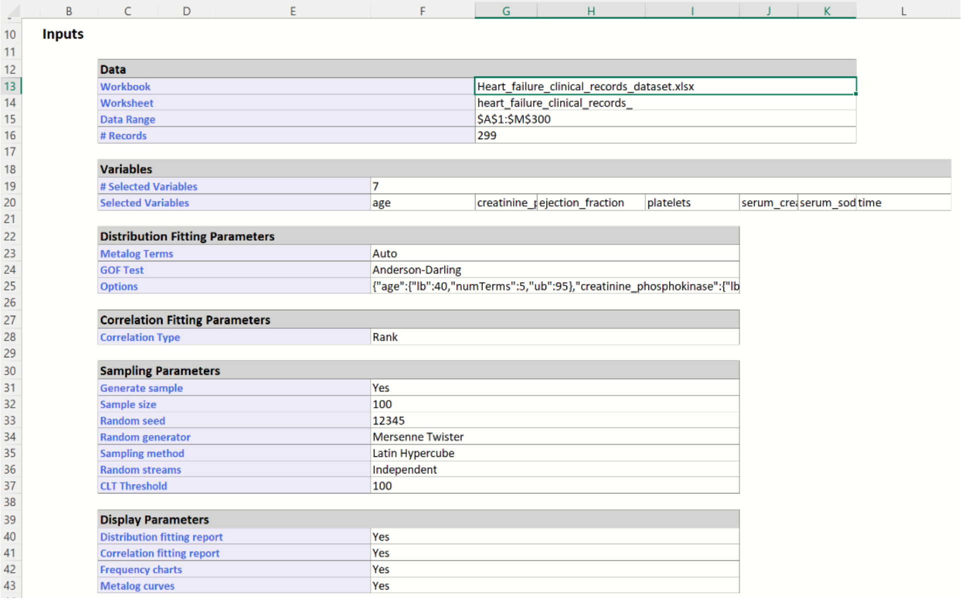
Output Navigator			
Inputs	Metalog Bounds	Best Metalog Fit	Metalog Coefficients
Metalog Goodness of F	Correlation Matrix	Synthetic Sample	

Elapsed Times in Milliseconds			
Data Reading Time	gorithm Tin	Report Time	Total
4	81	26	111

SynthethicData_Output

Scroll down to the Inputs section of the SyntheticData_Output worksheet to view all user inputs including the data source, the selected variables, the distribution fitting parameters, correlation fitting parameters, sampling parameters and display parameters.

SyntheticData_Output: Inputs



Inputs

Data	
Workbook	Heart_failure_clinical_records_dataset.xlsx
Worksheet	heart_failure_clinical_records_
Data Range	$A$1:$M$300
# Records	299

Variables							
# Selected Variables	7						
Selected Variables	age	creatinine_p	ejection_fraction	platelets	serum_crea	serum_sod	time

Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"age":{"lb":40,"numTerms":5,"ub":95},"creatinine_phosphokinase":{"lb

Correlation Fitting Parameters	
Correlation Type	Rank

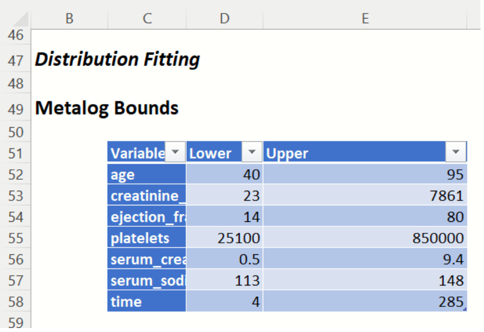
Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
CLT Threshold	100

Display Parameters	
Distribution fitting report	Yes
Correlation fitting report	Yes
Frequency charts	Yes
Metalog curves	Yes

Distribution Fitting Report: If Distribution Fitting Report was selected on the Parameters tab on the Generate Data dialog, the Distribution Fitting Report will be inserted into the SyntheticData_Output worksheet. This report contains information for each of the fitted Metalog distributions, such as:

- Metalog Bounds: This table shows the lower and upper bounds as entered in the Fitting Options section of the Generate Synthetic Data dialog, Parameters tab.

  SyntheticData_Output: Metalog Bounds report

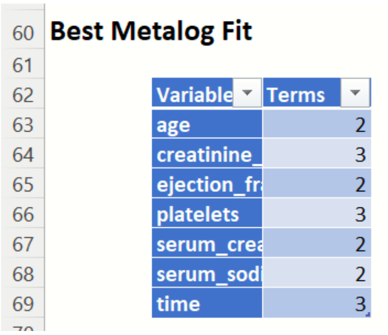
  Distribution Fitting

  Metalog Bounds

Variable	Lower	Upper
age	40	95
creatinine_	23	7861
ejection_fr	14	80
platelets	25100	850000
serum_crea	0.5	9.4
serum_sod	113	148
time	4	285

- Best Metalog Fit:
  - If **Auto** is selected for Metalog Terms, this table displays the number of terms for the best Metalog distribution for each variable, as decided by the selected Goodness-of-Fit test.
  - If **Fixed** is selected for Metalog Terms, this table displays the number of terms for the (only) Metalog distribution fit for each variable.

SyntheticData_Output: Best Metalog Fit report



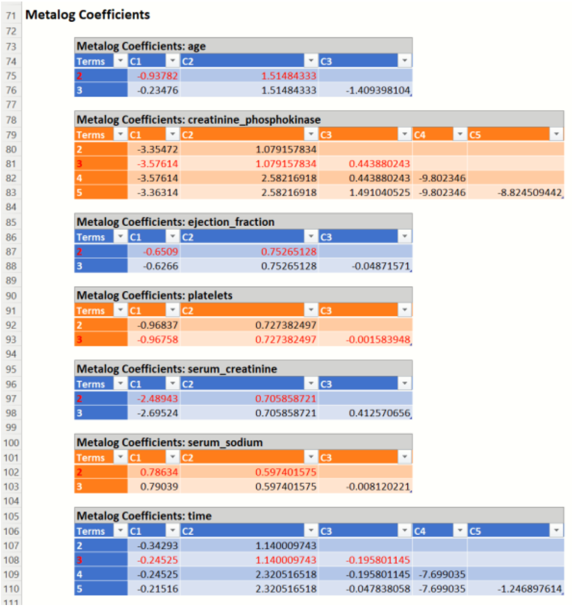
Best Metalog Fit

Variable	Terms
age	2
creatinine_	3
ejection_fr	2
platelets	3
serum_crea	2
serum_sodi	2
time	3

- Metalog Coefficients: This table shows the fitted coefficients for all feasible Metalog distributions that Analytic Solver Data Science attempted to fit for each variable. The best distribution (as decided by the chosen Goodness-of-Fit test) that will be used in sample generation (if requested) is highlighted in red.

  Note that it is not guaranteed that all possible Metalog distributions will be fit. As shown in the screenshot below, not all variables have exactly 5 Metalog distributions for 1, 2, 3, 4 and 5 terms.

SyntheticData_Output: Metalog Coefficients report



Metalog Coefficients

Metalog Coefficients: age

Terms	C1	C2	C3
2	-0.93782	1.51484333	
3	-0.23476	1.51484333	-1.409398104

Metalog Coefficients: creatinine_phosphokinase

Terms	C1	C2	C3	C4	C5
2	-3.35472	1.079157834			
3	-3.57614	1.079157834	0.443880243		
4	-3.57614	2.58216918	0.443880243	-9.802346	
5	-3.36314	2.58216918	1.491040525	-9.802346	-8.824509442

Metalog Coefficients: ejection_fraction

Terms	C1	C2	C3
2	-0.6509	0.75265128	
3	-0.6266	0.75265128	-0.04871571

Metalog Coefficients: platelets

Terms	C1	C2	C3
2	-0.96837	0.727382497	
3	-0.96758	0.727382497	-0.001583948

Metalog Coefficients: serum_creatinine

Terms	C1	C2	C3
2	-2.48943	0.705858721	
3	-2.69524	0.705858721	0.412570656

Metalog Coefficients: serum_sodium

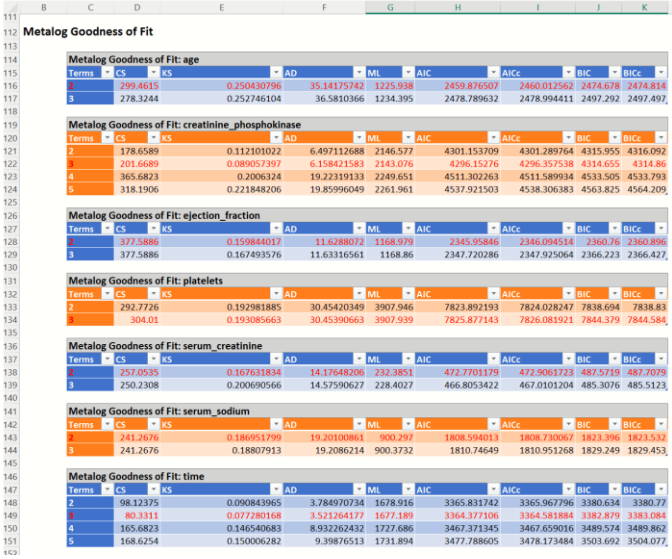
Terms	C1	C2	C3
2	0.78634	0.597401575	
3	0.79039	0.597401575	-0.008120221

Metalog Coefficients: time

Terms	C1	C2	C3	C4	C5
2	-0.34293	1.140009743			
3	-0.24525	1.140009743	-0.195801145		
4	-0.24525	2.320516518	-0.195801145	-7.699035	
5	-0.21516	2.320516518	-0.047838058	-7.699035	-1.246897614

- Metalog Goodness-of-Fit: This table shows the Goodness-of-Fit test statistics for all feasible Metalog distributions that Analytic Solver Data Science attempted to fit for each variable. As in the Metalog Coefficients report, the best distribution (as decided by the chosen Goodness-of-Fit test) that will be used in sample generation (if requested) is highlighted in red.

SyntheticData_Output: Metalog Goodness of Fit report



Metalog Goodness of Fit

Metalog Goodness of Fit: age

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	299.4615	0.250430796	35.14175742	1225.938	2459.876507	2460.012562	2474.678	2474.814
3	278.3244	0.252746104	36.5810366	1234.395	2478.789632	2478.994411	2497.292	2497.497

Metalog Goodness of Fit: creatinine_phosphokinase

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	178.6589	0.112101022	6.497112688	2146.577	4301.153709	4301.289764	4315.955	4316.092
3	201.6689	0.089057397	6.158421583	2143.076	4296.15276	4296.357538	4314.655	4314.86
4	365.6823	0.2006324	19.22319133	2249.651	4511.302263	4511.589934	4533.505	4533.793
5	318.1906	0.221848206	19.85996049	2261.961	4537.921503	4538.306383	4563.825	4564.209

Metalog Goodness of Fit: ejection_fraction

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	377.5886	0.159844017	11.6288072	1168.979	2345.95846	2346.094514	2360.76	2360.896
3	377.5886	0.167493576	11.63316561	1168.86	2347.720286	2347.925064	2366.223	2366.427

Metalog Goodness of Fit: platelets

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	292.7726	0.192981885	30.45420349	3907.946	7823.892193	7824.028247	7838.694	7838.83
3	304.01	0.193085663	30.45390663	3907.939	7825.877143	7826.081921	7844.379	7844.584

Metalog Goodness of Fit: serum_creatinine

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	257.0535	0.167631834	14.17648206	232.3851	472.7701179	472.9061723	487.5719	487.7079
3	250.2308	0.200690566	14.57590627	228.4027	466.8053422	467.0101204	485.3076	485.5123

Metalog Goodness of Fit: serum_sodium

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	241.2676	0.186951799	19.20100861	900.297	1808.594013	1808.730067	1823.396	1823.532
3	241.2676	0.18807913	19.2086214	900.3732	1810.74649	1810.951268	1829.249	1829.453

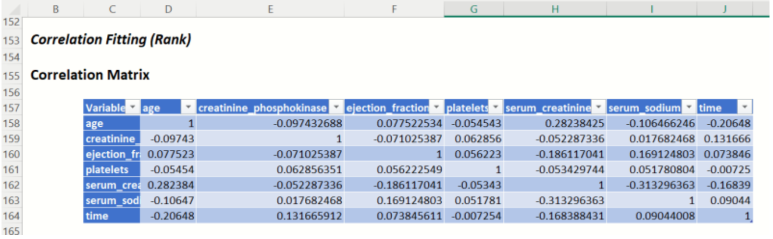
Metalog Goodness of Fit: time

Terms	CS	KS	AD	ML	AIC	AICc	BIC	BICc
2	98.12375	0.090843965	3.784970734	1678.916	3365.831742	3365.967796	3380.634	3380.77
3	80.3311	0.077280168	3.521264177	1677.189	3364.377106	3364.581884	3382.879	3383.084
4	165.6823	0.146540683	8.932262432	1727.686	3467.371345	3467.659016	3489.574	3489.862
5	168.6254	0.150006282	9.39876513	1731.894	3477.788605	3478.173484	3503.692	3504.077

Correlation Fitting Report: If Correlation Fitting Report is selected on the Parameters tab of the Generate Data dialog, the Correlation Fitting Report will be inserted on the SyntheticData_Output worksheet.

This report contains the Correlation Matrix containing the correlations between variables using the correlation technique selected, Rank or Copula.

SyntheticData_Output: Correlation Matrix.



Correlation Fitting (Rank)

Correlation Matrix

Variable	age	creatinine_phosphokinase	ejection_fraction	platelets	serum_creatinine	serum_sodium	time
age	1	-0.097432688	0.077522534	-0.054543	0.28238425	-0.106466246	-0.20648
creatinine_	-0.09743	1	-0.071025387	0.062856	-0.052287336	0.017682468	0.131666
ejection_fr	0.077523	-0.071025387	1	0.056223	-0.186117041	0.169124803	0.073846
platelets	-0.05454	0.062856351	0.056222549	1	-0.053429744	0.051780804	-0.00725
serum_crea	0.282384	-0.052287336	-0.186117041	-0.05343	1	-0.313296363	-0.16839
serum_sod	-0.10647	0.017682468	0.169124803	0.051781	-0.313296363	1	0.09044
time	-0.20648	0.131665912	0.073845611	-0.007254	-0.168388431	0.09044008	1

SynthethicData_Sample

Click the SyntheticData_Sample worksheet to view the synthetic data for each selected variable. (Recall that this data is available because Generate Data was selected on the Parameters tab on the Generate Data dialog.) This worksheet also includes the Output Navigator as described above for the SyntheticData_Output worksheet.

Recall that to produce this synthetic data, Analytic Solver:

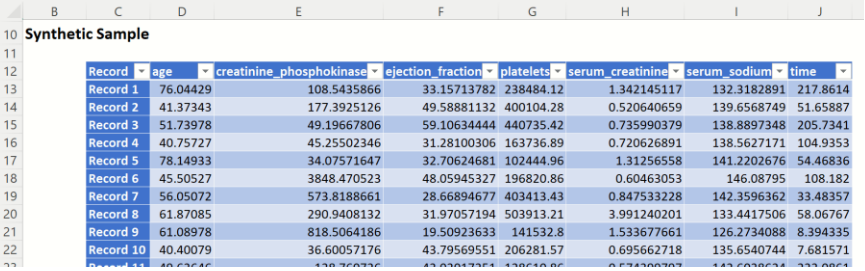
1. For each selected variable, Analytic Solver first fit the original data to a Metalog distribution by using either a fixed number of terms or the automatic search option.

2. Afterwards, Analytic Solver fit a correlation to all variables by using either Rank Correlation or one of the five available copulas.

3. Lastly, the trial values (i.e. synthetic data) were generated using the fitted distribution and correlations.

The screenshot below displays the first 10 trial values generated for each selected variable: age, creatinine_phosphokinase, ejection_fraction, platelets, serum_creatinine, serum_sodium and time.

SytheticData: Sample: Synthetic Sample



Synthetic Sample

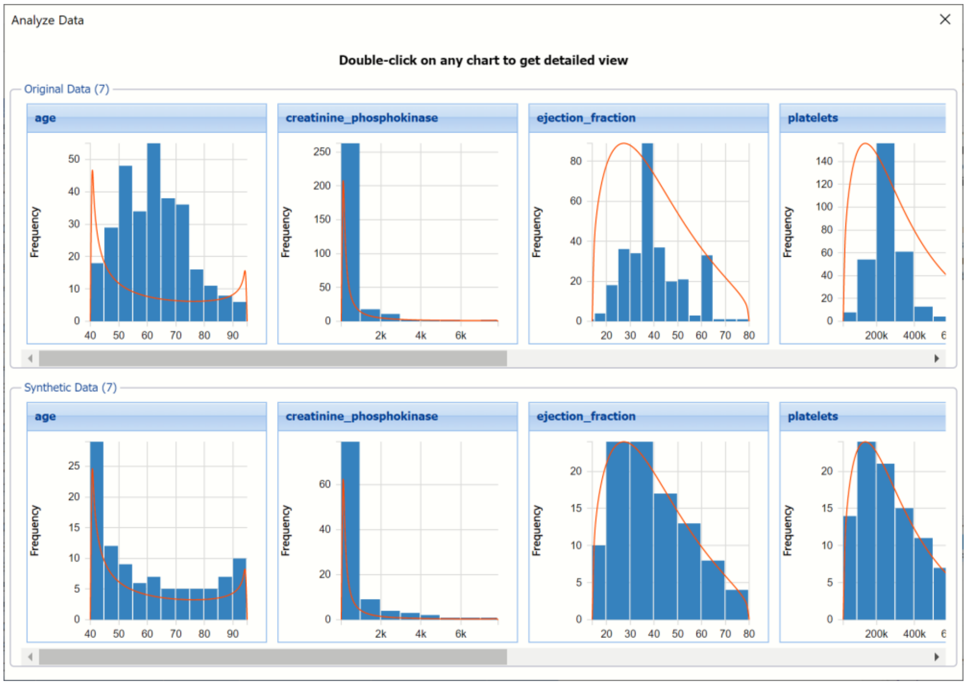
Record	age	creatinine_phosphokinase	ejection_fraction	platelets	serum_creatinine	serum_sodium	time
Record 1	76.04429	108.5435866	33.15713782	238484.12	1.342145117	132.3182891	217.8614
Record 2	41.37343	177.3925126	49.58881132	400104.28	0.520640659	139.6568749	51.65887
Record 3	51.73978	49.19667806	59.10634444	440735.42	0.735990379	138.8897348	205.7341
Record 4	40.75727	45.25502346	31.28100306	163736.89	0.720626891	138.5627171	104.9353
Record 5	78.14933	34.07571647	32.70624681	102444.96	1.31256558	141.2202676	54.46836
Record 6	45.50527	3848.470523	48.05945327	196820.86	0.60463053	146.08795	108.182
Record 7	56.05072	573.8188661	28.66894677	403413.43	0.847533228	142.3596362	33.48357
Record 8	61.87085	290.9408132	31.97057194	503913.21	3.991240201	133.4417506	58.06767
Record 9	61.08978	818.5064186	19.50923633	141532.8	1.533677661	126.2734088	8.394335
Record 10	40.40079	36.60057176	43.79569551	206281.57	0.695662718	135.6540744	7.681571

If the Frequency Chart Option was selected on the Parameters tab of the Generate Data dialog, a dialog containing frequency charts for both the *original data and the synthetic data*, for each of the selected variables is displayed immediately when the SyntheticData_Sample worksheet is opened. This chart is discussed in-depth in the Aanlyze Data section of the Exploring

From here, each chart may be selected to open a larger, more detailed, interactive chart. If this chart is closed, by clicking the X in the upper right hand corner, simply click to a different tab in the worksheet and then back to the SyntheticData_Sample tab to reopen.

Since Metalog Curves was selected on the Parameters tab, on the Generate Data dialog, the curve for each fitted Metalog function is displayed on each.

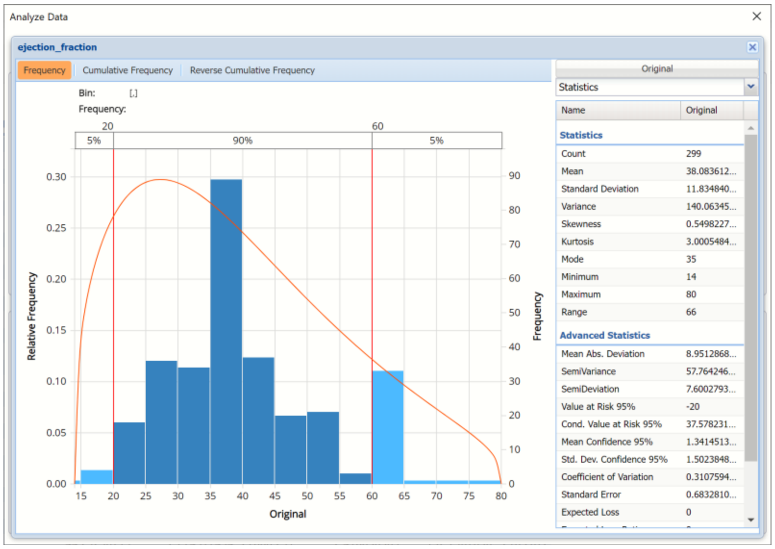
Analyze Data Preview Chart dialog



Double-click on any chart to get detailed view

Double click any of the charts, for this example double click the original data ejection_fraction chart, to open a detailed, interactive frequency chart for the Original variable data.

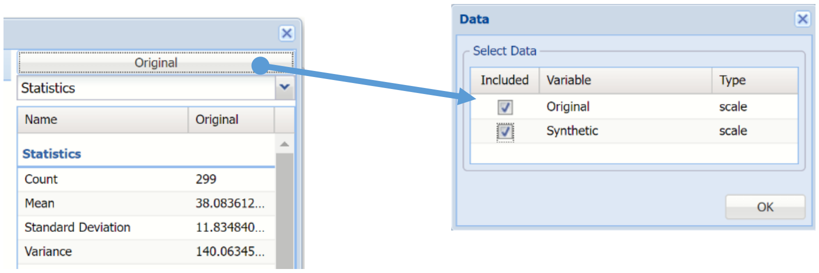
Analyze Data dialog displaying original data for ejection_fraction variable.



Name	Original
Statistics	
Count	299
Mean	38.083612...
Standard Deviation	11.834840...
Variance	140.06345...
Skewness	0.5498227...
Kurtosis	3.0005484...
Mode	35
Minimum	14
Maximum	80
Range	66
Advanced Statistics	
Mean Abs. Deviation	8.9512868...
SemiVariance	57.764246...
SemiDeviation	7.6002793...
Value at Risk 95%	-20
Cond. Value at Risk 95%	37.578231...
Mean Confidence 95%	1.3414513...
Std. Dev. Confidence 95%	1.5023848...
Coefficient of Variation	0.3107594...
Standard Error	0.6832810...
Expected Loss	0

To overlay the generated synthetic data on top of the Original data, click Original in the upper right hand corner and select both checkboxes in the Data dialog.

Click Original to add Synthetic data to the interactive chart.



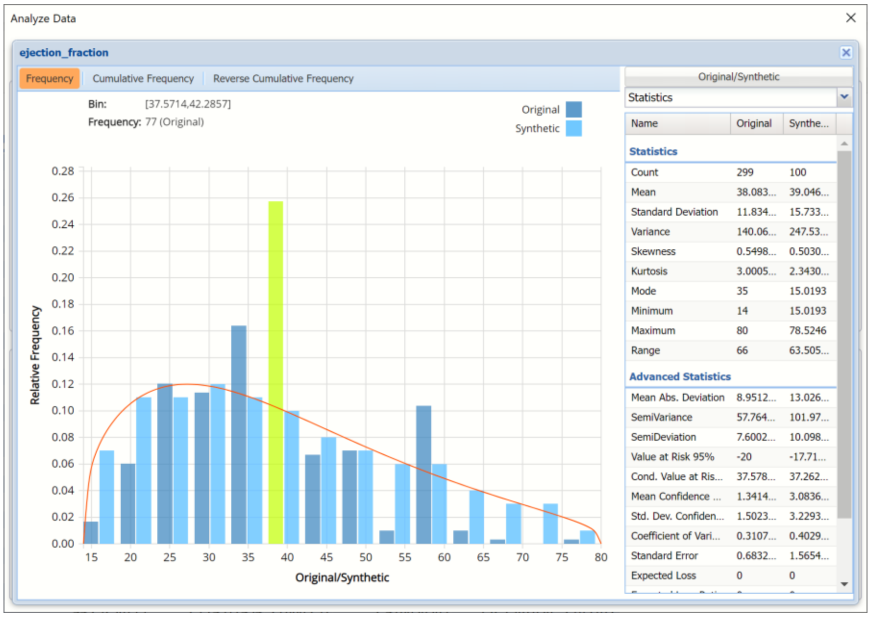
Name	Original
Statistics	
Count	299
Mean	38.083612...
Standard Deviation	11.834840...
Variance	140.06345...

Included	Variable	Type
☑	Original	scale
☑	Synthetic	scale

Notice in the screenshot below that both the Original and Synthetic data appear in the chart together, and statistics for both data appear on the right.

To remove either the Original or the Synthetic data from the chart, click Original/Synthetic in the top right and then uncheck the data type to be removed.

Analyze Data dialog shown with Frequency chart and Statistics displayed.



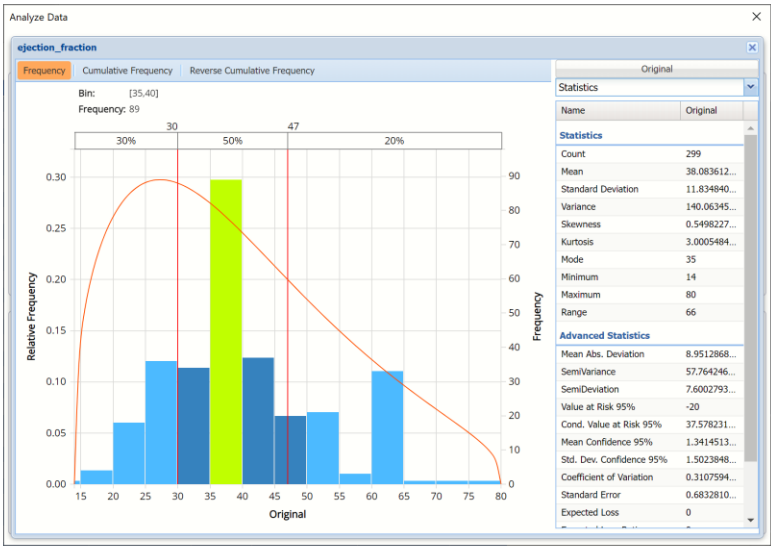
Bin: [37.5714,42.2857]
Frequency: 77 (Original)

Name	Original	Synthe...
Statistics		
Count	299	100
Mean	38.083...	39.046...
Standard Deviation	11.834...	15.733...
Variance	140.06...	247.53...
Skewness	0.5498...	0.5030...
Kurtosis	3.0005...	2.3430...
Mode	35	15.0193
Minimum	14	15.0193
Maximum	80	78.5246
Range	66	63.505...
Advanced Statistics		
Mean Abs. Deviation	8.9512...	13.026...
SemiVariance	57.764...	101.97...
SemiDeviation	7.6002...	10.098...
Value at Risk 95%	-20	-17.71...
Cond. Value at Ris...	37.578...	37.262...
Mean Confidence ...	1.3414...	3.0836...
Std. Dev. Confiden...	1.5023...	3.2293...
Coefficient of Vari...	0.3107...	0.4029...
Standard Error	0.6832...	1.5654...
Expected Loss	0	0

This chart behaves the same as the interactive chart in the Analyze Data feature found on the Explore menu.

- Use the mouse to hover over any of the bars in the graph to populate the Bin and Frequency headings at the top of the chart.

- When displaying either Original or Synthetic data (not both), red vertical lines will appear at the 5% and 95% percentile values in all three charts (Frequency, Cumulative Frequency and Reverse Cumulative Frequency) effectively displaying the 90th confidence interval. The middle percentage is the percentage of all the variable values that lie within the 'included' area, i.e. the darker shaded area. The two percentages on each end are the percentage of all variable values that lie outside of the 'included' area or the "tails". i.e. the lighter shaded area. Percentile values can be altered by moving either red vertical line to the left or right.



Bin: [35,40]

Frequency: 89

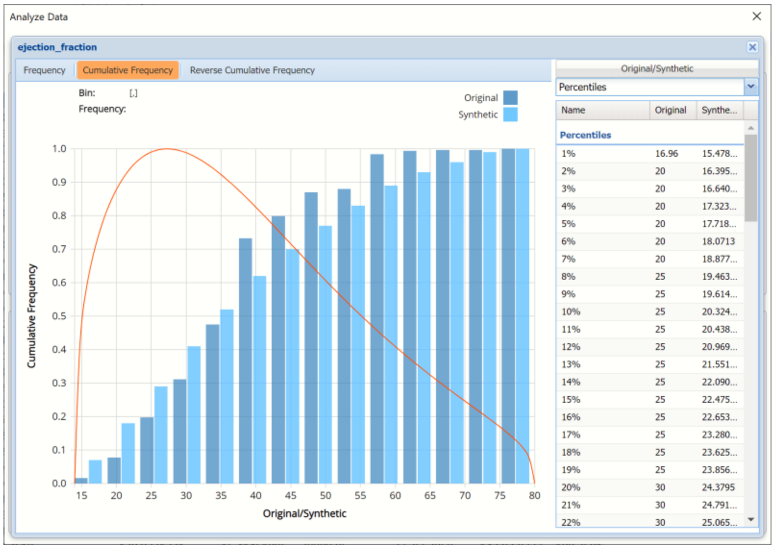
Original

Statistics

Name	Original
Statistics	
Count	299
Mean	38.083612...
Standard Deviation	11.834840...
Variance	140.06345...
Skewness	0.5498227...
Kurtosis	3.0005484...
Mode	35
Minimum	14
Maximum	80
Range	66
Advanced Statistics	
Mean Abs. Deviation	8.9512868...
SemiVariance	57.764246...
SemiDeviation	7.6002793...
Value at Risk 95%	-20
Cond. Value at Risk 95%	37.578231...
Mean Confidence 95%	1.3414513...
Std. Dev. Confidence 95%	1.5023848...
Coefficient of Variation	0.3107594...
Standard Error	0.6832810...
Expected Loss	0

- Click Cumulative Frequency and Reverse Cumulative Frequency tabs to see the Cumulative Frequency and Reverse Cumulative Frequency charts, respectively.

Analyze Data dialog shown with Cumulative Frequency chart and Percentiles displayed



Bin: [,]

Frequency:

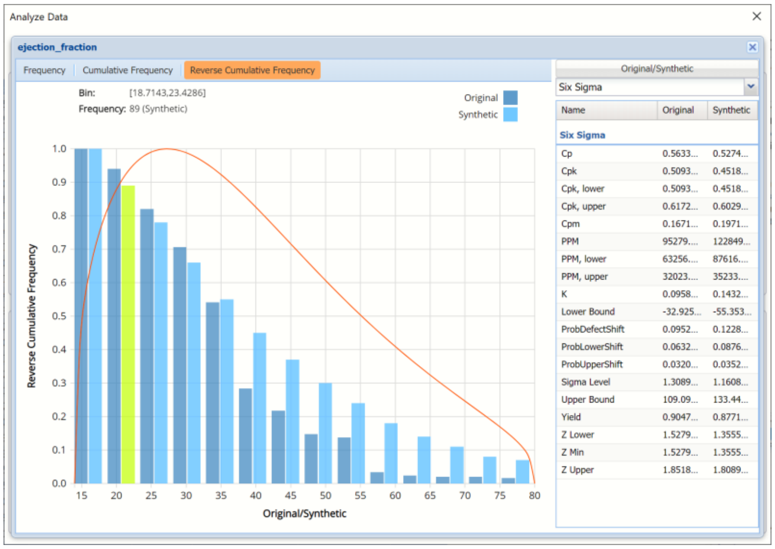
Original/Synthetic

Percentiles

Name	Original	Synthe...
Percentiles		
1%	16.96	15.478...
2%	20	16.395...
3%	20	16.640...
4%	20	17.323...
5%	20	17.718...
6%	20	18.0713
7%	20	18.877...
8%	25	19.463...
9%	25	19.614...
10%	25	20.324...
11%	25	20.438...
12%	25	20.969...
13%	25	21.551...
14%	25	22.090...
15%	25	22.475...
16%	25	22.653...
17%	25	23.280...
18%	25	23.625...
19%	25	23.856...
20%	30	24.3795
21%	30	24.791...
22%	30	25.065...

- Click the down arrow next to Statistics to view Percentiles for each type of data along with Six Sigma indices. Use the Chart Options view to manually select the number of bins to use in the chart, as well as to set personalization options.

Analyze Data dialog shown with Reverse Cumulative Frequency chart and Six Sigma indices displayed.



Bin: [18.7143,23.4286]
Frequency: 89 (Synthetic)

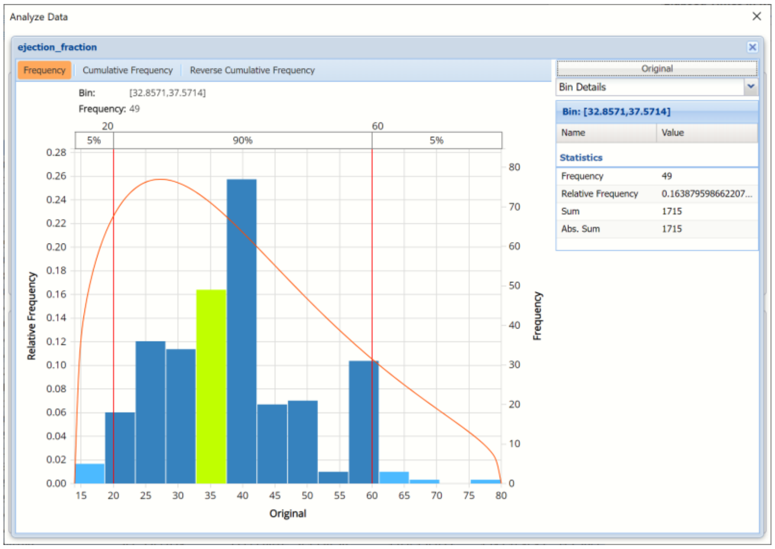
Six Sigma

Name	Original	Synthetic
Six Sigma		
Cp	0.5633...	0.5274...
Cpk	0.5093...	0.4518...
Cpk, lower	0.5093...	0.4518...
Cpk, upper	0.6172...	0.6029...
Cpm	0.1671...	0.1971...
PPM	95279....	122849...
PPM, lower	63256....	87616....
PPM, upper	32023....	35233....
K	0.0958...	0.1432...
Lower Bound	-32.925...	-55.353...
ProbDefectShift	0.0952...	0.1228...
ProbLowerShift	0.0632...	0.0876...
ProbUpperShift	0.0320...	0.0352...
Sigma Level	1.3089...	1.1608...
Upper Bound	109.09...	133.44...
Yield	0.9047...	0.8771...
Z Lower	1.5279...	1.3555...
Z Min	1.5279...	1.3555...
Z Upper	1.8518...	1.8089...

- Click the down arrow next to Statistics to view Bin Details for each bin in the chart.

  Bin: If viewing a chart with a single variable, only one grid will be displayed on the Bin Details pane. This grid displays important bin statistics such as frequency, relative frequency, sum and absolute sum.

  Bin Details View with continuous (scale) variables



Bin: [32.8571,37.5714]
Frequency: 49

Bin Details

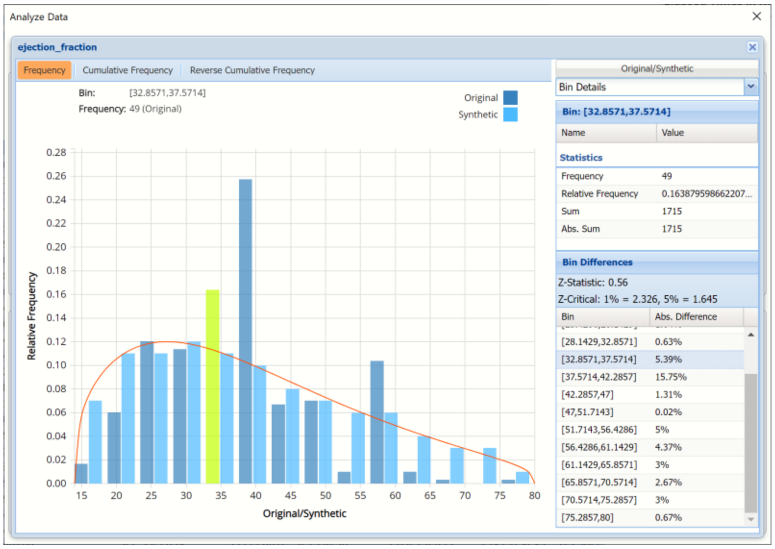
Bin: [32.8571,37.5714]

Name	Value
Statistics	
Frequency	49
Relative Frequency	0.163879598662207...
Sum	1715
Abs. Sum	1715

- Frequency is the number of observations assigned to the bin.
- Relative Frequency is the number of observations assigned to the bin divided by the total number of observations.
- Sum is the sum of all observations assigned to the bin.
- Absolute Sum is the sum of the absolute value of all observations assigned to the bin, i.e. |observation 1| + |observation 2| + |observation 3| + …

Bin Differences: If viewing a chart with two variables, two grids will be displayed, Bin and Bin Differences.

Bin Differences displays the differences between the relative frequencies of each bin for the two histograms, sorted in the same order as the bins listed in the chart. The computed Z-Statistic as well as the critical values, are displayed in the title of the grid.



Bin: [32.8571,37.5714]
Frequency: 49 (Original)

Bin: [32.8571,37.5714]

Name	Value
Statistics	
Frequency	49
Relative Frequency	0.163879598662207...
Sum	1715
Abs. Sum	1715

Bin Differences

Z-Statistic: 0.56
Z-Critical: 1% = 2.326, 5% = 1.645

Bin	Abs. Difference
[28.1429,32.8571]	0.63%
[32.8571,37.5714]	5.39%
[37.5714,42.2857]	15.75%
[42.2857,47]	1.31%
[47,51.7143]	0.02%
[51.7143,56.4286]	5%
[56.4286,61.1429]	4.37%
[61.1429,65.8571]	3%
[65.8571,70.5714]	2.67%
[70.5714,75.2857]	3%
[75.2857,80]	0.67%

ⓘ Use the Z-Statistic to quickly compare the Original and Synthetic data.

The Z Statistic, or Z score, can be used to compare both samples by using the following formula.

$$Z = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\sigma_{x1}^2 + \sigma_{x2}^2}}$$

where:

X_i is the mean of the sample, i = 1, 2

σ_{xi}^2 is the standard deviation of the sample, divided by the square root of the number of data points in the sample, i = 1,2.

A Z score gives the location of the score on a normal, or bell, curve. A score equal to 0 means that the values are in the center of the curve, or exactly average. A Z score of 1 means that the score is more than 1 standard deviation above the mean and a Z score of 2 means that the score is more than 2 standard deviations above the mean. Conversely, a score of -1 means that the score is 1 standard deviation *below* the mean and a score of -2 means that the score is 2 standard deviations *below* the mean.

In this example, the Z score for the two sets of data is 0.56. Since this score is less than both critical values (1% and 5%), the distributions of both datasets are statistically the same.

As discussed above, see the Analyze Data section of the Exploring Data chapter for an in-depth discussion of this chart as well as descriptions of all statistics, percentiles, bin details and six sigma indices.

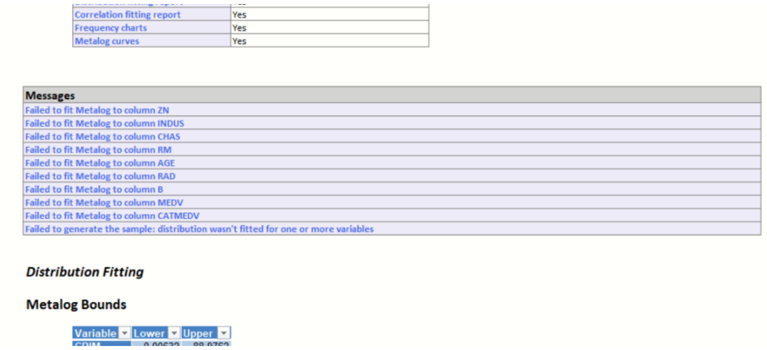
If something goes wrong…

Any errors or warnings (i.e. that a Metalog distribution was not able to be fitted for one or more vars, correlation fitting failed, a sample could not be generated etc.) produced during data generation will be reported on the SyntheticData_Output worksheet in the Messages section, beneath Inputs.

If the error is not fatal, Generate Data tries to complete as many tasks as possible. For example, if all Metalog distribution fittings attempted were infeasible for at least one variable (more likely if Fixed is selected for Metalog Terms), Analytic Solver Data Science will continue with distribution fitting for all other variables and it will attempt to fit a correlation matrix (if requested). However, Analytic Solver would not be able to generate synthetic data since one feasible Metalog distribution for each variable is required in order to generate a sample. Or, if at least one feasible Metalog distribution for each variable is fit successfully but Analytic Solver Data Science fails to fit a correlation or copula, sample data will still be generated (if requested) but the sample will be uncorrelated.

The screenshot below displays multiple "Failed to fit" messages and one "Failed to generate the sample" error.

Failure messages as produced and reported by Generate Data



Correlation fitting report	Yes
Frequency charts	Yes
Metalog curves	Yes

Messages
Failed to fit Metalog to column ZN
Failed to fit Metalog to column INDUS
Failed to fit Metalog to column CHAS
Failed to fit Metalog to column RM
Failed to fit Metalog to column AGE
Failed to fit Metalog to column RAD
Failed to fit Metalog to column B
Failed to fit Metalog to column MEDV
Failed to fit Metalog to column CATMEDV
Failed to generate the sample: distribution wasn't fitted for one or more variables

Distribution Fitting

Metalog Bounds

Variable	Lower	Upper

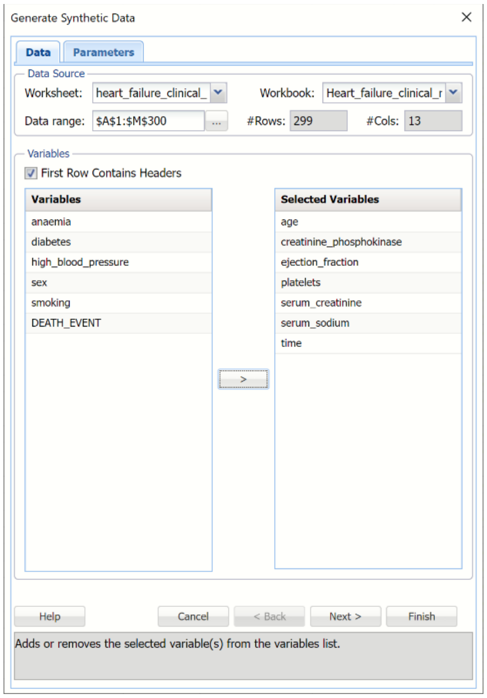
Generate Data Options

The following options appear on the Generate Data tabs, Data and Parameters.

Generate Synthetic Data dialog, Data tab

The following options appear on the Data tab of the Generate Synthetic Data dialog. These options pertain to the data source and the variables included.

Generate Synthetic Data dialog, Data tab



Worksheet: heart_failure_clinical_
Workbook: Heart_failure_clinical_r
Data range: $A$1:$M$300
#Rows: 299
#Cols: 13

☑ First Row Contains Headers

Variables	Selected Variables
anaemia	age
diabetes	creatinine_phosphokinase
high_blood_pressure	ejection_fraction
sex	platelets
smoking	serum_creatinine
DEATH_EVENT	serum_sodium
	time

Adds or removes the selected variable(s) from the variables list.

Variables

All variables in the data source data range are listed in this field. If the first row in the dataset contains headings, select *First Row Contains Headers*.

Selected Variables

Select a variable(s) in the *Variables* field, then click > to move the variable(s) to the *Selected Variables* field. Synthetic data will be generated for the variables appearing in this field.

Generate Synthetic Data dialog, Parameters tab

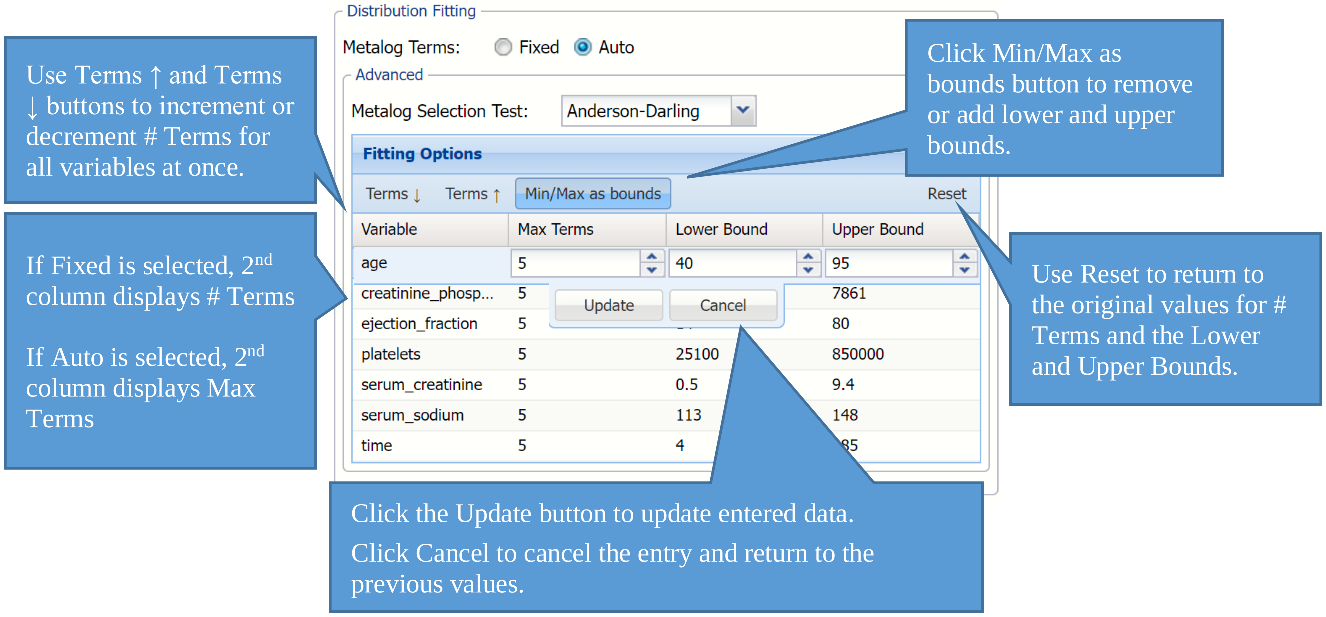
The following options appear on the *Parameters* tab of the *Generate Synthetic Data* dialog. These options pertain to the Distribution Terms, Correlation Fitting and available output.

Metalog Terms

- If Fixed is selected, Analytic Solver will attempt to fit and use the Metalog distribution with the specified number of terms entered into the # Terms column. (Only 1 distribution will be fit.) If Fixed is selected, Metalog Selection Test is disabled.
- If Auto is selected, Analytic Solver will attempt to fit *all possible* Metalog distributions, up to the entered value for Max Terms, and select and utilize the best Metalog distribution according to the goodness-of-fit test selected in the Metalog Selection Test menu.

Click the down arrow on the right of Fitting Options to enter either the maximum number of terms (if Auto is selected) or the exact number of terms (if Fixed is selected) for each variable as well as a lower and/or upper bound. By default the lower and upper bounds are set to the variable's minimum and maximum values, respectively. If no lower or upper bound is entered, Analytic Solver will fit a semi- (with one bound present) or unbounded (with no bounds present) Metalog function.

Distribution Fitting section of the Generate Data dialog



Use Terms ↑ and Terms ↓ buttons to increment or decrement # Terms for all variables at once.

If Fixed is selected, 2nd column displays # Terms

If Auto is selected, 2nd column displays Max Terms

Click Min/Max as bounds button to remove or add lower and upper bounds.

Use Reset to return to the original values for # Terms and the Lower and Upper Bounds.

Click the Update button to update entered data.

Click Cancel to cancel the entry and return to the previous values.

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

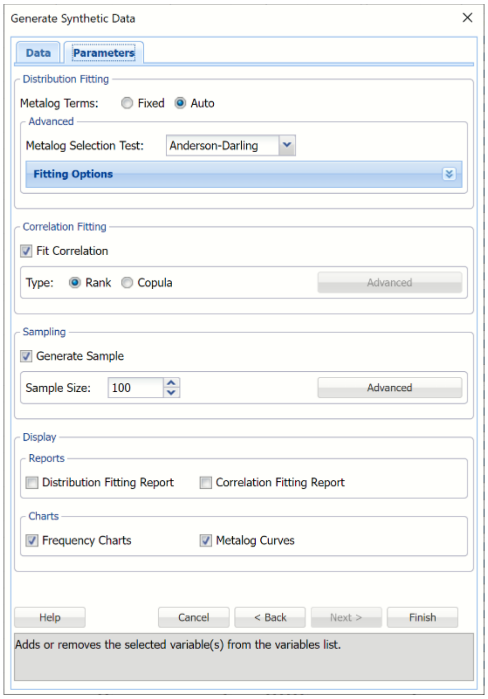
Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5	40	95
creatinine_phosp...	5		7861
ejection_fraction	5		80
platelets	5	25100	850000
serum_creatinine	5	0.5	9.4
serum_sodium	5	113	148
time	5	4	85

Generate Synthetic Data dialog, Parameters tab



Generate Synthetic Data

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

☑ Generate Sample

Sample Size: 100

Display

Reports

☐ Distribution Fitting Report ☐ Correlation Fitting Report

Charts

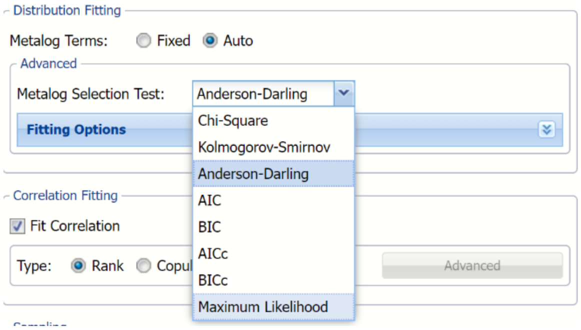
☑ Frequency Charts ☑ Metalog Curves

Adds or removes the selected variable(s) from the variables list.

Metalog Selection Test

Click the down arrow to select the desired Goodness-of-Fit test used by Analytic Solver. The Goodness of Fit test is used to select the best Metalog form for each data variable among the candidate distributions containing a different number of terms, from 2 to the value entered for Max Terms. The default Goodness-of-Fit test is Anderson-Darling.

Metalog Selection Test menu



Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copul

Goodness of Fit Tests:

- Chi Square – Uses the chi-square statistic and distribution to rank the distributions. Sample data is first divided into intervals using either equal probability, then the number of points that fall into each interval are compared with the expected number of points in each interval. The null hypotheses is rejected using a 90% significance level, if the chi-squared test statistic is greater than the critical value statistic.

  Note: The Chi Square test is used indirectly in continuous fitting as a support in the AIC test. The AIC test must succeed in fitting as this is a necessary condition as well as the fitting of at least one of the tests, Chi Squared, Kolmogorov-Smirnoff, or Anderson-Darling.

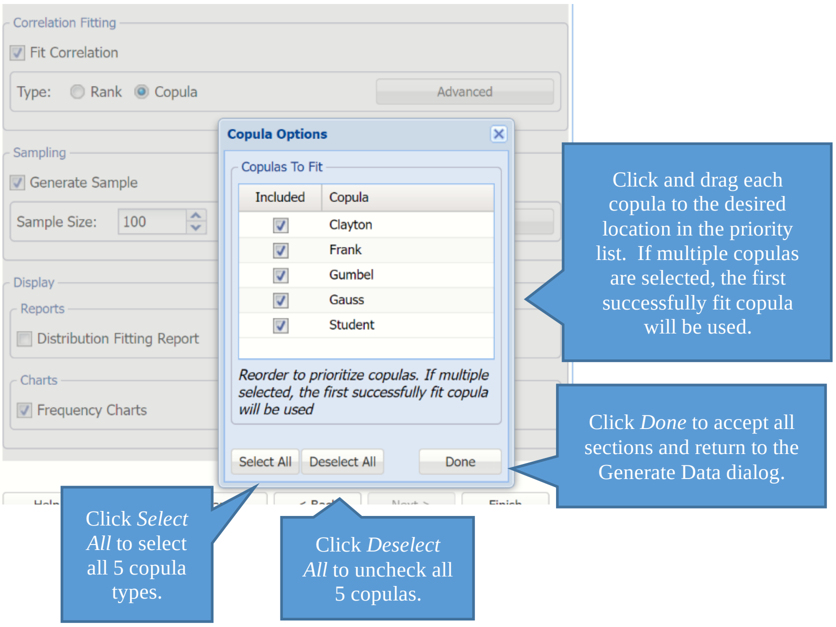
- Kolmogorov-Smirnoff –This test computes the difference (D) between the continuous distribution function (CDF) and the empirical cumulative distribution function (ECDF). The null hypothesis is rejected if, at the 90% significance level, D is larger than the critical value statistic.
- Anderson (Default) -Darling –Ranks the fitted distribution using the Anderson Darling statistic, A2 . The null hypothesis is rejected using a 90% significance level, if A2 is larger than the critical value statistic. This test awards more weight to the distribution tails then the Kolmogorov-Smirnoff test.
- AIC – The AIC test is a Chi Squared test corrected for the number of distribution parameters and sample size. AIC = 2 * p – 2 + ln(L) where p is the number of distribution parameters, n is the fitted sample size (number of data points) and ln(L) is the log-likelihood function computed on the fitted data.
- AICc –When the sample size is small, there is a significant chance that the AIC test will select a model with a large number of parameters. In other words, AIC will overfit the data. AICc was developed to reduce the possibility of overfitting by applying a penalty to the number of parameters. Assuming that the model is univariate, is linear in the parameters and has normally-distributed residuals, the formula for AICc is: AICc = AIC + 2 * p *(p + 1) / (n – p − 1) where n = sample size, p = # of parameters. As the sample size approaches infinity, the penalty on the number of parameters converges to 0 resulting in AICc converging to AIC.
- BIC – The Bayesian information criterion (BIC) is defined as: BIC = ln(n) * p -2 * ln(L) where p is the number of distribution parameters, n is the fitted sample size (number of data points) and ln(L) is the log-likelihood function computed on the fitted data.
- BICc – The BICc is the alternative version of BIC, corrected for the sample size BICc = BIC + 2 * p * (p + 1) / (n – p - 1).
- Maximum Likelihood – The (negated) raw value of the estimated maximum log likelihood utilized in tests described above.

Fit Correlation

Select **Fit Correlation** to fit a correlation between the variables. If this option is left unchecked, correlation fitting will not be performed.

- If *Rank* is selected Analytic Solver will use the *Spearman rank order correlation coefficient* to compute a correlation matrix that includes all included variables.
- Selecting *Copula* opens the Copula Options dialog where you can select and drag five types of copulas into a desired order of priority.

  Correlation Fitting section of the Generate Data dialog



Correlation Fitting

☑ Fit Correlation

Type: ○ Rank ● Copula

Sampling

☑ Generate Sample

Sample Size: 100

Display

Reports

☐ Distribution Fitting Report

Charts

☑ Frequency Charts

Copula Options

Copulas To Fit

Included	Copula
☑	Clayton
☑	Frank
☑	Gumbel
☑	Gauss
☑	Student

Reorder to prioritize copulas. If multiple selected, the first successfully fit copula will be used

Click and drag each copula to the desired location in the priority list. If multiple copulas are selected, the first successfully fit copula will be used.

Click *Done* to accept all sections and return to the Generate Data dialog.

Click *Select All* to select all 5 copula types.

Click *Deselect All* to uncheck all 5 copulas.

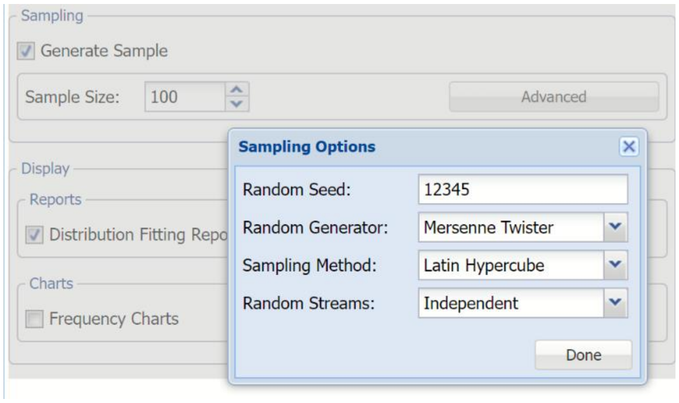
Generate Sample

Select **Generate Sample** to generate synthetic data for each selected variable. Use the Sample Size field to increase the size of the sample generated.

If this option is left unchecked, variable data will be fitted to a Metalog distribution and also correlations, if Fit Correlation is selected, but no synthetic data will be generated.

Click Advanced to open the Sampling Options dialog.

Sampling Options dialog



Sampling

☑ Generate Sample

Sample Size: 100

Display

Reports

☑ Distribution Fitting Repo

Charts

☐ Frequency Charts

Sampling Options

Random Seed: 12345

Random Generator: Mersenne Twister

Sampling Method: Latin Hypercube

Random Streams: Independent

From this dialog, users can set the Random Seed, Random Generator, Sampling Method and Random Streams.

Random Seed

Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the *same* sequence of random numbers is used for each simulation. When the seed is zero or the field is left empty, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each simulation. If you need the results from one simulation to another to be strictly comparable, you should set the seed. To do this, simply type the desired number into the box. (Default Value = 12345)

Random Generator

Use this menu to select a random number generation algorithm. Analytic Solver Data Science includes an advanced set of random number generation capabilities.

Computer-generated numbers are never truly "random," since they are always computed by an algorithm – they are called *pseudorandom* numbers. A random number generator is designed to quickly generate sequences of numbers that are as close to statistically independent as possible. Eventually, an algorithm will generate the same number seen sometime earlier in the sequence, and at this point the sequence will begin to repeat. The *period* of the random number generator is the number of values it can generate before repeating.

A long period is desirable, but there is a tradeoff between the length of the period and the degree of statistical independence achieved within the period. Hence, Analytic Solver Data Science offers a choice of four random number generators:

- *Park-Miller* "Minimal" Generator with Bayes-Durham shuffle and safeguards. This generator has a period of 2^{31}-2. Its properties are good, but the following choices are usually better.
- Combined Multiple Recursive Generator of L'Ecuyer (*L'Ecuyer-CMRG*). This generator has a period of 2^{191}, and excellent statistical independence of samples within the period.
- Well Equidistributed Long-period Linear (*WELL*) generator of Panneton, L'Ecuyer and Matsumoto. This generator combines a long period of 2^{1024} with very good statistical independence.
- *Mersenne Twister* (default setting) generator of Matsumoto and Nishimura. This generator has the longest period of 2^{19937}-1, but the samples are not as "equidistributed" as for the WELL and L-Ecuyer-CMRG generators.
- *HDR* Random Number Generator, designed by Doug Hubbard. Permits data generation running on various computer platforms to generate identical or independent streams of random numbers.

Sampling Method

Use this option group to select *Monte Carlo, Latin Hypercube*, or *Sobol RQMC* sampling.

- *Monte Carlo:* In standard Monte Carlo sampling, numbers generated by the chosen random number generator are used directly to obtain sample values. With this method, the variance or estimation error in computed samples is inversely proportional to the square root of the number of trials (controlled by the Sample Size); hence to cut the error in half, four times as many trials are required.

Analytic Solver Data Science provides two other sampling methods than can significantly improve the 'coverage' of the sample space, and thus reduce the variance in computed samples. This means that you can achieve a given level of accuracy (low variance or error) with fewer trials.

- *Latin Hypercube (default):* Latin Hypercube sampling begins with a stratified sample in each dimension (one for each selected variable), which constrains the random numbers drawn to lie in a set of subintervals from 0 to 1. Then these one-dimensional samples are combined and randomly permuted so that they 'cover' a unit hypercube in a stratified manner.
- *Sobol RQMC (Randomized QMC)***.** Sobol numbers are an example of so-called "Quasi Monte Carlo" or "low-discrepancy numbers," which are generated with a goal of coverage of the sample space rather than "randomness" and statistical independence. Analytic Solver Data Science adds a "random shift" to Sobol numbers, which improves their statistical independence.

Random Streams

Use this option group to select a *Single Stream* for each variable or an *Independent Stream (the default)* for each variable.

If *Single Stream* is selected, a single sequence of random numbers is generated. Values are taken consecutively from this sequence to obtain samples for each selected variable. This introduces a subtle dependence between the samples for all distributions in one trial. In many applications, the effect is too small to make a difference – but in some cases, better results are obtained if *independent* random number sequences (streams) are used for each distribution in the model. Analytic Solver Data Science offers this capability for Monte Carlo sampling and Latin Hypercube sampling; it does not apply to Sobol numbers.

Reports and Charts

Distribution Fitting: Report included on the SyntheticData_Output worksheet includes the number of terms, the coefficients for each term, the lower and upper bounds and the goodnesss of fit statistics used when fitting each Metalog distribution.

Correlation Fitting Report: Displays the correlation matrix on the SyntheticData_Output worksheet

Frequency Charts: Displays the multivariate chart produced by the Analyze Data feature. Double click each chart to view an interactive chart and detailed data (statistics, percentiles and six sigma indices) about each variable included in the analysis. For more information on the Analyze Data feature included in the latest version of Analytic Solver Data Science, see the Exploring Data chapter that appears later in this guide.

Metalog Curves: Select this Chart option to add Metalog distribution curves to each variable displayed in the multivariate chart and interactive charts described above for Frequency Charts.

Exploring Data

Introduction

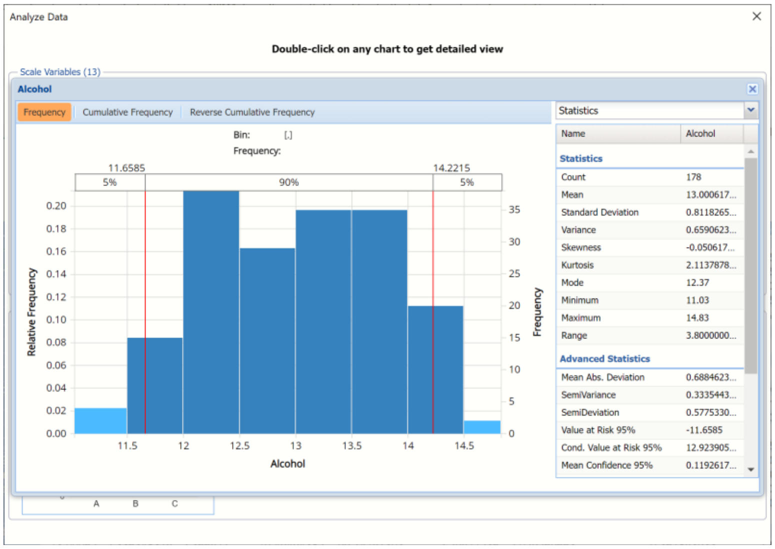
The Explore menu gives you access to Analytic Solver's new Data Analysis tool, Dimensionality Reduction via Feature Selection and the ability to explore your data using charts such as Bar charts, Line Charts, Scatterplots, Boxplots, Histograms, Parallel Coordinates, Scatter Plot Matrices and Variable Plots.

Analyze Data

In the latest release of Analytic Solver, users now have access to the Analyze Data feature located on the Explore menu. With the Analyze Data application, users can generate a multivariate chart for any number of scale (continuous) or categorical variables. This feature can be used as a standalone application and can be particularly useful as a step in understanding your data while in the process of building your data science model. This feature allows you to look at your data not as static historical data but as a realization of an uncertain variable such as what you would encounter in simulation modeling.

Double-clicking the preview chart will display a detailed view of a histogram for scale (continuous) variables and bar charts for categorical variables, allowing users to view their data from a frequency perspective, such as a historical sample from a possible probability distribution. Various statistics pertaining to each variable's range of values are displayed on the right, such as count, mean, standard deviation, etc.

Detailed view for the Alcohol scale (continuous) variable



Name	Alcohol
Statistics	
Count	178
Mean	13.000617...
Standard Deviation	0.8118265...
Variance	0.6590623...
Skewness	-0.050617...
Kurtosis	2.1137878...
Mode	12.37
Minimum	11.03
Maximum	14.83
Range	3.8000000...
Advanced Statistics	
Mean Abs. Deviation	0.6884623...
SemiVariance	0.3335443...
SemiDeviation	0.5775330...
Value at Risk 95%	-11.6585
Cond. Value at Risk 95%	12.923905...
Mean Confidence 95%	0.1192617...

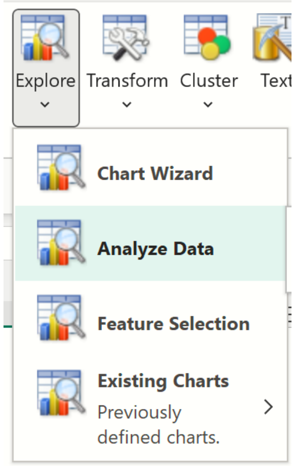
Analyze Data Example

This example utilizes the Wine dataset to display the various features of the Analyze Data application embedded in Analytic Solver.

1. Open the Wine dataset by clicking Help – Example Models – Forecasting/Data Science Examples -- Wine (at the bottom of the list). This example dataset contains information pertaining to wine from three different wineries located in the same region. Thirteen variables describe various characteristics among three classes of wine: A, B, and C.

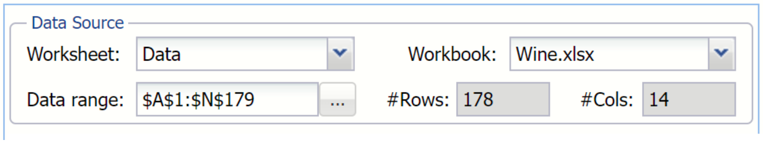
2. Confirm that the Data tab is selected, and click Data Analysis – Explore – Analyze Data to bring up the Analyze Data dialog

   Explore menu



3. The top of the dialog displays the information for the Data Source: worksheet name, *Data*, the workbook name, *Wine.xlsx*, the data range, *A1:N179* and the number of rows and columns in the dataset.

   Data Source section on the Analyze Data dialog



Data Source

Worksheet: Data

Workbook: Wine.xlsx

Data range: $A$1:$N$179

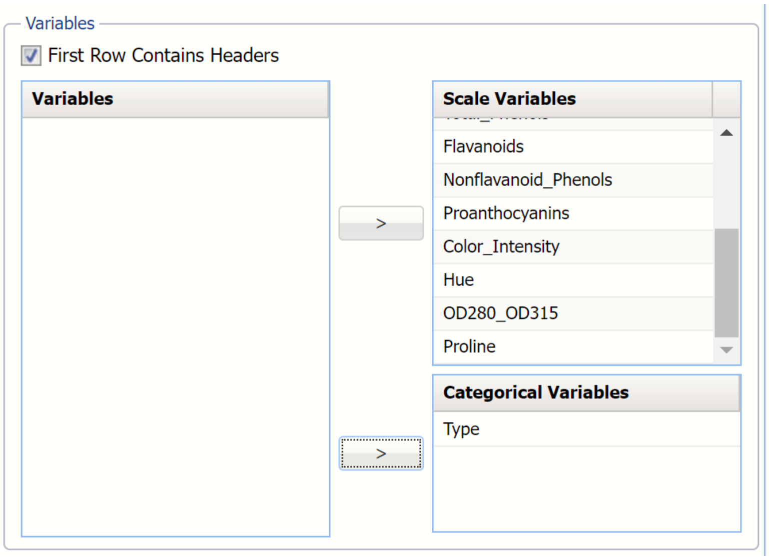
#Rows: 178

#Cols: 14

4. Select the variables to be included in the analysis.

   A. Select Type as a *Category Variable*.

B. Select the remaining variables, except Partition Variable, as *Scale Variables.* Variables section on the Analyze Data dialog



Variables

☑ First Row Contains Headers

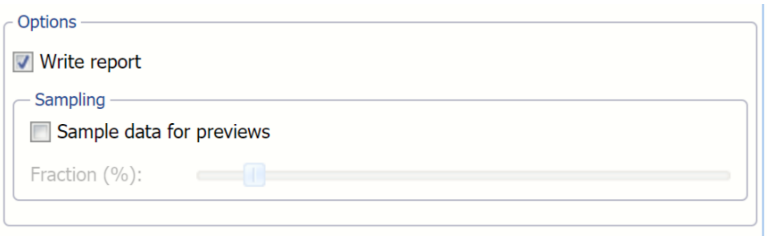
Variables

Scale Variables
Flavanoids
Nonflavanoid_Phenols
Proanthocyanins
Color_Intensity
Hue
OD280_OD315
Proline

Categorical Variables
Type

5. Click *Write report* to write all computed statistics for each variable to the Statistics worksheet. (The Statistics worksheet will be inserted to the right of the Data tab.) If this checkbox is left unchecked, no report will be inserted into the workbook. The preview dialog will be displayed, and detailed charts will be available if double-clicked. Once the dialog is closed, it will not persist in the workbook. (The application would have to be re-run to re-open the chart.)

   Options section on the Analyze Data dialog



Options

☑ Write report

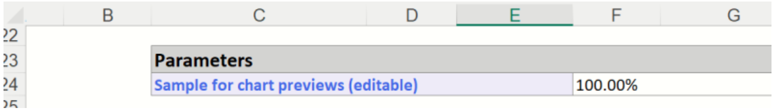
Sampling

☐ Sample data for previews

Fraction (%):

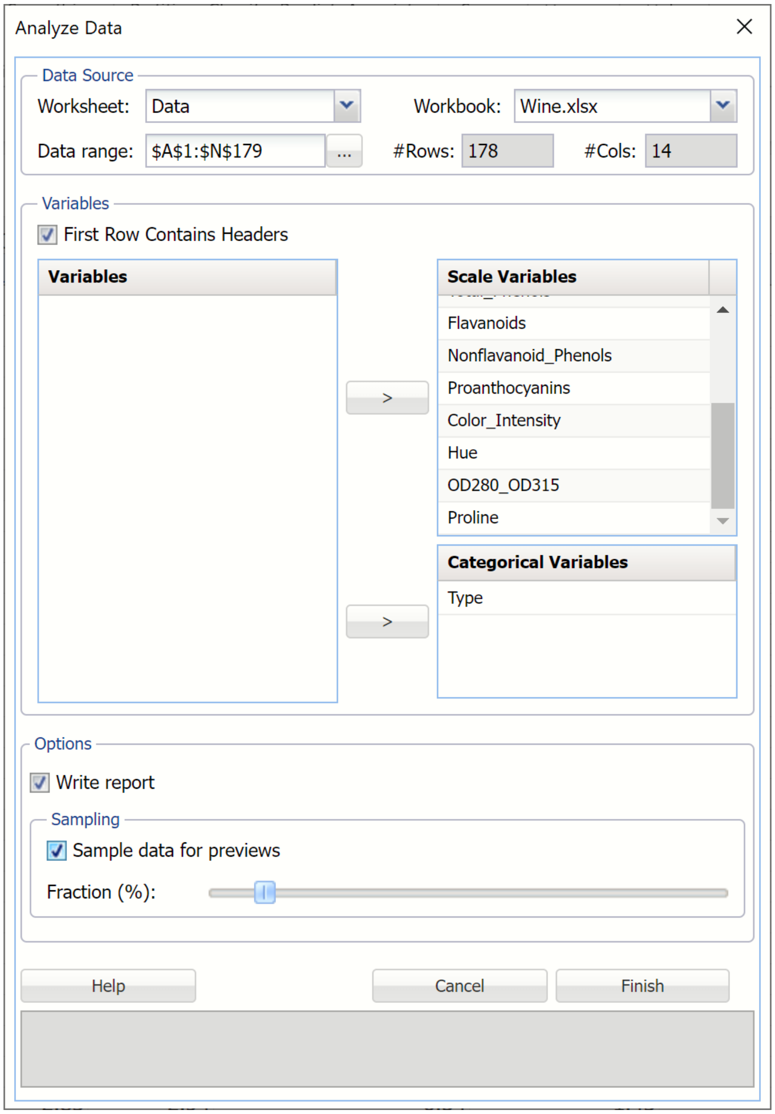
6. If the worksheet contains an extremely large dataset, users can select *Sample data for previews* and use the *Fraction %* slider to limit the amount of data utilized when creating the chart **previews**. This option has no bearing on the amount of data included in the detailed charts. Detailed charts always use the full amount of data to produce the interactive chart and statistics. The Fraction % may be changed after the report is created by editing the Sample for chart previews cell. For more information see below.

Parameters section as shown on the Statistics Report.



Parameters	
Sample for chart previews (editable)	100.00%

Analyze Data dialog as shown with all variables included in the analysis.



Analyze Data

Data Source

Worksheet: Data
Workbook: Wine.xlsx
Data range: $A$1:$N$179
#Rows: 178
#Cols: 14

Variables

☑ First Row Contains Headers

Variables

Scale Variables: Flavanoids, Nonflavanoid_Phenols, Proanthocyanins, Color_Intensity, Hue, OD280_OD315, Proline

Categorical Variables: Type

Options

☑ Write report

Sampling

☑ Sample data for previews

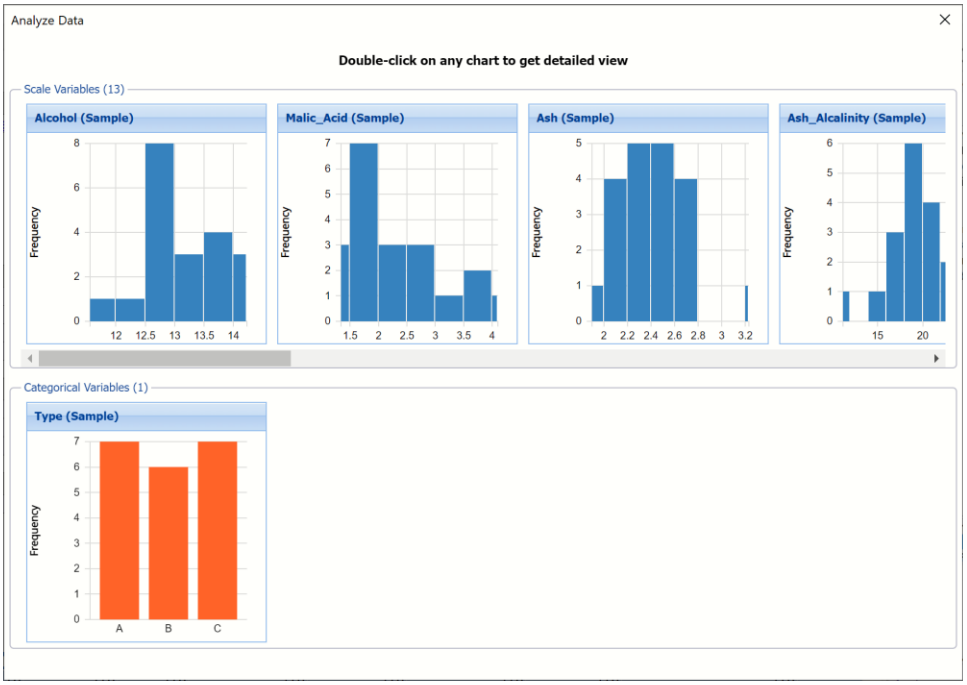
Fraction (%):

7. Click Finish.

Results

After Finish is clicked in the Analyze Data dialog. A new Statistics worksheet is inserted to the left of the Data tab and an Analyze Data Results dialog appears displaying a bar chart (for categorical variables) or histogram (for continuous or scale variables) for each variable included in the analysis.

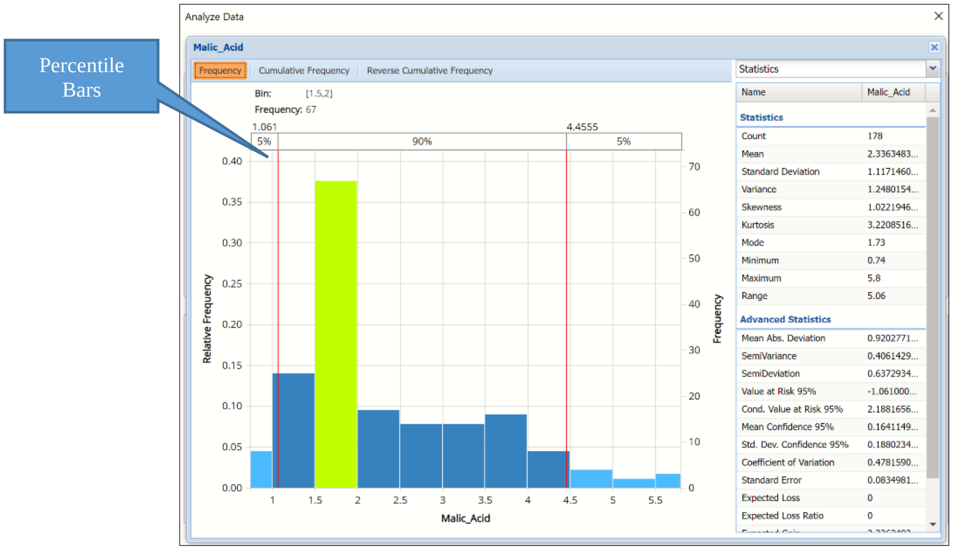
Analyze Data: Multivariate chart dialog



Double - click any chart to display a more detailed view of the chart and various computed statistics, including Six Sigma, and percentiles.

Display Placement: Click the title bar of the multivariate dialog to drag to a new location.

Malic_Acid chart view



Name	Malic_Acid
Statistics	
Count	178
Mean	2.3363483...
Standard Deviation	1.1171460...
Variance	1.2480154...
Skewness	1.0221946...
Kurtosis	3.2208516...
Mode	1.73
Minimum	0.74
Maximum	5.8
Range	5.06
Advanced Statistics	
Mean Abs. Deviation	0.9202771...
SemiVariance	0.4061429...
SemiDeviation	0.6372934...
Value at Risk 95%	-1.061000...
Cond. Value at Risk 95%	2.1881656...
Mean Confidence 95%	0.1641149...
Std. Dev. Confidence 95%	0.1880234...
Coefficient of Variation	0.4781590...
Standard Error	0.0834981...
Expected Loss	0
Expected Loss Ratio	0

Tabs: The Analyze Data dialog contains three tabs: Frequency, Cumulative Frequency, and Reverse Cumulative Frequency. Each tab displays different information about the distribution of variable values.

Hovering over a bar in either of the three charts will populate the Bin and Frequency headings at the top of the chart. In the Frequency chart above, the bar for the [1.5, 2] Bin is selected. This bar has a frequency of 67 and a relative frequency of about 38%.

By default, red vertical lines will appear at the 5% and 95% percentile values in all three charts, effectively displaying the 90th confidence interval. The middle percentage is the percentage of all the variable values that lie within the 'included' area, i.e. the darker shaded area. The two percentages on each end are the percentage of all variable values that lie outside of the 'included' area or the "tails". i.e. the lighter shaded area. Percentile values can be altered by moving either red vertical line to the left or right.

Click the "X" in the upper right corner of the detailed chart dialog to return to the Preview dialog. Click the "X" in the upper right corner of the preview chart dialog to close the dialog. To re-open the preview dialog, click a new tab, say the Data tab in this example, and then click the Statistics tab. The preview dialog will be displayed.

Frequency Tab: When the Analyze Data dialog is first displayed, the Frequency tab is selected by default.

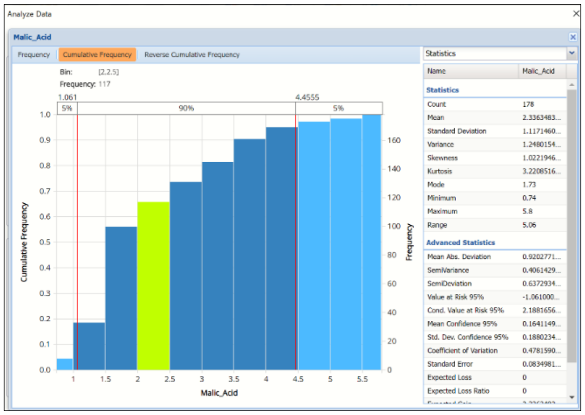
- For **continuous** variables, this tab displays a histogram of the variable's values.
- For **categorical** variables, this tab displays a bar chart.

Bins containing the range of values for the variable appear on the horizontal axis, the relative frequency of occurrence of the bin values appears on the left vertical axis while the actual frequency of the bin values appear on the right vertical axis.

Cumulative Frequency / Reverse Cumulative Frequency

The Cumulative Frequency tab displays a chart of the cumulative form of the frequency chart, as shown below. Hover over each bar to populate the Bin and Frequency headings at the top of the chart. In this screenshot below, the bar for the [2.0, 2.5] Bin is selected in the Cumulative Frequency Chart. This bar has a frequency of 117 and a relative frequency of about 68%.

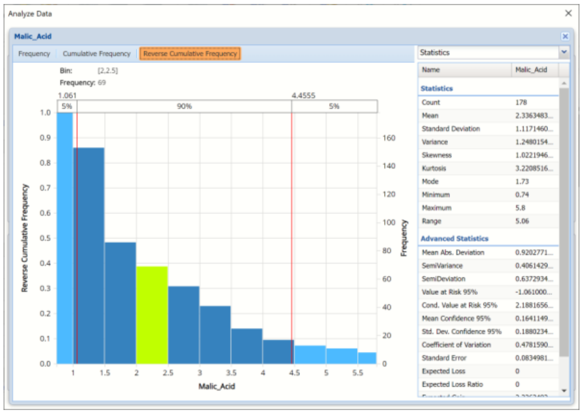
Cumulative Frequency Chart



Bin: [2.2.5]
Frequency: 117

Name	Malic_Acid
Statistics	
Count	178
Mean	2.3363483...
Standard Deviation	1.1171460...
Variance	1.2480154...
Skewness	1.0221946...
Kurtosis	3.2208516...
Mode	1.73
Minimum	0.74
Maximum	5.8
Range	5.06
Advanced Statistics	
Mean Abs. Deviation	0.9202771...
SemiVariance	0.4061429...
SemiDeviation	0.6372934...
Value at Risk 95%	-1.061000...
Cond. Value at Risk 95%	2.1881656...
Mean Confidence 95%	0.1641149...
Std. Dev. Confidence 95%	0.1880234...
Coefficient of Variation	0.4781590...
Standard Error	0.0834981...
Expected Loss	0
Expected Loss Ratio	0

Reverse Cumulative Frequency Chart



Bin: [2.2.5]
Frequency: 69

Name	Malic_Acid
Statistics	
Count	178
Mean	2.3363483...
Standard Deviation	1.1171460...
Variance	1.2480154...
Skewness	1.0221946...
Kurtosis	3.2208516...
Mode	1.73
Minimum	0.74
Maximum	5.8
Range	5.06
Advanced Statistics	
Mean Abs. Deviation	0.9202771...
SemiVariance	0.4061429...
SemiDeviation	0.6372934...
Value at Risk 95%	-1.061000...
Cond. Value at Risk 95%	2.1881656...
Mean Confidence 95%	0.1641149...
Std. Dev. Confidence 95%	0.1880234...
Coefficient of Variation	0.4781590...
Standard Error	0.0834981...
Expected Loss	0
Expected Loss Ratio	0

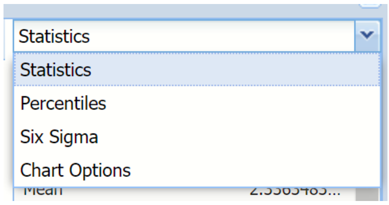
Cumulative Frequency Chart: Bins containing the range of values for the variable appear on the horizontal axis, the cumulative frequency of occurrence of the bin values appear on the left vertical axis while the actual cumulative frequency of the bin values appear on the right vertical axis.

Reverse Cumulative Frequency Chart: Bins containing the range of values for the variable appear on the horizontal axis, similar to the Cumulative Frequency

Chart. The reverse cumulative frequency of occurrence of the bin values appear on the left vertical axis while the actual reverse cumulative frequency of the bin values appear on the right vertical axis.

Click the drop down menu on the upper right of the dialog to display additional panes: Statistics, Six Sigma and Percentiles.

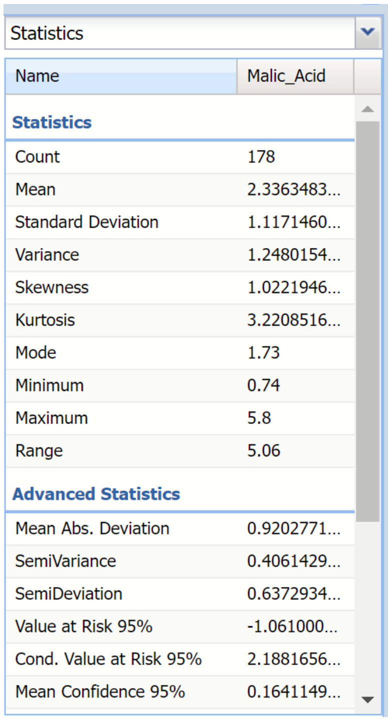
Drop down menu



Statistics View

The Statistics tab displays numeric values for several summary statistics, computed from all values for the specified variable. The statistics shown on the pane below were computed for the Malic Acid variable.

Statistics Pane



Statistics

Name	Malic_Acid
Statistics	
Count	178
Mean	2.3363483...
Standard Deviation	1.1171460...
Variance	1.2480154...
Skewness	1.0221946...
Kurtosis	3.2208516...
Mode	1.73
Minimum	0.74
Maximum	5.8
Range	5.06
Advanced Statistics	
Mean Abs. Deviation	0.9202771...
SemiVariance	0.4061429...
SemiDeviation	0.6372934...
Value at Risk 95%	-1.061000...
Cond. Value at Risk 95%	2.1881656...
Mean Confidence 95%	0.1641149...

All statistics appearing on the Statistics pane are briefly described below.

Statistics

- **Mean**, the average of all the values.
- **Standard Deviation**, the square root of variance.
- **Variance,** describes the spread of the distribution of values.
- **Skewness**, which describes the *asymmetry* of the distribution of values.

- **Kurtosis**, which describes the *peakedness* of the distribution of values.
- **Mode**, the most frequently occurring single value.
- **Minimum**, the minimum value attained.
- **Maximum**, the maximum value attained.
- **Range**, the difference between the maximum and minimum values.

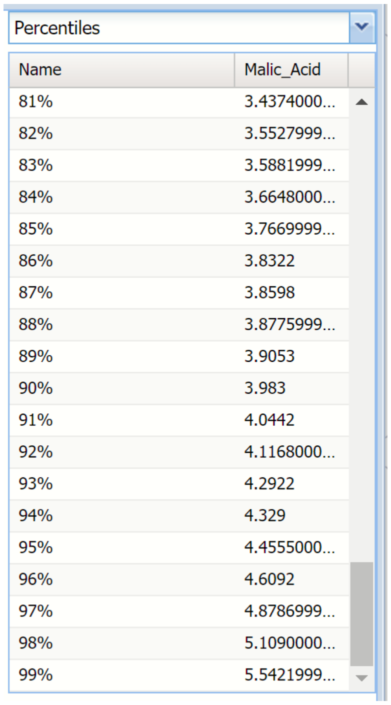
Advanced Statistics

- **Mean Abs. Deviation**, returns the average of the absolute deviations.
- **SemiVariance**, measure of the dispersion of values.
- **SemiDeviation**, *one-sided* measure of dispersion of values.
- **Value at Risk 95%**, the maximum loss that can occur at a given confidence level.
- **Cond. Value at Risk**, is defined as the *expected value* of a loss *given that* a loss at the specified percentile occurs.
- **Mean Confidence**, returns the confidence "half-interval" for the estimated mean value (returned by the PsiMean() function.
- **Std. Dev. Confidence 95%**, returns the confidence 'half-interval' for the estimated standard deviation of the simulation trials (returned by the PsiStdDev() function).
- **Coefficient of Variation**, is defined as the ratio of the standard deviation to the mean.
- **Standard Error**, defined as the standard deviation of the sample mean.
- **Expected Loss**, returns the average of all negative data multiplied by the percentrank of 0 among all data.
- **Expected Loss Ratio**, returns the expected loss ratio.
- **Expected Gain** returns the average of all positive data multiplied by 1 - percentrank of 0 among all data.
- **Expected Gain Ratio**, returns the expected gain ratio**.**
- **Expected Value Margin**, returns the expected value margin.

Percentiles View

Selecting Percentiles from the menu displays numeric percentile values (from 1% to 99%) computed using all values for the variable. The percentiles shown below were computed using the values for the Malic_Acid variable.

Percentiles Pane



Percentiles

Name	Malic_Acid
81%	3.4374000...
82%	3.5527999...
83%	3.5881999...
84%	3.6648000...
85%	3.7669999...
86%	3.8322
87%	3.8598
88%	3.8775999...
89%	3.9053
90%	3.983
91%	4.0442
92%	4.1168000...
93%	4.2922
94%	4.329
95%	4.4555000...
96%	4.6092
97%	4.8786999...
98%	5.1090000...
99%	5.5421999...

The values displayed here represent 99 equally spaced points on the Cumulative Frequency chart: In the Percentile column, the numbers rise smoothly on the vertical axis, from 0 to 1.0, and in the Value column, the corresponding values from the horizontal axis are shown. For example, the 75th Percentile value is a number such that three-quarters of the values occurring in the last simulation are less than or equal to this value.

Six Sigma View

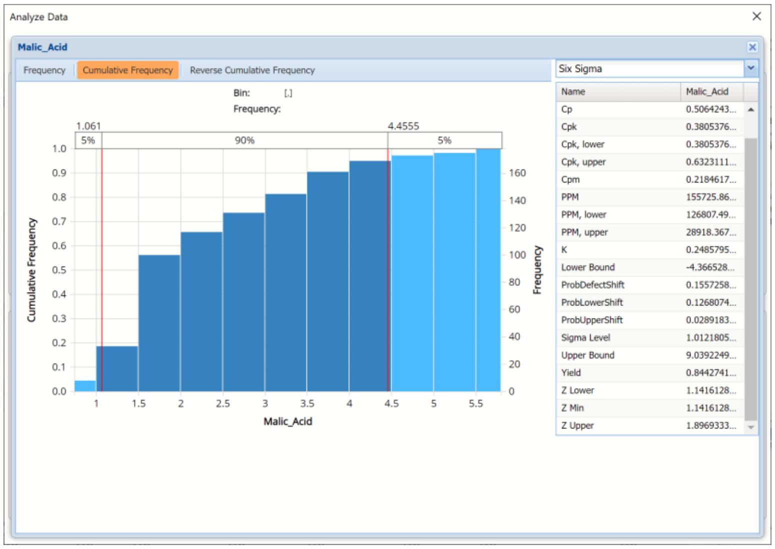
Selecting Six Sigma from the menu displays various computed Six Sigma measures. In this display, the red vertical lines on the chart are the Lower Specification Limit (LSL) and the Upper Specification Limit (USL) which are initially set equal to the 5th and 95th percentile values, respectively.

These functions compute values related to the Six Sigma indices used in manufacturing and process control. For more information on these functions, see the Appendix located at the end of this guide.

- **SigmaCP** calculates the Process Capability.
- **SigmaCPK** calculates the Process Capability Index.
- **SigmaCPKLower** calculates the one-sided Process Capability Index based on the Lower Specification Limit.
- **SigmaCPKUpper** calculates the one-sided Process Capability Index based on the Upper Specification Limit.
- **SigmaCPM** calculates the Taguchi Capability Index.
- **SigmaDefectPPM** calculates the Defect Parts per Million statistic.
- **SigmaDefectShiftPPM** calculates the Defective Parts per Million statistic with a Shift.

- **SigmaDefectShiftPPMLower** calculates the Defective Parts per Million statistic with a Shift below the Lower Specification Limit.
- **SigmaDefectShiftPPMUpper** calculates the Defective Parts per Million statistic with a Shift above the Upper Specification Limit.
- **SigmaK** calculates the Measure of Process Center.
- **SigmaLowerBound** calculates the Lower Bound as a specific number of standard deviations below the mean.
- **SigmaProbDefectShift** calculates the Probability of Defect with a Shift outside the limits.
- **SigmaProbDefectShiftLower** calculates the Probability of Defect with a Shift below the lower limit.
- **SigmaProbDefectShiftUpper** calculates the Probability of Defect with a Shift above the upper limit.
- **SigmaSigmaLevel** calculates the Process Sigma Level with a Shift.
- **SigmaUpperBound** calculates the Upper Bound as a specific number of standard deviations above the mean.
- **SigmaYield** calculates the Six Sigma Yield with a shift, i.e. the fraction of the process that is free of defects.
- **SigmaZLower** calculates the number of standard deviations of the process that the lower limit is below the mean of the process.
- **SigmaZMin** calculates the minimum of ZLower and ZUpper.
- **SigmaZUpper** calculates the number of standard deviations of the process that the upper limit is above the mean of the process.

Six Sigma Pane

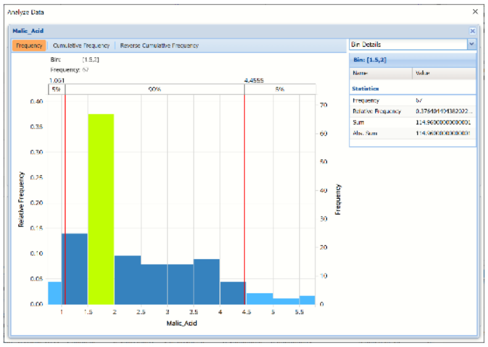


Name	Malic_Acid
Cp	0.5064243...
Cpk	0.3805376...
Cpk, lower	0.3805376...
Cpk, upper	0.6323111...
Cpm	0.2184617...
PPM	155725.86...
PPM, lower	126807.49...
PPM, upper	28918.367...
K	0.2485795...
Lower Bound	-4.366528...
ProbDefectShift	0.1557258...
ProbLowerShift	0.1268074...
ProbUpperShift	0.0289183...
Sigma Level	1.0121805...
Upper Bound	9.0392249...
Yield	0.8442741...
Z Lower	1.1416128...
Z Min	1.1416128...
Z Upper	1.8969333...

Bin Details View

Click the down arrow next to Statistics to view Bin Details for each bin in the chart.

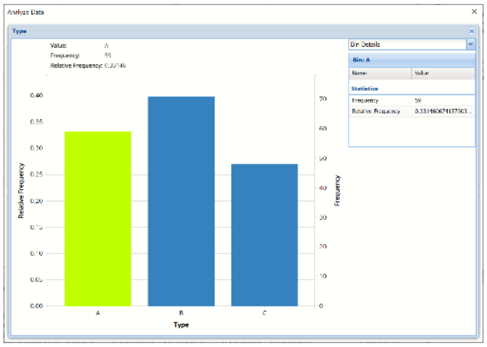
Frequency chart for scale (continuous) variable



Bin: [1.5,2]

Name	Value
Statistics	
Frequency	67
Relative Frequency	0.376404494382022...
Sum	114.960000000001
Abs. Sum	114.960000000001

Frequency chart for categorical variable



Bin: A

Name	Value
Statistics	
Frequency	59
Relative Frequency	0.331460674157303...

- Frequency is the number of observations assigned to the bin. (Scale and categorical variables)
- Relative Frequency is the number of observations assigned to the bin divided by the total number of observations. (Scale and categorical variables)
- Sum is the sum of all observations assigned to the bin. (Scale variables)
- Absolute Sum is the sum of the absolute value of all observations assigned to the bin, i.e. |observation 1| + |observation 2| + |observation 3| + … (Scale variables)

Chart Settings View

The Chart Options view contains controls that allow you to customize the appearance of the charts that appear in the dialog. When you change option selections or type numerical values in these controls, the chart area is instantly updated.

Chart Options Pane

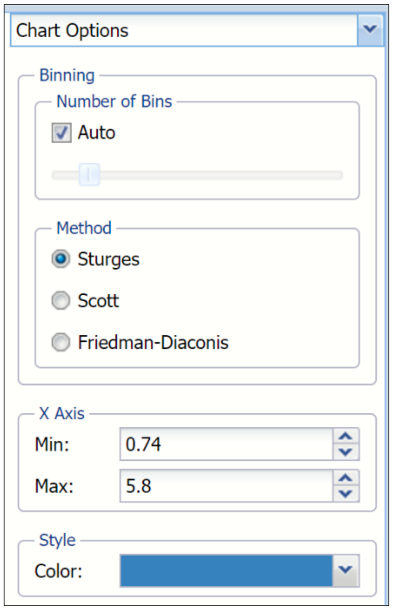


Chart Options

Binning

Number of Bins

☑ Auto

Method

● Sturges

○ Scott

○ Friedman-Diaconis

X Axis

Min: 0.74

Max: 5.8

Style

Color:

The controls are divided into three groups: Binning, Method and Style.

- **Binning**: Applies to the number of bins in the chart.

- ○ **Auto:** Select Auto to allow Analytic Solver to automatically select the appropriate number of bins to be included in the frequency charts. See Method below for information on how to change the bin generator used by Analytic Solver when this option is selected.
- ○ **Manually select # of Bins:** To manually select the number of bins used in the frequency charts, uncheck "Auto" and drag the slider to the right to increase the number of bins or to the left to decrease the number of bins.

- **Method:** Three generators are included in the Analyze Data application to generate the "optimal" number of bins displayed in the chart. All three generators implicitly assume a normal distribution. Sturges is the default setting. The Scott generator should be used with random samples of normally distributed data. The Freedman-Diaconis' generator is less sensitive than the standard deviation to outliers in the data.

- **X Axis:** Analytic Solver allows users to manually set the Min and Max values for the X Axis. Simply type the desired value into the appropriate text box.

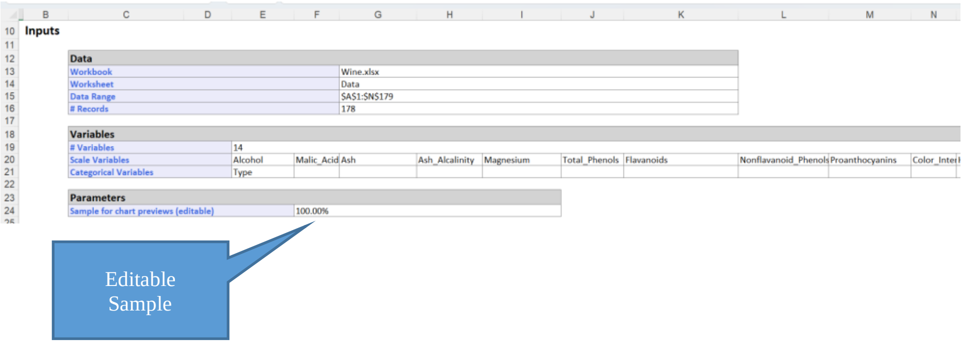
- **Style**:
  - ○ **Color**: Select a color, to apply to the entire variable graph, by clicking the down arrow next to Color and then selecting the desired hue.

Analyze Data Report

Click the "X" in the upper right hand corner to return to the preview dialog and then again to exit to the Statistics worksheet. This worksheet contains all computed statistics, percentiles and Six Sigma indices for each variable included in the report.

The top of the report contains the Output Navigator and the Inputs sections of the report.

Analyze Data Report: Inputs



Inputs

Data	
Workbook	Wine.xlsx
Worksheet	Data
Data Range	$A$1:$N$179
# Records	178

Variables										
# Variables	14									
Scale Variables	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Inte
Categorical Variables	Type									

Parameters	
Sample for chart previews (editable)	100.00%

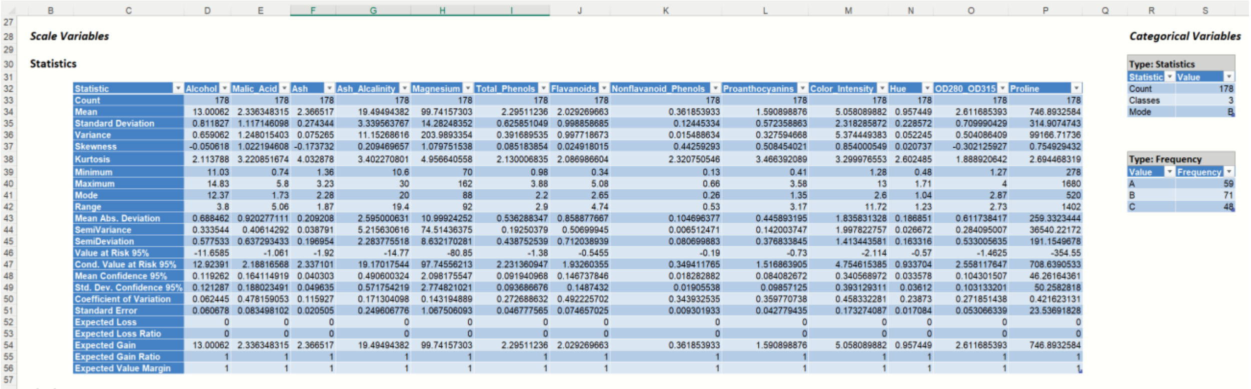
- **Output Navigator**: Click any of the links to jump to that section of the report.

- **Inputs**: This section contains information pertaining to the data source and the variables included in the data analysis.

- **Parameters**: If you find that the Preview dialog is taking a long time to open, you can edit the Sample Data Fraction % here. Simply enter a smaller percentage to speed up the opening of the dialog.

The next section, Statistics, lists each computed statistic found in the detailed chart view for each variable included in the analysis, scale and categorical.

Analyze Data Report: Statistics



Scale Variables

Statistics

Statistic	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
Count	178	178	178	178	178	178	178	178	178	178	178	178	178
Mean	13.00062	2.336348315	2.366517	19.49494382	99.74157303	2.29511236	2.029269663	0.361853933	1.590898876	5.058089882	0.957449	2.611685393	746.8932584
Standard Deviation	0.811827	1.117146098	0.274344	3.339563767	14.28248352	0.625851049	0.998858685	0.12445334	0.572358863	2.318285872	0.228572	0.709990429	314.9074743
Variance	0.659062	1.248015403	0.075265	11.15268616	203.9893354	0.391689535	0.997718673	0.015488634	0.327594668	5.374449383	0.052245	0.504086409	99166.71736
Skewness	-0.050618	1.022194608	-0.173732	0.209469657	1.079751538	0.085183854	0.024918015	0.44259293	0.508454021	0.854000549	0.020737	-0.302125927	0.754929432
Kurtosis	2.113788	3.220851674	4.032878	3.402270801	4.956640558	2.130006835	2.086986604	2.320750546	3.466392089	3.299976553	2.602485	1.888920642	2.694468319
Minimum	11.03	0.74	1.36	10.6	70	0.98	0.34	0.13	0.41	1.28	0.48	1.27	278
Maximum	14.83	5.8	3.23	30	162	3.88	5.08	0.66	3.58	13	1.71	4	1680
Mode	12.37	1.73	2.28	20	88	2.2	2.65	0.26	1.35	2.6	1.04	2.87	520
Range	3.8	5.06	1.87	19.4	92	2.9	4.74	0.53	3.17	11.72	1.23	2.73	1402
Mean Abs. Deviation	0.688462	0.920277111	0.209208	2.595000631	10.99924252	0.536288347	0.858877667	0.104696377	0.445893195	1.835831328	0.186851	0.611738417	259.3323444
SemiVariance	0.333544	0.40614292	0.038791	5.215630616	74.51436375	0.19250379	0.50699945	0.006512471	0.142003747	1.997822757	0.026672	0.284095007	36540.22172
SemiDeviation	0.577533	0.637293433	0.196954	2.283775518	8.632170281	0.438752539	0.712038939	0.080699883	0.376833845	1.413443581	0.163316	0.533005635	191.1549678
Value at Risk 95%	-11.6585	-1.061	-1.92	-14.77	-80.85	-1.38	-0.5455	-0.19	-0.73	-2.114	-0.57	-1.4625	-354.55
Cond. Value at Risk 95%	12.92391	2.18816568	2.337101	19.17017544	97.74556213	2.231360947	1.93260355	0.349411765	1.516863905	4.754615385	0.933704	2.558117647	708.6390533
Mean Confidence 95%	0.119262	0.164114919	0.040303	0.490600324	2.098175547	0.091940968	0.146737846	0.018282882	0.084082672	0.340568972	0.033578	0.104301507	46.26164361
Std. Dev. Confidence 95%	0.121287	0.188023491	0.049635	0.571754219	2.774821021	0.093686676	0.1487432	0.01905538	0.09857125	0.393129311	0.03612	0.103133201	50.2582818
Coefficient of Variation	0.062445	0.478159053	0.115927	0.171304098	0.143194889	0.272688632	0.492225702	0.343932535	0.359770738	0.458332281	0.23873	0.271851438	0.421623131
Standard Error	0.060678	0.083498102	0.020505	0.249606776	1.067506093	0.046777565	0.074657025	0.009301933	0.042779435	0.173274087	0.017084	0.053066339	23.53691828
Expected Loss	0	0	0	0	0	0	0	0	0	0	0	0	0
Expected Loss Ratio	0	0	0	0	0	0	0	0	0	0	0	0	0
Expected Gain	13.00062	2.336348315	2.366517	19.49494382	99.74157303	2.29511236	2.029269663	0.361853933	1.590898876	5.058089882	0.957449	2.611685393	746.8932584
Expected Gain Ratio	1	1	1	1	1	1	1	1	1	1	1	1	1
Expected Value Margin	1	1	1	1	1	1	1	1	1	1	1	1	1

Categorical Variables

Type: Statistics

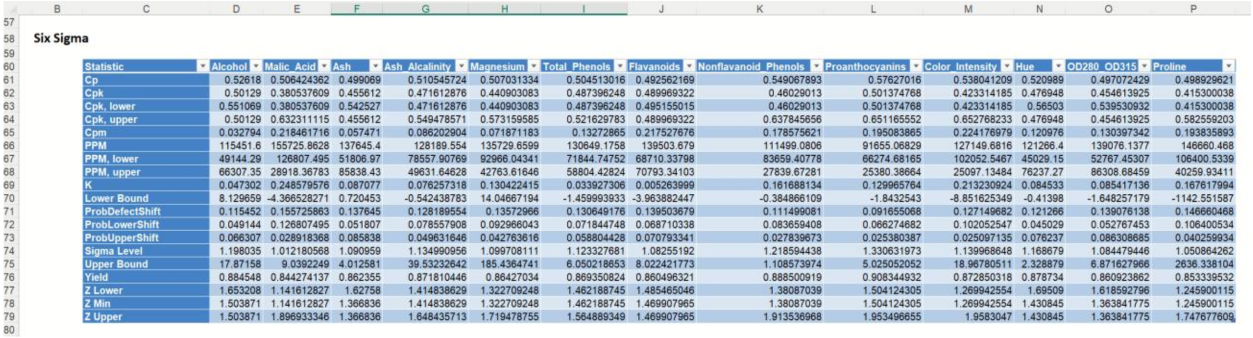
Statistic	Value
Count	178
Classes	3
Mode	B

Type: Frequency

Value	Frequency
A	59
B	71
C	48

Scroll down to the Six Sigma section of the report to see all 19 Six Sigma statistics and indices.

Analyze Data Report: Six Sigma

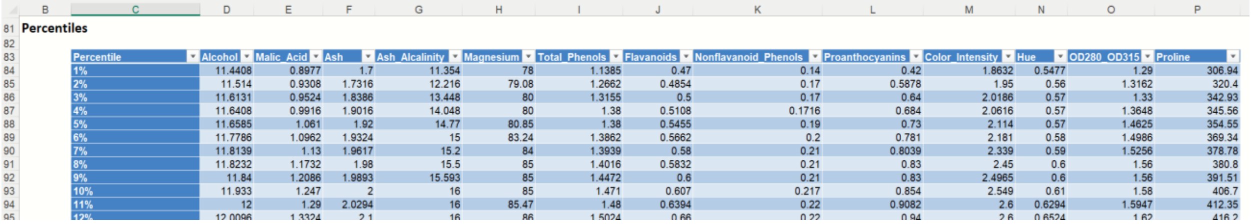


Six Sigma

Statistic	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
Cp	0.52618	0.506424362	0.499069	0.510545724	0.507031334	0.504513016	0.492562169	0.549067893	0.57627016	0.538041209	0.520989	0.497072429	0.498929621
Cpk	0.50129	0.380537609	0.455612	0.471612876	0.440903083	0.487396248	0.489969322	0.46029013	0.501374768	0.423314185	0.476948	0.454613925	0.415300038
Cpk, lower	0.551069	0.380537609	0.542527	0.471612876	0.440903083	0.487396248	0.495155015	0.46029013	0.501374768	0.423314185	0.56503	0.539530932	0.415300038
Cpk, upper	0.50129	0.632311115	0.455612	0.549478571	0.573159585	0.521629783	0.489969322	0.637845656	0.651165552	0.652768233	0.476948	0.454613925	0.582559203
Cpm	0.032794	0.218461716	0.057471	0.086202904	0.071871183	0.13272865	0.217527676	0.178575621	0.195083865	0.224176979	0.120976	0.130397342	0.193835893
PPM	115451.6	155725.8628	137645.4	128189.554	135729.6599	130649.1758	139503.679	111499.0806	91655.06829	127149.6816	121266.4	139076.1377	146660.468
PPM, lower	49144.29	126807.495	51806.97	78557.90769	92966.04341	71844.74752	68710.33798	83659.40778	66274.68165	102052.5467	45029.15	52767.45307	106400.5339
PPM, upper	66307.35	28918.36783	85838.43	49631.64628	42763.61646	58804.42824	70793.34103	27839.67281	25380.38664	25097.13484	76237.27	86308.68459	40259.93411
K	0.047302	0.248579576	0.087077	0.076257318	0.130422415	0.033927306	0.005263999	0.161688134	0.129965764	0.213230924	0.084533	0.085417136	0.167617994
Lower Bound	8.129659	-4.366528271	0.720453	-0.542438783	14.04667194	-1.459993933	-3.963882447	-0.384866109	-1.8432543	-8.851625349	-0.41398	-1.648257179	-1142.551587
ProbDefectShift	0.115452	0.155725863	0.137645	0.128189554	0.13572966	0.130649176	0.139503679	0.111499081	0.091655068	0.127149682	0.121266	0.139076138	0.146660468
ProbLowerShift	0.049144	0.126807495	0.051807	0.078557908	0.092966043	0.071844748	0.068710338	0.083659408	0.066274682	0.102052547	0.045029	0.052767453	0.106400534
ProbUpperShift	0.066307	0.028918368	0.085838	0.049631646	0.042763616	0.058804428	0.070793341	0.027839673	0.025380387	0.025097135	0.076237	0.086308685	0.040259934
Sigma Level	1.198035	1.012180568	1.090959	1.134990956	1.099708111	1.123327681	1.08255192	1.218594438	1.330631973	1.139968648	1.168679	1.084479446	1.050864262
Upper Bound	17.87158	9.0392249	4.012581	39.53232642	185.4364741	6.050218653	8.022421773	1.108573974	5.025052052	18.96780511	2.328879	6.871627966	2636.338104
Yield	0.884548	0.844274137	0.862355	0.871810446	0.86427034	0.869350824	0.860496321	0.888500919	0.908344932	0.872850318	0.878734	0.860923862	0.853339532
Z Lower	1.653208	1.141612827	1.62758	1.414838629	1.322709248	1.462188745	1.485465046	1.38087039	1.504124305	1.269942554	1.69509	1.618592796	1.245900115
Z Min	1.503871	1.141612827	1.366836	1.414838629	1.322709248	1.462188745	1.469907965	1.38087039	1.504124305	1.269942554	1.430845	1.363841775	1.245900115
Z Upper	1.503871	1.896933346	1.366836	1.648435713	1.719478755	1.564889349	1.469907965	1.913536968	1.953496655	1.9583047	1.430845	1.363841775	1.747677609

Finally, scroll down to Percentiles to view all 99 percentile values from 0.01 to .99.

Analyze Data Report: Percentiles



Percentiles

Percentile	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
1%	11.4408	0.8977	1.7	11.354	78	1.1385	0.47	0.14	0.42	1.8632	0.5477	1.29	306.94
2%	11.514	0.9308	1.7316	12.216	79.08	1.2662	0.4854	0.17	0.5878	1.95	0.56	1.3162	320.4
3%	11.6131	0.9524	1.8386	13.448	80	1.3155	0.5	0.17	0.64	2.0186	0.57	1.33	342.93
4%	11.6408	0.9916	1.9016	14.048	80	1.38	0.5108	0.1716	0.684	2.0616	0.57	1.3648	345.56
5%	11.6585	1.061	1.92	14.77	80.85	1.38	0.5455	0.19	0.73	2.114	0.57	1.4625	354.55
6%	11.7786	1.0962	1.9324	15	83.24	1.3862	0.5662	0.2	0.781	2.181	0.58	1.4986	369.34
7%	11.8139	1.13	1.9617	15.2	84	1.3939	0.58	0.21	0.8039	2.339	0.59	1.5256	378.78
8%	11.8232	1.1732	1.98	15.5	85	1.4016	0.5832	0.21	0.83	2.45	0.6	1.56	380.8
9%	11.84	1.2086	1.9893	15.593	85	1.4472	0.6	0.21	0.83	2.4965	0.6	1.56	391.51
10%	11.933	1.247	2	16	85	1.471	0.607	0.217	0.854	2.549	0.61	1.58	406.7
11%	12	1.29	2.0294	16	85.47	1.48	0.6394	0.22	0.9082	2.6	0.6294	1.5947	412.35
12%	12.0096	1.3324	2.1	16	86	1.5024	0.66	0.22	0.94	2.6	0.6524	1.62	416.2

Analyze Data Options

Descriptions of each option features in the Analyze Data application may be found below.

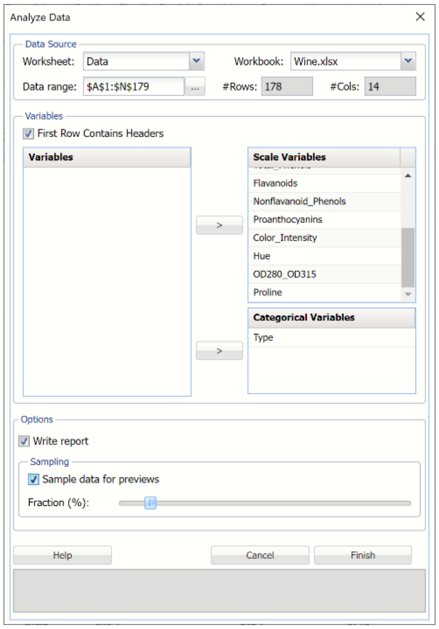
Data Source

This portion of the dialog includes data relevant to the data source such as the Worksheet name, the Workbook name, the Data range in the worksheet and the number of rows and columns in the data range.

Variables

Select the variable(s) to be included in the data analysis under Variables, then click the arrow to move the variable to either Scale Variables or Categorical Variables, depending on the type of variable selected.

Analyze Data dialog



Analyze Data

Data Source

Worksheet: Data

Workbook: Wine.xlsx

Data range: $A$1:$N$179

#Rows: 178

#Cols: 14

Variables

☑ First Row Contains Headers

Variables

Scale Variables: Flavanoids, Nonflavanoid_Phenols, Proanthocyanins, Color_Intensity, Hue, OD280_OD315, Proline

Categorical Variables: Type

Options

☑ Write report

Sampling

☑ Sample data for previews

Fraction (%):

Options: Write Report

If *Write report* is selected, Analytic Solver Data Science will insert a report containing all statistics in the detailed chart dialog. This includes all Statistics, Percentiles and Six Sigma indices. (The Statistics worksheet will be inserted to the right of the Data tab.) If this checkbox is left unchecked, no report will be inserted into the workbook. The preview dialog will be displayed, and detailed charts will be available if double-clicked. Once the dialog is closed, it will not persist in the workbook. (The application would have to be re-run to re-open the chart.)

Options: Sample data for previews and Fraction %

If the worksheet contains an extremely large dataset, users can select *Sample data for previews* and use the *Fraction %* slider to limit the amount of data utilized when creating the chart **previews**. This option has no bearing on the amount of data included in the detailed charts. Detailed charts always use the full amount of data to produce the interactive chart and statistics. The Fraction % may be changed after the report is created by editing the Sample for chart previews cell. For more information see below.

Feature Selection

Dimensionality Reduction is the process of deriving a lower-dimensional representation of original data, that still captures the most significant relationships, to be used to represent the original data in a model. This domain can be divided into two branches, feature selection and feature extraction. Feature selection attempts to discover a subset of the original variables while Feature Extraction attempts to map a high – dimensional model to a lower dimensional space. In past versions, Analytic Solver Data Science only contained one feature extraction tool which could be used outside of a classification or regression method, Principal Components Analysis (Transform – Principal Components on the Data Science ribbon). However, in V2015, a new feature selection tool was added, Feature Selection. For more information on Principal Components Analysis, please see the chapter of the same name.

In V2015, a new tool for Dimensionality Reduction was introduced, Feature Selection. Feature Selection attempts to identify the *best* subset of variables (or features) out of the available variables (or features) to be used as input to a classification or regression method. The main goal of Feature Selection is threefold – to "clean" the data, to eliminate redundancies, and to quickly identify the most relevant and useful information hidden within the data thereby reducing the scale or dimensionality of the data. Feature Selection results in an enhanced ability to explore the data, visualize the data and in some cases to make some previously infeasible analytic models feasible.

One important issue in Feature Selection is how to define the "best" subset. If using a supervised learning technique (classification/regression model), the "best" subset would result in a model with the lowest misclassification rate or residual error. This presents a different question – which classification method should we use? A given subset (of variables) may be optimal for one method but not for another. One might answer, “try all possible subsets”. Unfortunately, the number of all possible combinations of variables can quickly grow to an exponential number making the problem of finding the best subset (of variables) infeasible for even a moderate number of variables. Even trying to find the best subset of 10 variables out of a total of 50 would lead to 10,272,278,170 combinations!

Feature Selection methods are divided into 3 major categories: filters, wrappers, and embedded approaches. Analytic Solver Data Science’s new Feature Selection tool uses Filter Methods which provide the mechanisms to rank variables according to one or more univariate measure and to select the top-ranked variables to represent the data in the model. In Analytic Solver Data Science, Feature Selection is only supported in supervised learning methods; the importance of a variable is based on its relation, or ability to predict the value of, the output variable. The measures used to rank the variables can be divided into three main categories: correlation – based, statistical tests, and information – theoretic measures. The definitive characteristic of Filter methods is their independence of any particular model, therefore making them widely applicable as a preprocessing step for supervised learning algorithms. Usually, filter methods are much less computationally expensive than other Feature Selection approaches. This means that when faced with a big data problem, these methods are sometimes the only methods that are computationally feasible. The major drawback is that filters do not examine subsets containing multiple variables, they only rank them individually. Sometimes, individual features, not important by themselves, could become relevant when combined with other feature(s).

Feature Selection is a very important topic that becomes more relevant as the number of variables in a model increases. See the example below for a walk through of this significant new feature.

Feature Selection Example

Analytic Solver Data Science’s Feature Selection tool gives users the ability to rank and select the most relevant variables for inclusion in a classification or regression model. In many cases the most accurate models, or the models with the lowest misclassification or residual errors, have benefited from better feature selection, using a combination of human insights and automated methods. Analytic Solver Data Science provides a facility to compute all of the following metrics, described in the literature, to give users information on what features should be included, or excluded, from their models.

- **Correlation-based**
  - Pearson product-moment correlation
  - Spearman rank correlation
  - Kendall concordance
- **Statistical/probabilistic independence metrics**
  - Chi-square statistic
  - Cramer’s V
  - F-statistic
  - Fisher score
  - Welch’s statistic
- **Information-theoretic metrics**
  - Gain Ratio
  - Mutual Information (Information Gain)

Only some of these metrics can be used in any given application, depending on the characteristics of the input variables (features) and the type of problem. In a supervised setting, if we classify data science problems as follows:

- $\mathbb{R}^n \rightarrow \mathbb{R}$: real-valued features, regression problem
- $\mathbb{R}^n \rightarrow \{0, 1\}$: real-valued features, binary classification problem
- $\mathbb{R}^n \rightarrow \{1..C\}$: real-valued features, multi-class classification problem
- $\{1..C\}^n \rightarrow \mathbb{R}^n$: nominal categorical features, regression problem
- $\{1..C\}^n \rightarrow \{0, 1\}$: nominal categorical features, binary classification problem
- $\{1..C\}^n \rightarrow \{1..C\}$: nominal categorical features, multi-class classification problem

then we can describe the applicability of the Feature Selection metrics by the following table:

	R-R	R-{0,1}	R-{1..C}	{1..C}-R	{1..C}-{0,1}	{1..C}-{1..C}
Pearson	**N**					
Spearman	**N**					
Kendall	**N**					
Welch's	D	**N**				
F-Test	D	**N**	**N**			
Chi-squared	D	D	D	D	**N**	**N**
Mutual Info	D	D	D	D	**N**	**N**
Gain Ratio	D	D	D	D	**N**	**N**
Fisher	D	**N**	**N**			
Gini	D	**N**	**N**			

"N" means that metrics can be applied naturally, and “D” means that features and/or the outcome variable must be discretized before applying the particular filter.

As a result, depending on the variables (features) selected and the type of problem chosen in the first dialog, various metrics will be available or disabled in the second dialog.

Feature Selection Example

The goal of this example is three-fold: 1. To use Feature Selection as a tool for exploring relationships between features and the outcome variable, 2. Reducing the dimensionality based on the Feature Selection results and 3. Evaluating the performance of a supervised learning algorithm (a classification algorithm) for different feature subsets.

This example uses the Heart Failure Clinical Records Dataset[1], which contains thirteen variables describing 299 patients experiencing heart failure. The journal article referenced here discusses how the authors analyzed the dataset to first rank the features (variables) by significance and then used the Random Trees machine learning algorithm to fit a model to the dataset. This example attempts to emulate their results.

A description of each variable contained in the dataset appears in the table below.

VARIABLE	DESCRIPTION
AGE	Age of patient
ANAEMIA	Decrease of red blood cells or hemoglobin (boolean)
CREATINE_PHOSPHOKINASE	Level of the CPK enzyme in the blood (mcg/L)
DIABETES	If the patient has diabetes (boolean)
EJECTION_FRACTION	Percentage of blood leaving the heart at each contraction (percentage)
HIGH_BLOOD_PRESSURE	If the patient has hypertension (boolean)
PLATELETS	Platelets in the blood (kiloplatelets/mL)
SERUM_CREATININE	Level of serum creatinine in the blood (mg/dL)
SERUM_SODIUM	Level of serum sodium in the blood (mEq/L)
SEX	Woman (0) or man (1)
SMOKING	If the patient smokes or not (boolean)
TIME	Follow-up period (days)
DEATH_EVENT	If the patient deceased during the follow-up period (boolean)

To open the example dataset, click **Help – Example Models – Forecasting/Data Science Examples** – **Heart Failure Clinical Records**.

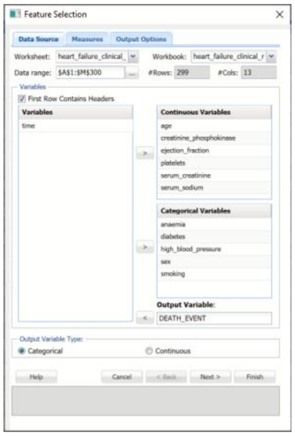
Select a cell within the data (say A2), then click **Explore – Feature Selection** to bring up the first dialog.

[1] Davide Chicco, Giuseppe Jurman: Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. BMC Medical Informatics and Decision Making 20, 16 (2020). (link)

- Select all the following variables as Continuous Variables: age, creatinine_phosphokinase, ejection_fraction, platelets, serum_creatinine and serum_sodium.
- Select the following variables as Categorical Variables: anaemia, diabetes, high_blood_pressure, sex and smoking.
- Select DEATH_EVENT as the Output Variable
- Confirm that Categorical is selected for Output Variable Type. This setting denotes that the Output Variable is a categorical variable. If the number of unique values in the *Output variable* is greater than 10, then *Continuous* will be selected by default. However, at any time the User may override the default choice based on his or her own knowledge of the variable.

This analysis omits the time variable. The Feature Selection dialog should look similar to the screenshot below.

Figure 1: Feature Selection Data Source dialog



Worksheet: heart_failure_clinical_
Workbook: heart_failure_clinical_r
Data range: $A$1:$M$300
#Rows: 299
#Cols: 13

Variables

☑ First Row Contains Headers

Variables: time

Continuous Variables: age, creatinine_phosphokinase, ejection_fraction, platelets, serum_creatinine, serum_sodium

Categorical Variables: anaemia, diabetes, high_blood_pressure, sex, smoking

Output Variable: DEATH_EVENT

Output Variable Type: ● Categorical ○ Continuous

Click the **Measures** tab or click **Next** to open the Measures dialog.

Since we have continuous variables, *Discretize predictors* is enabled. When this option is selected, Analytic Solver Data Science will transform continuous variables into discrete, categorical data in order to be able to calculate statistics, as shown in the table in the Introduction to this chapter.

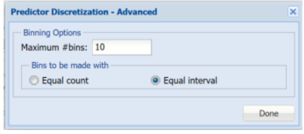
This dataset contains both continuous (or real-valued) features and categorical features which puts this dataset into the following category.

$\mathbb{R}^n \rightarrow \{0, 1\}$: real-valued features, binary classification problem

As a result, if interested in evaluating the relevance of features according to the Chi-Squared Test or measures available in the Information Theory group (Mutual Information and Gain ratio), the variables must first be discretized.

Select **Discretize predictors**, then click **Advanced**. Leave the defaults of **10** for *Maximum # bins* and **Equal Interval** for *Bins to be made with*. Analytic Solver Data Science will create 10 bins and will assign records to the bins based on if the variable's value falls in the interval of the bin. This will be performed for each of the Continuous Variables.

Figure 2: Predictor Discretization - Advanced



Predictor Discretization - Advanced

Binning Options

Maximum #bins: 10

Bins to be made with

○ Equal count ● Equal interval

Note: Discretize output variable is disabled because our output variable, DEATH_EVENT, is already a categorical nominal variable. If we had no Continuous Variables and all Categorical Variables, *Discretize predictors* would be disabled.

Select **Chi-squared** and **Cramer's V** under *Chi-Squared Test*. The Chi-squared test statistic is used to assess the statistical independence of two events. When applied to Feature Selection, it is used as a test of independence to assess whether the assigned class is independent of a particular variable. The minimum value for this statistic is 0. The higher the Chi-Squared statistic, the more independent the variable.

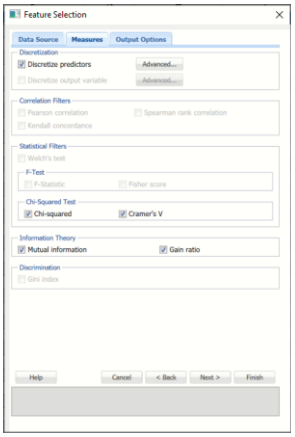
Cramer's V is a variation of the Chi-Squared statistic that also measures the association between two discrete nominal variables. This statistic ranges from 0 to 1 with 0 indicating no association between the two variables and 1 indicating complete association (the two variables are equal).

Select *Mutual information* and *Gain ratio* within the *Information Theory* frame. Mutual information is the degree of a variables' mutual dependence or the amount of uncertainty in variable 1 that can be reduced by incorporating knowledge about variable 2. Mutual Information is non-negative and is equal to zero if the two variables are statistically independent. Also, it is always less than the entropy (amount of information contained) in each individual variable.

The *Gain Ratio*, ranging from 0 and 1, is defined as the mutual information (or information gain) normalized by the feature entropy. This normalization helps address the problem of overemphasizing features with many values but the normalization results in an overestimate of the relevance of features with low entropy. It is a good practice to consider both mutual information and gain ratio for deciding on feature rankings. The larger the gain ratio, the larger the evidence for the feature to be relevant in a classification model.

For more information on the remaining options on this dialog, see the Using Feature Selection below.

Figure 3: Feature Selection Measures dialog



Feature Selection

Discretization

☑ Discretize predictors

☐ Discretize output variable

Correlation Filters

☐ Pearson correlation ☐ Spearman rank correlation

☐ Kendall concordance

Statistical Filters

☐ Welch's test

F-Test

☐ F-Statistic ☐ Fisher score

Chi-Squared Test

☑ Chi-squared ☑ Cramer's V

Information Theory

☑ Mutual information ☑ Gain ratio

Discrimination

☐ Gini index

Click the **Output Options** tab or click **Next** to open the Output Options dialog. *Table of all produced measures* is selected by default. When this option is selected, Analytic Solver Data Science will produce a report containing all measures selected on the Measures tab.

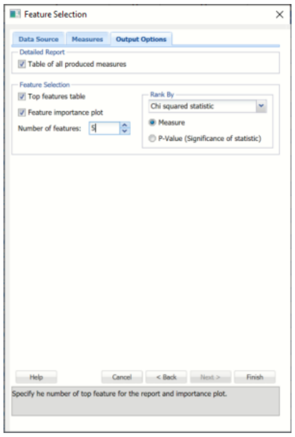
Top Features table is selected by default. This option produces a report containing only top variables as indicated by the Number of features edit box.

Select **Feature importance plot**. This option produces a graphical representation of variable importance based on the measure selected in the *Rank By* drop down menu.

Enter **5** for *Number of features*. Analytic Solver Data Science will display the top 5 most important or most relevant features (variables) as ranked by the statistic displayed in the *Rank By* drop down menu.

Keep *Chi squared statistic* selected for the *Rank By* option. Analytic Solver Data Science will display all measures and rank them by the statistic chosen in this drop down menu.

Figure 4: Feature Selection Output Options dialog



Detailed Report

☑ Table of all produced measures

Feature Selection

☑ Top features table

☑ Feature importance plot

Number of features: 5

Rank By: Chi squared statistic

● Measure

○ P-Value (Significance of statistic)

Specify he number of top feature for the report and importance plot.

Click **Finish**.

Two worksheets are inserted to the right of the heart_failure_clinical_records worksheet: FS_Output and FS_Top_Features.

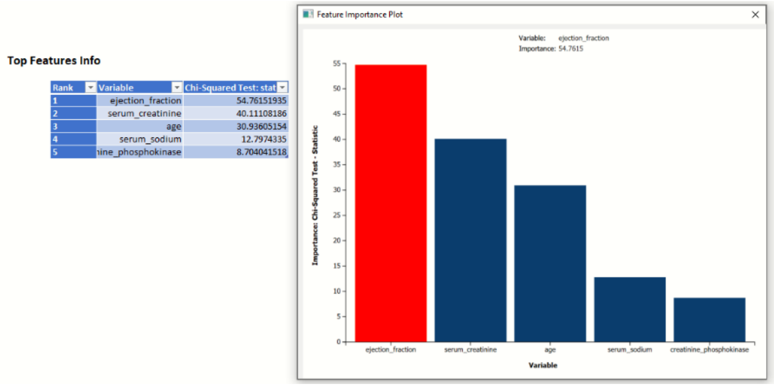
Click the FS_Top_Features tab.

In the Data Science Cloud app, click the Charts icon on the Ribbon to open the Charts dialog, then select **FS_Top_Features** for *Worksheet* and **Feature Importance Chart** for *Chart*.

The Feature Importance Plot ranks the variables by most important or relevant according to the selected measure. In this example, we see that the ejection_fraction, serum_creatinine, age, serum_sodium and creatinine_phosphokinase are the top five most important or relevant variables according to the Chi-Squared statistic. It's beneficial to examine the Feature Selection Importance Plot in order to quickly identify the largest drops or "elbows" in feature relevancy (importance) and select the optimal number of variables for a given classification or regression model.

Note: We could have limited the number of variables displayed on the plot to a specified number of variables (or features) by selecting Number of features and then specifying the number of desired variables. This is useful when the number of input variables is large or we are particularly interested in a specific number of highly – ranked features.

Figure 5: Top Features Plot and Table



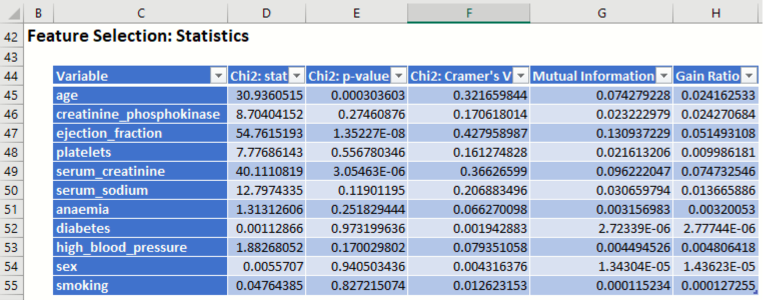
Top Features Info

Rank	Variable	Chi-Squared Test: stat
1	ejection_fraction	54.76151935
2	serum_creatinine	40.11108186
3	age	30.93605154
4	serum_sodium	12.7974335
5	nine_phosphokinase	8.704041518

Run your mouse over each bar in the graph to see the Variable name and Importance factor, in this case Chi-Square, in the top of the dialog.

Click the **X** in the upper right hand corner to close the dialog, then click *FS_Output* tab to open the Feature Selection report.

Figure 6: Feature Selection: Statistics Table



Feature Selection: Statistics

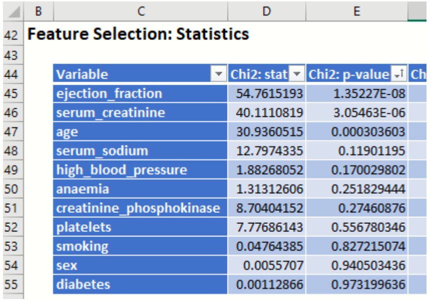
Variable	Chi2: stat	Chi2: p-value	Chi2: Cramer's V	Mutual Information	Gain Ratio
age	30.9360515	0.000303603	0.321659844	0.074279228	0.024162533
creatinine_phosphokinase	8.70404152	0.27460876	0.170618014	0.023222979	0.024270684
ejection_fraction	54.7615193	1.35227E-08	0.427958987	0.130937229	0.051493108
platelets	7.77686143	0.556780346	0.161274828	0.021613206	0.009986181
serum_creatinine	40.1110819	3.05463E-06	0.36626599	0.096222047	0.074732546
serum_sodium	12.7974335	0.11901195	0.206883496	0.030659794	0.013665886
anaemia	1.31312606	0.251829444	0.066270098	0.003156983	0.00320053
diabetes	0.00112866	0.973199636	0.001942883	2.72339E-06	2.77744E-06
high_blood_pressure	1.88268052	0.170029802	0.079351058	0.004494526	0.004806418
sex	0.0055707	0.940503436	0.004316376	1.34304E-05	1.43623E-05
smoking	0.04764385	0.827215074	0.012623153	0.000115234	0.000127255

The Detailed Feature Selection Report displays each computed metric selected on the Measures tab: Chi-squared statistic, Chi-squared P-Value, Cramer's V, Mutual Information, and Gain Ratio.

Chi2: Statistic and p-value

Click the down arrow next to Chi2: p-value to sort the table according to this statistic going from smallest p-value to largest.

Figure 7: Statistics sorted by Chi2:p-value



Feature Selection: Statistics

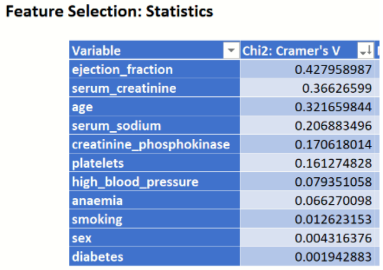
Variable	Chi2: stat	Chi2: p-value
ejection_fraction	54.7615193	1.35227E-08
serum_creatinine	40.1110819	3.05463E-06
age	30.9360515	0.000303603
serum_sodium	12.7974335	0.11901195
high_blood_pressure	1.88268052	0.170029802
anaemia	1.31312606	0.251829444
creatinine_phosphokinase	8.70404152	0.27460876
platelets	7.77686143	0.556780346
smoking	0.04764385	0.827215074
sex	0.0055707	0.940503436
diabetes	0.00112866	0.973199636

According to the Chi-squared test, ejection_fraction, serum_creatinine and age are the 3 most relevant variables for predicting the outcome of a patient in heart failure.

Chi2: Cramer's V

Recall that the Cramer's V statistic ranges from 0 to 1 with 0 indicating no association between the two variables and 1 indicating complete association (the two variables are equal). Sort the Cramer's V statistic from largest to smallest.

Figure 8: Chi2: Cramer's V Statistic



Feature Selection: Statistics

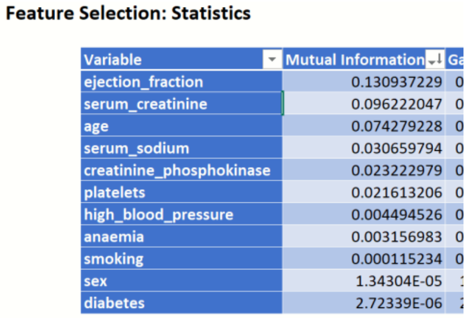
Variable	Chi2: Cramer's V
ejection_fraction	0.427958987
serum_creatinine	0.36626599
age	0.321659844
serum_sodium	0.206883496
creatinine_phosphokinase	0.170618014
platelets	0.161274828
high_blood_pressure	0.079351058
anaemia	0.066270098
smoking	0.012623153
sex	0.004316376
diabetes	0.001942883

Again, this statistics ranks the same four variables, ejection_fraction, serum_creatinine, age and serum_sodium, as the Chi^2 statistic.

Mutual Information

Sort the Mutual Information column by largest to smallest value. This statistic measures how much information the presence/absence of a term contributes to making the correct classification decision.[2] The closer the value to 1, the more contribution the feature provides.

Figure 9: Mutual Information Statistic



Feature Selection: Statistics

Variable	Mutual Information	Ga
ejection_fraction	0.130937229	0
serum_creatinine	0.096222047	0
age	0.074279228	0
serum_sodium	0.030659794	0
creatinine_phosphokinase	0.023222979	0
platelets	0.021613206	0
high_blood_pressure	0.004494526	0
anaemia	0.003156983	0
smoking	0.000115234	0
sex	1.34304E-05	1
diabetes	2.72339E-06	2

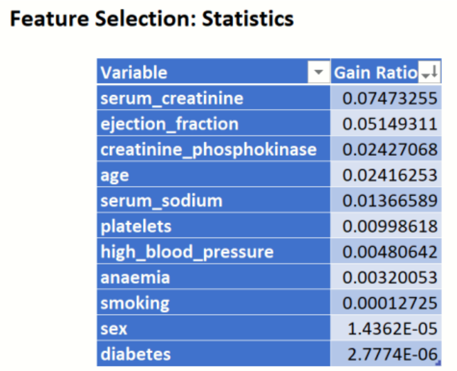
When compared to the Chi2 and Cramer's V statistic, the top four most significant variables calculated for Mututal Information are the same: ejection_fraction, serum_creatinine, age, and serum_sodium.

Gain Ratio

Finally, sort the Gain Ratio from largest to smallest. (Recall that the larger the gain ratio value, the larger the evidence for the feature to be relevant in the classification model.)

[2] https://nlp.stanford.edu/IR-book/html/htmledition/mutual-information-1.html

Figure 10: Gain Ratio



Feature Selection: Statistics

Variable	Gain Ratio
serum_creatinine	0.07473255
ejection_fraction	0.05149311
creatinine_phosphokinase	0.02427068
age	0.02416253
serum_sodium	0.01366589
platelets	0.00998618
high_blood_pressure	0.00480642
anaemia	0.00320053
smoking	0.00012725
sex	1.4362E-05
diabetes	2.7774E-06

While this statistic's rankings differ from the first 4 statistic's rankings, ejection_fraction, age and serum_creatinine are still ranked in the top four positions.

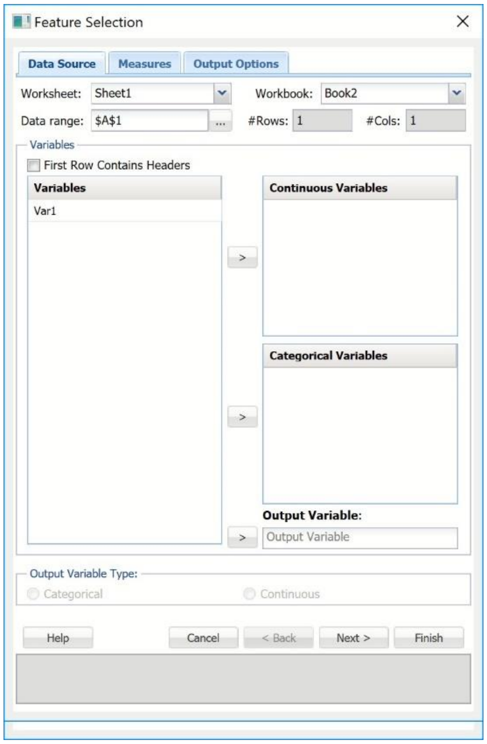
The Feature Selection tool has allowed us to quickly explore and learn about our data. We now have a pretty good idea of which variables are the most relevant or most important to our classification or prediction model, how our variables relate to each other and to the output variable, and which data attributes would be worth extra time and money in future data collection. Interestingly, for this example, most of our ranking statistics have agreed (mostly) on the most important or relevant features with strong evidence. We computed and examined various metrics and statistics and for some (where p-values can be computed) we've seen a statistical evidence that the test of interest succeeded with definitive conclusion. In this example, we've observed that several variable (or features) were consistently ranked in the top 3-4 most important variables by most of the measures produced by Analytic Solver Data Science's Feature Selection tool. However, this will not always be the case. On some datasets you will find that the ranking statistics and metrics compete on rankings. In cases such as these, further analysis may be required.

Fitting the Model

See the Analytic Solver User Guide for an extension of this example which fits a model to the heart_failure dataset using the top variables found by Feature Selection and compares that model to a model fit using all variables in the dataset.

Feature Selection Options

This section gives an explanation on each option located on each of the three Feature Selection tabs.



Worksheet: Sheet1 Workbook: Book2
Data range: $A$1 #Rows: 1 #Cols: 1
□ First Row Contains Headers
Variables: Var1

Variables listbox

Variables (or features) included in the dataset are listed here.

Continuous Variables listbox

Place continuous variables from the Variables listbox to be included in Feature Selection by clicking the > command button. Feature Selection will accept all values for continuous variables except non-numeric values.

Categorical Variables listbox

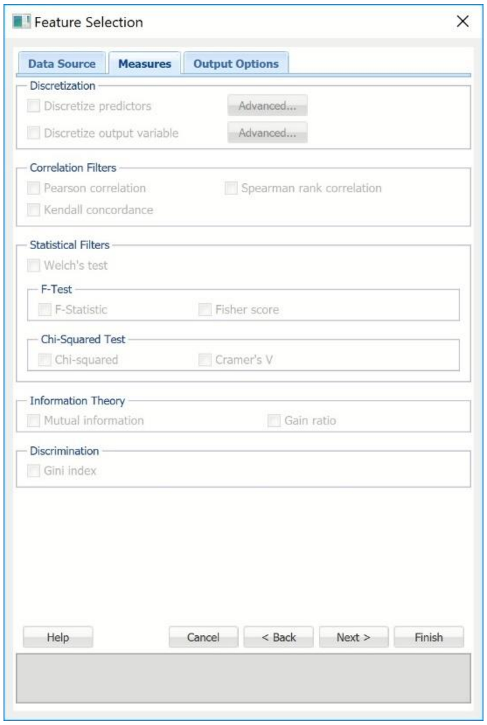
Place categorical variables from the Variables listbox to be included in Feature Selection by clicking the > command button. Feature Selection will accept non-numeric categorical variables.

Output Variable

Click the > command button to select the Output Variable.   This variable may be continuous or categorical. If the variable contains more than 10 unique values, the output variable will be considered “continuous”. If the variable contains less than 10 unique values, the output variable will be considered “categorical”.

Output Variable Type

If the Output Variable contains more than 10 unique values, Continuous will be automatically selected and options relevant to this type of variable will be offered on the Measures tab. If the Output Variable contains 10 or less unique values, Categorical will be automatically selected and options relevant to this type of variable will be offered on the Measures tab. The default selection can always be overridden by the user based on his/her knowledge of the output variable.



In a supervised setting, if we classify data science problems as follows:

- $\mathbb{R}^n \rightarrow \mathbb{R}$: real-valued features, regression problem
- $\mathbb{R}^n \rightarrow \{0,1\}$: real-valued features, binary classification problem
- $\mathbb{R}^n \rightarrow \{1..C\}$: real-valued features, multi-class classification problem
- $\{1..C\}^n \rightarrow \mathbb{R}^n$: nominal categorical features, regression problem
- $\{1..C\}^n \rightarrow \{0,1\}$: nominal categorical features, binary classification problem
- $\{1..C\}^n \rightarrow \{1..C\}$: nominal categorical features, multi-class classification problem

then we can describe the applicability of the Feature Selection metrics by the following table:

	R-R	R-{0,1}	R-{1..C}	{1..C}-R	{1..C}-{0,1}	{1..C}-{1..C}
Pearson	**N**					
Spearman	**N**					
Kendall	**N**					

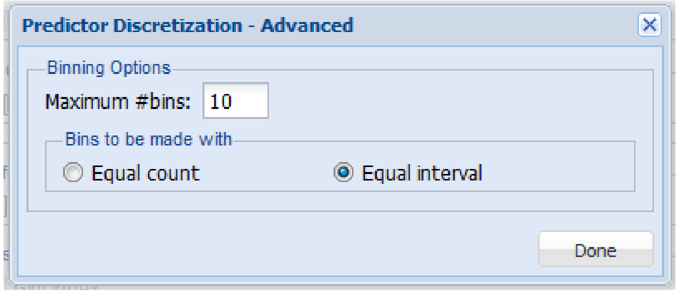
Welch's	D	**N**				
F-Test	D	**N**	**N**			
Chi-squared	D	D	D	D	**N**	**N**
Mutual Info	D	D	D	D	**N**	**N**
Gain Ratio	D	D	D	D	**N**	**N**
Fisher	D	**N**	**N**			
Gini	D	**N**	**N**			

"N" means that metrics can be applied naturally, and “D” means that features and/or the outcome variable must be discretized before applying the particular filter. As a result, depending on the variables (features) selected and the type of problem chosen in the first dialog, various metrics will be available or disabled in this dialog.

Discretize predictors

When this option is selected, Analytic Solver Data Science will transform continuous variables listed under *Continuous Variables* on the *Data Source* tab into categorical variables.

Click the *Advanced* command button to open the *Predictor Discretization - Advanced* dialog. Here the *Maximum number of bins* can be selected. Analytic Solver Data Science will assign records to the bins based on if the variable’s value falls within the interval of the bin (if *Equal interval* is selected for *Bins to be made with*) or on an equal number of records in each bin (if *Equal Count* is selected for *Bins to be made with*). These settings will be applied to each of the variables listed under *Continuous Variables* on the *Data Source* tab.



Predictor Discretization - Advanced

Binning Options

Maximum #bins: 10

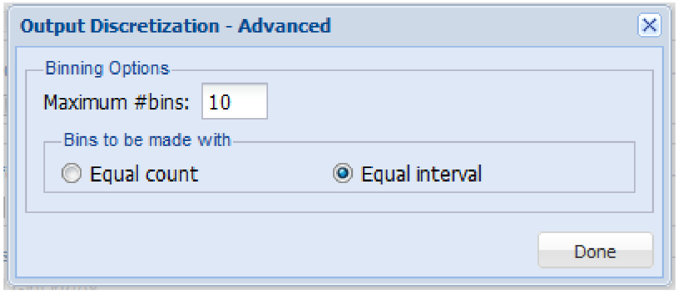
Bins to be made with

○ Equal count ● Equal interval

Discretize output variable

When this option is selected, Analytic Solver Data Science will transform the continuous output variable, listed under *Output Variable* on the *Data Source* tab, into a categorical variable.

Click the *Advanced* command button to open the *Output Discretization - Advanced* dialog. Here the *Maximum number of bins* can be selected. Analytic Solver Data Science will assign records to the bins based on if the output variable’s value falls within the interval of the bin (if *Equal interval* is selected for *Bins to be made with*) or on an equal number of records in each bin (if *Equal Count* is selected for *Bins to be made with*). These settings will be applied to the variable selected for *Output Variable* in the *Data Source* tab.



Output Discretization - Advanced

Binning Options

Maximum #bins: 10

Bins to be made with

○ Equal count ● Equal interval

Pearson correlation

The *Pearson product-moment correlation* coefficient is a widely used statistic that measures the closeness of the linear relationship between two variables, with a value between +1 and −1 inclusive, where 1 indicates complete positive correlation, 0 indicates no correlation, and −1 indicates complete negative correlation.

Spearman rank correlation

The *Spearman rank correlation coefficient* is a nonparametric measure that assesses the relationship between two variables. This measure calculates the correlation coefficient between the ranked values of the two variables. If data values are repeated, the Spearman rank correlation coefficient will be +1 or -1, if each of the variables is a perfect monotone (or non-varying) function of the other.

Kendall concordance

Kendall concordance, also known as Kendall's tau coefficient, is also used to measure the level of association between two variables. A tau value of +1 signifies perfect agreement and a -1 indicates complete disagreement. If a variable and the outcome variable are independent, then one could expect the Kendall tau to be approximately zero.

Welch's Test

Welch's Test is a two-sample test (i.e. applicable for binary classification problems) that is used to check the hypothesis that two populations with possibly unequal variances have equal means. When used with the Feature Selection tool, a large T-statistic value (in conjunction with a small p-value) would provide sufficient evidence that the Distribution of values for each of the two classes are distinct and the variable may have enough discriminative power to be included in the classification model.

F-Statistic

F-Test tests the hypothesis of at least one sample mean being different from other sample means assuming equal variances among all samples. If the variance between the two samples is large with respect to the variance within the sample, the F-statistic will be large. Specifically for Feature Selection purposes, it is used to test if a particular feature is able to separate the records from different target classes by examining between-class and within-class variances.

Fisher score

Fisher Score is a variation of the F-Statistic. It chooses (or assigns higher values) to variables that assign similar values to samples from the same class and different values to samples from different *classes*. The larger the Fisher Score value, the more relevant or important the variable (or feature).

Chi-Squared

The Chi-squared test statistic is used to assess the statistical independence of two events. When applied to Feature Selection, it is used as a test of independence to assess whether the assigned class is independent of a particular variable. The minimum value for this statistic is 0. The higher the Chi-Squared statistic, the more independent the variable.

Cramer's V

Cramer's V is a variation of the Chi-Squared statistic that also measures the association between two discrete nominal variables. This statistic ranges from 0 to 1 with 0 indicating no association between the two variables and 1 indicating complete association (the two variables are equal).

Mutual information

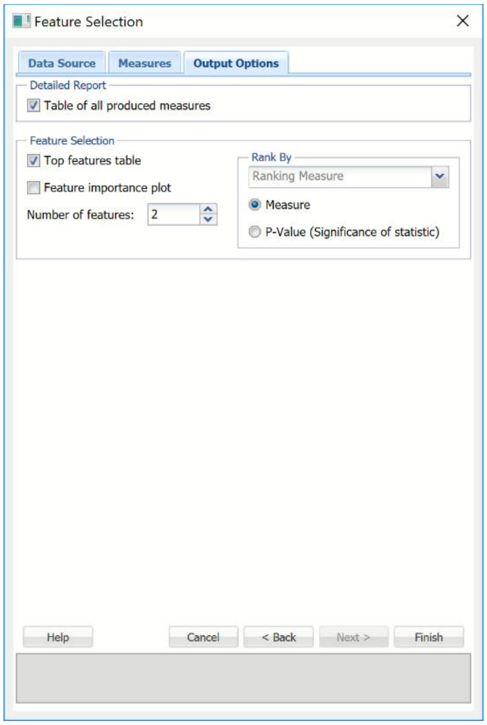
Mutual information is the degree of a variables' mutual dependence or the amount of uncertainty in variable 1 that can be reduced by incorporating knowledge about variable 2. Mutual Information is non-negative and is equal to zero if the two variables are statistically independent. Also, it is always less than the entropy (amount of information contained) in each individual variable.

Gain ratio

This ratio, ranging from 0 and 1, is defined as the mutual information (or information gain) normalized by the feature entropy. This normalization helps address the problem of overestimating features with many values but the normalization overestimates the relevance of features with low entropy. It is a good practice to consider both mutual information and gain ratio for deciding on feature rankings. The larger the gain ratio, the larger the evidence for the feature to be relevant in a classification model.

Gini index

The Gini index measures a variable's ability to distinguish between classes. The maximum value of the index for binary classification is 0.5. The smaller the Gini index, the more relevant the variable.


Feature Selection

Data Source | Measures | Output Options

Detailed Report

☑ Table of all produced measures

Feature Selection

☑ Top features table

□ Feature importance plot

Number of features: 2

Rank By

Ranking Measure

● Measure

○ P-Value (Significance of statistic)

Table of all produced measures

If this option is selected, Analytic Solver Data Science will produce a table containing all of the selected measures from the *Measures* tab. This option is selected by default.

Top features table

If this option is selected, Analytic Solver Data Science will produce a table containing the top number of features as determined by the *Number of features* edit box and the *Rank By* option. This option is not selected by default.

Feature importance plot

If this option is selected, Analytic Solver Data Science will plot the top most important or relevant features as determined by the value entered for the *Number of features* option and the *Rank By* option. This feature is selected by default. To open this plot in the Cloud app, click Charts on the Ribbon.

Number of features

Enter a value here ranging from 1 to the number of features selected in the *Continuous* and *Categorical Variables* listboxes on the *Data Source* tab. This value, along with the *Rank By* option setting, will be used to determine the variables included in the *Top Features Table* and *Feature Importance Plot*. This option has a default setting of “2”.

Rank By

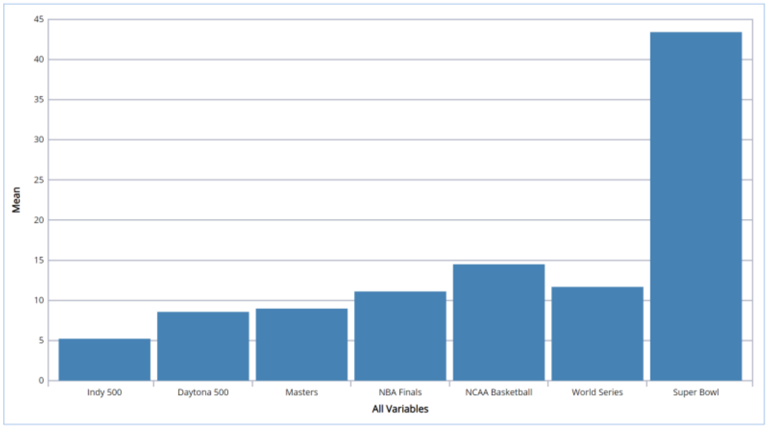
Select *Measure* or *P-Value*, then select the measure from the *Rank By* drop down menu to rank the variables by most important or relevant to least important or relevant in the *Top Features Table* and *Feature Importance* Plot. If *Measure* is selected, then the variables will be ranked by the actual value of the measure or statistic selected, depending on the interpretation (either largest to smallest or smallest to largest). If *P-Value* is selected, then the variables will be ranked from smallest to largest using the P-value of the measure or statistic selected.

Chart Wizard

To create a chart, you can invoke the Chart Wizard by clicking Explore on the Data Science ribbon. A description of each chart type follows.

Bar Chart

The bar chart is one of the easiest and effective plots to create and understand. The best application for this type of chart is comparing an individual statistic (i.e. mean, count, etc.) across a group of variables. The bar height represents the statistic while the bars represent the different variable groups. An example of a bar chart is shown below.



Box Whisker Plot

A box plot graph summarizes a dataset and is often used in exploratory data analysis. This type of graph illustrates the shape of the distribution, its central value, and the range of the data. The plot consists of the most extreme values in the data set (maximum and minimum values), the lower and upper quartiles, and the median.

Box plots are very useful when large numbers of observations are involved or when two or more data sets are being compared. In addition, they are also helpful for indicating whether a distribution is skewed and whether there are any unusual observations (outliers) in the data set. The most important trait of the box plot is its failure to be strongly influenced extreme values, or outliers.

A box plot includes the following statistical features.

Median: The median value in a dataset is the value that appears in the middle of a sorted dataset.   If the dataset has an even number of values then the median is the average of the two middle values in the dataset.

Quartiles: Quartiles, by definition, separate a quarter of data points from the rest. This roughly means that the first quartile is the value **under** which 25% of the data lie and the third quartile is the value **over** which 25% of the data are found. (Note: This indicates that the second quartile is the median itself.)

First Quartile, Q1: Concluding from the definitions above, the first quartile is the median of the lower half of the data. If the number of data points is odd, the lower half includes the median.

Third Quartile, Q3: Third quartile is the median of the upper half of the data. If the number of data points is odd, the upper half of the data includes the median. See the following example.

Consider the following dataset **--**

52, 57, 60, 63, 71, 72, 73, 76, 98, 110, 120, 121, 124

The dataset has 13 values sorted in ascending order. The median is the middle value, (i.e. 7th value in this case.)

Median = 73

Q1 is the median of the first 7 values

25th Percentile = 63

Q3 is the median of the last 7 values.

75th Percentile = 110

The mean is the average of all the data values ((52 + 57 + 60 + 63 + 71 + 72 + 73 + 76 + 98 + 110 + 120 + 121 + 124) / 13).

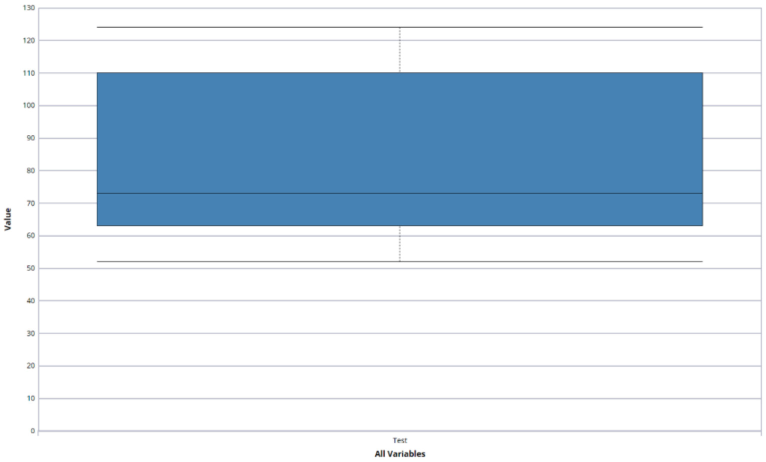
Mean = 84.38

Interquartile Range (IQR) = 47 The Interquartile range is a useful measure of the amount of variation in a set of data and is simply the 75th Percentile – 25th Percentile (110 – 63 = 47).

- The box extends from Q1 to Q3 and includes Q2.
- The median is denoted with a solid line through the box.
- The “whiskers” denote the extreme range of the data, not including any outliers.
    - Min range: 25th percentile **-** 1.5 * IQR
    - Max range: 75th percentile + 1.5 * IQR
- Outliers are denoted by a circle.

Min: 52

Max: 124



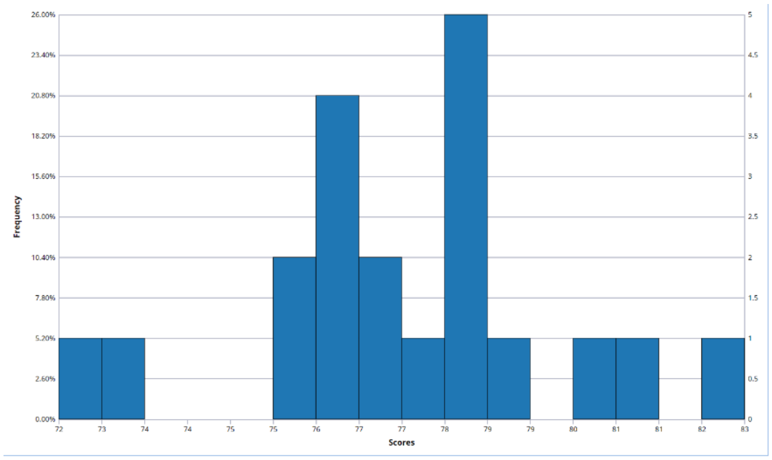
Histogram

A Histogram, or a Frequency Histogram is a bar graph which depicts the range and scale of the observations on the x axis and the number of data points (or frequency) of the various intervals on the y axis. These types of graphs are popular among statisticians. Although these types of graphs do not show the exact values of the data points, they give a very good idea about the spread and shape of the data.

Consider the percentages below from a college final exam.

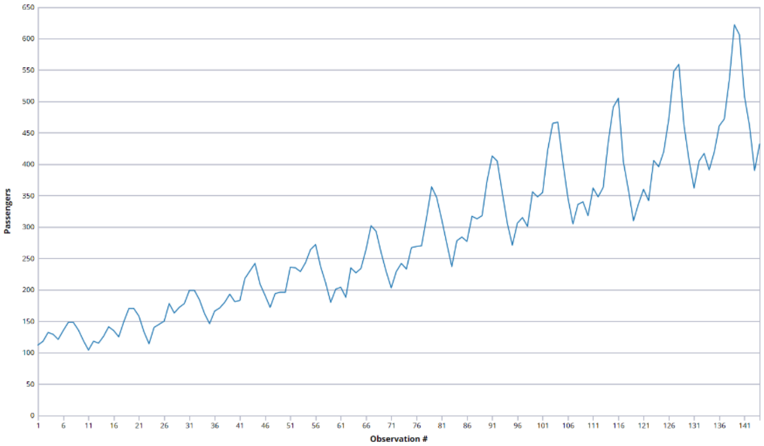
82.5, 78.3, 76.2, 81.2, 72.3, 73.2, 76.3, 77.3, 78.2, 78.5, 75.6, 79.2, 78.3, 80.2, 76.4, 77.9, 75.8, 76.5, 77.3, 78.2

One can immediately see the value of a histogram by taking a quick glance at the graph below. This plot quickly and efficiently illustrates the shape and size of the dataset above.



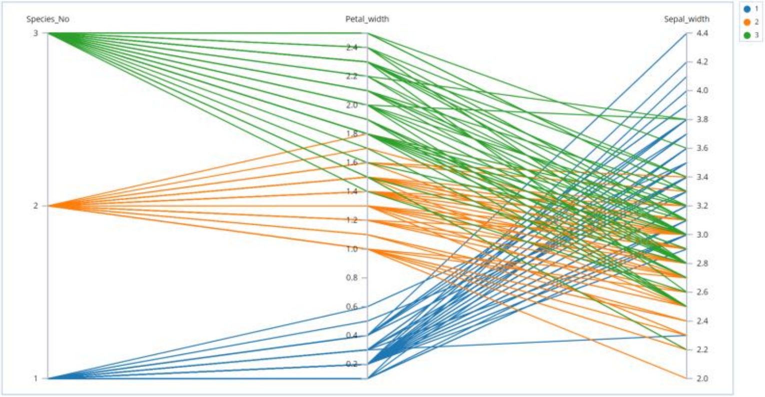
Line Chart

A line chart is best suited for time series datasets. In the example below, the line chart plots the number of airline passengers from January 1949 to December 1960. (The X – axis is the number of months starting with January 1949 as “1”.)



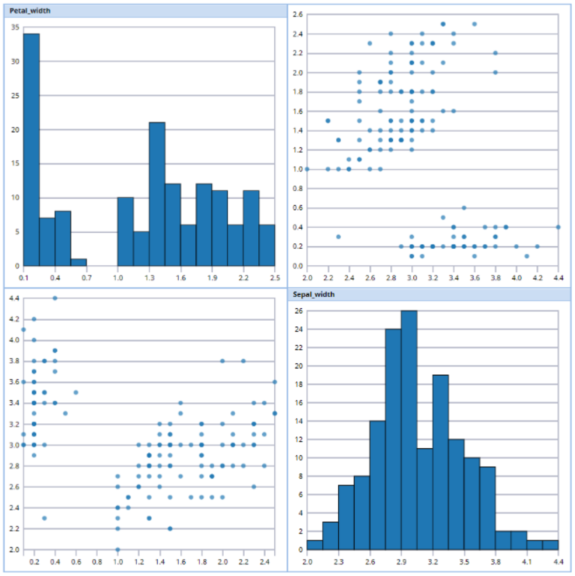
Parallel Coordinates

A Parallel Coordinates plot consists of N number of vertical axes where N is the number of variables selected to be included in the plot. A line is drawn connecting the observation's values for each different variable (each different axis) creating a "multivariate profile". These types of graphs can be useful for prediction and possible data binning. In addition, these graphs can expose clusters, outliers and variable "overlap". An example of a Parallel Coordinates plot is shown below.



Scatterplot

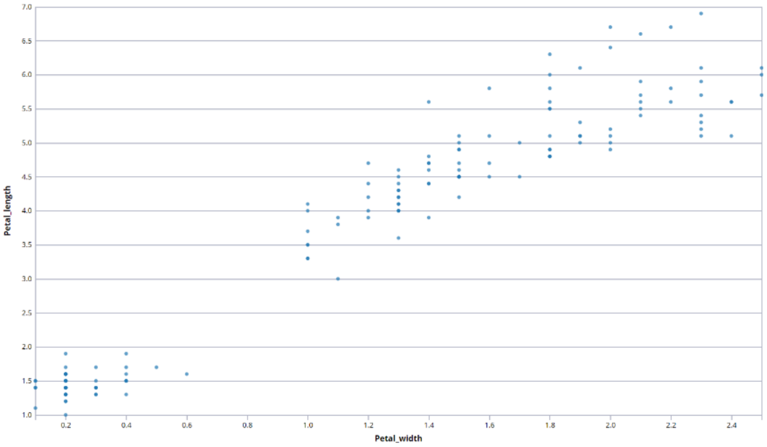
One of the most common, effective and easy to create plots is the scatterplot. These graphs are used to compare the relationships between two variables and are useful in identifying clusters and variable "overlap".



Scatterplot Matrix

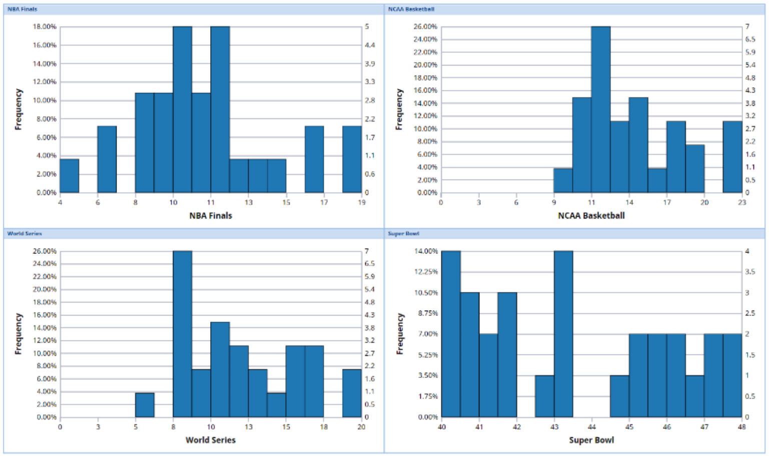
A Matrix plot combines several scatterplots into one panel enabling the user to see pairwise relationships between variables. Given a set of variables Var1, Var2, Var3, ..., Var N the matrix plot contains all the pairwise scatter plots of the variables on a single page in a matrix format. The names of the variables are on the diagonals. In other words, if there are k variables, there will be k rows and k columns in the matrix and the ith row and jth column will be the plot of Var_i versus Var_j.

The axes titles and the values of the variables appear at the edge of the respective row or column. The comparison of the variables and their interactions with one another can be studied easily and with a simple glance which is why matrix plots are becoming increasingly common in general purpose statistical software programs. An example is shown below.



Variable Plot

Analytic Solver Data Science's Variables graph simply plots each selected variable's distribution. See below for an example.



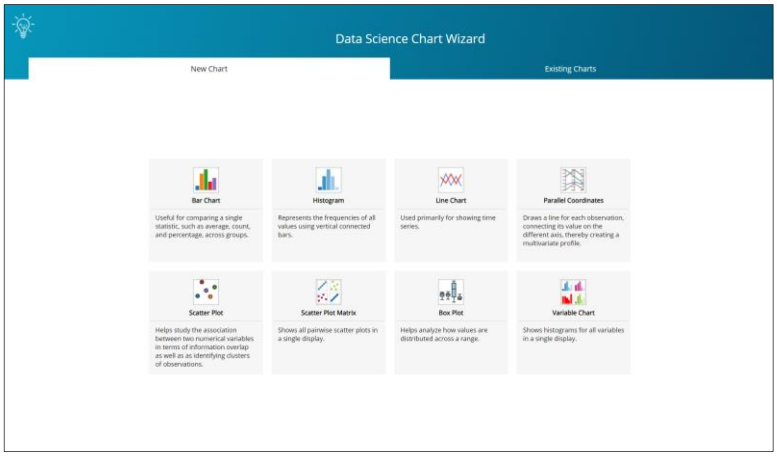
Using the Chart Wizard to create a Bar Chart

This example illustrates how easy it is to create a chart using the chart wizard. This example creates a chart bar chart but any chart can be created using the same steps. See below for examples for each chart type.

1. Click **Help – Example Models** on the Data Science ribbon, then click **Forecasting / Data Science Models** and open the example file, **SportsTVRatings.xlsx**.

2. Select a cell within the data (say A2), then click **Explore – Chart Wizard** to bring up the first dialog of the Chart Wizard.

   Note: When a single cell is selected that is located inside or adjacent to a dataset, the entire dataset will be used for the "Data" cell range. If the user has selected multiple cells in a single rectangular region, those cells, excluding any bounding empty cells, will be used for the "Data" cell range. This behavior was added to allow users to create a chart on a subset of a worksheet dataset without the User being forced to edit the Data address after the chart has been created.



Useful for comparing a single statistic, such as average, count, and percentage, across groups.

Represents the frequencies of all values using vertical connected bars.

Used primarily for showing time series.

Draws a line for each observation, connecting its value on the different axis, thereby creating a multivariate profile.

Helps study the association between two numerical variables in terms of information overlap as well as as identifying clusters of observations.

Shows all pairwise scatter plots in a single display.

Helps analyze how values are distributed across a range.

Shows histograms for all variables in a single display.

3. Select Bar Chart on the New Chart tab.

   Note: Since the variable Year is not a descriptive metric, uncheck "Year", in the Filters pane, to remove the Year from the chart.

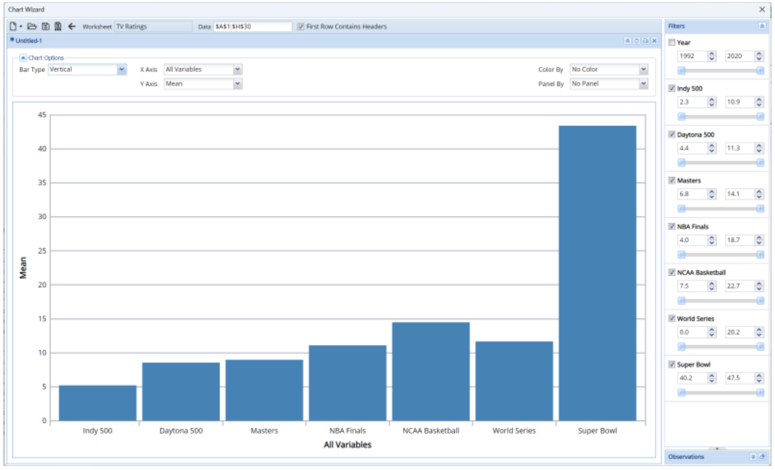


Chart Wizard

Worksheet TV Ratings Data $A$1:$H$30 ☑ First Row Contains Headers

Untitled-1

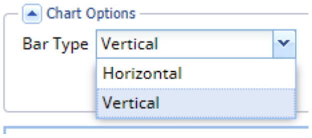
Chart Options

Bar Type Vertical X Axis All Variables Y Axis Mean Color By No Color Panel By No Panel

Filters

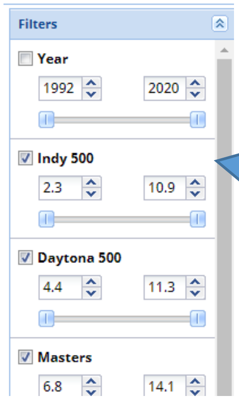
□ Year	1992	2020
☑ Indy 500	2.3	10.9
☑ Daytona 500	4.4	11.3
☑ Masters	6.8	14.1
☑ NBA Finals	4.0	18.7
☑ NCAA Basketball	7.5	22.7
☑ World Series	0.0	20.2
☑ Super Bowl	40.2	47.5

- ○ To change the chart to a horizontal bar chart, click the down arrow next to Bar Type and select Horizontal.



- ○ To change the variables plotted on the X Axis, use the Filters pane. Uncheck the variable to remove from the chart. Select to include in the chart. The data range may be altered by clicking the up an down buttons on the spinner fields or by moving the sliders left (to decrement) or right (to increment).

  On charts that explicitly include the variable tag field, use these tags to add to or remove variables from the chart. For more information on the tag field, see below.

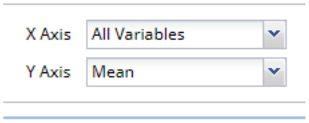


Filters

□ Year	1992	2020
☑ Indy 500	2.3	10.9
☑ Daytona 500	4.4	11.3
☑ Masters	6.8	14.1

Filter the data range by incrementing or decrementing the spinner controls for each variable or by sliding the sliders left and right.

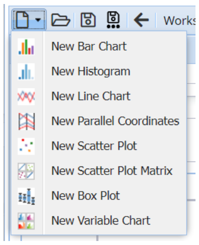
- ○ To change the plotted metric on the Y Axis, click the down arrow next to Y Axis.



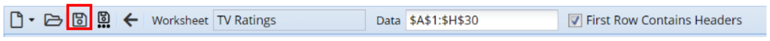
X Axis All Variables

Y Axis Mean

- ○ To add a 2nd chart to the window, simply click the New Chart icon at the top, left of the dialog.

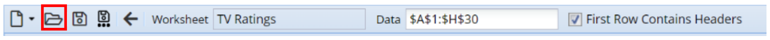


- To save the chart, click the Save icon on the top, left of the dialog.

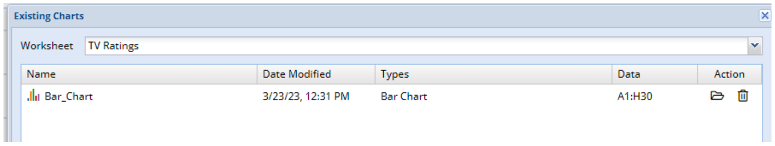
  Worksheet TV Ratings Data $A$1:$H$30 ☑ First Row Contains Headers

- To open an existing chart, click the open file icon.

  Worksheet TV Ratings Data $A$1:$H$30 ☑ First Row Contains Headers

  Click the 🗁 icon to open the chart and the 🗑 icon to delete the chart.

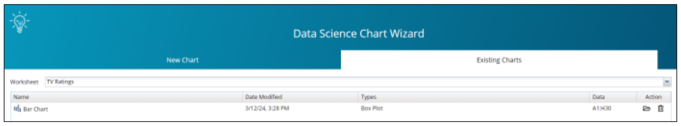
  

  Existing Charts

  Worksheet: TV Ratings

Name	Date Modified	Types	Data	Action
Bar_Chart	3/23/23, 12:31 PM	Bar Chart	A1:H30	🗁 🗑

  You can also open an existing chart by clicking the Existing Charts tab on the opening screen of the Chart Wizard.

  Data Science Chart Wizard

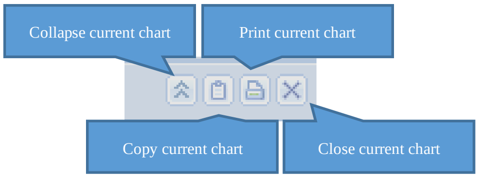
  New Chart | Existing Charts

  Worksheet: TV Ratings

Name	Date Modified	Types	Data	Action
Bar Chart	3/12/24, 3:28 PM	Box Plot	A1:H30	🗁 🗑

Charts are saved by worksheet. Select the desired worksheet to view the saved charts.

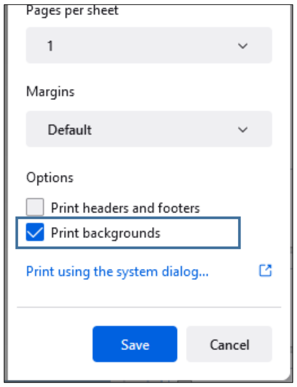
- Click the back arrow to go back to the chart type selection dialog. Note that the current chart will be lost if not saved.
- Chart changes are tracked. Show confirm dialogs will be displayed when going back, closing or opening an existing chart when the current chart has been changed. The chart title will display an * when unsaved changes have been applied to the chart.
- Use the Data field to control which data to include in the chart.
- First Row Contains Headers is selected by default. This option should be selected if the first row in the data range contains column headings.
- Use the icons on the top right of the chart to :



  - Collapse the current chart while keeping it open.
  - Copy the current chart to the clipboard.

    Note: Mozilla's Firefox web browser uses a default configuration that does not allow images to be copied to the clipboard. This configuration may be changed by navigating to *about:config*, within Firefox, and setting *dom.events.asyncClipboard.clipboardItem* to *True*.
  - Sent the current chart to the printer.

    Note: If using Mozilla's Firefox web browser, "Print backgrounds" must be selected in the Print Options pane, as shown in the screenshot below.



Pages per sheet

1

Margins

Default

Options

☐ Print headers and footers

☑ Print backgrounds

Print using the system dialog...

- Close the current chart window.

- To exit the graph, click the X in the upper right hand corner of the Chart Wizard window.
- Note: Categorical variables are avariables with less than or equal to 12 unique values.

Box Whisker Plot Example

This example describes the use of the Boxplot chart to illustrate the characteristics of the dataset.

Click **Help – Example Models** on the Data Science ribbon to open the **BoxPlot.xlsx** example file under *Forecasting/ Data Science Examples.*

Select on a cell within the data (say A2), then click **Explore – Chart Wizard** to bring up the first dialog of the Chart Wizard. Select **BoxPlot**.

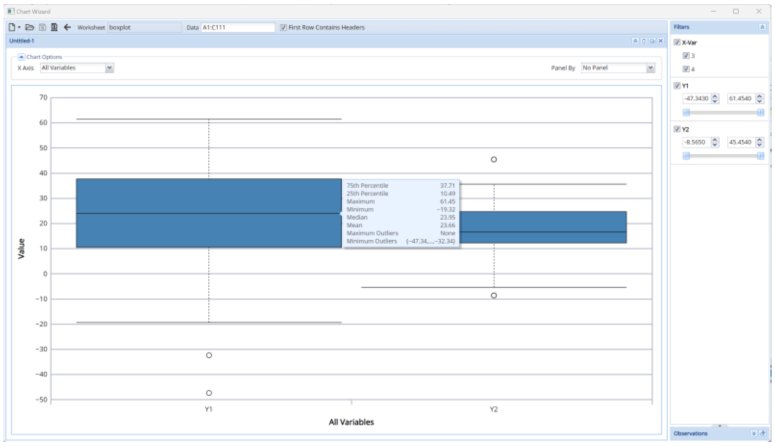


Chart Wizard

Worksheet: boxplot Data: A1:C111 ☑ First Row Contains Headers

Untitled-1

Chart Options

X Axis: All Variables Panel By: No Panel

Filters

☑ X-Var

☑ 3

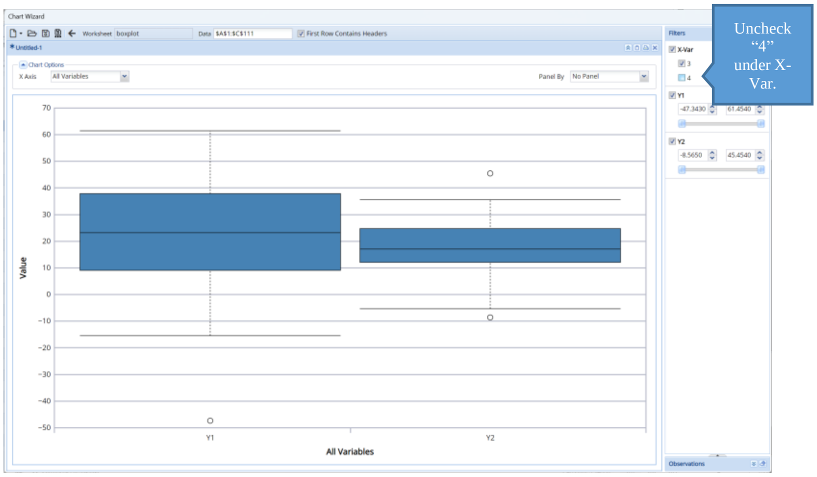
☑ 4

☑ Y1 -47.3430 61.4540

☑ Y2 -8.5650 45.4540

75th Percentile	37.71
25th Percentile	10.49
Maximum	61.45
Minimum	−19.32
Median	23.95
Mean	23.66
Maximum Outliers	None
Minimum Outliers	{−47.34,...,−32.34}

Uncheck class 4 under the X-Var filter to remove this class from the plot.

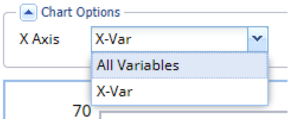


Uncheck "4" under X-Var.

X Axis: All Variables

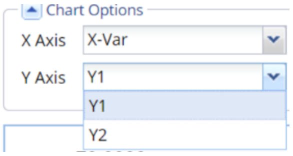
Panel By: No Panel

- To select a different variable on the X-axis, click the down arrow net to X-Axis and select the desired variable from the menu, say X-Var.



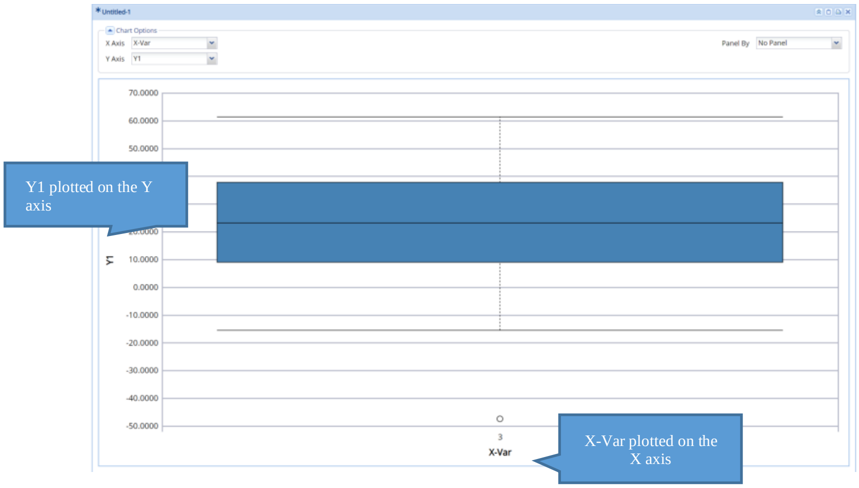
X Axis: X-Var

- Once a specific variable is selcted for X Axis, a new menu, Y Axis, appears. Click the down arrow next to Y Axis and select Y1 from the menu. The graph changes to display the range of values in the Y1 column for X variable 3.



X Axis: X-Var

Y Axis: Y1



Y1 plotted on the Y axis

X-Var plotted on the X axis

X Axis: X-Var

Y Axis: Y1

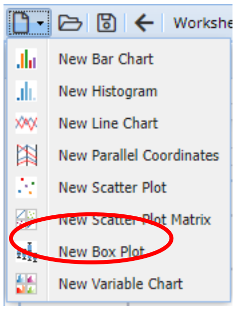
Panel By: No Panel

The solid line denotes the Median. The box reaches from the 25th Percentile to the 75th Percentile. The upper "whiskers" denote the extreme minimum and maximum values, excluding outliers. In this example, there is one outlier below the minimum "whisker". Outliers are values that fall outside of the ranges:

- Min: 25th Percentile – 1.5 * IQR – Any data points falling below this limit is considered an outlier. In this example, there is one outlier, shown above.

- Max: 75th Percentile + 1.5 * IQR – Any data points appearing above this limit is considered an outlier.

To add a 2nd boxplot, click the New File icon, top, left, and select New Box Plot from the menu.



A second chart is added to the Chart Wizard dialog.

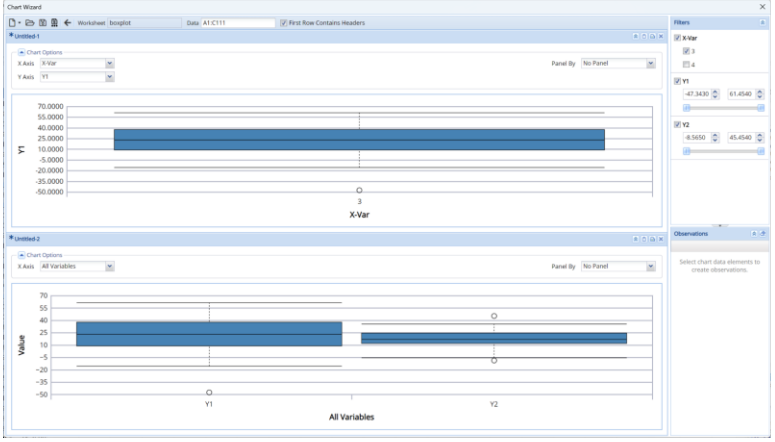


Chart Wizard

Worksheet boxplot   Data A1:C111   ☑ First Row Contains Headers

*Untitled-1

Chart Options

X Axis X-Var

Y Axis Y1

Panel By No Panel

*Untitled-2

Chart Options

X Axis All Variables

Panel By No Panel

Filters

☑ X-Var

☑ 3

☐ 4

☑ Y1

-47.3430   61.4540

☑ Y2

-8.5650   45.4540

Observations

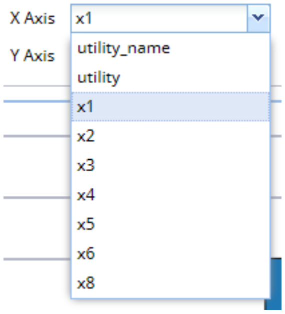
Select chart data elements to create observations.

Note that the Filters pane applies to *both* charts. The bottom graph charts the Y1 and Y2 values for (only) X variable 3.

Histogram Example

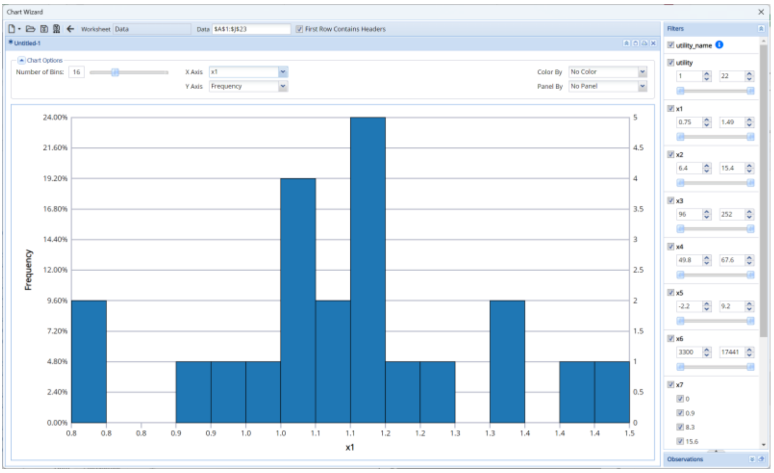
The example below illustrates the use of Analytic Solver Data Science’s chart wizard in drawing a histogram of the Utilities dataset. Click **Help – Example Models** on the Data Science ribbon to open the example dataset, **Utilities.xlsx** under *Forecasting/Data Science Examples*. Select a cell within the dataset, say A2, and then click **Explore – Chart Wizard** on the Data Science ribbon. Select **Histogram**, and then click **Next**.

- On the opening chart, select X1 for the X Axis to create a histogram of the values for the x1 variable.



X Axis: x1

Move the slider right and left to increase or decrease the number of bins in the chart.



Number of Bins: 16

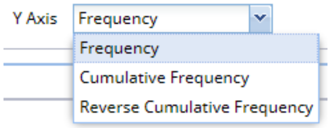
X Axis: x1

Y Axis: Frequency

Color By: No Color

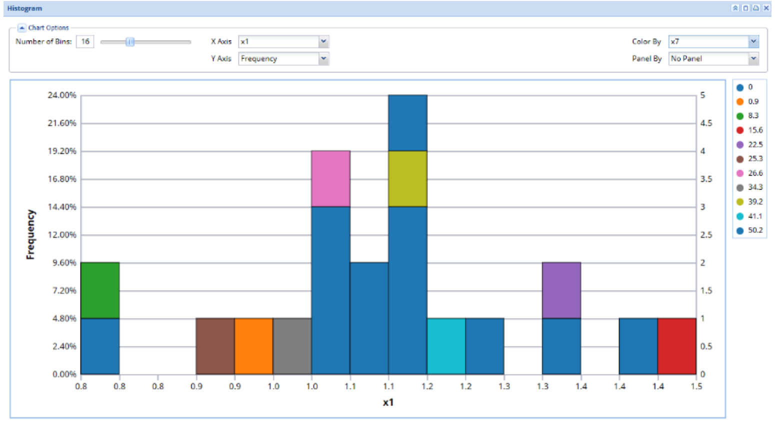
Panel By: No Panel

- Number of Bins: Move the slider right and left to right and left to increase or decrease the number of bins in the chart.
- For the Y Axis, select Frequency, Cumulative Frequency or Reverse Cumulative Frequency.



Y Axis: Frequency

Set Color By to x7. This new chart indicates the corresponding value for x7 for each utility. For example, the 5th bin consists of 1 value, 0.96 for the Pacific utility. The color of this bin indicates that the value of x7 for this specific utility is 0.9.



Number of Bins: 16

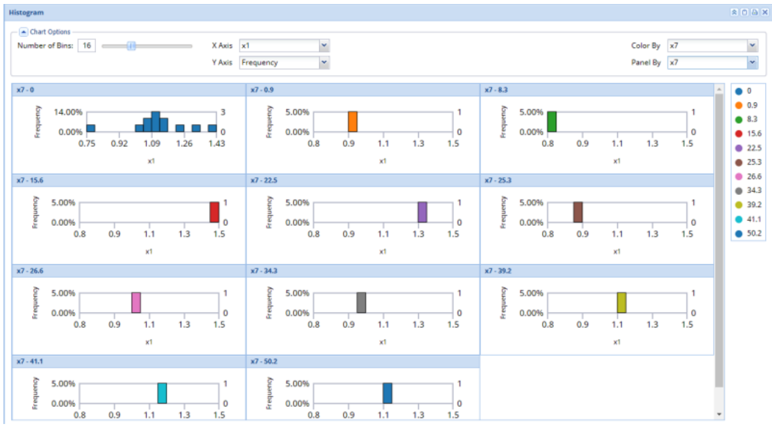
X Axis: x1

Y Axis: Frequency

Color By: x7

Panel By: No Panel

Set Panel by to X7. This chart creates 11 different graphs; one graph for each of the values for x7: 0, 0.9, 8.3, 15.6, 22.5, 25.3, 26.6, 34.3, 39.2, 41.1, 50.2. Notice that labels for each of the graphs, i.e. x7-0, x7-0.9, x7 – 8.3, etc. Each chart plots the corresponding value for the x1 variable, for example the SanDiego utility has a value of 8.3 for x7. The value for x1 (0.76) is plotted in the x7-8.3 chart.



Histogram

Chart Options

Number of Bins: 16

X Axis: x1

Y Axis: Frequency

Color By: x7

Panel By: x7

Line Chart Example

The example below illustrates the use of Analytic Solver Data Science's chart wizard in drawing a Line Chart using the Airpass.xlsx dataset. Click **Help – Example Models** on the Data Science ribbon to open the example dataset, **Airpass.xlsx** under *Forecasting/Data Science Examples*. Select a cell within the dataset, say A2, and then click **Explore – Chart Wizard** on the Data Science ribbon, then select **Line Chart**.

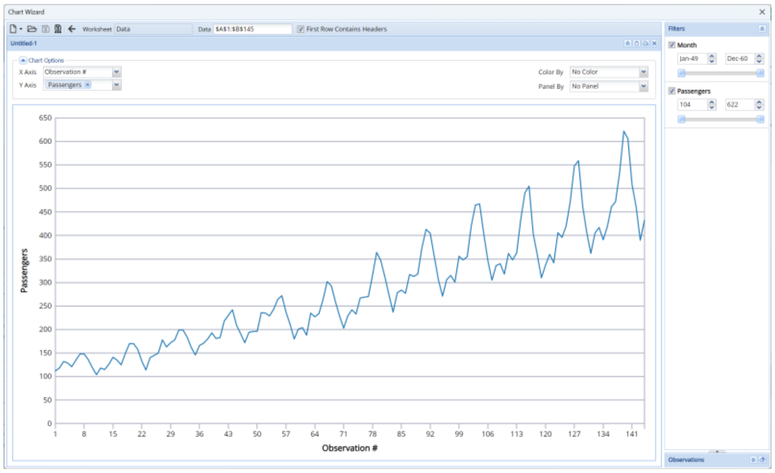


Chart Wizard

Worksheet: Data

Data: $A$1:$B$145

☑ First Row Contains Headers

Untitled-1

Chart Options

X Axis: Observation #

Y Axis: Passengers

Color By: No Color

Panel By: No Panel

Filters

☑ Month: Jan-49 to Dec-60

☑ Passengers: 104 to 622

The y-axis plots the number of passengers and the x-axis plots the month. This plot shows that as the months progress, the number of airline passengers appear to be in an increasing trend with yearly seasonality dips.

Parallel Coordinates Chart Example

The example below illustrates the use of Analytic Solver Data Science's chart wizard in drawing a Parallel Coordinates Plot using the dataset.

A parallel coordinates plot allows the exploration of high dimensional datasets, or datasets with a large number of features (variables). This type of graph starts with a set of vertically drawn parallel lines, equally spaced, which corresponds to the features included in the graph. Observations for each feature are recorded as dots on the vertical line. Observations that are contained within the same record are connected by a line.

Click **Help – Example Models** on the Data Science ribbon to open the example dataset, **SportsTVRatings.xlsx** under *Forecasting/Data Science Examples*. Select a cell within the dataset, say A2, then click **Explore – Chart Wizard** on the Data Science ribbon. Select **Parallel Coordinates**.

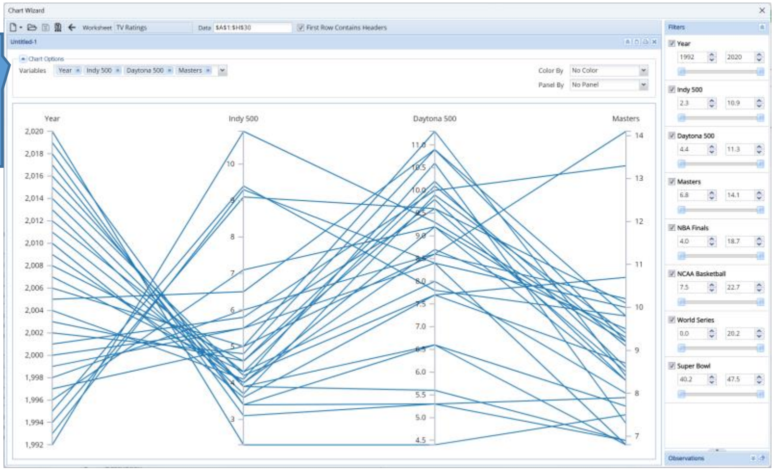


Chart Wizard

Worksheet: TV Ratings Data: $A$1:$H$30 ☑ First Row Contains Headers

Untitled-1

Chart Options
Variables: Year × Indy 500 × Daytona 500 × Masters ×
Color By: No Color
Panel By: No Panel

Filters
☑ Year 1992 2020
☑ Indy 500 2.3 10.9
☑ Daytona 500 4.4 11.3
☑ Masters 6.8 14.1
☑ NBA Finals 4.0 18.7
☑ NCAA Basketball 7.5 22.7
☑ World Series 0.0 20.2
☑ Super Bowl 40.2 47.5

Remove Year and Masters by clicking the “x” next to each variable under Variable Options.

Remove all variables from the chart except **Indy500** and **Daytona 500**.

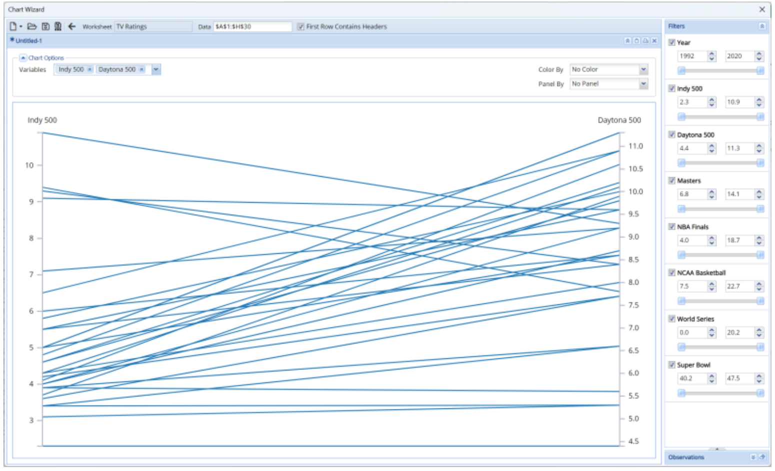


Chart Wizard

Worksheet: TV Ratings Data: $A$1:$H$30 ☑ First Row Contains Headers

* Untitled-1

Chart Options
Variables: Indy 500 × Daytona 500 ×
Color By: No Color
Panel By: No Panel

Filters
☑ Year 1992 2020
☑ Indy 500 2.3 10.9
☑ Daytona 500 4.4 11.3
☑ Masters 6.8 14.1
☑ NBA Finals 4.0 18.7
☑ NCAA Basketball 7.5 22.7
☑ World Series 0.0 20.2
☑ Super Bowl 40.2 47.5

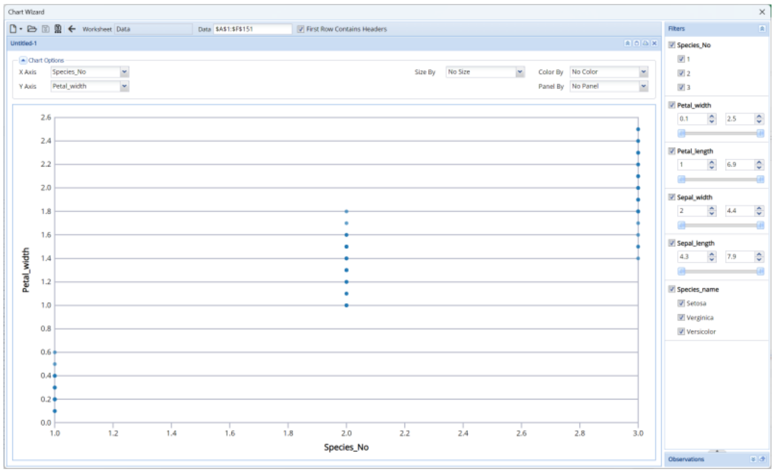
The first thing that we notice is the range of each of the races that are indicated at the top and bottom of each vertical line. The range of ratings for the Indy 500 has a high of 10.9 and a low of 2.3 whereas the range for the Daytona 500 is 11.3 to 4.4. As a result, this chart already conveys that the viewership for the Daytona 500 is larger than the viewship for the Indy 500.

When looking at the observations for each feature, this chart shows that in most years, the viewship of the Indy 500 was low whereas the viewship for the Daytona 500 was high. There are just four years wheree high ratings for the Indy 500 was recorded. In these same years, the ratings for the Daytona 500 was correspondingly lower.

Note: For the parallel coordinates chart, when all values of a category are numeric, the y-axis tick values are determined using D3 fixed precision. This shortens the axis tick labels by rounding values using a decimal precision automatically selected for the data set. If for some reason the user requires more precision, the column can be formatted using a number formatter in the grid view before creating the chart.

ScatterPlot Example

The example below illustrates the use of Analytic Solver Data Science's chart wizard in drawing a Scatterplot using the Iris.xlsx dataset. Click **Help – Example Models** on the Data Science ribbon to open the example dataset, **Iris.xlsx** from *Forecasting/Data Science Examples*. Select a cell within the dataset, say A2, and then click **Explore – Chart Wizard** on the Data Science ribbon. Select Scatter Plot.



Worksheet: Data

Data: $A$1:$F$151

☑ First Row Contains Headers

X Axis: Species_No

Y Axis: Petal_width

Size By: No Size

Color By: No Color

Panel By: No Panel

Filters:

☑ Species_No: ☑ 1 ☑ 2 ☑ 3

☑ Petal_width: 0.1 – 2.5

☑ Petal_length: 1 – 6.9

☑ Sepal_width: 2 – 4.4

☑ Sepal_length: 4.3 – 7.9

☑ Species_name: ☑ Setosa ☑ Verginica ☑ Versicolor

Note that the graph shows that Iris flowers with small, medium or large pedal widths belong to the same Iris species.

Click the down arrow next to X Axis (at the top, left) and select Petal_length from the menu.

Notice that Iris flowers with small, medium or large pedal widths generally have the same (small, medium or large) pedal lengths.

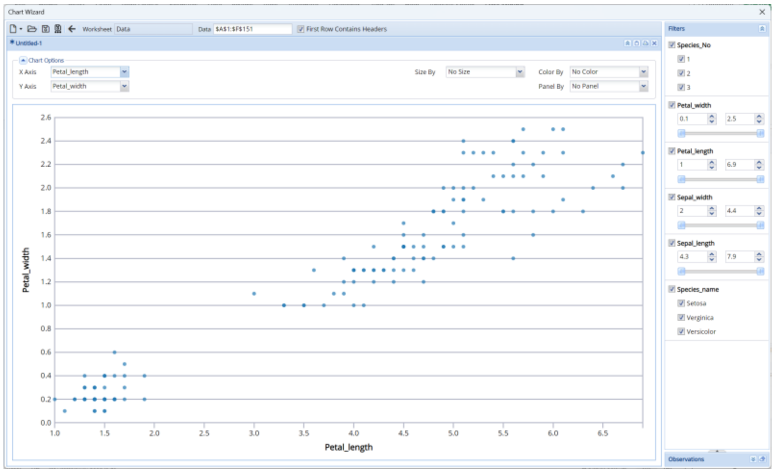


Chart Wizard

Worksheet Data

Data $A$1:$F$151

☑ First Row Contains Headers

Untitled-1

Chart Options

X Axis Petal_length

Y Axis Petal_width

Size By No Size

Color By No Color

Panel By No Panel

Filters

☑ Species_No

☑ 1

☑ 2

☑ 3

☑ Petal_width 0.1 2.5

☑ Petal_length 1 6.9

☑ Sepal_width 2 4.4

☑ Sepal_length 4.3 7.9

☑ Species_name

☑ Setosa

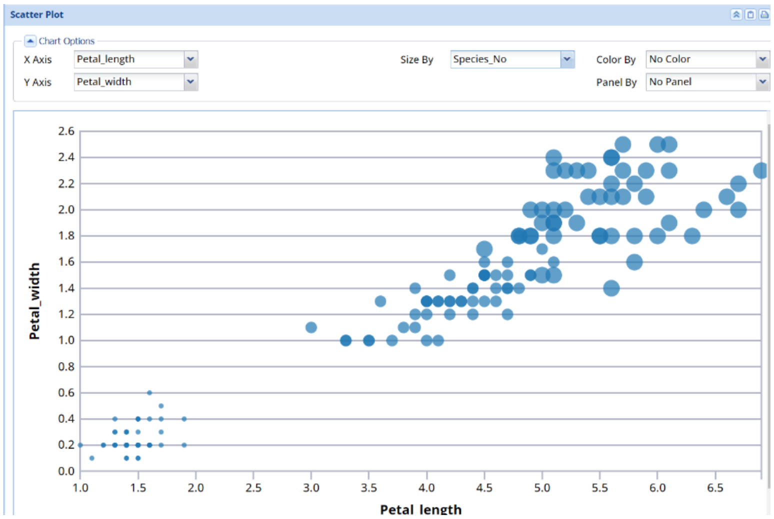
☑ Verginica

☑ Versicolor

Observations

Click the down arrow next to Size By and select Species_No. Now the chart clearly shows which data points on the chart belong to each species.

This chart clearly shows that, out of the three varieties of iris flowers, the Setosa variety has the smallest petal widths and lengths, the Verginica has the latest petal widths and lengths and the Versicolor variety is in between the Setosa and Verginica.



Scatter Plot

Chart Options

X Axis Petal_length

Y Axis Petal_width

Size By Species_No

Color By No Color

Panel By No Panel

Alternatively, select the down arrow next to Color By and select Species_No to color each data point by species type.

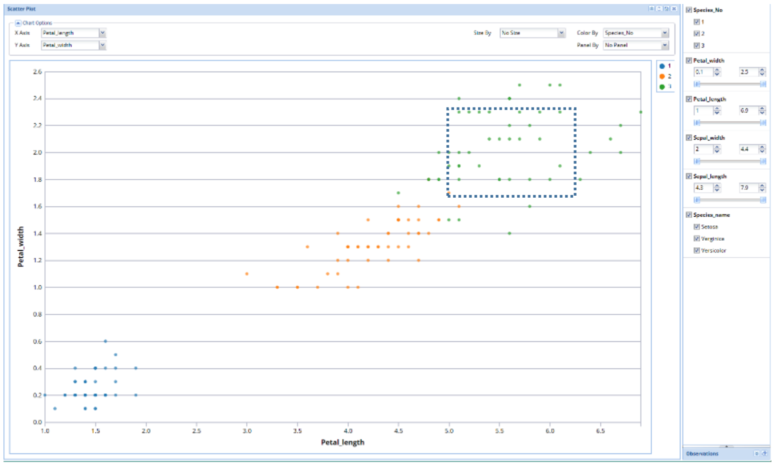

Scatter Plot

Chart Options

X Axis: Petal_length

Y Axis: Petal_width

Size By: No Size

Color By: Species_No

Panel By: No Panel

☑ Species_No
- ☑ 1
- ☑ 2
- ☑ 3

☑ Petal_width: 0.1 – 2.5

☑ Petal_length: 1 – 6.9

☑ Sepal_width: 2 – 4.4

☑ Sepal_length: 4.3 – 7.9

☑ Species_name
- ☑ Setosa
- ☑ Verginica
- ☑ Versicolor

Click Observations on the bottom right to open the Observations pane. Then use the mouse to draw a square in the middle of the data points. The data points contained within the square immediately appear in the Observations pane. Move the bottom scroll bar to the right to view all data for each record in the dataset.

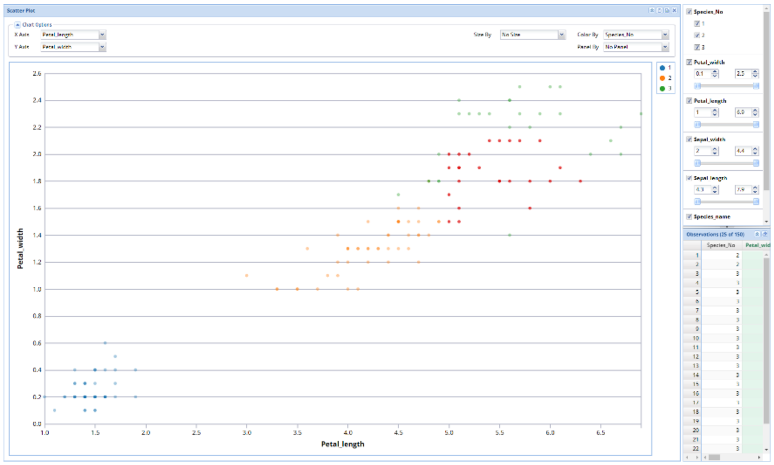

Scatter Plot

Chart Options

X Axis: Petal_length

Y Axis: Petal_width

Size By: No Size

Color By: Species_No

Panel By: No Panel

☑ Species_No
- ☑ 1
- ☑ 2
- ☑ 3

☑ Petal_width: 0.1 – 2.5

☑ Petal_length: 1 – 6.9

☑ Sepal_width: 2 – 4.4

☑ Sepal_length: 4.3 – 7.9

☑ Species_name

Observations (25 of 150)

	Species_No	Petal_wid
1	2	
2	2	
3	3	
4	3	
5	3	
6	3	
7	3	
8	3	
9	3	
10	3	
11	3	
12	3	
13	3	
14	3	
15	3	
16	3	
17	3	
18	3	
19	3	
20	3	
21	3	
22	3	

Scatterplot Matrix Plot Example

The example below illustrates the use of Analytic Solver Data Science’s chart wizard in drawing a Scatterplot Matrix using the Iris.xlsx dataset. Click **Help – Example Models** on the Data Science ribbon to open the example dataset, **Iris.xlsx**. Select a cell within the dataset, say A2, then click **Explore – Chart Wizard** on the Data Science ribbon. Select **Scatterplot Matrix**.

Click the “x” to remove the variable from the chart.

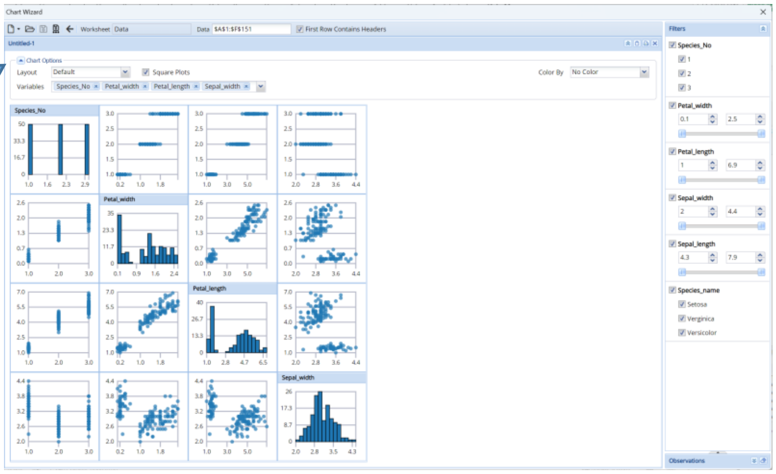


Chart Wizard

Worksheet Data   Data $A$1:$F$151   ☑ First Row Contains Headers

Layout Default   ☑ Square Plots   Color By No Color

Variables Species_No × Petal_width × Petal_length × Sepal_width ×

Histograms of all variables appear on the diagonal. To remove a variable from the matrix, next to Variables (top, left), click the “x” next to the variable to be removed.   To add a variable to the matrix, click the down arrow and select the desired variable.

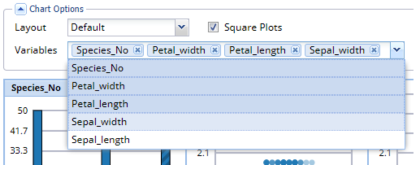


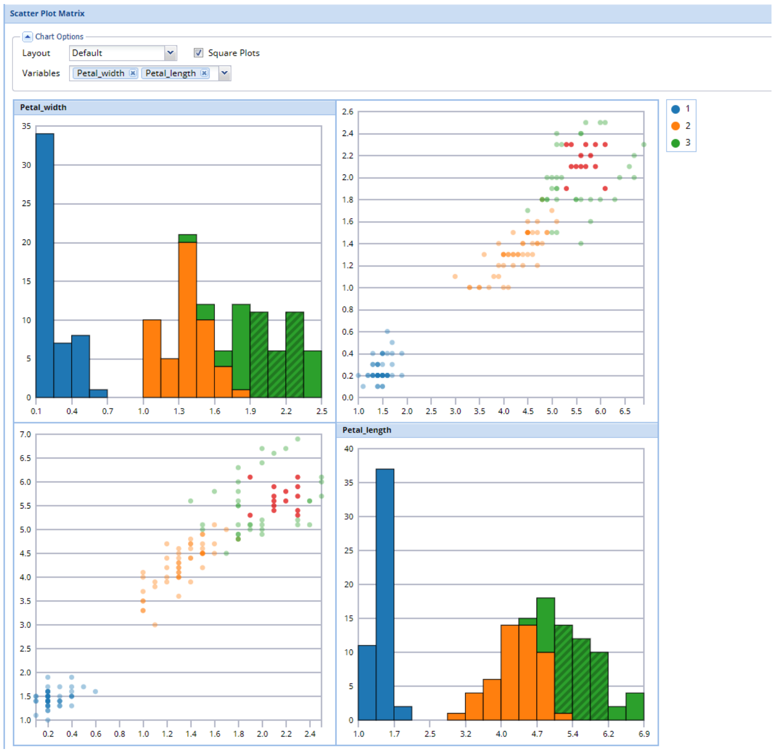
Chart Options

Layout Default   ☑ Square Plots

Variables Species_No × Petal_width × Petal_length × Sepal_width ×

Graphs on either side of the diagonal represent a pairwise relationship between all variables. For example, the graph in the 2nd column, 1st row, shows the relationship between Species_no and Petal_width. The Y-axis for either plot can be found on the far left of the chart and the X-axis for either plot can be found at the far bottom.

Remove all variablees except Petal_width and Petal_length, then use the Color By feature to easily see the petal characterisitics of each species of iris.



Scatter Plot Matrix

Chart Options

Layout: Default

☑ Square Plots

Variables: Petal_width × Petal_length ×

Variable Plot Example

The example below illustrates the use of Analytic Solver Data Science's chart wizard in drawing a Variable plot using the Universal Bank Main.xlsx dataset. Click **Help – Example Models** on the Data Science ribbon to open the example dataset, **Universal Bank Main.xlsx**. Select a cell within the dataset, say A2, and then click **Explore – Chart Wizard** on the Data Science ribbon. Select **Variable Chart**.

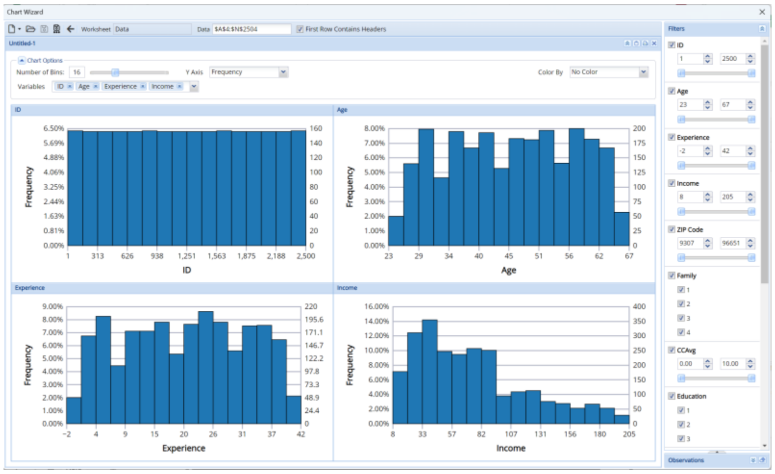


Chart Wizard

Worksheet: Data Data: $A$4:$N$2504 ☑ First Row Contains Headers

Untitled-1

Chart Options

Number of Bins: 16 Y Axis: Frequency Color By: No Color

Variables: ID × Age × Experience × Income ×

Filters

☑ ID: 1 – 2500

☑ Age: 23 – 67

☑ Experience: -2 – 42

☑ Income: 8 – 205

☑ ZIP Code: 9307 – 96651

☑ Family: ☑ 1 ☑ 2 ☑ 3 ☑ 4

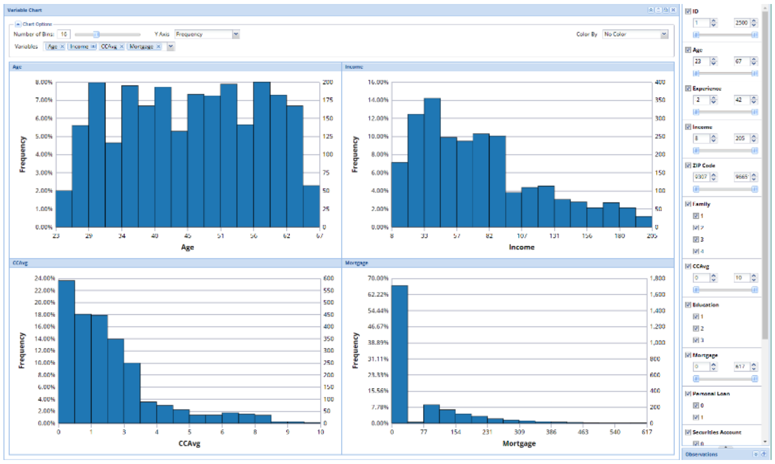
☑ CCAvg: 0.00 – 10.00

☑ Education: ☑ 1 ☑ 2 ☑ 3

The first four continous varaibles in the dataset are included in the chart by default. (Note: Only continuous variables may be included in the Variable chart.)

The distributions of each variable are shown in separate histograms. To remove a variable from the matrix, click the “x” next to each included variable at the top of the chart. To add a variable to the matrix, click the down arrow next to Variables and select the desired variable under from the menu.

Remove Frequency and ID variables from the chart, leaving Age and Income. Then add the CCAvg and Mortgage variables to the chart. From looking at these charts, a user could immediately distinguish the range of values for each continuous variable included in the chart.



Variable Chart

Chart Options

Number of Bins: 10

Y Axis: Frequency

Color By: No Color

Variables: Age × Income × CCAvg × Mortgage ×

☑ ID 1 2500

☑ Age 23 67

☑ Experience 2 42

☑ Income 8 205

☑ ZIP Code 9307 9665

☑ Family
☑ 1
☑ 2
☑ 3
☑ 4

☑ CCAvg 0 10

☑ Education
☑ 1
☑ 2
☑ 3

☑ Mortgage 0 617

☑ Personal Loan
☑ 0
☑ 1

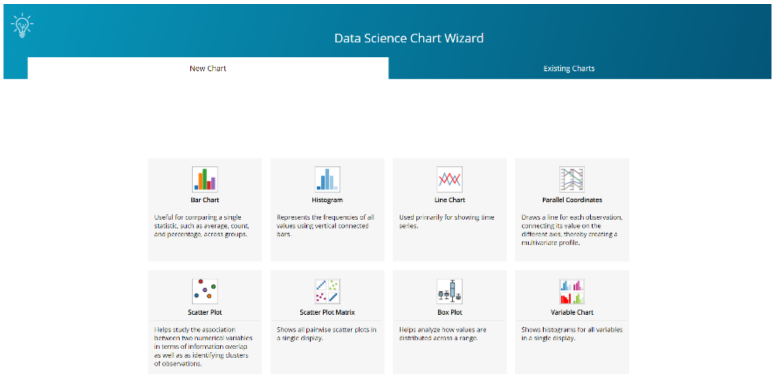
☑ Securities Account
☑ 0

Observations

Data Science Chart Options

See below for documentation on all chart options.

Click Explore – Chart Wizard to open the Chart Wizard.



Bar Chart

Useful for comparing a single statistic, such as average, count, and percentage, across groups.

Histogram

Represents the frequencies of all values using vertical connected bars.

Line Chart

Used primarily for showing time series.

Parallel Coordinates

Draws a line for each observation, connecting its value on the different axis, thereby creating a multivariate profile.

Scatter Plot

Helps study the association between two numerical variables in terms of information overlap as well as as identifying clusters of observations.

Scatter Plot Matrix

Shows all pairwise scatter plots in a single display.

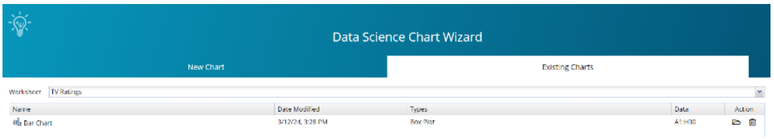
Box Plot

Helps analyze how values are distributed across a range.

Variable Chart

Shows histograms for all variables in a single display.

To open a previously saved chart, click the Existing Charts tab and then select the desired chart.

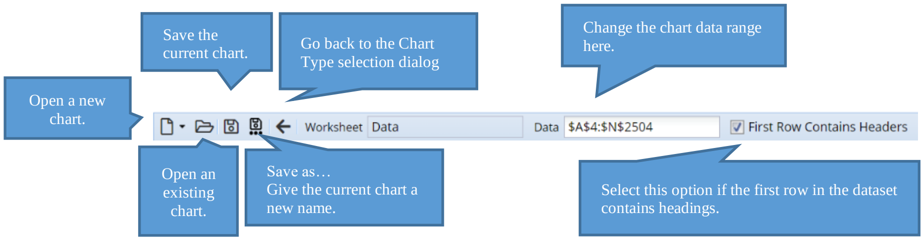


Worksheet: TV Ratings

Name	Date Modified	Types	Data	Action
Bar Chart	3/12/24, 3:28 PM	Box Plot	A1:H30	

Charts are saved by worksheet. Click the down arrow next to Worksheet to view all previously saved charts.

The following bar appears at the top of each chart. From here, you can open a new or existing chart, save the current chart, go back to the chart selection dialog or change the data range.



Save the current chart.

Go back to the Chart Type selection dialog

Change the chart data range here.

Open a new chart.

Worksheet Data

Data $A$4:$N$2504

☑ First Row Contains Headers

Open an existing chart.

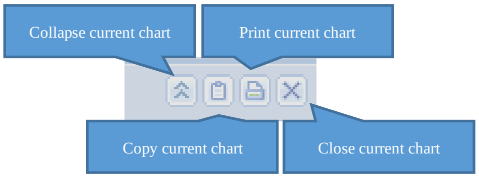
Save as… Give the current chart a new name.

Select this option if the first row in the dataset contains headings.

Note on data selection: When a single cell is selected that is located inside or adjacent to a dataset, the entire dataset will be used for the "Data" cell range. If the user has selected multiple cells in a single rectangular region, those cells, excluding any bounding empty cells, will be used for the "Data" cell range. This behavior was added to allow users to create a chart on a subset of a worksheet dataset without the User being forced to edit the Data address after the chart has been created.

Note on chart changes: Chart changes are tracked. Show confirm dialogs will be displayed when going back, closing or opening an existing chart when the current chart has been changed. The chart title will display an * when unsaved changes have been applied to the chart.

Use the icons on the top right of the chart to :



Collapse current chart

Print current chart

Copy current chart

Close current chart

- Collapse the current chart while keeping it open.
- Copy the current chart to the clipboard.

  Note: Mozilla's Firefox web browser uses a default configuration that does not allow images to be copied to the clipboard. This configuration may be changed by navigating to *about:config*, within Firefox, and setting *dom.events.asyncClipboard.clipboardItem* to *True*.
- Sent the current chart to the printer.
- Close the current chart window

Chart Options Pane

The following options may appear within the Chart Options pane in each chart.

The following option appears only with bar charts.

Bar Type: Select the Bar Type: Vertical or Horizontal. This option only appears when a bar chart is created.

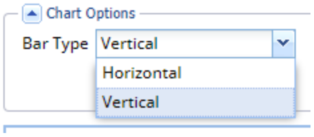
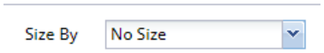


Chart Options

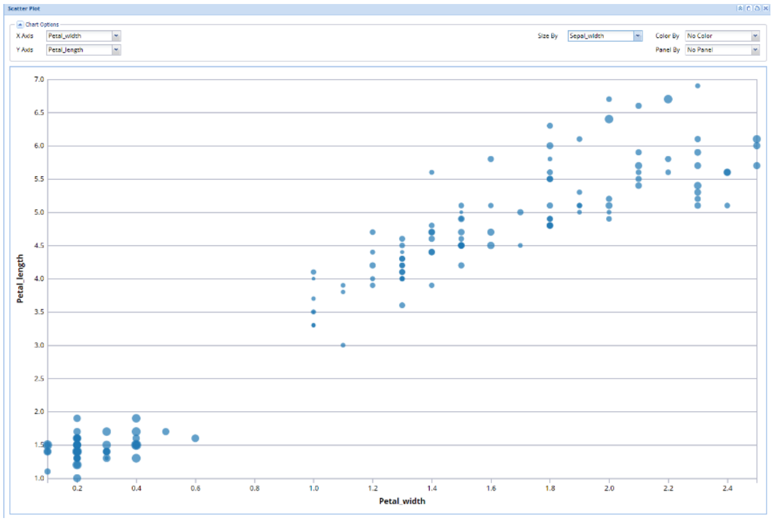
Bar Type: Vertical

The following option appears only with Scatterplot Matrix charts.

Size By: Click the down arrow to select a variable to determine the size of the points on the chart. In the chart below, for example, the size of the dots are determined by the size of the Sepal_width for each record in the dataset. This option appears in a scatter plot.



Size By No Size



Scatter Plot

Chart Options

X Axis Petal_width

Y Axis Petal_length

Size By Sepal_width

Color By No Color

Panel By No Panel

The following two options appear on Scatterplot Matrix charts.

Layout: Choose Fit to Width to fit the chart to the width of the chart dialog area. Select Fit to Page to fit all scatter plots onto the chart dialog area.

Square Plots: Select this option to generate square scatter plots. Uncheck this option to generate rectangular scatter plots.

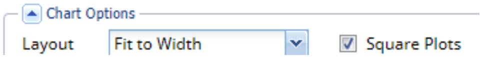


Chart Options

Layout Fit to Width ☑ Square Plots

The following option appears on two chart types: parallel coordinates, scatterplot matrix and variable.

Variable Tag Field: Click the down arrow to select a variable to include in the chart. Click the x in the upper right of the variable tab to remove the variable from the chart.

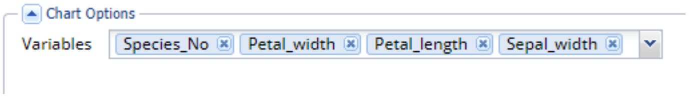


Chart Options

Variables Species_No ☒ Petal_width ☒ Petal_length ☒ Sepal_width ☒

The following option appears on histogram and variable charts.

Number of Bins: Use the slider to increase or decrease the number of bins.

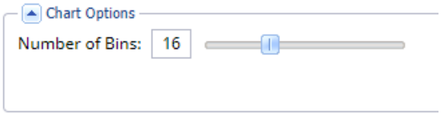


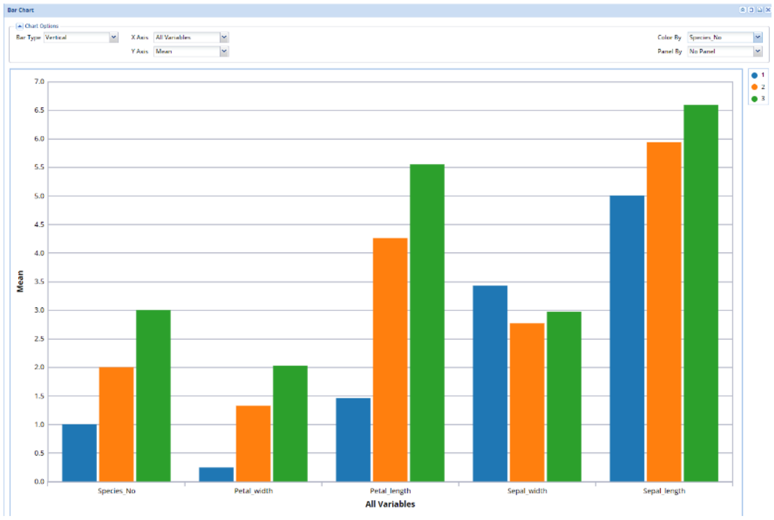
Chart Options

Number of Bins: 16

The following options appear on the following chart types: bar, box plot, histogram, line, scatter plot and variable.
X Axis: Click the down arrow select the variable(s) to appear on the x-axis.
YAxis: Click the down arrow to select the metric to appear on the y-axis.

The following options appear on all charts.

Color By: If the dataset contains categorical variables, click the down arrow next to Color By to display the data by category. A different color will be applied to each category, as shown below.

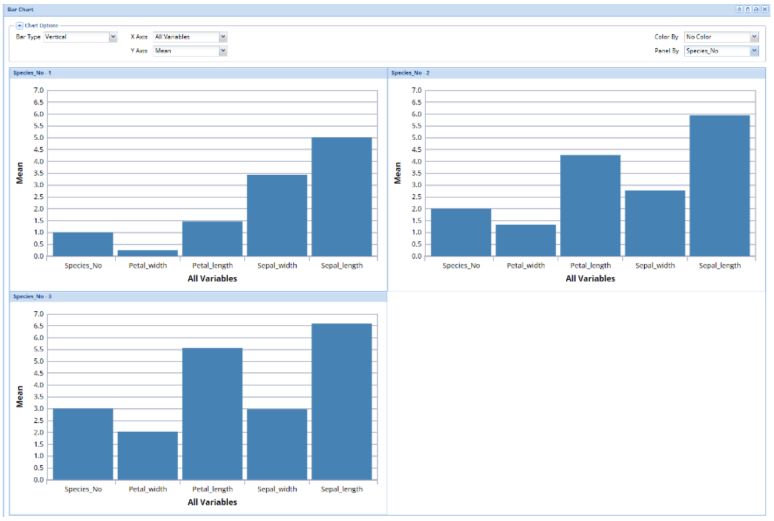


Bar Chart

Chart Options

Bar Type: Vertical — X Axis: All Variables — Y Axis: Mean — Color By: Species_No — Panel By: No Panel

Panel By: If the dataset contains categorical variables, click the down arrow next to Panel By to display the data by category in separate charts. Each category will be displayed in a separate chart, as shown below.



Bar Chart

Chart Options

Bar Type: Vertical — X Axis: All Variables — Y Axis: Mean — Color By: No Color — Panel By: Species_No

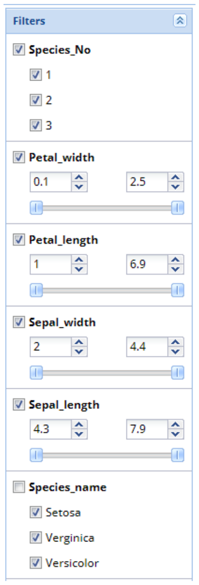
Filters Pane and Data Requirements

Use the Filters pane to add and remove variables when a Variable Tag Field is not present. (For more information on the Variable Tag Field, see below.

Uncheck the variable to remove from the chart. Select to include in the chart. The data range may be altered by clicking the up an down buttons on the spinner fields or by moving the sliders left (to decrement) or right (to increment).

Missing Data, Text and Error Values

- If a column contains numeric data, missing values are allowed but no error values, i.e. #Value, #N/A, etc.
- When there are <= 12 unique values in a column, that column is treated as categorical.
- Data can be text if there are <= 12 unique values.
- If text appears in a column containing > 12 unique values, the entire column is treated as text.



Filters

☑ Species_No

☑ 1

☑ 2

☑ 3

☑ Petal_width

0.1 2.5

☑ Petal_length

1 6.9

☑ Sepal_width

2 4.4

☑ Sepal_length

4.3 7.9

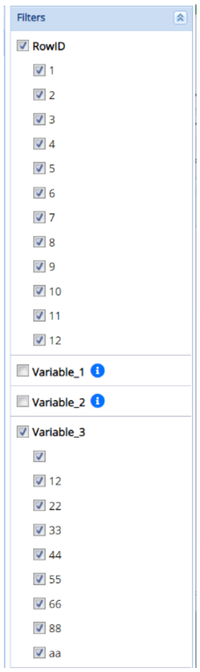
□ Species_name

☑ Setosa

☑ Verginica

☑ Versicolor

Note that variables with error data (#NA, #DIV/0, #NAME, etc.) are flagged in the filters list and excluded from chart elements.



Filters

☑ RowID

☑ 1

☑ 2

☑ 3

☑ 4

☑ 5

☑ 6

☑ 7

☑ 8

☑ 9

☑ 10

☑ 11

☑ 12

□ Variable_1 ℹ

□ Variable_2 ℹ

☑ Variable_3

☑

☑ 12

☑ 22

☑ 33

☑ 44

☑ 55

☑ 66

☑ 88

☑ aa

Transforming Datasets with Missing or Invalid Data

Introduction

Analytic Solver Data Science's Missing Data Handling utility allows users to detect missing values in the dataset and handle them in a way you specify. Analytic Solver Data Science considers an observation to be *missing* if the cell is empty or contains an invalid formula. Analytic Solver Data Science also allows you to indicate specific data that you want designated as "missing" or "corrupt".

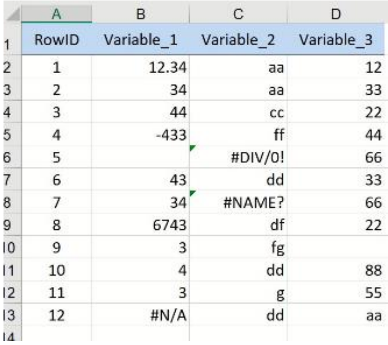
Analytic Solver Data Science offers several different methods for dealing with missing values. Each variable can be assigned a different "treatment". For example, if there is a missing value, then the entire record could be deleted or the missing value could be replaced by an estimated mean/median/mode of the bin or even with a value that you specify. The available options depend on the variable type.

In the following examples, we will explore the various ways in which Analytic Solver Data Science can treat missing or invalid values in a dataset.

Missing Data Handling Examples

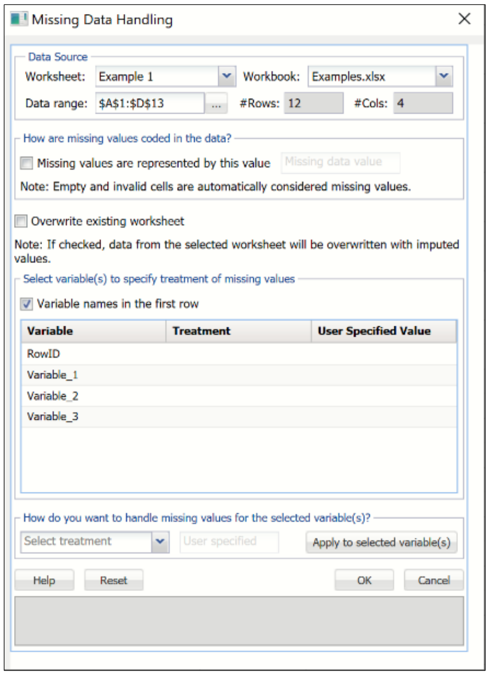
To open the Examples.xlsx workbook, click **Help – Example Models** on the Desktop or AnalyticSolver.com ribbon, click **Forecasting/Data Science Examples**, and open the dataset, **Examples**.

This workbook contains six worksheets containing small sample datasets. The Example 1 dataset contains empty cells (cells B6 and D10), cells containing invalid formulas (B13, C6, & C8), cells containing non numeric characters (D13), etc. Analytic Solver Data Science will treat each of these as missing values.



	A	B	C	D
1	RowID	Variable_1	Variable_2	Variable_3
2	1	12.34	aa	12
3	2	34	aa	33
4	3	44	cc	22
5	4	-433	ff	44
6	5		#DIV/0!	66
7	6	43	dd	33
8	7	34	#NAME?	66
9	8	6743	df	22
10	9	3	fg	
11	10	4	dd	88
12	11	3	g	55
13	12	#N/A	dd	aa
14				

Open the Missing Data Handling dialog by clicking Data Science – Transform – Missing Data Handling. Confirm that *Example 1* is displayed for *Worksheet.*



Missing Data Handling

Data Source

Worksheet: Example 1 Workbook: Examples.xlsx

Data range: $A$1:$D$13 #Rows: 12 #Cols: 4

How are missing values coded in the data?

□ Missing values are represented by this value Missing data value

Note: Empty and invalid cells are automatically considered missing values.

□ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

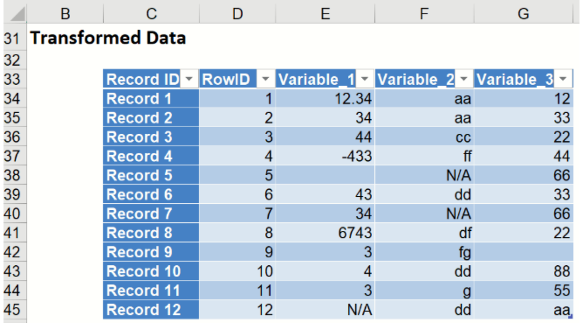
Select variable(s) to specify treatment of missing values

☑ Variable names in the first row

Variable	Treatment	User Specified Value
RowID		
Variable_1		
Variable_2		
Variable_3		

How do you want to handle missing values for the selected variable(s)?

Click **OK**. The results of the data transformation are inserted into the Imputation worksheet. Since no treatment was specified for any of the variables, none of the missing or invalid values were replaced.

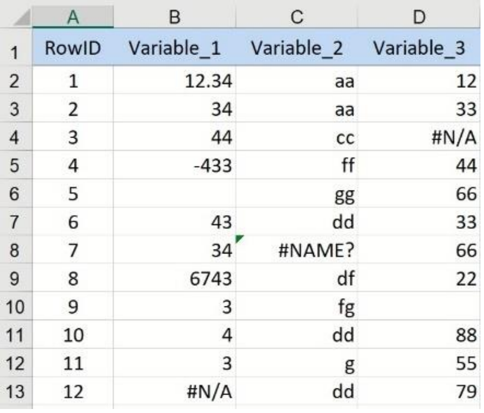


Transformed Data

Record ID	RowID	Variable_1	Variable_2	Variable_3
Record 1	1	12.34	aa	12
Record 2	2	34	aa	33
Record 3	3	44	cc	22
Record 4	4	-433	ff	44
Record 5	5		N/A	66
Record 6	6	43	dd	33
Record 7	7	34	N/A	66
Record 8	8	6743	df	22
Record 9	9	3	fg	
Record 10	10	4	dd	88
Record 11	11	3	g	55
Record 12	12	N/A	dd	aa

If "Overwrite existing worksheet" is selected in the Missing Data Handling dialog, Analytic Solver will overwrite the existing data with the treatment option specified. Note: You must *save* the workbook in order for these changes to be saved.

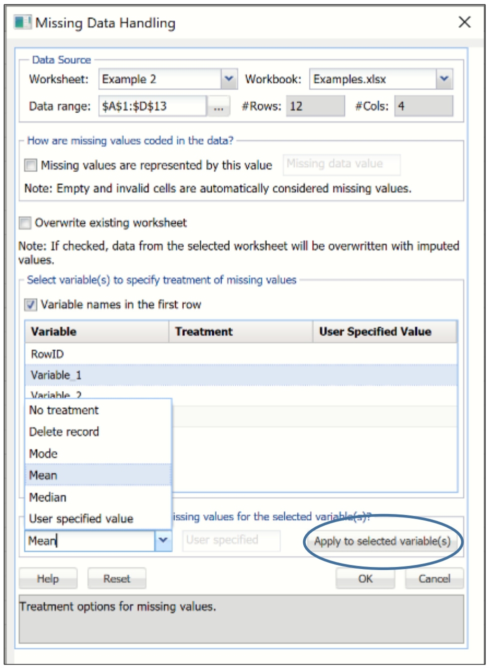
The Example 2 dataset is similar to the Example 1 dataset in that this dataset contains empty cells (cells B6 and D10), cells containing invalid formulas (B13, C8 & D4), cells containing non numeric characters (column C), etc. In this example we will see how the missing values in the Variable_1 and Variable_3 columns can be replaced by Mean and Median, respectively.



	A	B	C	D
1	RowID	Variable_1	Variable_2	Variable_3
2	1	12.34	aa	12
3	2	34	aa	33
4	3	44	cc	#N/A
5	4	-433	ff	44
6	5		gg	66
7	6	43	dd	33
8	7	34	#NAME?	66
9	8	6743	df	22
10	9	3	fg	
11	10	4	dd	88
12	11	3	g	55
13	12	#N/A	dd	79

To start, open the **Missing Data Handling** dialog. Confirm that *Example 2* is displayed for *Worksheet.*

Select **Variable_1** in the *Variables* field then click the down arrow next to *Select Treatment* in the section under *How do you want to handle missing values for the selected variable(s)* and select **Mean**.



Missing Data Handling

Data Source

Worksheet: Example 2 Workbook: Examples.xlsx

Data range: $A$1:$D$13 #Rows: 12 #Cols: 4

How are missing values coded in the data?

☐ Missing values are represented by this value Missing data value

Note: Empty and invalid cells are automatically considered missing values.

☐ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

Select variable(s) to specify treatment of missing values

☑ Variable names in the first row

Variable	Treatment	User Specified Value
RowID		
Variable_1		
Variable_2		

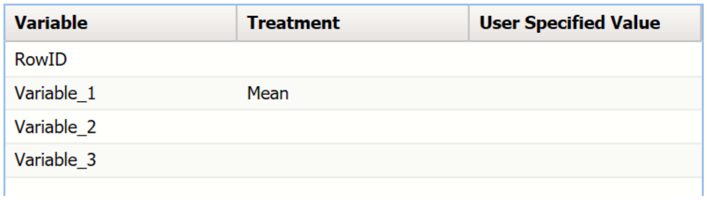
...issing values for the selected variable(s)?

Mean

Treatment options for missing values.

Click **Apply to selected variable(s)**.

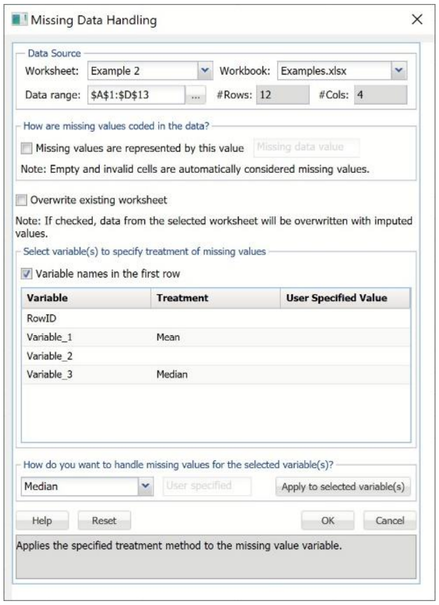
The next tab shows “Mean” appearing under “Treatment” for *Variable_1*.



Variable	Treatment	User Specified Value
RowID		
Variable_1	Mean	
Variable_2		
Variable_3		

Now select **Variable_3** in the *Variables* field and click the down arrow next to *Mean* under *How do you want to handle missing values for the selected variable(s).*

Select Median, then click **Apply to selected variable(s)**



Missing Data Handling

Data Source

Worksheet: Example 2 Workbook: Examples.xlsx

Data range: $A$1:$D$13 #Rows: 12 #Cols: 4

How are missing values coded in the data?

□ Missing values are represented by this value

Note: Empty and invalid cells are automatically considered missing values.

□ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

Select variable(s) to specify treatment of missing values

☑ Variable names in the first row

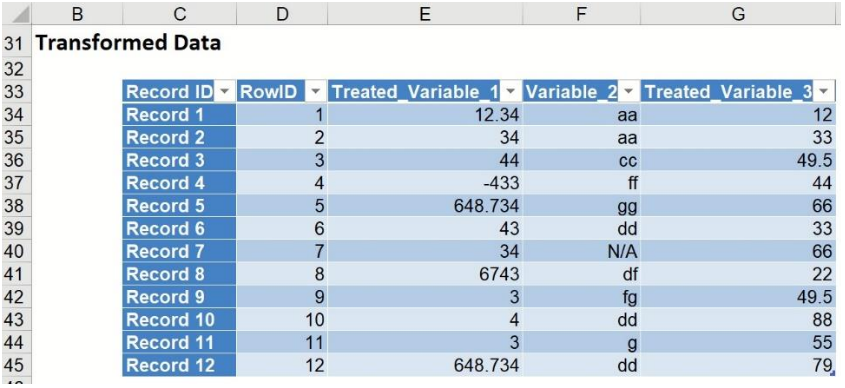
Variable	Treatment	User Specified Value
RowID		
Variable_1	Mean	
Variable_2		
Variable_3	Median	

How do you want to handle missing values for the selected variable(s)?

Median

Applies the specified treatment method to the missing value variable.

Click **OK** to transform the data. See the newly inserted Imputation1 worksheet for the results, shown below.

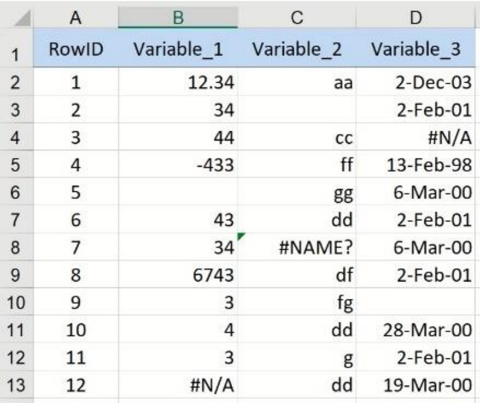


Transformed Data

Record ID	RowID	Treated_Variable_1	Variable_2	Treated_Variable_3
Record 1	1	12.34	aa	12
Record 2	2	34	aa	33
Record 3	3	44	cc	49.5
Record 4	4	-433	ff	44
Record 5	5	648.734	gg	66
Record 6	6	43	dd	33
Record 7	7	34	N/A	66
Record 8	8	6743	df	22
Record 9	9	3	fg	49.5
Record 10	10	4	dd	88
Record 11	11	3	g	55
Record 12	12	648.734	dd	79

In the *Variable_1* column, invalid or missing values have been replaced with the mean calculated from the remaining values in the column. (12.34, 34, 44, -433, 43, 34, 6743, 3, 4 & 3). The cells containing missing values or invalid values in the *Variable_3* column, have been replaced by the median of the remaining values in that column (12, 33, 44, 66, 33, 66, 22, 88, 55 & 79). The invalid data for *Variable_2* remains since no treatment was selected for this variable.

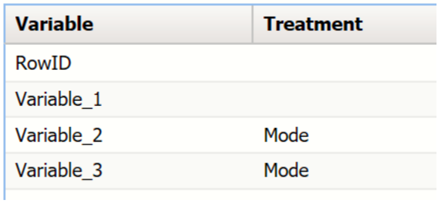
In the Example 3 dataset, *Variable_3* has been replaced with date values.



	A	B	C	D
1	RowID	Variable_1	Variable_2	Variable_3
2	1	12.34	aa	2-Dec-03
3	2	34		2-Feb-01
4	3	44	cc	#N/A
5	4	-433	ff	13-Feb-98
6	5		gg	6-Mar-00
7	6	43	dd	2-Feb-01
8	7	34	#NAME?	6-Mar-00
9	8	6743	df	2-Feb-01
10	9	3	fg	
11	10	4	dd	28-Mar-00
12	11	3	g	2-Feb-01
13	12	#N/A	dd	19-Mar-00

Open the **Missing Data Handling** dialog. Confirm that *Example 3* is displayed for *Worksheet*. In this example, we will replace the missing / invalid values for *Variable_2* and *Variable_3* with the mode of each column.

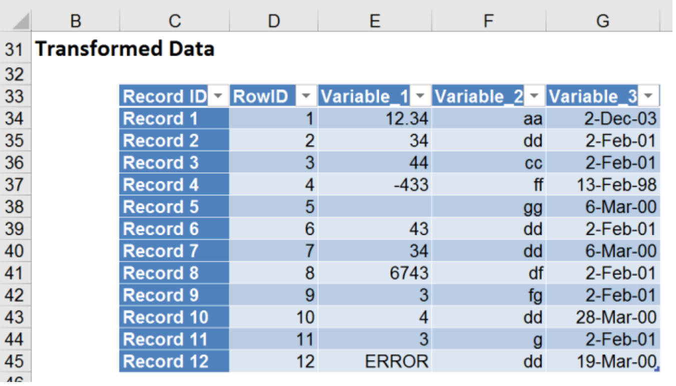
On the Missing Data Handling dialog select **Variable_2**, click the down arrow next to *Select treatment* under *How do you want to handle missing values for the selected variable(s)*, then select **Mode**. (The options Mean and Median do not appear in the list since Variable_2 contains non-numeric values.) Click on **Apply to selected variable(s)**. Repeat these steps for *Variable_3*.



Variable	Treatment
RowID	
Variable_1	
Variable_2	Mode
Variable_3	Mode

Then click **OK**.

Results within the Imputation2 worksheet are shown below.



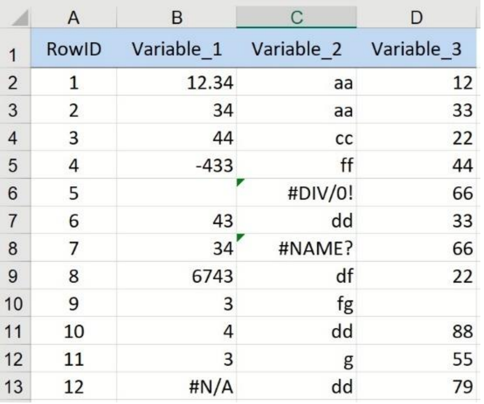
Transformed Data

	B	C	D	E	F	G
31	Transformed Data					
32						
33		Record ID	RowID	Variable_1	Variable_2	Variable_3
34		Record 1	1	12.34	aa	2-Dec-03
35		Record 2	2	34	dd	2-Feb-01
36		Record 3	3	44	cc	2-Feb-01
37		Record 4	4	-433	ff	13-Feb-98
38		Record 5	5		gg	6-Mar-00
39		Record 6	6	43	dd	2-Feb-01
40		Record 7	7	34	dd	6-Mar-00
41		Record 8	8	6743	df	2-Feb-01
42		Record 9	9	3	fg	2-Feb-01
43		Record 10	10	4	dd	28-Mar-00
44		Record 11	11	3	g	2-Feb-01
45		Record 12	12	ERROR	dd	19-Mar-00

The missing values in the Variable_2 column have been replaced by the mode of the valid values (dd) even though, in this instance, the data is non-numeric. (Remember, the mode is the most frequently occurring value in the *Variable_2* column.)

In the *Variable_3* column, the third and ninth records contained missing values. As you can see, they have been replaced by the mode for that column, 2 – Feb – 01.

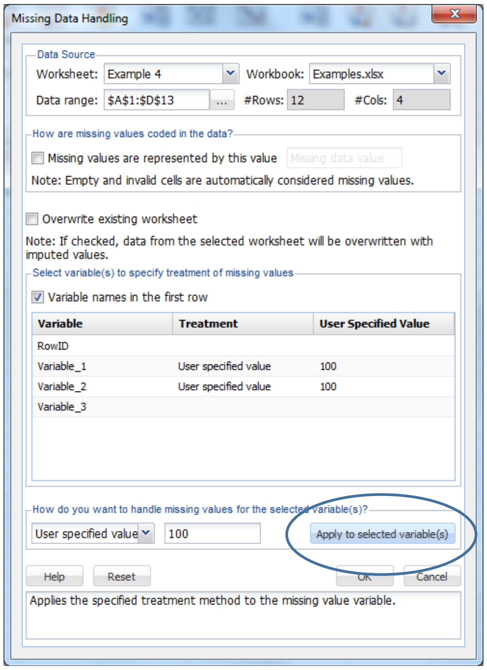
The Example 4 dataset again contains missing and invalid data for all three variables: missing data in cells B6 and D10 and Excel errors in cells B13, C6, and C8. In this example, we will demonstrate Analytic Solver Data Science's ability to replace missing values with User Specified Values.



	A	B	C	D
1	RowID	Variable_1	Variable_2	Variable_3
2	1	12.34	aa	12
3	2	34	aa	33
4	3	44	cc	22
5	4	-433	ff	44
6	5		#DIV/0!	66
7	6	43	dd	33
8	7	34	#NAME?	66
9	8	6743	df	22
10	9	3	fg	
11	10	4	dd	88
12	11	3	g	55
13	12	#N/A	dd	79

Open the **Missing Data Handling** dialog. Confirm that *Example 4* is displayed for *Worksheet.*

Select **Variable_1**, then click the down arrow next to *Select treatment* under *How do you want to handle missing values for the selected variable(s)*, then select **User specified value**. In the field that appears directly to the right of *User specified value*, enter **100**, then click **Apply to selected variable(s)**. Repeat these steps for *Variable_2*. Then click **OK**.



Missing Data Handling

Data Source

Worksheet: Example 4   Workbook: Examples.xlsx

Data range: $A$1:$D$13   #Rows: 12   #Cols: 4

How are missing values coded in the data?

□ Missing values are represented by this value   Missing data value

Note: Empty and invalid cells are automatically considered missing values.

□ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

Select variable(s) to specify treatment of missing values

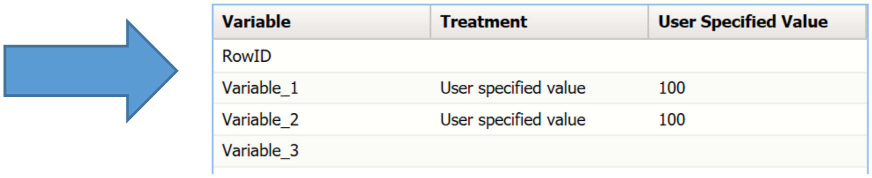
☑ Variable names in the first row

Variable	Treatment	User Specified Value
RowID		
Variable_1	User specified value	100
Variable_2	User specified value	100
Variable_3		

How do you want to handle missing values for the selected variable(s)?

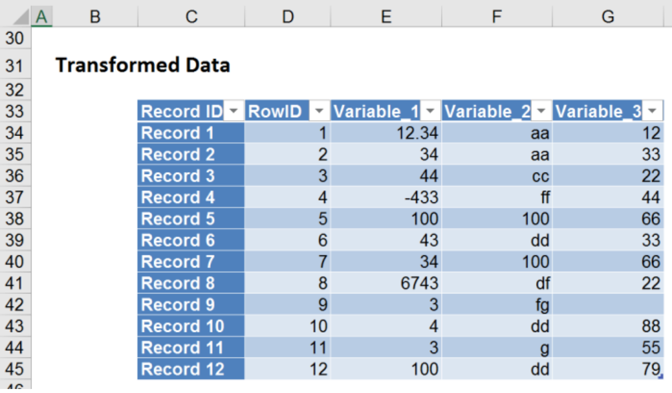
User specified value   100

Applies the specified treatment method to the missing value variable.



Variable	Treatment	User Specified Value
RowID		
Variable_1	User specified value	100
Variable_2	User specified value	100
Variable_3		

The results are shown below.

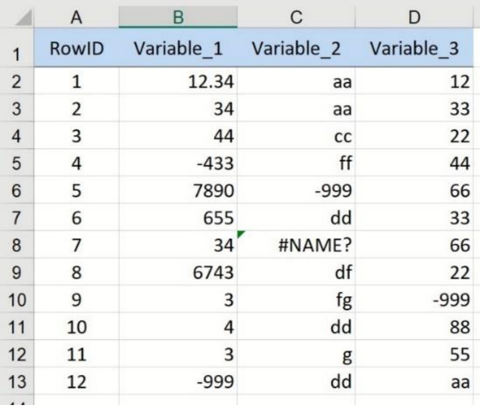


Transformed Data

Record ID	RowID	Variable_1	Variable_2	Variable_3
Record 1	1	12.34	aa	12
Record 2	2	34	aa	33
Record 3	3	44	cc	22
Record 4	4	-433	ff	44
Record 5	5	100	100	66
Record 6	6	43	dd	33
Record 7	7	34	100	66
Record 8	8	6743	df	22
Record 9	9	3	fg	
Record 10	10	4	dd	88
Record 11	11	3	g	55
Record 12	12	100	dd	79

The missing values for *Variable_1* in records 5 and 12 and in records 5 and 7 for *Variable_2*, have been replaced by 100 while the empty cells for *Variable_3* remain untouched.

In the Example 5 dataset, the value -999 appears in all three columns. This example will illustrate Analytic Solver Data Science's ability to detect a given value and replace that value with a user specified value.



RowID	Variable_1	Variable_2	Variable_3
1	12.34	aa	12
2	34	aa	33
3	44	cc	22
4	-433	ff	44
5	7890	-999	66
6	655	dd	33
7	34	#NAME?	66
8	6743	df	22
9	3	fg	-999
10	4	dd	88
11	3	g	55
12	-999	dd	aa

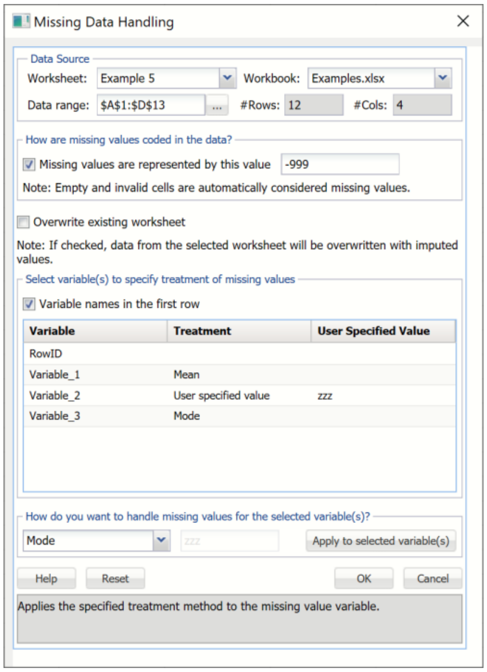
Open the **Missing Data Handling** dialog. Confirm that *Example 5* is displayed for *Worksheet* or *Data Source* within the *Data Source* group.

Select **Missing values are represented by this value** and enter **-999** in the field that appears directly to the right of the option.

Select **Variable_1** in the *Variables* field, click the down arrow next to *Select treatment* and choose **Mean** from the menu, then click **Apply to selected variable(s)**.

Select **Variable_2** in the *Variables* field, click the down arrow next to *No Treatment* and choose **User specified value** from the menu. Enter "zzz" for the value then click **Apply to selected variable(s)**.

Finally, select **Variable_3** in the *Variables* field, click the down arrow next to *User specified value* and choose **Mode** from the menu. Click A**pply to selected variable(s)**.



Missing Data Handling

Data Source

Worksheet: Example 5   Workbook: Examples.xlsx

Data range: $A$1:$D$13   #Rows: 12   #Cols: 4

How are missing values coded in the data?

☑ Missing values are represented by this value -999

Note: Empty and invalid cells are automatically considered missing values.

☐ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

Select variable(s) to specify treatment of missing values

☑ Variable names in the first row

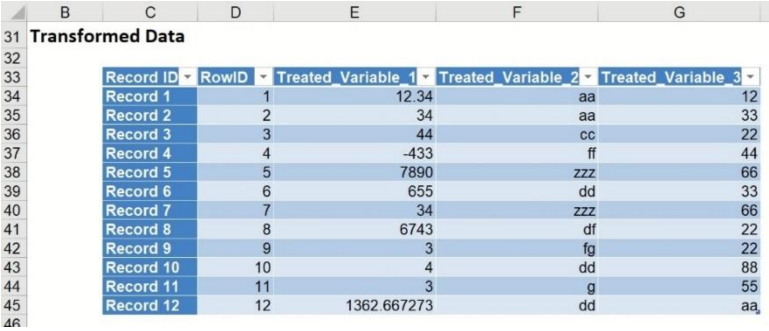
Variable	Treatment	User Specified Value
RowID		
Variable_1	Mean	
Variable_2	User specified value	zzz
Variable_3	Mode	

How do you want to handle missing values for the selected variable(s)?

Mode   zzz

Applies the specified treatment method to the missing value variable.

Click **OK** to transform the data. The results are shown below.

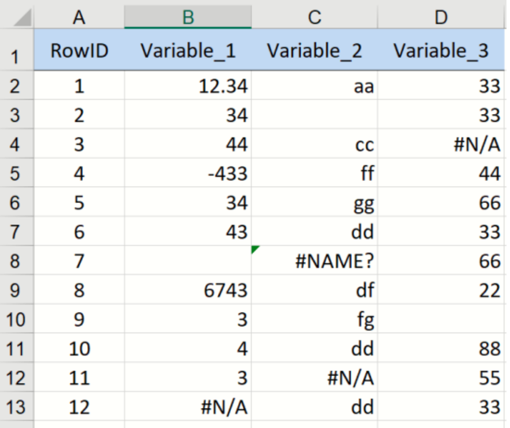


Transformed Data

Record ID	RowID	Treated_Variable_1	Treated_Variable_2	Treated_Variable_3
Record 1	1	12.34	aa	12
Record 2	2	34	aa	33
Record 3	3	44	cc	22
Record 4	4	-433	ff	44
Record 5	5	7890	zzz	66
Record 6	6	655	dd	33
Record 7	7	34	zzz	66
Record 8	8	6743	df	22
Record 9	9	3	fg	22
Record 10	10	4	dd	88
Record 11	11	3	g	55
Record 12	12	1362.667273	dd	aa

Note that in the *Variable_1* column, the specified missing code (-999) was replaced by the mean of the column (in record 12). In the *Variable_2* column, the missing values have been replaced by the user specified value of "zzz" in records 5 and 7, and for *variable_3*, by the mode of the column in record 9.

Let's take a look at one more dataset, *Example 6*, of Examples.xlsx.



RowID	Variable_1	Variable_2	Variable_3
1	12.34	aa	33
2	34		33
3	44	cc	#N/A
4	-433	ff	44
5	34	gg	66
6	43	dd	33
7		#NAME?	66
8	6743	df	22
9	3	fg	
10	4	dd	88
11	3	#N/A	55
12	#N/A	dd	33

Open the **Missing Data Handling** dialog, confirm that *Example 6* is displayed for *Worksheet* or *Data Source* within the *Data Source* group, then apply the following procedures to the indicated columns.

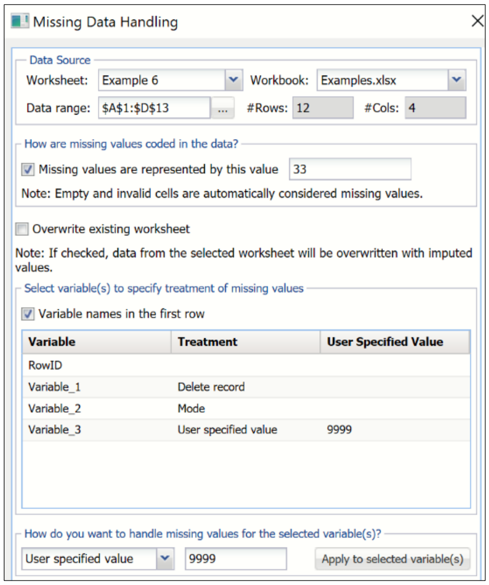
A. Select **Missing values are represented by this value** and enter **33** in the field that appears directly to the right of the option.

B. Select **Variable_1**, select **Delete record** for *How do you want to handle missing values for the selected variable(s)?*, then click **Apply to selected variable(s)**.

C. Select **Variable_2**, select **Mode** for *How do you want to handle missing values for the selected variable(s)?*, then click **Apply to selected variable(s)**.

D. Select **Variable_3**, select **User specified value** for *How do you want to handle missing values for the selected variable(s)?*, enter **9999**, then click **Apply to selected variable(s)**.

Click **OK** to transform the data.



Missing Data Handling

Data Source

Worksheet: Example 6 Workbook: Examples.xlsx

Data range: $A$1:$D$13 #Rows: 12 #Cols: 4

How are missing values coded in the data?

☑ Missing values are represented by this value 33

Note: Empty and invalid cells are automatically considered missing values.

□ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

Select variable(s) to specify treatment of missing values

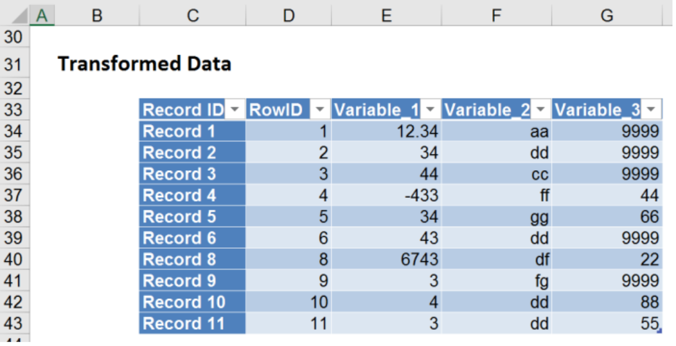
☑ Variable names in the first row

Variable	Treatment	User Specified Value
RowID		
Variable_1	Delete record	
Variable_2	Mode	
Variable_3	User specified value	9999

How do you want to handle missing values for the selected variable(s)?

User specified value 9999 Apply to selected variable(s)

See the output in the Imputation5 worksheet.



Transformed Data

Record ID	RowID	Variable_1	Variable_2	Variable_3
Record 1	1	12.34	aa	9999
Record 2	2	34	dd	9999
Record 3	3	44	cc	9999
Record 4	4	-433	ff	44
Record 5	5	34	gg	66
Record 6	6	43	dd	9999
Record 8	8	6743	df	22
Record 9	9	3	fg	9999
Record 10	10	4	dd	88
Record 11	11	3	dd	55

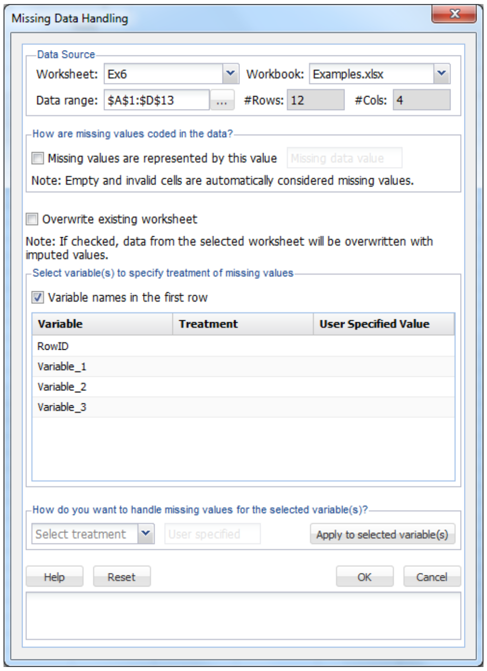
Records 7 and 12 have been deleted since *Delete Record* was chosen for the treatment of missing values for *Variable_1*. In the *Variable_2* column, the missing values in records 2 and 11 have been replaced by the mode of the column, "dd". (Remember, record 7 (which included #NAME for Variable_2) was deleted.) It is important to note that **"Delete record" holds priority over any other instruction in the Missing Data Handling feature.**

In the *Variable_3* column, we instructed Analytic Solver Data Science to treat 33 as a missing value. As a result, the value of “33” in records 1, 2, 3, 6 and 9, were replaced by the user specified value of “9999”.

Note: The value for *Variable_3* for record 12 was 33 which should have been replaced by 9999. However, since *Variable_1* contained a missing value for this record, the instruction "Delete record" was executed first.

Options for Missing Data Handling

The following options appear on the *Missing Data* Handling dialog.



Missing Data Handling

Data Source

Worksheet: Ex6 Workbook: Examples.xlsx

Data range: $A$1:$D$13 #Rows: 12 #Cols: 4

How are missing values coded in the data?

□ Missing values are represented by this value

Note: Empty and invalid cells are automatically considered missing values.

□ Overwrite existing worksheet

Note: If checked, data from the selected worksheet will be overwritten with imputed values.

Select variable(s) to specify treatment of missing values

☑ Variable names in the first row

Variable	Treatment	User Specified Value
RowID		
Variable_1		
Variable_2		
Variable_3		

How do you want to handle missing values for the selected variable(s)?

Missing Values are represented by this value

If this option is selected, a value (either non-numeric or numeric) must be provided in the field that appears directly to the right of the option. Analytic Solver Data Science will treat this value as “missing” and will be handled per the instructions applied in the Missing Data Handling dialog.

Note: Analytic Solver Data Science treats empty and invalid cells as missing values automatically.

Overwrite existing worksheet

If checked, Analytic Solver Data Science overwrites the data set with the new dataset in which all the missing values are appropriately treated.

Variable names in the first Row

When this option is selected, Analytic Solver Data Science will list each variable according to the first row in the selected data range. When the box is unchecked, Analytic Solver Data Science follows the default naming convention, i.e., the variable in the first column of the selected range will be called "Var1", the second column "Var2," etc.

Variables

Each variable and its selected treatment option are listed here.

How do you want to handle missing values for the selected variable(s)?

When a variable in the Variables field is selected, this option is enabled. Click the down arrow to display the following options.

Delete record - If this option is selected, Analytic Solver Data Science will delete the entire record if a missing or invalid value is found for that variable.

Mode - All missing values in the column for the variable specified will be replaced by the mode **-** the value occurring most frequently in the remainder of the column.

Mean - All missing values in the column for the variable specified will be replaced by the mean **-** the average of the values in the remainder of the column.

Median - All missing values in the column for the variable specified will be replaced by the median **-** the number that would appear in the middle of the remaining column values if all values were written in ascending order.

User specified value – If selected, a value must be entered in the field that appears directly to the right of this menu. Analytic Solver Data Science will replace all missing / invalid values with this specified value.

No treatment - If this option is selected, no treatment will be applied to the missing / invalid values for the selected variable.

Apply to selected variable(s)

Clicking this command button will apply the treatment option to the selected variable.

Reset

Resets treatment for all variables listed in the *Variables* field. Also, deselects the *Overwrite Existing Worksheet* option if selected.

OK

Click to run the Missing Data Handling feature of Analytic Solver Data Science.

Transform Continuous Data

Introduction

Analytic Solver Data Science contains two techniques for transforming continuous data: Binning and Rescaling.

Bin Continuous Data

Binning a dataset is a process of grouping measured data into data classes which can reduce the effect of minor errors in the dataset leading to better understanding and visualization. For example, consider exact ages versus the categories, "child", "adult", and "elderly". The three categories would suffice in most analysis rather than using exact ages which are less visual. In Analytic Solver Data Science, the user decides what values the binned variable should take.

A variable can be binned in the following ways.

Equal count: When using this option, the data is binned in such a way that each bin contains the same number of records. When this option is selected, the options Rank of the bin, Mean of the bin, and Median of the bin are enabled.

Rank of the bin: In this option each value in the variable is assigned a rank according to the start and interval values as specified by the user.

Mean of the bin: The mean is calculated as the average of the values lying in the bin interval. This mean value is assigned to each record that lies in that interval.

Median of the bin: Records with the same binning value are counted and the median is calculated on the input value. The median value is then assigned to the binned variable.

Equal Interval: Equal interval is based on bin size. When this method is selected, the whole range is divided into bins with bin sizes specified by the user. The options of Rank and Mid value are available with this method.

Rank of the bin: In this option each value in the variable is assigned a rank according to the start and increment value. Users can specify the starting and increment value.

Mid value: The mean is calculated as the average of the values lying in the bin interval. This mean value is assigned to each value of the variable that lies in that interval.

Rescale Continuous Data

The Rescaling utility was introduced in Analytic Solver Data Science V2017. Use this utility to normalize one or more features in your data. Many Data Science workflows include feature scaling/normalization during the data preprocessing stage. Along with this general-purpose facility, you can access rescaling functionality directly from the dialogs for Supervised Algorithms available in Analytic Solver Data Science application.

Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm.

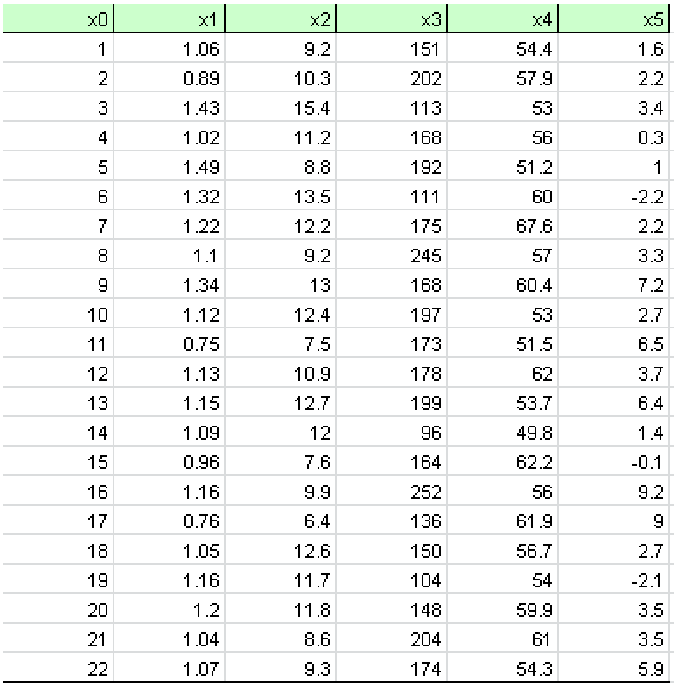
- **Standardization** makes the feature values have zero mean and unit variance. (x−mean)/std.dev.
- **Normalization** scales the data values to the [0,1] range. (x−min)/(max−min)

  The **Correction** option specifies a small positive number ε that is applied as a correction to the formula. The corrected formula is widely used in Neural Networks when Logistic Sigmoid function is used to activate the neurons in hidden layers – it ensures that the data values never reach the asymptotic limits of the activation function. The corrected formula is [x−(min−ε)]/[(max+ε)−(min−ε)].
- **Adjusted Normalization** scales the data values to the [-1,1] range. [2(x−min)/(max−min)]−1

  The **Correction** option specifies a small positive number ε that is applied as a correction to the formula. The corrected formula is widely used in Neural Networks when Hyperbolic Tangent function is used to activate the neurons in hidden layers – it ensures that the data values never reach the asymptotic limits of the activation function. The corrected formula is {2[(x−(min−ε))/((max+ε)−(min−ε))]}−1.
- **Unit Normalization** is another frequently used method to scale the data such that the feature vector has a unit length. This usually means dividing each value by the Euclidean length (L2-norm) of the vector. In some applications, it can be more practical to use the Manhattan Distance (L1-norm).

Examples for Binning Continuous Data

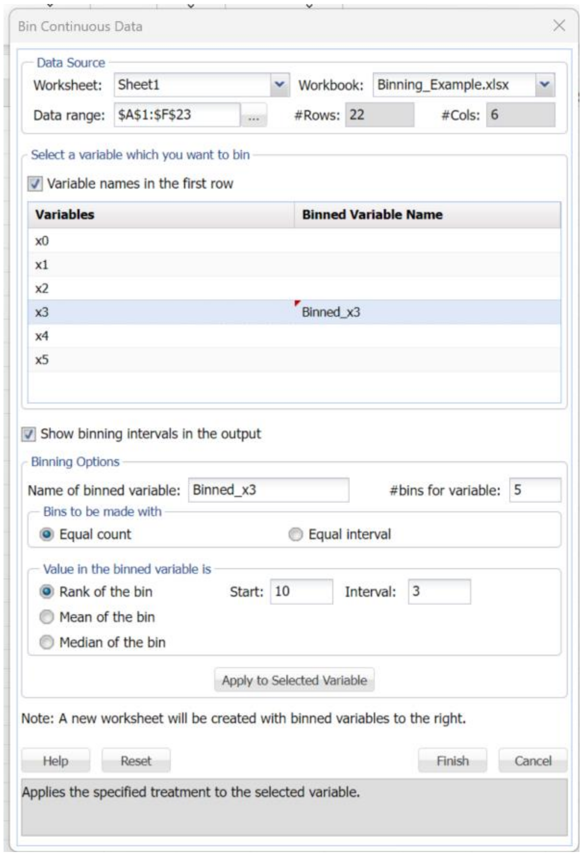
The next four examples illustrate usage of the binning utility included within Analytic Solver Data Science. These examples all use the dataset within Binning_Example.xlsx. Open **Binning_Example.xlsx** by clicking **Help – Example Models**, then **Forecasting/Data Science Examples – Binning_Example.xlsx.** A portion of the dataset is shown below.



x0	x1	x2	x3	x4	x5
1	1.06	9.2	151	54.4	1.6
2	0.89	10.3	202	57.9	2.2
3	1.43	15.4	113	53	3.4
4	1.02	11.2	168	56	0.3
5	1.49	8.8	192	51.2	1
6	1.32	13.5	111	60	-2.2
7	1.22	12.2	175	67.6	2.2
8	1.1	9.2	245	57	3.3
9	1.34	13	168	60.4	7.2
10	1.12	12.4	197	53	2.7
11	0.75	7.5	173	51.5	6.5
12	1.13	10.9	178	62	3.7
13	1.15	12.7	199	53.7	6.4
14	1.09	12	96	49.8	1.4
15	0.96	7.6	164	62.2	-0.1
16	1.16	9.9	252	56	9.2
17	0.76	6.4	136	61.9	9
18	1.05	12.6	150	56.7	2.7
19	1.16	11.7	104	54	-2.1
20	1.2	11.8	148	59.9	3.5
21	1.04	8.6	204	61	3.5
22	1.07	9.3	174	54.3	5.9

Click **Transform -- Transform Continuous Data – Bin** on the desktop Data Science or **Transform – Bin Continuous Data** on the Cloud app Ribbon, to open the *Bin Continuous Data* dialog.

Select **x3** in the *Variables* field. The options are immediately activated. Enter **5** for *#bins for variable*. Under *Value in the binned variable is*, enter **10** for *Start* and **3** for *Interval,* then click **Apply to selected variable**. The variable, *x3,* will appear in the field labeled, *Binned Variable Name.*



Bin Continuous Data

Data Source

Worksheet: Sheet1 Workbook: Binning_Example.xlsx

Data range: $A$1:$F$23 #Rows: 22 #Cols: 6

Select a variable which you want to bin

☑ Variable names in the first row

Variables	Binned Variable Name
x0	
x1	
x2	
x3	Binned_x3
x4	
x5	

☑ Show binning intervals in the output

Binning Options

Name of binned variable: Binned_x3 #bins for variable: 5

Bins to be made with

● Equal count ○ Equal interval

Value in the binned variable is

● Rank of the bin Start: 10 Interval: 3

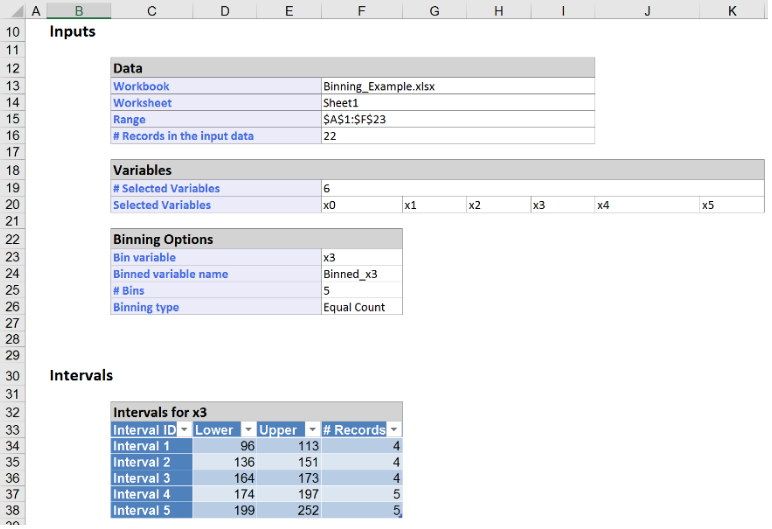
○ Mean of the bin

○ Median of the bin

Note: A new worksheet will be created with binned variables to the right.

Applies the specified treatment to the selected variable.

Now click **Finish**. Analytic Solver Data Science reports the binning intervals in the Bin_Output. This report (pictured below), displays the number of intervals, or bins, created, along with the lower value, upper value and number of records assigned to each bin. As specified, 5 bins, or intervals, were created.



Inputs

Data	
Workbook	Binning_Example.xlsx
Worksheet	Sheet1
Range	$A$1:$F$23
# Records in the input data	22

Variables						
# Selected Variables	6					
Selected Variables	x0	x1	x2	x3	x4	x5

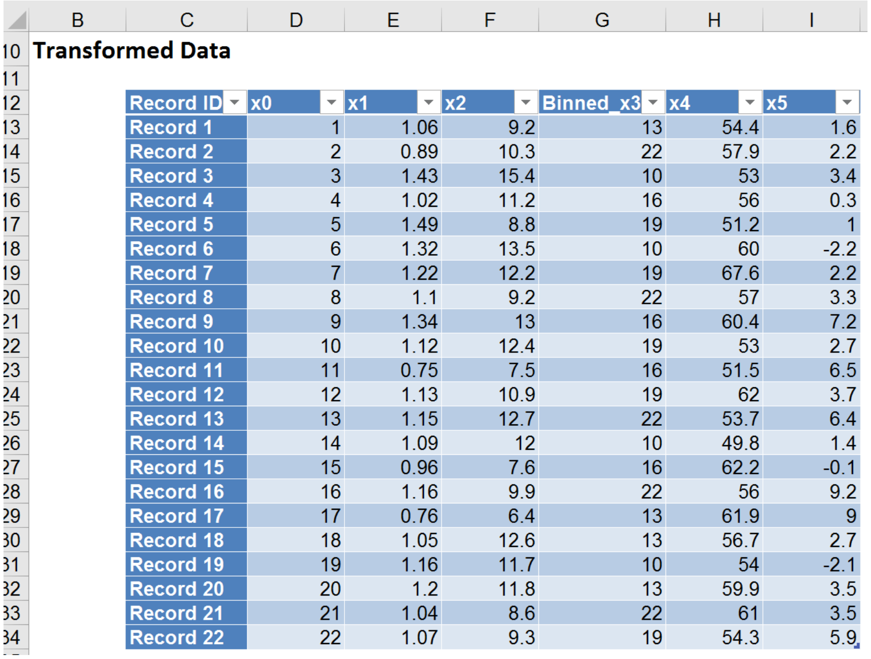
Binning Options	
Bin variable	x3
Binned variable name	Binned_x3
# Bins	5
Binning type	Equal Count

Intervals

Intervals for x3

Interval ID	Lower	Upper	# Records
Interval 1	96	113	4
Interval 2	136	151	4
Interval 3	164	173	4
Interval 4	174	197	5
Interval 5	199	252	5

Click Bin_Transform to see the records assigned to each bin.



Transformed Data

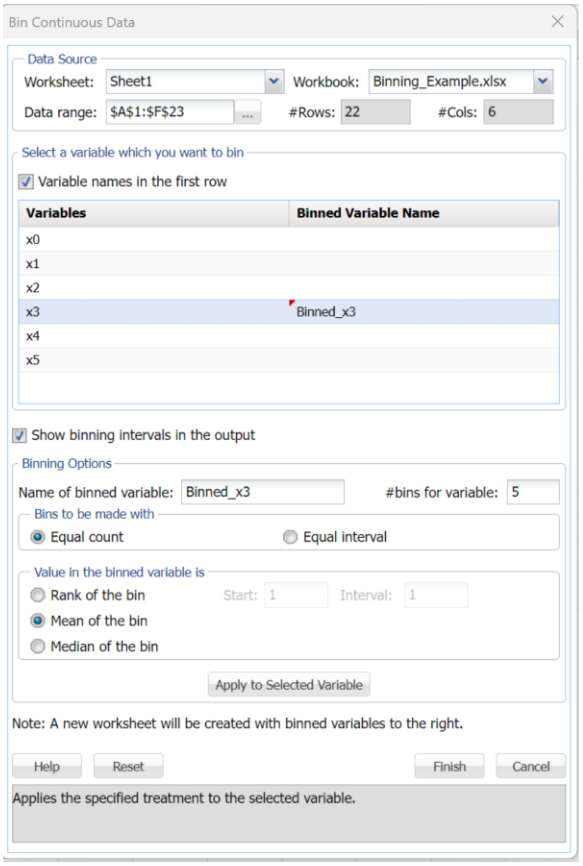
Record ID	x0	x1	x2	Binned_x3	x4	x5
Record 1	1	1.06	9.2	13	54.4	1.6
Record 2	2	0.89	10.3	22	57.9	2.2
Record 3	3	1.43	15.4	10	53	3.4
Record 4	4	1.02	11.2	16	56	0.3
Record 5	5	1.49	8.8	19	51.2	1
Record 6	6	1.32	13.5	10	60	-2.2
Record 7	7	1.22	12.2	19	67.6	2.2
Record 8	8	1.1	9.2	22	57	3.3
Record 9	9	1.34	13	16	60.4	7.2
Record 10	10	1.12	12.4	19	53	2.7
Record 11	11	0.75	7.5	16	51.5	6.5
Record 12	12	1.13	10.9	19	62	3.7
Record 13	13	1.15	12.7	22	53.7	6.4
Record 14	14	1.09	12	10	49.8	1.4
Record 15	15	0.96	7.6	16	62.2	-0.1
Record 16	16	1.16	9.9	22	56	9.2
Record 17	17	0.76	6.4	13	61.9	9
Record 18	18	1.05	12.6	13	56.7	2.7
Record 19	19	1.16	11.7	10	54	-2.1
Record 20	20	1.2	11.8	13	59.9	3.5
Record 21	21	1.04	8.6	22	61	3.5
Record 22	22	1.07	9.3	19	54.3	5.9

As specified, 5 bins were created for the Binned_x3 variable starting with a rank of 10 and an interval of 3: 10, 13 (10 + 3), 16 (13 + 3), 19 (16 + 3), and 22 (19 + 3). The first four smallest values (96, 104, 111, 113 in records 14, 19, 6, and 3, respectively) have been assigned to Bin 10. The next four values in ascending order (136, 148, 150, 151 in records 17, 20, 18, and 1, respectively) have been assigned to Bin 13. The next four values in ascending order (164, 168, 168, 173 in records 15, 9, 4, and 11, respectively) have been assigned to Bin 16. The next five values in ascending order (174, 175, 178, 192, 197 in records 22, 7, 12, 5, and 10, respectively) have been assigned to Bin 19 and the last five values (199, 202, 204, 245, 252 in records 13, 2, 21, 8, and 16, respectively) have been assigned to Bin 22.

Though *Binning Type* is set to *Equal Count*, the number of records in each interval may not be essentially the same. Factors such as border values, total number of records, etc. influence the number of records assigned to each bin.

The next example bins the value of the variable to the mean of the bin rather than the rank of the bin.

Click back to Sheet1 and open the Bin Continuous Data dialog. Select variable “x3”, then select **Mean of the bin**, rather than *Rank of the bin,* for *Value in the binned variable is.* Again enter **5** for *#bins for variable*. Click **Apply to selected variable** then click **Finish**.



Bin Continuous Data

Data Source

Worksheet: Sheet1 Workbook: Binning_Example.xlsx

Data range: $A$1:$F$23 #Rows: 22 #Cols: 6

Select a variable which you want to bin

☑ Variable names in the first row

Variables	Binned Variable Name
x0	
x1	
x2	
x3	Binned_x3
x4	
x5	

☑ Show binning intervals in the output

Binning Options

Name of binned variable: Binned_x3 #bins for variable: 5

Bins to be made with

● Equal count ○ Equal interval

Value in the binned variable is

○ Rank of the bin Start: 1 Interval: 1

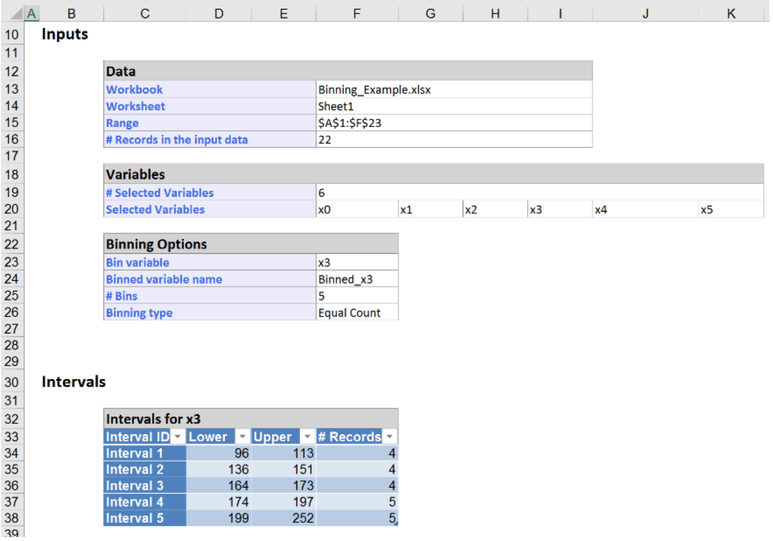
● Mean of the bin

○ Median of the bin

Note: A new worksheet will be created with binned variables to the right.

Applies the specified treatment to the selected variable.

The Bin_Output1 worksheet displays the number of bins created, the minimum and maximum values and the number of records assigned to each bin.



Inputs

Data						
Workbook	Binning_Example.xlsx					
Worksheet	Sheet1					
Range	$A$1:$F$23					
# Records in the input data	22					

Variables						
# Selected Variables	6					
Selected Variables	x0	x1	x2	x3	x4	x5

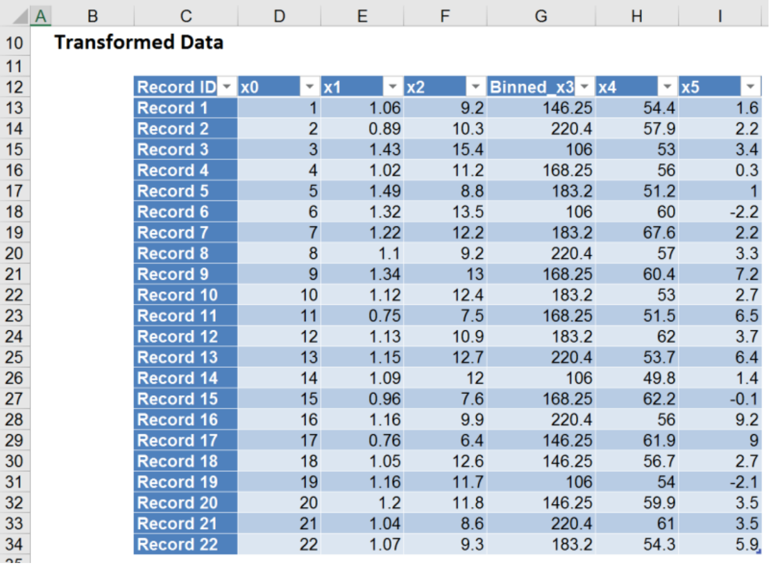
Binning Options	
Bin variable	x3
Binned variable name	Binned_x3
# Bins	5
Binning type	Equal Count

Intervals

Intervals for x3

Interval ID	Lower	Upper	# Records
Interval 1	96	113	4
Interval 2	136	151	4
Interval 3	164	173	4
Interval 4	174	197	5
Interval 5	199	252	5

Click the Bin_Transform1 output sheet.



Transformed Data

Record ID	x0	x1	x2	Binned_x3	x4	x5
Record 1	1	1.06	9.2	146.25	54.4	1.6
Record 2	2	0.89	10.3	220.4	57.9	2.2
Record 3	3	1.43	15.4	106	53	3.4
Record 4	4	1.02	11.2	168.25	56	0.3
Record 5	5	1.49	8.8	183.2	51.2	1
Record 6	6	1.32	13.5	106	60	-2.2
Record 7	7	1.22	12.2	183.2	67.6	2.2
Record 8	8	1.1	9.2	220.4	57	3.3
Record 9	9	1.34	13	168.25	60.4	7.2
Record 10	10	1.12	12.4	183.2	53	2.7
Record 11	11	0.75	7.5	168.25	51.5	6.5
Record 12	12	1.13	10.9	183.2	62	3.7
Record 13	13	1.15	12.7	220.4	53.7	6.4
Record 14	14	1.09	12	106	49.8	1.4
Record 15	15	0.96	7.6	168.25	62.2	-0.1
Record 16	16	1.16	9.9	220.4	56	9.2
Record 17	17	0.76	6.4	146.25	61.9	9
Record 18	18	1.05	12.6	146.25	56.7	2.7
Record 19	19	1.16	11.7	106	54	-2.1
Record 20	20	1.2	11.8	146.25	59.9	3.5
Record 21	21	1.04	8.6	220.4	61	3.5
Record 22	22	1.07	9.3	183.2	54.3	5.9

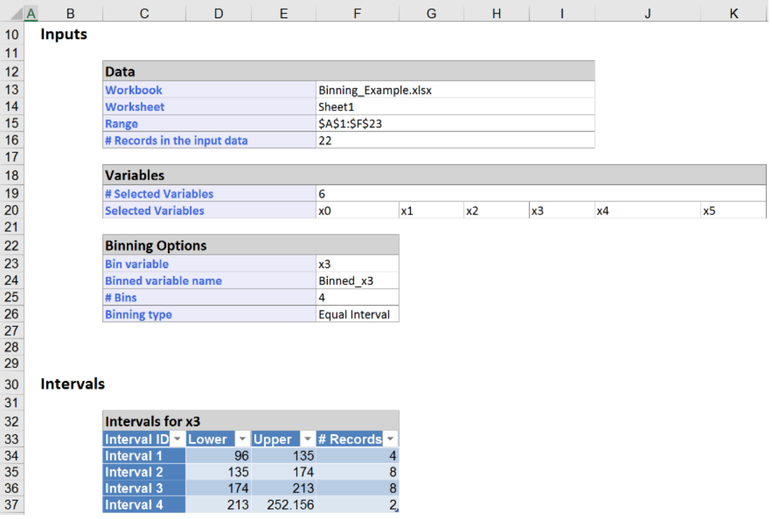
In the output, the **Binned_x3** variable is equal to the mean of all the **x3** variables assigned to that bin. Let's take the first record for an example. Recall, from the previous example, the values from Bin 13: 136, 148, 150, 151. The mean of these values is 146.25 ((136 + 148 + 150 + 151) / 4) which is the value for the *Binned_x3* variable for the first record.

Similarly, if we were to select the *Median of the bin* option, the *Binned_x3* variable would equal the median of all x3 variables assigned to each bin.

The next example explores the *Equal interval option*.

Click back to Sheet1 and open the Bin Continuous Data dialog. Select **x3** in the *Variables* field, enter **4** for *#bins for variable*, select **Equal interval** under *Bins to be made with*, enter **12** for *Start* and **3** for *Interval* under *Value in the binned variable is*, then click **Apply to selected variable**.

Click **Finish**. The Bin_Output2 output sheet displays the number of bins created and the number of records assigned to each bin.



Inputs

Data	
Workbook	Binning_Example.xlsx
Worksheet	Sheet1
Range	$A$1:$F$23
# Records in the input data	22

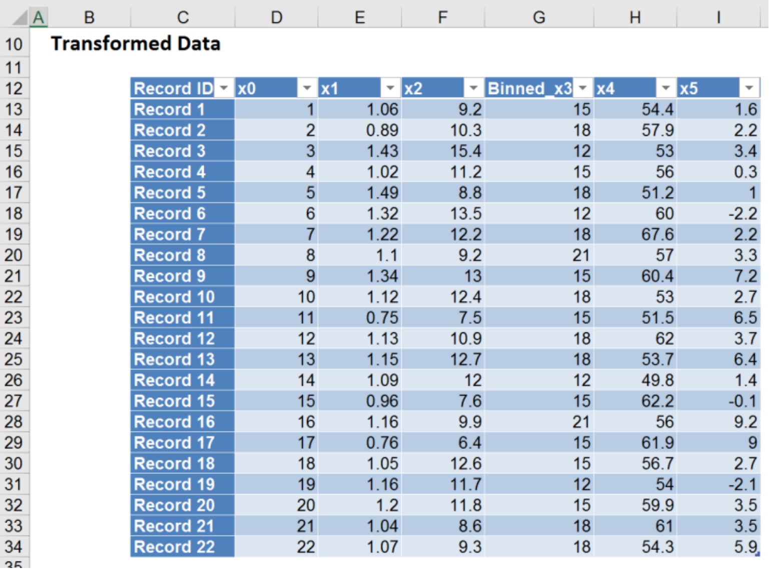
Variables						
# Selected Variables	6					
Selected Variables	x0	x1	x2	x3	x4	x5

Binning Options	
Bin variable	x3
Binned variable name	Binned_x3
# Bins	4
Binning type	Equal Interval

Intervals

Intervals for x3

Interval ID	Lower	Upper	# Records
Interval 1	96	135	4
Interval 2	135	174	8
Interval 3	174	213	8
Interval 4	213	252.156	2



Transformed Data

Record ID	x0	x1	x2	Binned_x3	x4	x5
Record 1	1	1.06	9.2	15	54.4	1.6
Record 2	2	0.89	10.3	18	57.9	2.2
Record 3	3	1.43	15.4	12	53	3.4
Record 4	4	1.02	11.2	15	56	0.3
Record 5	5	1.49	8.8	18	51.2	1
Record 6	6	1.32	13.5	12	60	-2.2
Record 7	7	1.22	12.2	18	67.6	2.2
Record 8	8	1.1	9.2	21	57	3.3
Record 9	9	1.34	13	15	60.4	7.2
Record 10	10	1.12	12.4	18	53	2.7
Record 11	11	0.75	7.5	15	51.5	6.5
Record 12	12	1.13	10.9	18	62	3.7
Record 13	13	1.15	12.7	18	53.7	6.4
Record 14	14	1.09	12	12	49.8	1.4
Record 15	15	0.96	7.6	15	62.2	-0.1
Record 16	16	1.16	9.9	21	56	9.2
Record 17	17	0.76	6.4	15	61.9	9
Record 18	18	1.05	12.6	15	56.7	2.7
Record 19	19	1.16	11.7	12	54	-2.1
Record 20	20	1.2	11.8	15	59.9	3.5
Record 21	21	1.04	8.6	18	61	3.5
Record 22	22	1.07	9.3	18	54.3	5.9

Analytic Solver Data Science calculates the interval roughly as the (Maximum value for the x3 variable **-** Minimum value for the x3 variable) / #bins specified by the user **-** or in this instance (252 – 96) / 4 which equals 39. This means that the bins will be assigned x3 variables in accordance to the following rules.

Bin 12: Values 96 **-** < 135

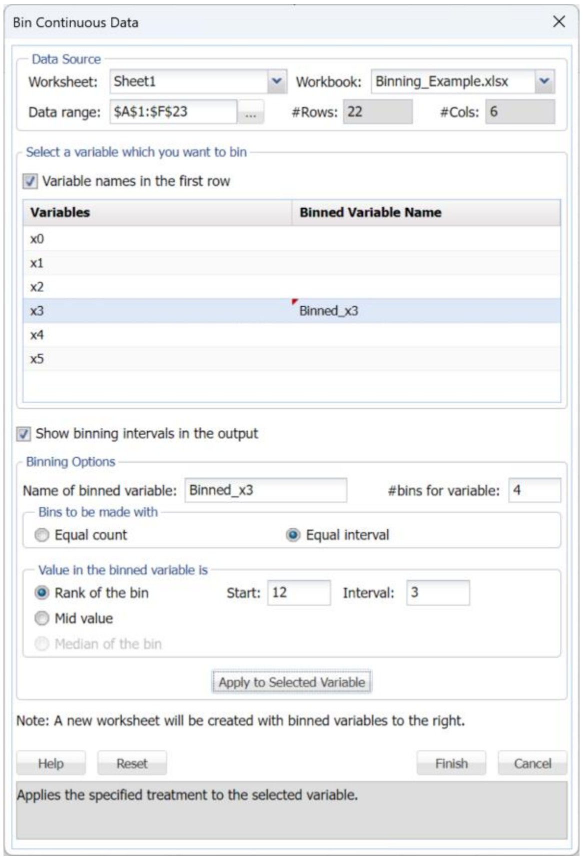
Bin 15: Values 135 **-** < 174

Bin 18: Values 174 **-** < 213

Bin 21: Values 213 **-** < 252

In the first record, x3 has a value of 151. As a result, this record has been assigned to Bin 15 since 151 lies in the interval of Bin 3.

Click back to Sheet1 and open the Bin Continuous Data dialog. Select **x3** in the *Variables* field, enter **4** for *#bins for variable*, select **Equal interval** under *Bins to be made with*, select **Mid Value** for *Value in the binned variable is*, then click **Apply to selected variable**.



Bin Continuous Data

Data Source

Worksheet: Sheet1 Workbook: Binning_Example.xlsx

Data range: $A$1:$F$23 #Rows: 22 #Cols: 6

Select a variable which you want to bin

☑ Variable names in the first row

Variables	Binned Variable Name
x0	
x1	
x2	
x3	Binned_x3
x4	
x5	

☑ Show binning intervals in the output

Binning Options

Name of binned variable: Binned_x3 #bins for variable: 4

Bins to be made with

○ Equal count ● Equal interval

Value in the binned variable is

● Rank of the bin Start: 12 Interval: 3

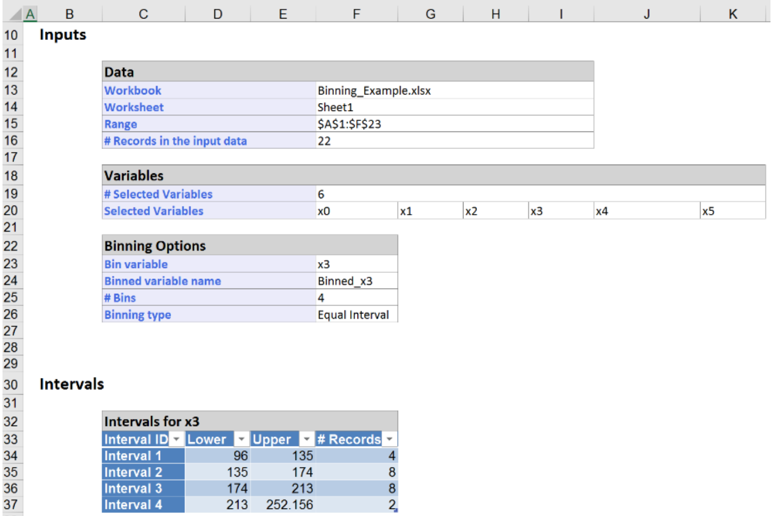
○ Mid value

○ Median of the bin

Note: A new worksheet will be created with binned variables to the right.

Applies the specified treatment to the selected variable.

Then click **Finish**.



Inputs

Data	
Workbook	Binning_Example.xlsx
Worksheet	Sheet1
Range	$A$1:$F$23
# Records in the input data	22

Variables						
# Selected Variables	6					
Selected Variables	x0	x1	x2	x3	x4	x5

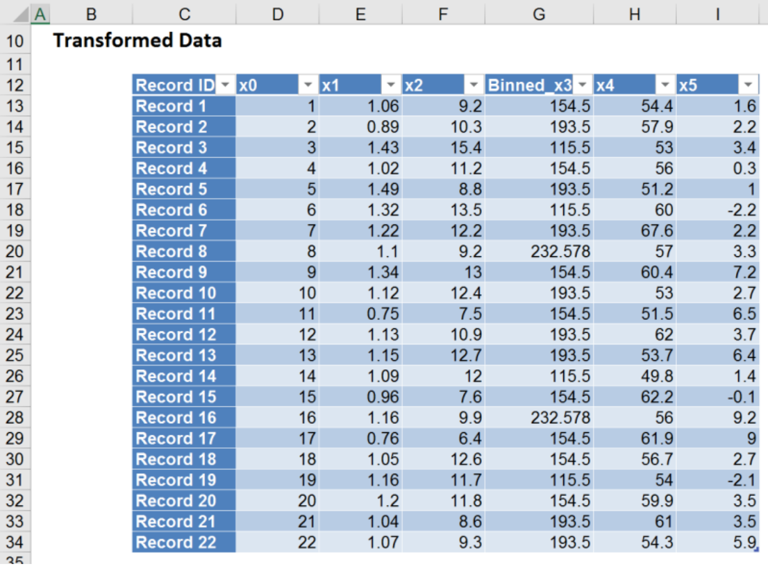
Binning Options	
Bin variable	x3
Binned variable name	Binned_x3
# Bins	4
Binning type	Equal Interval

Intervals

Intervals for x3

Interval ID	Lower	Upper	# Records
Interval 1	96	135	4
Interval 2	135	174	8
Interval 3	174	213	8
Interval 4	213	252.156	2

The Bin_Ouput3 worksheet displays the 4 intervals and the number of records along with the range of values assigned to each bin: Bin 1 (96 to 135), Bin 2 (135 – 174), Bin 3 (174 – 213), and Bin 4 (213 – 252).



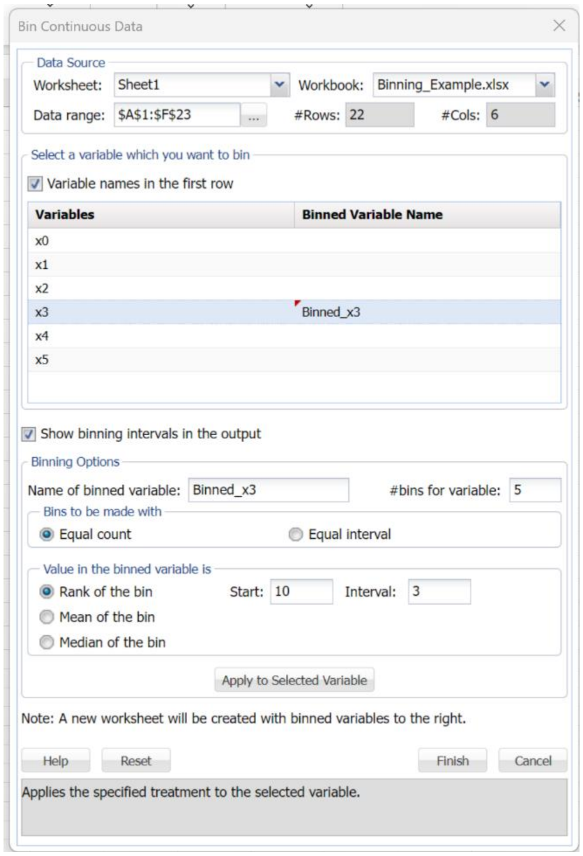
Transformed Data

Record ID	x0	x1	x2	Binned_x3	x4	x5
Record 1	1	1.06	9.2	154.5	54.4	1.6
Record 2	2	0.89	10.3	193.5	57.9	2.2
Record 3	3	1.43	15.4	115.5	53	3.4
Record 4	4	1.02	11.2	154.5	56	0.3
Record 5	5	1.49	8.8	193.5	51.2	1
Record 6	6	1.32	13.5	115.5	60	-2.2
Record 7	7	1.22	12.2	193.5	67.6	2.2
Record 8	8	1.1	9.2	232.578	57	3.3
Record 9	9	1.34	13	154.5	60.4	7.2
Record 10	10	1.12	12.4	193.5	53	2.7
Record 11	11	0.75	7.5	154.5	51.5	6.5
Record 12	12	1.13	10.9	193.5	62	3.7
Record 13	13	1.15	12.7	193.5	53.7	6.4
Record 14	14	1.09	12	115.5	49.8	1.4
Record 15	15	0.96	7.6	154.5	62.2	-0.1
Record 16	16	1.16	9.9	232.578	56	9.2
Record 17	17	0.76	6.4	154.5	61.9	9
Record 18	18	1.05	12.6	154.5	56.7	2.7
Record 19	19	1.16	11.7	115.5	54	-2.1
Record 20	20	1.2	11.8	154.5	59.9	3.5
Record 21	21	1.04	8.6	193.5	61	3.5
Record 22	22	1.07	9.3	193.5	54.3	5.9

The output sheet, Bin_Transform3, shows us which records have been assigned to each of the 4 bins. The value of the binned variable is the midpoint of each interval: 115.5 for Bin 1, 154.5 for Bin 2, 193.5 for Bin 3 and 232.5 for Bin 4. In the first record, x3's value is 154.4. Since this value lies in the interval for Bin 2 (135 – 174) the mid value of this interval is reported for the Binned_x3 variable, 154.5. In the last record, x3's value is 193.5. Since this value lies in the interval for Bin 3 (174 – 213), the mid value of this interval is reported for the Binned_x3 variable, 193.5.

Options for Binning Continuous Data

The following options appear on the *Bin Continuous Data* dialog.



Data Source

Worksheet: Sheet1 Workbook: Binning_Example.xlsx

Data range: $A$1:$F$23 #Rows: 22 #Cols: 6

Select a variable which you want to bin

☑ Variable names in the first row

Variables	Binned Variable Name
x0	
x1	
x2	
x3	Binned_x3
x4	
x5	

☑ Show binning intervals in the output

Binning Options

Name of binned variable: Binned_x3 #bins for variable: 5

Bins to be made with

● Equal count ○ Equal interval

Value in the binned variable is

● Rank of the bin Start: 10 Interval: 3

○ Mean of the bin

○ Median of the bin

Note: A new worksheet will be created with binned variables to the right.

Applies the specified treatment to the selected variable.

Data Source

Worksheet: The name of the worksheet containing the dataset.

Workbook: The name of the workbook containing the dataset.

Data range: The data range for the dataset on the Excel worksheet.

#Rows: The number of rows in the dataset.

#Cols: The number of columns in the dataset.

Select a variable which you want to bin:

Variable names in the first row: If this option is selected, the list of variables in the *Variables* field will be listed according to titles appearing in the first row of the dataset.

Binned Variable Name: Variable appearing here will be binned.

Show binning intervals in the output: Select this option to include the binning intervals in the output report.

Name of binned variable: The name displayed here will appear for the binned variable in the output report.

#bins for variable: Enter the number of desired bins here.

Equal Count: When this option is selected, the data is binned in such a way that each bin contains the same number of records. Note: There is a possibility that the number of records in a bin may not be equal due to factors such as border values, the number of records being divisible by the number of bins, etc.

The options for *Value of the binned* variable for this process are *Rank, Mean*, and *Median*. See below for explanations of each.

Equal Interval: When this option is selected, the binning procedure will assign records to bins if the record's value falls in the interval of the bin. Bin intervals are calculated by roughly subtracting the Minimum variable value from the Maximum variable value and dividing by the number of bins ((Max Value – Min Value) / # bins). The options for *Value of the binned variable* for this process are *Rank* and *Mid value*. See below for explanations of each.

Rank of the bin: When either the *Equal count* or the *Equal interval* option is selected, *Rank of the bin* is enabled. When selected, the User has the option to specify the *Start* value of the first bin and the *Interval* of each bin. Subsequent bin values will be calculated as the previous bin + interval value.

Mean of the bin: When the *Equal count* option is selected, *Mean of the bin* is enabled. Analytic Solver Data Science calculates the mean of all values in the bin and assigns that value to the binned variable.

Median of the bin: When the *Equal count* option is selected, *Median of the bin* is enabled. Analytic Solver Data Science finds the median of all values in the bin and assigns that value to the binned variable.

Mid Value: When the *Equal Interval* option is selected, this option is enabled. The mid value of the interval will be displayed on the output report for the assigned bin.

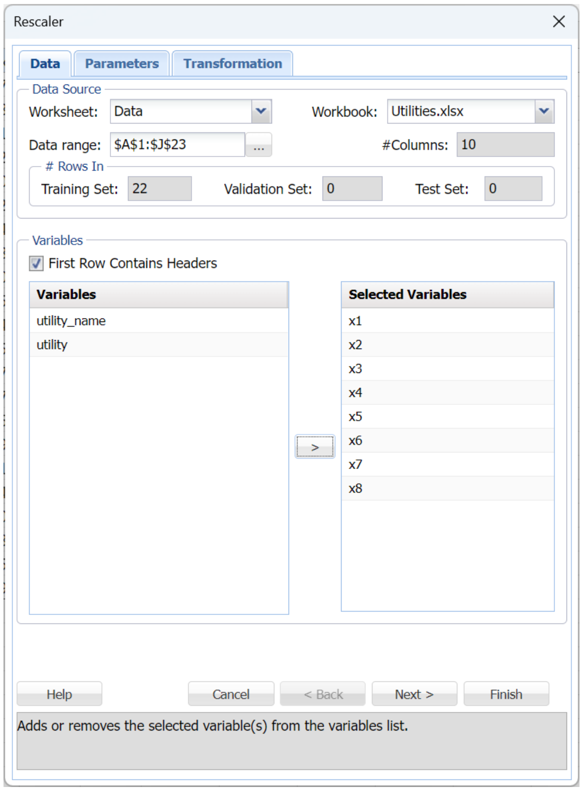
Apply to Selected Variable: Click this command button to apply the selected options to the selected variable.

Examples for Rescaling Continuous Data

The next example illustrates how to use the rescaling utility included within Analytic Solver Data Science. This example uses the Utilities.xlsx example dataset. Open **Utilities.xlsx** by clicking, **Help – Example Models**, then **Forecasting/Data Science Examples**.

Click **Transform -- Transform Continuous Data – Rescale** on the desktop Data Science or AnalyticSolver.com ribbon or **Transform - Rescale Continuous Data** on the Cloud app Ribbon, to open the *Rescaler* dialog.

Select **x1, x2, x3, x4, x5, x6, x7** and **x8** in the *Variables* field. then click > to add them as *Selected Variables.*



Data Source

Worksheet: Data

Workbook: Utilities.xlsx

Data range: $A$1:$J$23

#Columns: 10

Rows In

Training Set: 22

Validation Set: 0

Test Set: 0

Variables

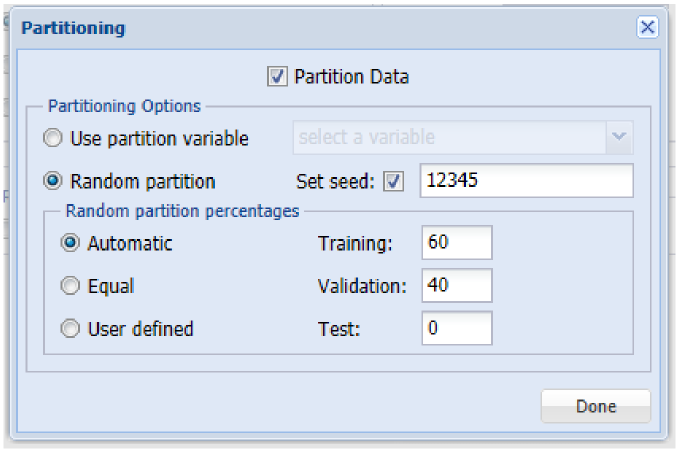
☑ First Row Contains Headers

Variables	Selected Variables
utility_name	x1
utility	x2
	x3
	x4
	x5
	x6
	x7
	x8

Adds or removes the selected variable(s) from the variables list.

Click **Next** to advance to the Parameters tab.

Click Partition Data to open the Partition Data dialog, then select the Partition Data option to enable the partition options.



☑ Partition Data

Partitioning Options

○ Use partition variable   select a variable

● Random partition   Set seed: ☑ 12345

Random partition percentages

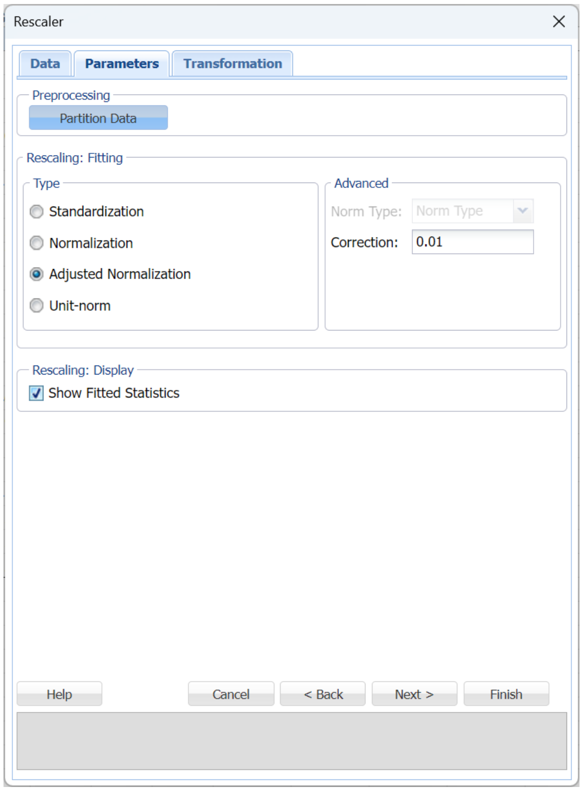
● Automatic   Training: 60

○ Equal   Validation: 40

○ User defined   Test: 0

Click Done to accept the random partition defaults. For more information on partition, see the Random Data Partitioning chapter that occurs later in this guide.

- Under Rescaling: Fitting, Select Adjusted Normalization.
- Leave the Correction option set to the default of 0.01.
- Select Show Fitted Statistics to include in the output.



Rescaler

Preprocessing

Rescaling: Fitting

Type

○ Standardization

○ Normalization

● Adjusted Normalization

○ Unit-norm

Advanced

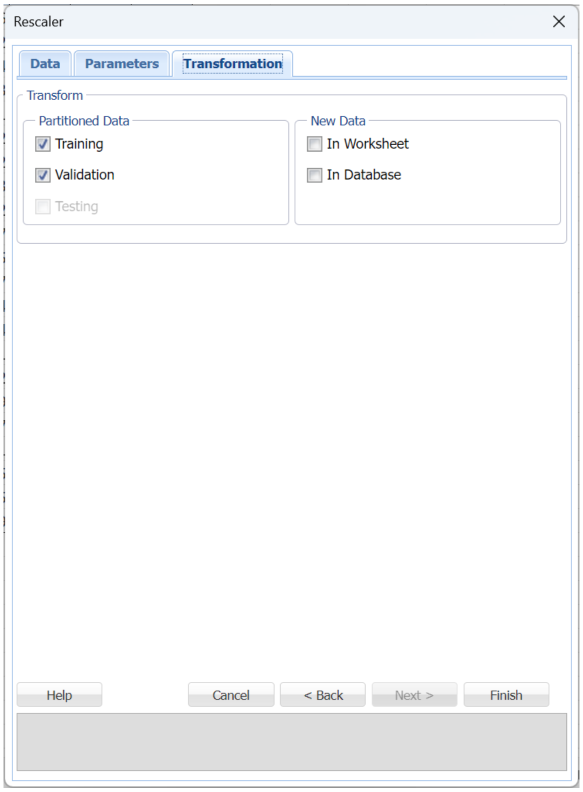
Norm Type: Norm Type

Correction: 0.01

Rescaling: Display

☑ Show Fitted Statistics

Click **Next** to advance to the *Transformation* dialog. Leave Training and Validation selected under Partition Data (the defaults), to rescale both partitions.



Rescaler

Transform

Partitioned Data

☑ Training

☑ Validation

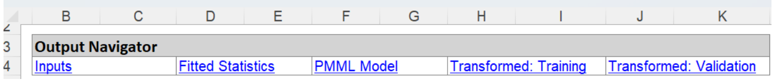
□ Testing

New Data

□ In Worksheet

□ In Database

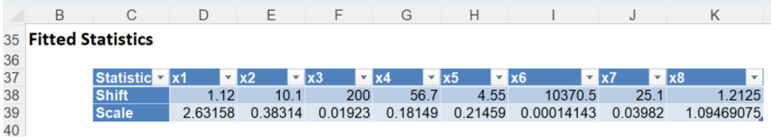
Click **Finish**. Four output sheets are inserted to the right of the Data tab: Rescaling, Rescaling_TrainingTransform, Rescaling_ValidationTransform and Rescaling_Stored. The Output Navigator appears at the top of each of these four sheets. (See the Scoring New Data chapter for information on how to score new data using Rescaling_Stored.)



Output Navigator

Inputs	Fitted Statistics	PMML Model	Transformed: Training	Transformed: Validation

Click the Fitted Statistics link to navigate to the Fitted Statistics table located on the Rescaling output sheet. Shift and Scale values are inferred from the training data. Each formula below can be rearranged into the form (x-shift)/scale. Then other partitions/new data is rescaled using the statistics of data features in the training set.

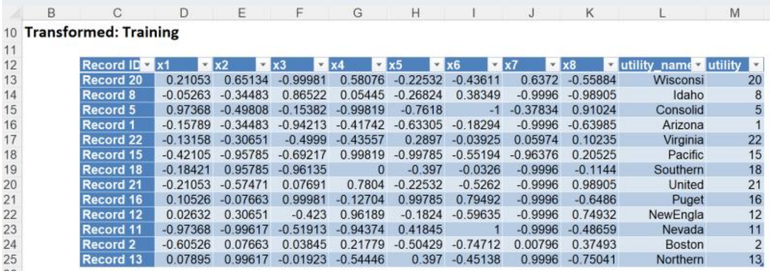


Fitted Statistics

Statistic	x1	x2	x3	x4	x5	x6	x7	x8
Shift	1.12	10.1	200	56.7	4.55	10370.5	25.1	1.2125
Scale	2.63158	0.38314	0.01923	0.18149	0.21459	0.00014143	0.03982	1.09469075

Click the Transformed: Training link on the Output Navigator to display the rescaled variable values for the Training partition.

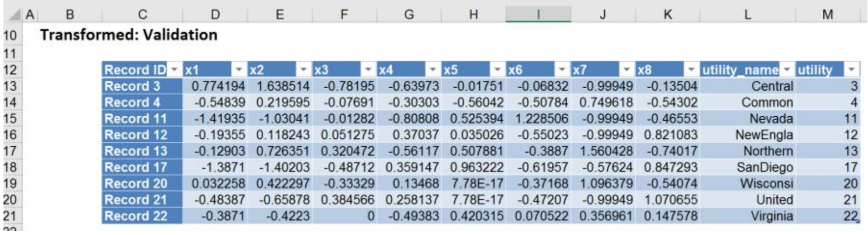
Note: Unselected variables are appended to the rescaled variables in the Transformed: Training and Transformed: Validation data tables to maintain the complete input data.



Transformed: Training

Record ID	x1	x2	x3	x4	x5	x6	x7	x8	utility_name	utility
Record 20	0.21053	0.65134	-0.99981	0.58076	-0.22532	-0.43611	0.6372	-0.55884	Wisconsi	20
Record 8	-0.05263	-0.34483	0.86522	0.05445	-0.26824	0.38349	-0.9996	-0.98905	Idaho	8
Record 5	0.97368	-0.49808	-0.15382	-0.99819	-0.7618	-1	-0.37834	0.91024	Consolid	5
Record 1	-0.15789	-0.34483	-0.94213	-0.41742	-0.63305	-0.18294	-0.9996	-0.63985	Arizona	1
Record 22	-0.13158	-0.30651	-0.4999	-0.43557	0.2897	-0.03925	0.05974	0.10235	Virginia	22
Record 15	-0.42105	-0.95785	-0.69217	0.99819	-0.99785	-0.55194	-0.96376	0.20525	Pacific	15
Record 18	-0.18421	0.95785	-0.96135	0	-0.397	-0.0326	-0.9996	-0.1144	Southern	18
Record 21	-0.21053	-0.57471	0.07691	0.7804	-0.22532	-0.5262	-0.9996	0.98905	United	21
Record 16	0.10526	-0.07663	0.99981	-0.12704	0.99785	0.79492	-0.9996	-0.6486	Puget	16
Record 12	0.02632	0.30651	-0.423	0.96189	-0.1824	-0.59635	-0.9996	0.74932	NewEngla	12
Record 11	-0.97368	-0.99617	-0.51913	-0.94374	0.41845	1	-0.9996	-0.48659	Nevada	11
Record 2	-0.60526	0.07663	0.03845	0.21779	-0.50429	-0.74712	0.00796	0.37493	Boston	2
Record 13	0.07895	0.99617	-0.01923	-0.54446	0.397	-0.45138	0.9996	-0.75041	Northern	13

Click the Transformed: Validation link on the Output Navigator to display the rescaled variable values for the Validation partition.



Transformed: Validation

Record ID	x1	x2	x3	x4	x5	x6	x7	x8	utility_name	utility
Record 3	0.774194	1.638514	-0.78195	-0.63973	-0.01751	-0.06832	-0.99949	-0.13504	Central	3
Record 4	-0.54839	0.219595	-0.07691	-0.30303	-0.56042	-0.50784	0.749618	-0.54302	Common	4
Record 11	-1.41935	-1.03041	-0.01282	-0.80808	0.525394	1.228506	-0.99949	-0.46553	Nevada	11
Record 12	-0.19355	0.118243	0.051275	0.37037	0.035026	-0.55023	-0.99949	0.821083	NewEngla	12
Record 13	-0.12903	0.726351	0.320472	-0.56117	0.507881	-0.3887	1.560428	-0.74017	Northern	13
Record 17	-1.3871	-1.40203	-0.48712	0.359147	0.963222	-0.61957	-0.57624	0.847293	SanDiego	17
Record 20	0.032258	0.422297	-0.33329	0.13468	7.78E-17	-0.37168	1.096379	-0.54074	Wisconsi	20
Record 21	-0.48387	-0.65878	0.384566	0.258137	7.78E-17	-0.47207	-0.99949	1.070655	United	21
Record 22	-0.3871	-0.4223	0	-0.49383	0.420315	0.070522	0.356961	0.147578	Virginia	22

Rescaling Options

See below for an explanation of options on all three tabs of the Rescaling dialog*: Data*, *Parameters* and *Transformation* tabs.

The following options appear on all three tabs of the Rescaler dialog.

Help: Click the Help button to access documentation on all k-Means Clustering options.

Cancel: Click the Cancel button to close the dialog without running k-Means Clustering.

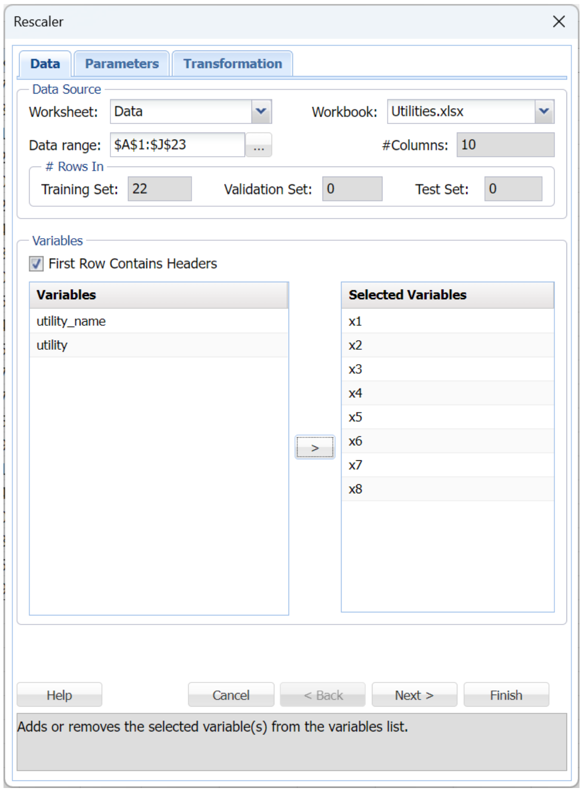
Next: Click the Next button to advance to the next tab.

Finish: Click Finish to accept all option settings on all three dialogs, and run k-Means Clustering.

Rescaling Data Tab

See below for documentation for all options appearing on the Rescaling Data tab.

Rescaling Data tab



Adds or removes the selected variable(s) from the variables list.

Data Source

Worksheet: Click the down arrow to select the desired worksheet where the dataset is contained.

Workbook: Click the down arrow to select the desired workbook where the dataset is contained.

Data range: Select or enter the desired data range within the dataset. This data range may either be a portion of the dataset or the complete dataset.

#Columns: Displays the number of columns in the data range. This option is read only.

#Rows In Training Set, Validation Set and Test Set: Displays the number of columns in training, validation and/or test partitions, if they exist. This option is read only.

Variables

First Row Contains Headers: Select this checkbox if the first row in the dataset contains column headings.

Variables: This field contains the list of the variables, or features, included in the data range.

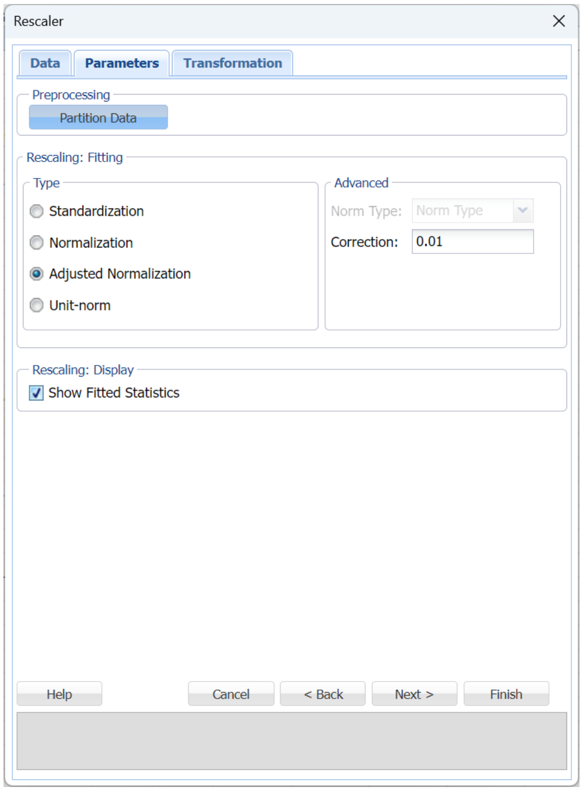
Selected Variables: This field contains the list of variables, or features, to be included in k-Means Clustering.

- To include a variable in k-Means Clustering, select the variable in the Variables list, then click > to move the variable to the Selected Variables list.
- To remove a variable as a selected variable, click the variable in the Selected Variables list, then click < to move the variable back to the Variables list.

Rescaling Parameters Tab

See below for documentation for all options appearing on the Rescaling Parameters tab.

Rescaling Parameters tab



Rescaling: Fitting

Type

○ Standardization

○ Normalization

● Adjusted Normalization

○ Unit-norm

Advanced

Norm Type: Norm Type

Correction: 0.01

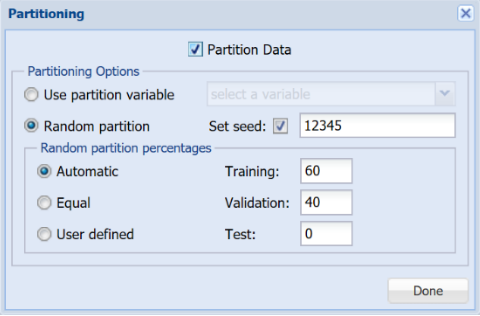
Rescaling: Display

☑ Show Fitted Statistics

Preprocessing

Analytic Solver Data Science allows partitioning to be performed on the Parameters tab for Rescaling, if the active data set is un-partitioned. If the active data set has already been partitioned, this button will be disabled. Clicking the Partition Data button opens the following dialog. Select Partition Data on the dialog to enable the partitioning options.   See the Partitioning chapter for descriptions of each Partitioning option shown in the dialog below.

Partitioning "On-the-fly" dialog



Partitioning

☑ Partition Data

Partitioning Options

○ Use partition variable   select a variable

● Random partition   Set seed: ☑ 12345

Random partition percentages

● Automatic   Training: 60

○ Equal   Validation: 40

○ User defined   Test: 0

Why use "on-the-fly" Partitioning?

If a data partition will be used to train and validate several different algorithms that will be compared for predictive power, it may be better to use the Ribbon Partition choices to create, rescale and/or partition the dataset. But if the rescaled data and/or data partition will be used with a single algorithm, or if it isn't crucial to compare algorithms on exactly the same data, "Partition-on-the-Fly" and "Rescale-on-the-fly" offers several advantages:

- User interface steps are saved, and the Analytic Solver task pane is not cluttered with partition and rescaling output.
- Partition-on-the-fly and Rescaling-on-the-fly is *much faster* than first rescaling the data, creating a standard partition and then running an algorithm.
- Partition-on-the-fly and Rescaling-on-the-fly can handle *larger* datasets without exhausting memory, since the intermediate partition results for the partitioned data is never created.

Rescaling: Fitting

Use Rescaling to normalize one or more features in your data. Many Data Science workflows include feature scaling/normalization during the data preprocessing stage. Along with this general-purpose facility, you can access rescaling functionality directly from the dialogs for Supervised Algorithms available in Analytic Solver Data Science application.

Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm.

- **Standardization** makes the feature values have zero mean and unit variance. (x−mean)/std.dev.
- **Normalization** scales the data values to the [0,1] range. (x−min)/(max−min)

The **Correction** option specifies a small positive number ε that is applied as a correction to the formula. The corrected formula is widely used in Neural Networks when Logistic Sigmoid function is used to activate the neurons in hidden layers – it ensures that the data values never reach the asymptotic limits of the activation function. The corrected formula is $[x-(\min-\varepsilon)]/[(\max+\varepsilon)-(\min-\varepsilon)]$.

- **Adjusted Normalization** scales the data values to the [-1,1] range. $[2(x-\min)/(\max-\min)]-1$

  The **Correction** option specifies a small positive number ε that is applied as a correction to the formula. The corrected formula is widely used in Neural Networks when Hyperbolic Tangent function is used to activate the neurons in hidden layers – it ensures that the data values never reach the asymptotic limits of the activation function. The corrected formula is $\{2[(x-(\min-\varepsilon))/((\max+\varepsilon)-(\min-\varepsilon))]\}-1$.

- **Unit Normalization** is another frequently used method to scale the data such that the feature vector has a unit length. This usually means dividing each value by the Euclidean length (L2-norm) of the vector. In some applications, it can be more practical to use the Manhattan Distance (L1-norm).

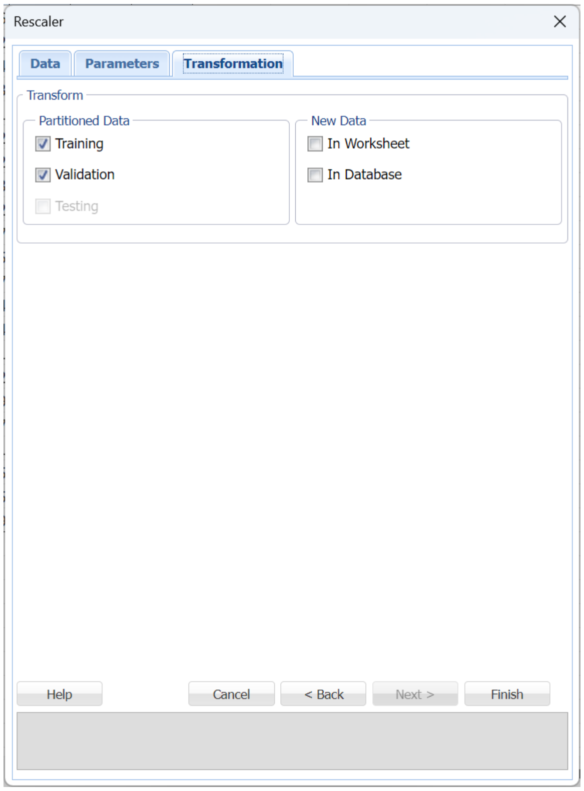
Show Fitted Statistics

Select Fitted Statistics to include in the Rescaler output. Shift and Scale values are inferred from the training data. Each formula in the data table can be rearranged into the form (x-shift)/scale. Then other partitions/new data is rescaled using the statistics of data features in the training set.

Transformation Parameters Tab

See below for documentation for all options appearing on the Rescaling Transformation tab.

Rescaling Transformation tab



Transform

Partitioned Data

☑ Training

☑ Validation

□ Testing

New Data

□ In Worksheet

□ In Database

Partitioned Data

Select Training to apply the Rescaler method to the Training Partition.

Select Validation to apply the Rescaler method to the Validation Partition, if one exists.

Select Testing to apply the Rescaler method to the Test Partition, if one exists.

New Data

See the Scoring New Data chapter in the Data Miing User Guide for more information on scoring new data within a worksheet or database.

Transforming Categorical Data

Introduction

Analysts often deal with data that is not numeric. Non numeric data values can be alphanumeric (mix of text and numbers) or numeric values with no numerical significance (such as a postal code). Such variables are called 'Categorical' variables, where every unique value of the variable is a separate 'category'.

Categorical variables can be nominal or ordinal. Nominal variable values have no order, for example, True or False or Male or Female. Values for an ordinal variable have a clear order but no fixed unit of measurement, i.e. Kinder, First, Second, Third, Fourth, and Fifth or a Size Chart of 1, 2, 3, 4, 5.

Dealing with categorical data poses some limitations. For example, if your data contains a multitude of categories, you might want to combine several categories into one or perhaps you may want to use a data science technique that does not directly handle untransformed categorical variables.

Analytic Solver Data Science provides options to transform data in the following ways:

1. **By Creating Dummy Variables:** When this feature is used, a non-numeric variable (column) is transformed into several new numeric or binary variables (columns).

   Imagine a variable called ***Language*** which has data values *English, French, German* and *Spanish*. Running this transformation will result in the creation of four new variables: ***Language_English, Language_French, Language_German,*** and ***Language_Spanish***. Each of these variables will take on values of either 0 or 1 depending on the value of the ***Language*** variable in the record. For instance, if in a particular record Language = *German,* then among the dummy variables created, ***Language_German*** will be 1 while the other Language_XXX variables will be set to zero.

2. **Create Category Scores:** In this feature, a string variable is converted into a new numeric, categorical variable.

3. **Reduce Categories:** This utility helps you create a new categorical variable that reduces the number of categories. You can reduce the number of categories “by frequency” or “manually”.

   There are two different options to choose from.

   A. Option 1 assigns categories 1 through n - 1 to the n - 1 most frequently occurring categories, and assigns category n to all remaining categories.

   B. Option 2 maps multiple distinct category values in the original column to a new category variable between 1 and n where n is the number of observations.

Note: See the Analytic Solver User Guide for data limitations in Analytic Solver Comprehensive/Data Science.

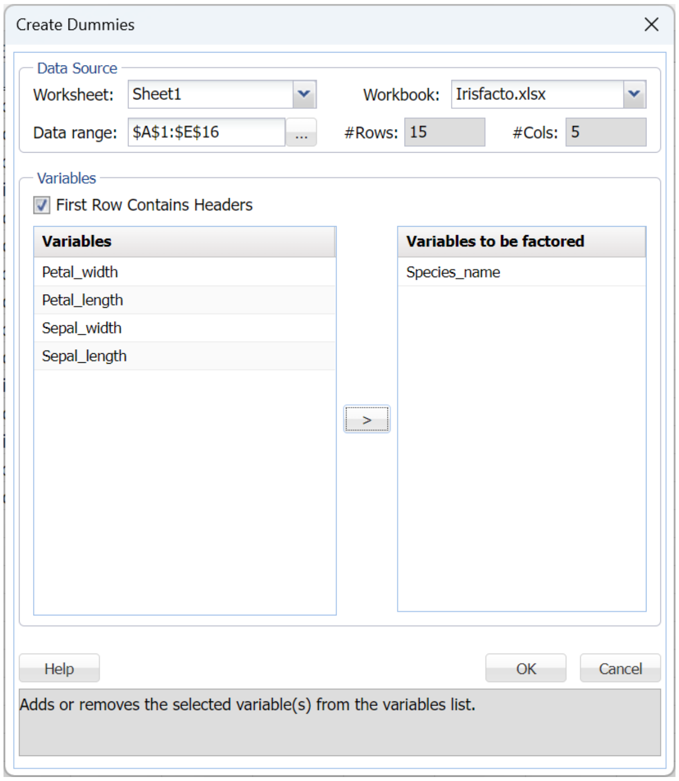
Transforming Categorical Data Examples

Four examples are presented in this section. The first example replaces one categorical variable with three binary variables, the second example replaces the same categorical variable with one ordinal variable and the last two examples illustrate how to use the Reduce Categories tool. The first two examples use the dataset contained within IrisFacto.xlsx, the last two use the Iris.xlsx dataset. (IrisFacto.xlsx is derived from the well-known dataset, *Iris.xlsx*.) Both of these datasets may be found in Help – Example Models – Forecasting/Datamining.

Example 1 Create Dummies

Open the **Irisfacto** example dataset by clicking **Help – Example Models – Forecasting/Data Science Examples**. Click Sheet1 and then **Transform -- Transform Categorical Data -- Create Dummies** in Desktop Analytic Solver or **Transform – Categorical Data – Create Dummies** in the Cloud app to bring up the *Create Dummies* dialog.

Select **Species_name** in the *Variables* field and then > to move the variable to the *Variables to be factored* field. Note that *Species_Name* is a string variable.



Create Dummies

Data Source

Worksheet: Sheet1

Workbook: Irisfacto.xlsx

Data range: $A$1:$E$16

#Rows: 15

#Cols: 5

Variables

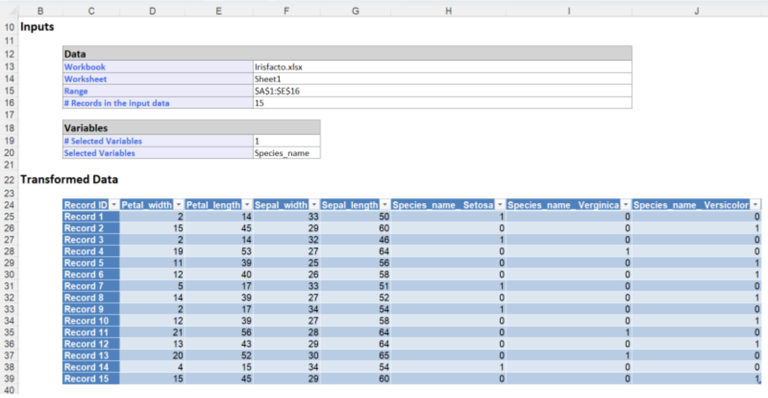
☑ First Row Contains Headers

Variables
Petal_width
Petal_length
Sepal_width
Sepal_length

Variables to be factored
Species_name

Adds or removes the selected variable(s) from the variables list.

Click **OK** and view the output, *Encoding*, which is inserted on the Model tab of the Analytic Solver Task Pane under Data Science – Transformations – Create Dummies.



Inputs

Data	
Workbook	Irisfacto.xlsx
Worksheet	Sheet1
Range	$A$1:$E$16
# Records in the input data	15

Variables	
# Selected Variables	1
Selected Variables	Species_name

Transformed Data

Record ID	Petal_width	Petal_length	Sepal_width	Sepal_length	Species_name_Setosa	Species_name_Verginica	Species_name_Versicolor
Record 1	2	14	33	50	1	0	0
Record 2	15	45	29	60	0	0	1
Record 3	2	14	32	46	1	0	0
Record 4	19	53	27	64	0	1	0
Record 5	11	39	25	56	0	0	1
Record 6	12	40	26	58	0	0	1
Record 7	5	17	33	51	1	0	0
Record 8	14	39	27	52	0	0	1
Record 9	2	17	34	54	1	0	0
Record 10	12	39	27	58	0	0	1
Record 11	21	56	28	64	0	1	0
Record 12	13	43	29	64	0	0	1
Record 13	20	52	30	65	0	1	0
Record 14	4	15	34	54	1	0	0
Record 15	15	45	29	60	0	0	1

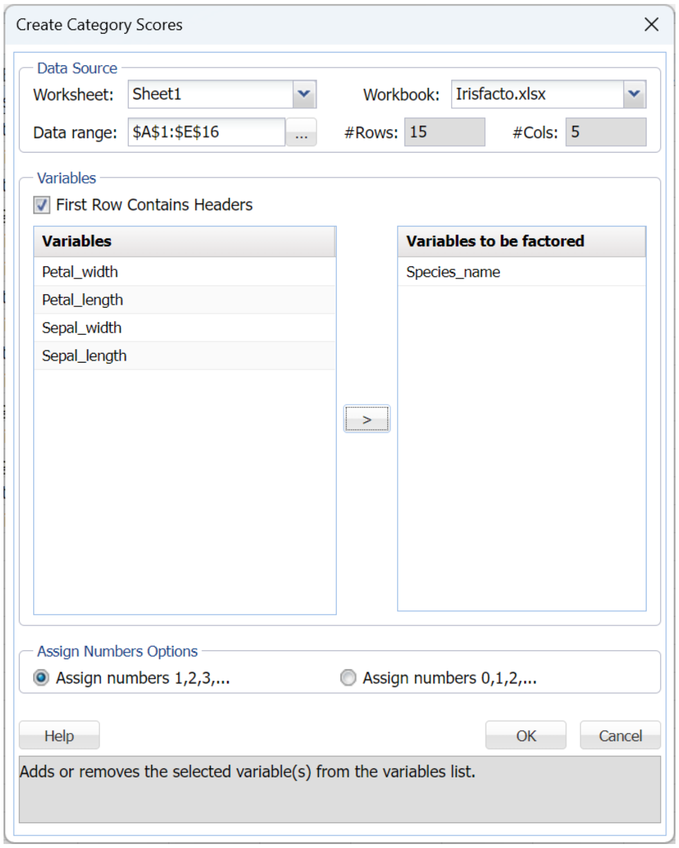
As shown in the output above, the variable, *Species_name*, is expressed as three binary dummy variables: *Species_name_Setosa*, *Species_name_Verginica* and *Species_name_Versicolor*. These new dummy variables are assigned values of either 1, to indicate that the record belongs, or 0, to indicate that the record does not belong. For example, *Species_name_Setosa* is assigned a value of 1 only when the value of Species_name="Setosa" is in the dataset. Otherwise, *Species_name_Setosa* = 0. The same is true for the two remaining dummy variables i.e. *Species_name_Verginica* and *Species_name_Versicolor*.

Analytic Solver Data Science converted the string variable into three categorical variables which resulted in a completely numeric dataset.

Example 2 Create Category Scores

Click back to the Sheet1 and then **Transform -- Transform Categorical Data -- Create Category Scores** in Desktop Analytic Solver or **Transform – Categorical Data – Create Category Scores** in the Cloud app to bring up the *Create Category Scores* dialog.

Select **Species_name** in the *Variables* field and click > to move the variable to the *Variables to be factored* field. Keep the default option of *Assign numbers 1,2,3...*.



Create Category Scores

Data Source

Worksheet: Sheet1

Workbook: Irisfacto.xlsx

Data range: $A$1:$E$16

#Rows: 15

#Cols: 5

Variables

☑ First Row Contains Headers

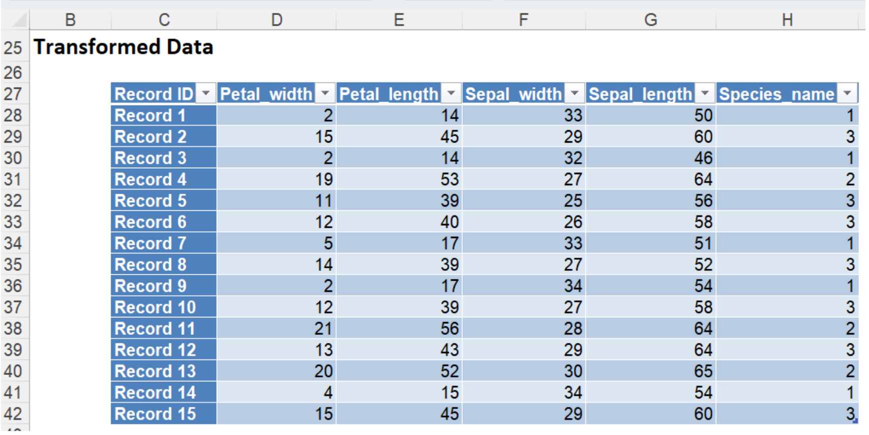
Variables	Variables to be factored
Petal_width	Species_name
Petal_length	
Sepal_width	
Sepal_length	

Assign Numbers Options

● Assign numbers 1,2,3,... ○ Assign numbers 0,1,2,...

Adds or removes the selected variable(s) from the variables list.

Click **OK**. Expand Data Science – Transformations – Create Category Scores to view the results contained within Factorization.



Transformed Data

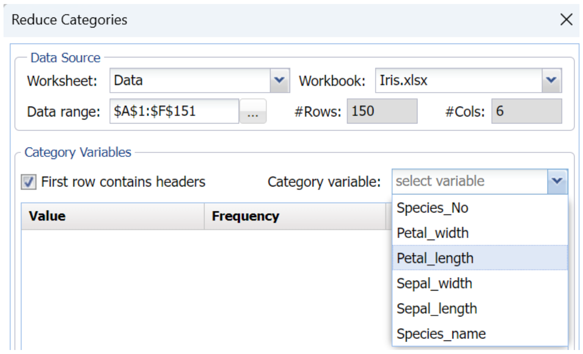
Record ID	Petal_width	Petal_length	Sepal_width	Sepal_length	Species_name
Record 1	2	14	33	50	1
Record 2	15	45	29	60	3
Record 3	2	14	32	46	1
Record 4	19	53	27	64	2
Record 5	11	39	25	56	3
Record 6	12	40	26	58	3
Record 7	5	17	33	51	1
Record 8	14	39	27	52	3
Record 9	2	17	34	54	1
Record 10	12	39	27	58	3
Record 11	21	56	28	64	2
Record 12	13	43	29	64	3
Record 13	20	52	30	65	2
Record 14	4	15	34	54	1
Record 15	15	45	29	60	3

Analytic Solver Data Science has sorted the values of the *Species_name* variable alphabetically and then assigned values of 1, 2 or 3 to each record depending on the species type. (Starting from 1 because we selected *Assign numbers 1,2,3....* To have Analytic Solver Data Science start from 0, select the option *Assign numbers 0, 1, 2,…* on the *Create Category Scores* dialog.) A variable, *Factorized_Species_name* is created to store these assigned numbers. Analytic Solver Data Science has converted this dataset to an entirely numeric dataset.

Example 3 Reduce Categories Manually

Open the Iris example dataset by clicking **Help – Example Models – Forecasting-Data Science Examples**. Select a cell within the dataset, say A1, then click **Transform -- Transform Categorical Data – Reduce Categories** in Desktop Analytic Solver or **Transform – Categorical Data – Reduce Categories** in the Cloud app to bring up the Reduce Categories dialog.

Select **Petal_length** as the variable…



Reduce Categories

Data Source

Worksheet: Data

Workbook: Iris.xlsx

Data range: $A$1:$F$151

#Rows: 150

#Cols: 6

Category Variables

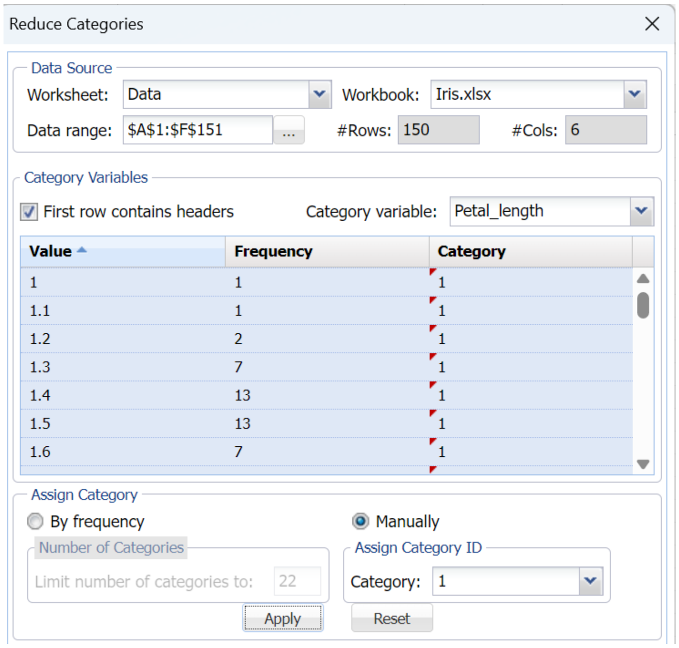
☑ First row contains headers

Category variable: select variable

Value	Frequency

…then select the **Manually** radio button under *Assign Category* heading.

All unique values of the *Petal_length* variable are now listed. Select all categories with Values less than 2 (so Value = 1 to 1.9); click the down arrow next to *Category* and select **1**, then click **Apply**.



Reduce Categories

Data Source

Worksheet: Data

Workbook: Iris.xlsx

Data range: $A$1:$F$151

#Rows: 150

#Cols: 6

Category Variables

☑ First row contains headers

Category variable: Petal_length

Value	Frequency	Category
1	1	1
1.1	1	1
1.2	2	1
1.3	7	1
1.4	13	1
1.5	13	1
1.6	7	1

Assign Category

○ By frequency

Number of Categories

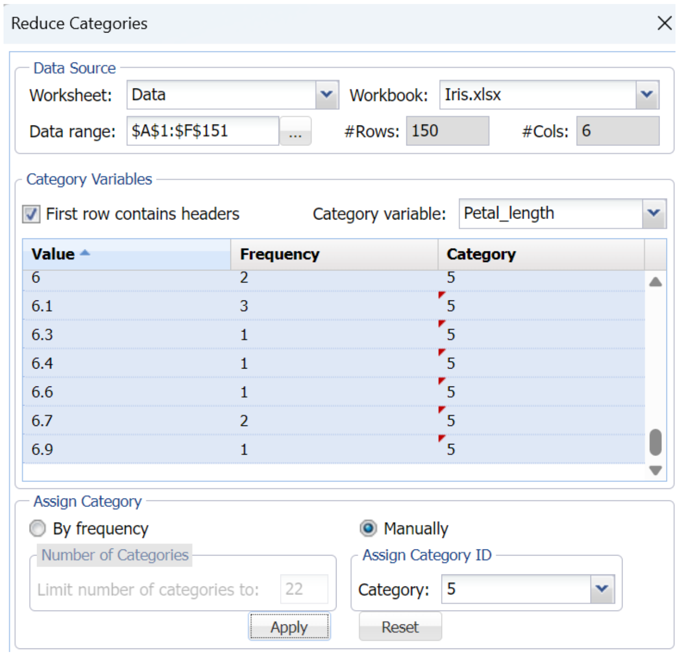
Limit number of categories to: 22

● Manually

Assign Category ID

Category: 1

Repeat these steps for categories with values from 3 to 3.9 and apply a Category number of 2. Continue repeating these steps until values ranging from 4 thru 4.9 are assigned a category number = 3, values ranging from 5 thru 5.9 are assigned a category number = 4, and values ranging from 6 thru 6.9 are assigned a category = 5.



Reduce Categories

Data Source

Worksheet: Data Workbook: Iris.xlsx

Data range: $A$1:$F$151 #Rows: 150 #Cols: 6

Category Variables

☑ First row contains headers Category variable: Petal_length

Value	Frequency	Category
6	2	5
6.1	3	5
6.3	1	5
6.4	1	5
6.6	1	5
6.7	2	5
6.9	1	5

Assign Category

○ By frequency ● Manually

Number of Categories

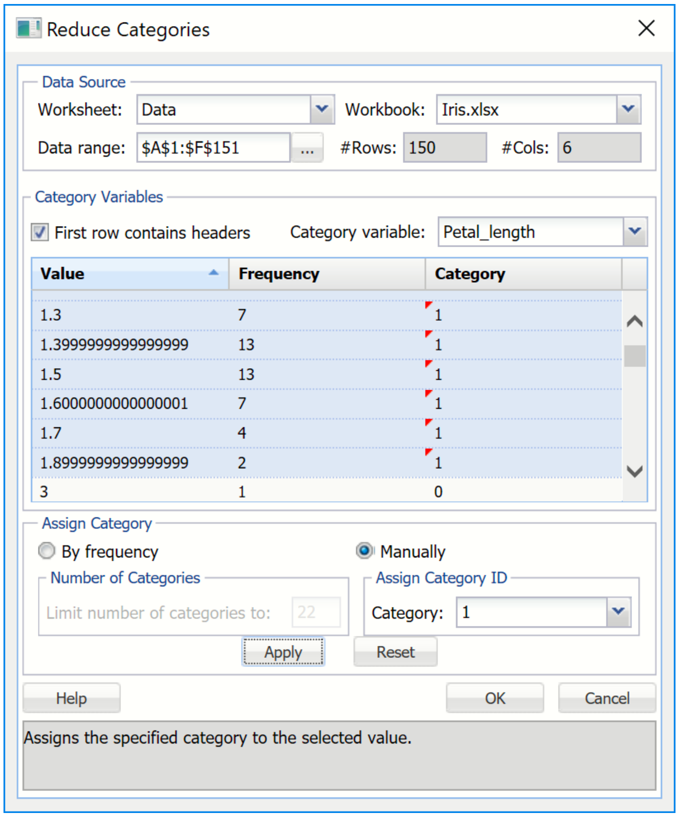
Limit number of categories to: 22

Assign Category ID

Category: 5

Notes:

- If using Analytic Solver Comprehensive or Data Science Cloud the maximum number of categories will be equal to the number of unique values for the selected variable. In this instance the *petal_length* variable contains 43 unique values**.**
- If *By Frequency* is selected, Analytic Solver Data Science assigns category numbers 1 through 29 to the most frequent 29 unique values; and category number 30 to all other unique values.



Reduce Categories

Data Source

Worksheet: Data | Workbook: Iris.xlsx

Data range: $A$1:$F$151 | #Rows: 150 | #Cols: 6

Category Variables

☑ First row contains headers | Category variable: Petal_length

Value	Frequency	Category
1.3	7	1
1.3999999999999999	13	1
1.5	13	1
1.6000000000000001	7	1
1.7	4	1
1.8999999999999999	2	1
3	1	0

Assign Category

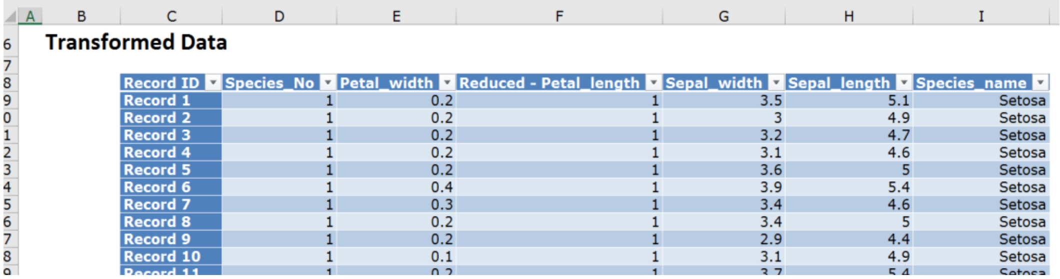
○ By frequency | ● Manually

Number of Categories: Limit number of categories to: 22

Assign Category ID: Category: 1

Assigns the specified category to the selected value.

Click **OK.** The output, Category_Reduction, is inserted into the Analytic Solver task pane under Data Science – Transformations – Reduce Categories.



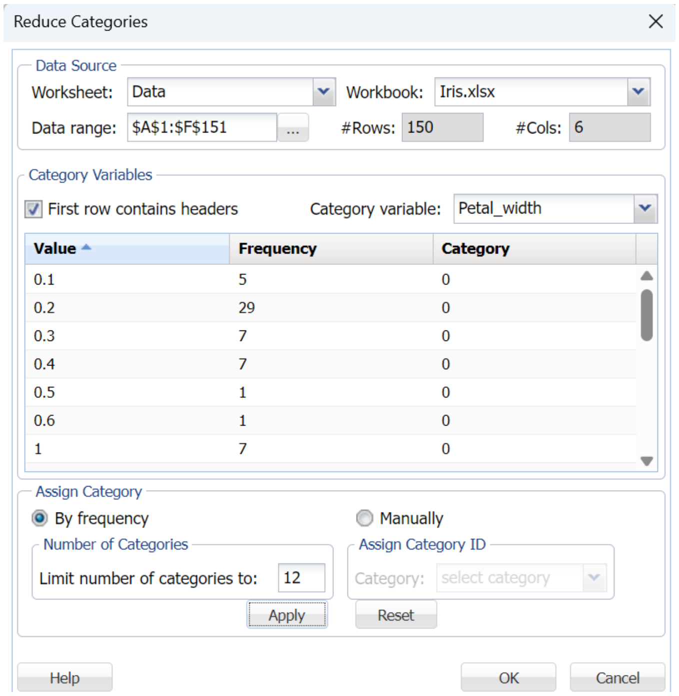
Transformed Data

Record ID	Species_No	Petal_width	Reduced - Petal_length	Sepal_width	Sepal_length	Species_name
Record 1	1	0.2	1	3.5	5.1	Setosa
Record 2	1	0.2	1	3	4.9	Setosa
Record 3	1	0.2	1	3.2	4.7	Setosa
Record 4	1	0.2	1	3.1	4.6	Setosa
Record 5	1	0.2	1	3.6	5	Setosa
Record 6	1	0.4	1	3.9	5.4	Setosa
Record 7	1	0.3	1	3.4	4.6	Setosa
Record 8	1	0.2	1	3.4	5	Setosa
Record 9	1	0.2	1	2.9	4.4	Setosa
Record 10	1	0.1	1	3.1	4.9	Setosa
Record 11	1	0.2	1	3.7	5.4	Setosa

In the output, Analytic Solver Data Science has assigned new categories as shown in the column, Reduced-*Petal_Length***,** based on the choices made in the *Reduce Categories* dialog.

Example 4 Reduce Categories By Frequency

Click back to the Data worksheet and once more open the Reduce Categories dialog. This time, select **Petal_width** in the *Category variable* field, leave the default setting of *By Frequency*, enter **12** for *Limit number of categories to,* click **Apply** then click **OK**.



Reduce Categories

Data Source

Worksheet: Data | Workbook: Iris.xlsx

Data range: $A$1:$F$151 | #Rows: 150 | #Cols: 6

Category Variables

☑ First row contains headers | Category variable: Petal_width

Value	Frequency	Category
0.1	5	0
0.2	29	0
0.3	7	0
0.4	7	0
0.5	1	0
0.6	1	0
1	7	0

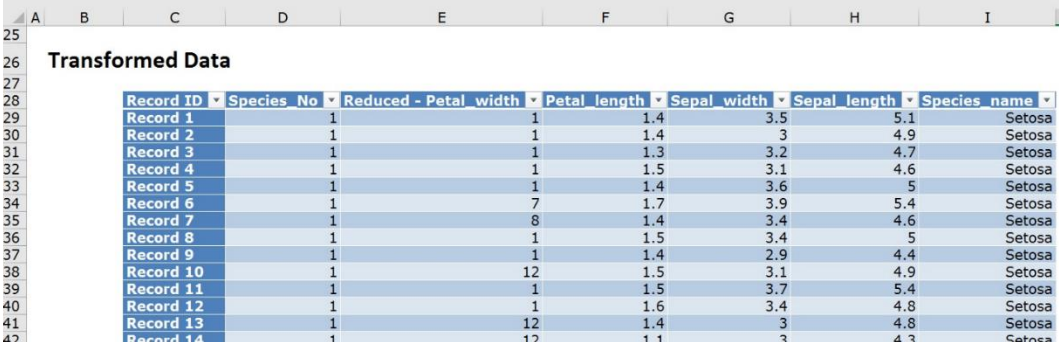
Assign Category

● By frequency | ○ Manually

Number of Categories: Limit number of categories to: 12

Assign Category ID: Category: select category

The output, *Category_Reduction1*, will be inserted into the task pane under Data Science – Transformations – Reduce Categories.



Transformed Data

Record ID	Species_No	Reduced - Petal_width	Petal_length	Sepal_width	Sepal_length	Species_name
Record 1	1	1	1.4	3.5	5.1	Setosa
Record 2	1	1	1.4	3	4.9	Setosa
Record 3	1	1	1.3	3.2	4.7	Setosa
Record 4	1	1	1.5	3.1	4.6	Setosa
Record 5	1	1	1.4	3.6	5	Setosa
Record 6	1	7	1.7	3.9	5.4	Setosa
Record 7	1	8	1.4	3.4	4.6	Setosa
Record 8	1	1	1.5	3.4	5	Setosa
Record 9	1	1	1.4	2.9	4.4	Setosa
Record 10	1	12	1.5	3.1	4.9	Setosa
Record 11	1	1	1.5	3.7	5.4	Setosa
Record 12	1	1	1.6	3.4	4.8	Setosa
Record 13	1	12	1.4	3	4.8	Setosa
Record 14	1	12	1.1	3	4.3	Setosa

There are 22 unique values for *Petal_width* and Analytic Solver Data Science has classified the *Petal_width* variable using 12 different categories. The most frequently appearing value is 0.2 (with 29 instances) which has been assigned to category 1. The second most frequently appearing value is 1.3 (with 13 instances) which has been assigned to category 2. See the chart below for all category assignments.

Value	Number of Instances	Assigned Category
0.2	29	1
1.3	13	2
1.8	12	3
1.5	12	4
2.3	8	5
1.4	8	6
0.4	7	7

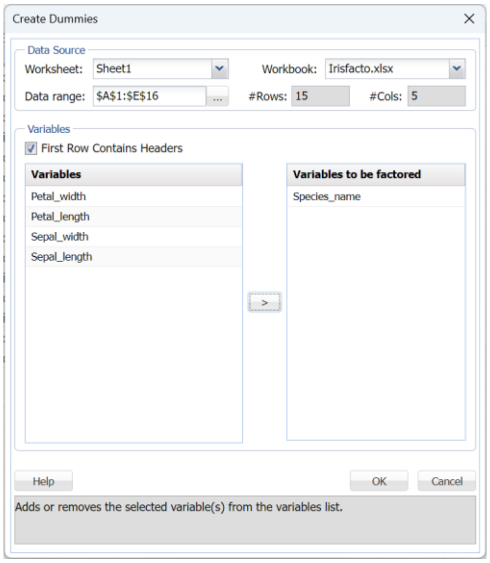
0.3	7	8
1.0	7	9
2.1	6	10
2.0	6	11
All Remaining Values	35	12

Incrementally increased category numbers are assigned to each value as the number of instances decreases until the 11th category is assigned. All remaining values are then lumped into the last category, 12.

Options for Transforming Categorical Data

Explanations for options that appear on one of the three Transform Categorical Data dialogs appear below.

Create Dummies Dialog



Worksheet: Sheet1
Workbook: Irisfacto.xlsx
Data range: $A$1:$E$16
#Rows: 15
#Cols: 5

☑ First Row Contains Headers

Variables	Variables to be factored
Petal_width	Species_name
Petal_length	
Sepal_width	
Sepal_length	

Adds or removes the selected variable(s) from the variables list.

Data Range

Either type the cell address directly into this field, or using the reference button; select the required data range from the worksheet or data set. If the cell pointer (active cell) is already somewhere in the data range, Analytic Solver Data Science automatically picks up the contiguous data range surrounding the active cell. When the data range is selected Analytic Solver Data Science displays the number of records in the selected range.

First row contains headers

When this box is checked, Analytic Solver Data Science lists the variables according to the first row of the selected data range. When the box is unchecked, Analytic Solver Data Science follows the default naming convention, i.e., the variable in the first column of the selected range will be called "Var1", the second column "Var2," etc.

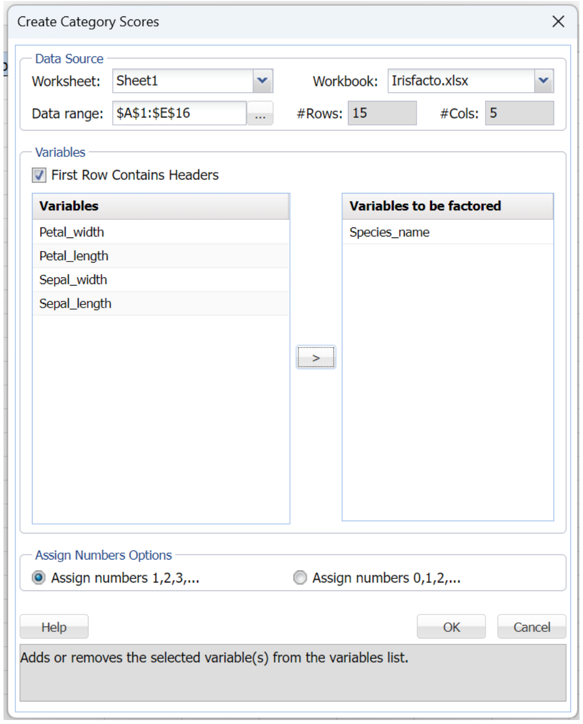
Variables

This list box contains the names of the variables in the selected data range. To select a variable, simply click to highlight, then click the > button. Use the CTRL or SHIFT keys to select multiple variables.

Variables to be factored

This list box contains the names of the input variables or the variables that will be replaced with dummy variables. To remove a variable, simply click to highlight, then click the < button. Use the CTRL or SHIFT keys to select multiple variables.

Create Category Scores Dialog



Data Source

Worksheet: Sheet1 | Workbook: Irisfacto.xlsx

Data range: $A$1:$E$16 | #Rows: 15 | #Cols: 5

Variables

☑ First Row Contains Headers

Variables	Variables to be factored
Petal_width	Species_name
Petal_length	
Sepal_width	
Sepal_length	

Assign Numbers Options

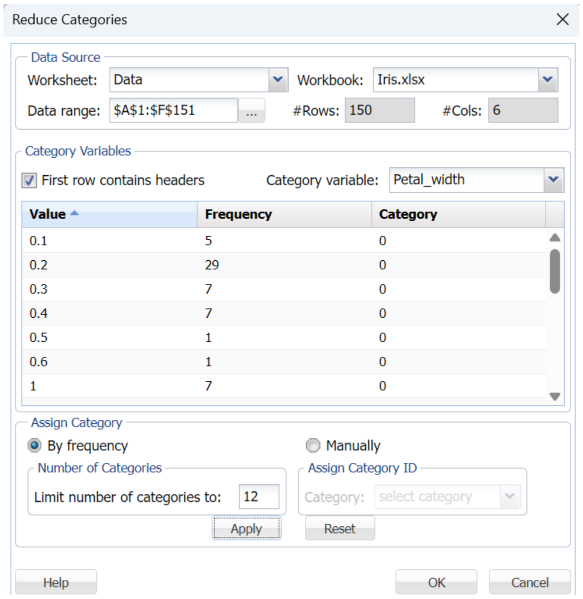
● Assign numbers 1,2,3,... ○ Assign numbers 0,1,2,...

Adds or removes the selected variable(s) from the variables list.

Assign Numbers Options

The user can specify the number with which to start categorization 0 or 1. Select the appropriate option.

Reduce Categories Dialog



Data Source

Worksheet: Data Workbook: Iris.xlsx

Data range: $A$1:$F$151 #Rows: 150 #Cols: 6

Category Variables

☑ First row contains headers Category variable: Petal_width

Value	Frequency	Category
0.1	5	0
0.2	29	0
0.3	7	0
0.4	7	0
0.5	1	0
0.6	1	0
1	7	0

Assign Category

● By frequency ○ Manually

Number of Categories

Limit number of categories to: 12

Assign Category ID

Category: select category

Category variable

Click the down arrow to select the desired variable for category reduction.

Assign Category

If *By frequency* is selected, incrementally increased category numbers will be assigned to each category as the number of instances decrease until the n - 1 category is assigned. All remaining values will then be lumped into the last category, n. If this option is selected, the *Limit number of categories to* option will be enabled.

If *Manually* is selected, Analytic Solver Data Science allows you to assign a specific category number to single or multiple categories using the *Assign Category ID* dropdown menu. If this option is selected, the *Category* option will be enabled.

Limit number of categories to

If *By frequency* is selected, *Limit number of categories to* is enabled. Enter a value from 1 to n-1 where n is the maximum number of unique values contained in the variable. Click **Apply** to apply this mapping, or **Reset** to start over.

Assign Category ID

If Manually is selected, *Assign Category ID* is enabled. Click the down arrow to select the Category number to assign to each unique value for the variable. This list will contain values from 1 to n where n is the maximum number of distinct values contained in the variable. Click **Apply** to apply this mapping, or **Reset** to start over.

Apply

Click the *Apply* command button to assign the specified category to the selected variable.

Reset

Click the *Reset* command button to reset all categories in the variable to unassigned.

Principal Components Analysis

Introduction

In the data science field, databases with large amounts of variables are routinely encountered. In most cases, the size of the database can be reduced by removing highly correlated or superfluous variables. The accuracy and reliability of a classification or regression model produced from this resultant database will be improved by the removal of these redundant and unnecessary variables. In addition, superfluous variables increase the data-collection and data-processing costs of deploying a model on a large database. As a result, one of the first steps in data science should be finding ways to reduce the number of independent or input variables used in the model (otherwise known as dimensionality) without sacrificing accuracy.

Dimensionality Reduction is the process of reducing the amount of variables to be used as input in a regression or classification model. This domain can be divided into two branches, feature selection and feature extraction. Feature selection attempts to discover a subset of the original variables while Feature Extraction attempts to map a high – dimensional model to a lower dimensional space. In the past, Analytic Solver (previously referred to as XLMiner) only contained a feature extraction tool, Principal Components Analysis (Transform – Principal Components). However, in V2015, a new feature selection tool was introduced, Feature Selection. This chapter explains Analytic Solver Data Science's Principal Components Analysis functionality. For more information on Analytic Solver Data Science's Feature Selection tool, please see the previous chapter, "Feature Selection".

Principal component analysis (PCA) is a mathematical procedure that transforms a number of (possibly) correlated variables into a smaller number of uncorrelated variables called principal components. The objective of principal component analysis is to reduce the dimensionality (number of variables) of the dataset but retain as much of the original variability in the data as possible. The first principal component accounts for the majority of the variability in the data, the second principal component accounts for the majority of the remaining variability, and so on.

A principal component analysis is concerned with explaining the variance covariance structure of a high dimensional random vector through a few linear combinations of the original component variables. Consider a database X with m rows and n columns (X_{4x3})

X_{11}	X_{12}	X_{13}
X_{21}	X_{22}	X_{23}
X_{31}	X_{32}	X_{33}
X_{41}	X_{42}	X_{43}

1. The first step in reducing the number of columns (variables) in the X matrix using the Principal Components Analysis algorithm is to find the mean of each column.

   $(X_{11} + X_{21} + X_{31} + X_{41})/4 = Mu_1$

$(X_{12} + X_{22} + X_{32} + X_{42})/4 = Mu_2$

$(X_{13} + X_{23} + X_{33} + X_{43})/4 = Mu_3$

2. Next, the algorithm subtracts each element in the database by the mean (Mu) thereby obtaining a new matrix, $\ddot{X}$, which also contains 4 rows and 3 columns.

$X_{11} – Mu_1 = \ddot{X}_{11}$	$X_{12} – Mu_2 = \ddot{X}_{12}$	$X_{13} – Mu_3 = \ddot{X}_{13}$
$X_{21} – Mu_1 = \ddot{X}_{21}$	$X_{22} – Mu_2 = \ddot{X}_{22}$	$X_{23} – Mu_3 = \ddot{X}_{23}$
$X_{31} – Mu_1 = \ddot{X}_{31}$	$X_{32} – Mu_2 = \ddot{X}_{32}$	$X_{33} – Mu_3 = \ddot{X}_{33}$
$X_{41} – Mu_1 = \ddot{X}_{41}$	$X_{42} – Mu_2 = \ddot{X}_{42}$	$X_{43} – Mu_3 = \ddot{X}_{43}$

3. Next, the PCA algorithm calculates the covariance or correlation matrix (depending on the user's preference) of the new $\ddot{X}$ matrix.

4. Afterwards the algorithm calculates eigenvalues and eigenvectors from the covariance matrix for each variable and lists these eigenvalues in order from largest to smallest.

   Larger eigenvalues denote that the variable should remain in the database. Variables with smaller eigenvalues will be removed according to the user's preference.

5. Analytic Solver Data Science allows users to choose between selecting a fixed number of components (variables) to be included in the "reduced" matrix (we will refer to this new matrix as the Y matrix) or the smallest subset of variables that "explains" or accounts for a certain percentage variance in the database. Variables with eigenvalues below the chosen threshold will not be included in the Y matrix. Assume that the user has chosen a fixed number of variables (2) to be included in the Y matrix.

6. A new matrix V (containing eigenvectors based on the selected eigenvalues) is formed.

7. The original matrix X which has 4 rows and 3 columns will be multiplied by the V matrix, containing 4 rows and 2 columns. This matrix multiplication results in the new reduced Y matrix, containing 4 rows and 2 columns.

In algebraic form, consider a p-dimensional random vector $\underline{X} = (X_1, X_2, ..., X_p)$ where p principal components of $\underline{X}$ are k univariate random variables $Y_1, Y_2, ..., Y_k$ which are defined by the following formulae:

$$Y_1 = l_1'\underline{X} = l_{11}X_1 + l_{12}X_2 + \ldots + l_{1p}X_p$$

$$Y_2 = l_2'\underline{X} = l_{21}X_1 + l_{22}X_2 + \ldots + l_{2p}X_p$$

$$\vdots$$

$$Y_k = l_k'\underline{X} = l_{k1}X_1 + l_{k2}X_2 + \ldots + l_{pk}X_p$$

where the coefficient vectors l_1, l_2 ,..etc. are chosen such that they satisfy the following conditions:

First Principal Component = Linear combination $l_1'\underline{X}$ that maximizes Var($l_1'\underline{X}$) and $\| l_1 \| =1$

Second Principal Component = Linear combination $l_2'\underline{X}$ that maximizes Var($l_2'\underline{X}$) and $\| l_2 \| =1$

and Cov($l_1'\underline{X}$, $l_2'\underline{X}$) =0 jth Principal Component = Linear combination $l_j'\underline{X}$ that maximizes Var($l_j'\underline{X}$) and $\| l_j \|$ =1

and Cov($l_k'\underline{X}$, $l_j'\underline{X}$) =0 for all k < j

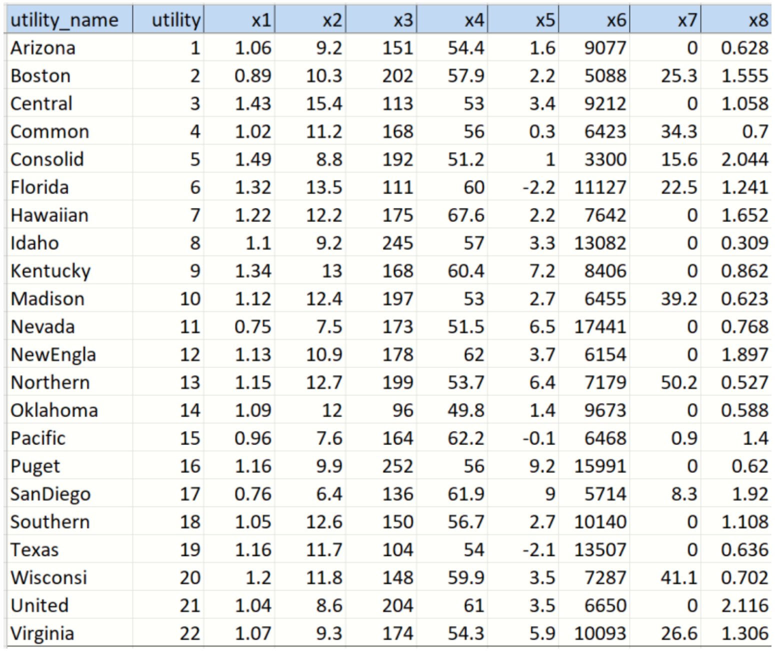
These functions indicate that the principal components are those linear combinations of the original variables which maximize the variance of the linear combination and which have zero covariance (and hence zero correlation) with the previous principal components.

It can be proved that there are exactly p such linear combinations. However, typically, the first few principal components explain most of the variance in the original data. As a result, instead of working with all the original variables X_1, X_2, ..., X_p, you would typically first perform PCA and then use only the first two or three principal components, say Y_1 and Y_2, in a subsequent analysis.

Examples for Principal Components

Two examples appear in this section to illustrate the Principal Components Analysis Tool in Analytic Solver Data Science. Each example uses the example file, Utilities.xlsx. This example dataset gives data on 22 public utilities within the US.

Open this dataset by clicking, Help – Example Models on the Data Science ribbon, then, Forecasting/Data Science Examples -- Utilities.xlsx

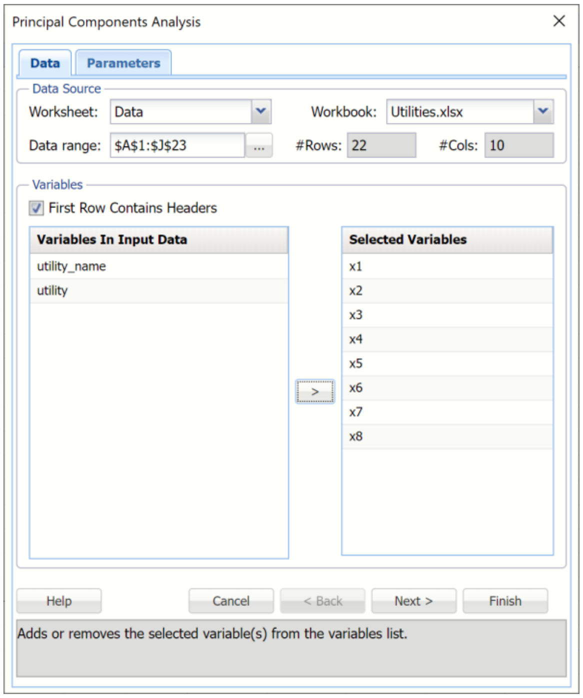


utility_name	utility	x1	x2	x3	x4	x5	x6	x7	x8
Arizona	1	1.06	9.2	151	54.4	1.6	9077	0	0.628
Boston	2	0.89	10.3	202	57.9	2.2	5088	25.3	1.555
Central	3	1.43	15.4	113	53	3.4	9212	0	1.058
Common	4	1.02	11.2	168	56	0.3	6423	34.3	0.7
Consolid	5	1.49	8.8	192	51.2	1	3300	15.6	2.044
Florida	6	1.32	13.5	111	60	-2.2	11127	22.5	1.241
Hawaiian	7	1.22	12.2	175	67.6	2.2	7642	0	1.652
Idaho	8	1.1	9.2	245	57	3.3	13082	0	0.309
Kentucky	9	1.34	13	168	60.4	7.2	8406	0	0.862
Madison	10	1.12	12.4	197	53	2.7	6455	39.2	0.623
Nevada	11	0.75	7.5	173	51.5	6.5	17441	0	0.768
NewEngla	12	1.13	10.9	178	62	3.7	6154	0	1.897
Northern	13	1.15	12.7	199	53.7	6.4	7179	50.2	0.527
Oklahoma	14	1.09	12	96	49.8	1.4	9673	0	0.588
Pacific	15	0.96	7.6	164	62.2	-0.1	6468	0.9	1.4
Puget	16	1.16	9.9	252	56	9.2	15991	0	0.62
SanDiego	17	0.76	6.4	136	61.9	9	5714	8.3	1.92
Southern	18	1.05	12.6	150	56.7	2.7	10140	0	1.108
Texas	19	1.16	11.7	104	54	-2.1	13507	0	0.636
Wisconsi	20	1.2	11.8	148	59.9	3.5	7287	41.1	0.702
United	21	1.04	8.6	204	61	3.5	6650	0	2.116
Virginia	22	1.07	9.3	174	54.3	5.9	10093	26.6	1.306

Fixed # Components Example

This example uses a fixed number of components in the ‘reduced’ model.

Click **Transform – Principal Components** on the Data Science ribbon to open the *Principal Components Analysis* dialog. On the Data tab, select variables **x1 to x8**, then click the > command button to move them to the *Selected Variables* field.



Data Source

Worksheet: Data

Workbook: Utilities.xlsx

Data range: $A$1:$J$23

#Rows: 22

#Cols: 10

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
utility_name	x1
utility	x2
	x3
	x4
	x5
	x6
	x7
	x8

Adds or removes the selected variable(s) from the variables list.

Click **Next** to move to the Parameters tab.

Analytic Solver Data Science provides two routines for specifying the number of principal components in the model: *# Components* and *Minimum Variance Cutoff*.

- The *# Components* (the default) method allows the user to specify a fixed number of components, or variables, to be included in the "reduced" model. The default setting for this option is equal to the number of selected variables. This value can be decreased to 1.
- The *Minimum Variance Cutoff* method allows the user to specify a percentage of the variance. When this method is selected, Analytic Solver Data Science will calculate the minimum number of principal components required to account for that percentage of the variance.

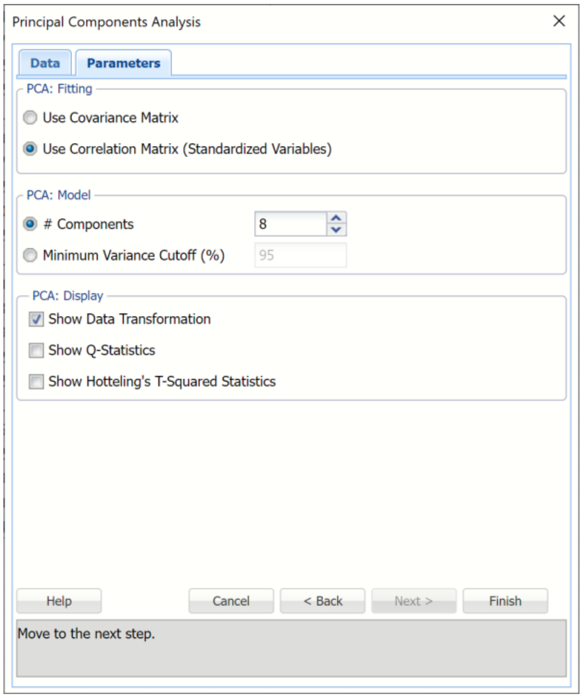
In addition, Analytic Solver Data Science provides two methods for calculating the principal components: using the covariance or the correlation matrix.

When using the **correlation** matrix method, the data will be normalized first before the method is applied. (The dataset is normalized by dividing each variable by its standard deviation.) Normalizing gives all variables equal importance in terms of variability. If the **covariance** method is selected, *the dataset should first be normalized.* Note: The Correlation Matrix is used only for internal calculations and is not included in the output results.

Under PCA: Display, confirm **Show Data Transformation** is selected by default. This option displays an output matrix where the columns are the principal components, the rows are the individual data records and the value in each cell is the calculated score for that record on the relevant principal component.

For a description of *Show Q-Statistics* and *Show Hotteling's T-Squared Statistics* options, please see the Principal Components Options section below.

Click **Finish** to run PCA.



Principal Components Analysis

PCA: Fitting

○ Use Covariance Matrix

● Use Correlation Matrix (Standardized Variables)

PCA: Model

● # Components: 8

○ Minimum Variance Cutoff (%): 95

PCA: Display

☑ Show Data Transformation

□ Show Q-Statistics

□ Show Hotteling's T-Squared Statistics

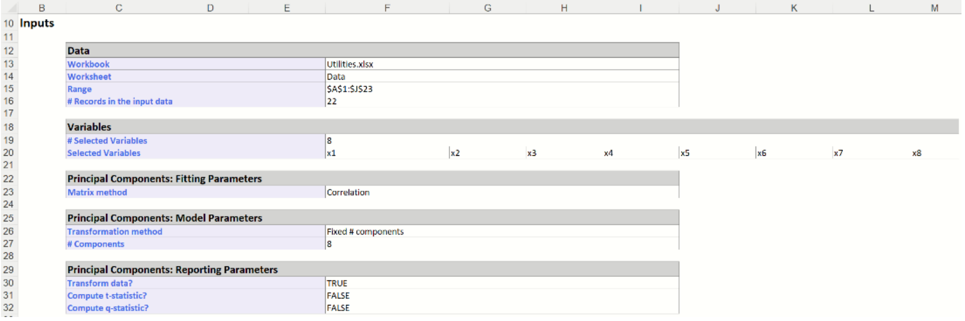
Move to the next step.

PCA Output Worksheets

Two worksheets are inserted to the right of the Data worksheet: PCA_Output and PCA_Scores. PCA_Output contains the Inputs, the Principal Components table and the Explained Variance table. PCA_Scores contains the Scores table.

PCA_Output

Click the *PCA_Output* tab. The inputs table is shown below. This table displays the options selected on both tabs of the Principal Components Analysis dialog.

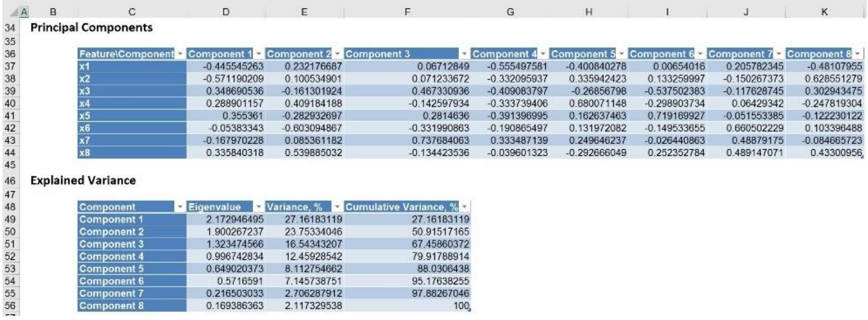


Inputs

Data								
Workbook	Utilities.xlsx							
Worksheet	Data							
Range	$A$1:$J$23							
# Records in the input data	22							
Variables								
# Selected Variables	8							
Selected Variables	x1	x2	x3	x4	x5	x6	x7	x8
Principal Components: Fitting Parameters								
Matrix method	Correlation							
Principal Components: Model Parameters								
Transformation method	Fixed # components							
# Components	8							
Principal Components: Reporting Parameters								
Transform data?	TRUE							
Compute t-statistic?	FALSE							
Compute q-statistic?	FALSE							

Further down the *PCA_Output* worksheet is the Principal Components table. The maximum *magnitude* element for Component1 corresponds to x2 (-0.5712). This signifies that the first principal component is measuring the effect of x2 on the utility companies. Likewise, the second component appears to be measuring the effect of x6 on the utility companies (maximum magnitude = |-0.6031|).

Take a look at the Explained Variance table. The Variance, % column displays Component1 as accounting for 27.16% of the variance while the second component accounts for 23.75%. Together, these two components account for more than 50% of the total variation. You can alternatively say the maximum magnitude element for component 1 corresponds to x2.



Principal Components

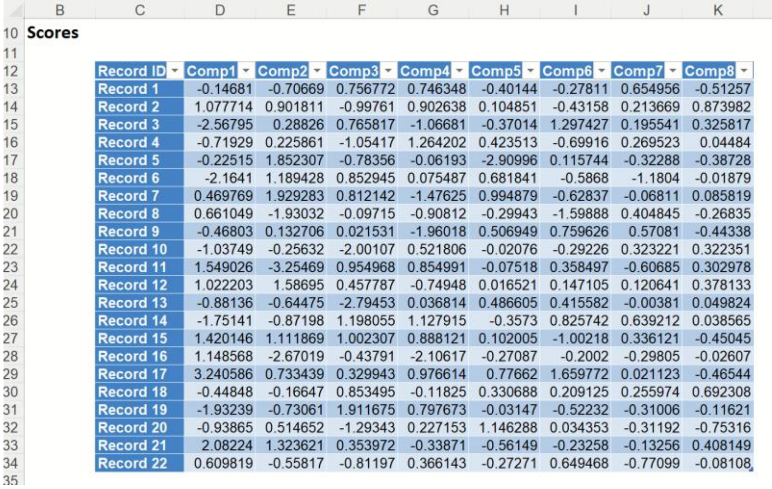
Feature\Component	Component 1	Component 2	Component 3	Component 4	Component 5	Component 6	Component 7	Component 8
x1	-0.445545263	0.232176687	0.06712849	-0.555497581	-0.400840278	0.00654016	0.205782345	-0.48107955
x2	-0.571190209	0.100534901	0.071233672	-0.332095937	0.335942423	0.133259997	-0.150267373	0.628551279
x3	0.348690536	-0.161301924	0.467330936	-0.409083797	-0.26856798	-0.537502383	-0.117628745	0.302943475
x4	0.288901157	0.409184188	-0.142597934	-0.333739406	0.680071148	-0.298903734	0.06429342	-0.247819304
x5	0.355361	-0.282932697	0.2814636	-0.391396995	0.162637463	0.719169927	-0.051553385	-0.122230122
x6	-0.05383343	-0.603094867	-0.331990863	-0.190865497	0.131972082	-0.149533655	0.660502229	0.103396488
x7	-0.167970228	0.085361182	0.737684063	0.333487139	0.249646237	-0.026440863	0.48879175	-0.084665723
x8	0.335840318	0.539885032	-0.134423536	-0.039601323	-0.292666049	0.252352784	0.489147071	0.43300956

Explained Variance

Component	Eigenvalue	Variance, %	Cumulative Variance, %
Component 1	2.172946495	27.16183119	27.16183119
Component 2	1.900267237	23.75334046	50.91517165
Component 3	1.323474566	16.54343207	67.45860372
Component 4	0.996742834	12.45928542	79.91788914
Component 5	0.649020373	8.112754662	88.0306438
Component 6	0.5716591	7.145738751	95.17638255
Component 7	0.216503033	2.706287912	97.88267046
Component 8	0.169386363	2.117329538	100

PCA_Scores

Double click *PCA_Scores* in the taskpane or click the *PCA_Scores* worksheet tab to view the Principal Components table. This table holds the weighted averages of the normalized variables (after each variable's mean is subtracted). (This matrix is described in the 2nd step of the PCA algorithm **-** see *Introduction* above.) Again, we are looking for the magnitude or absolute value of each figure in the table.



Scores

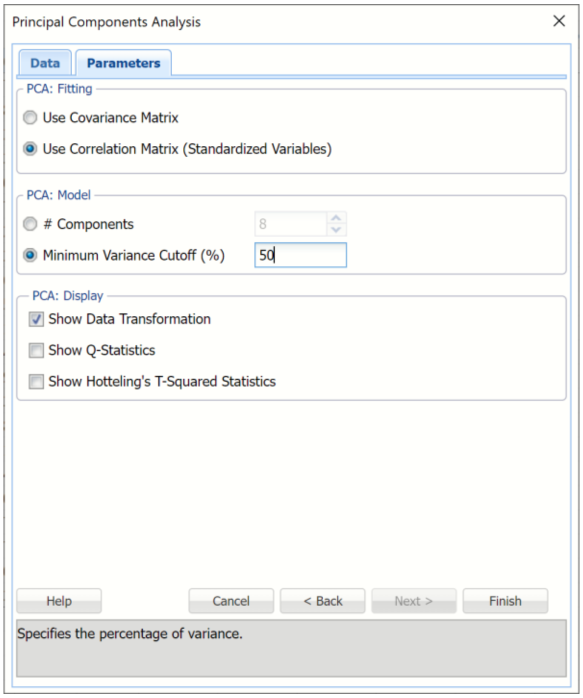
Record ID	Comp1	Comp2	Comp3	Comp4	Comp5	Comp6	Comp7	Comp8
Record 1	-0.14681	-0.70669	0.756772	0.746348	-0.40144	-0.27811	0.654956	-0.51257
Record 2	1.077714	0.901811	-0.99761	0.902638	0.104851	-0.43158	0.213669	0.873982
Record 3	-2.56795	0.28826	0.765817	-1.06681	-0.37014	1.297427	0.195541	0.325817
Record 4	-0.71929	0.225861	-1.05417	1.264202	0.423513	-0.69916	0.269523	0.04484
Record 5	-0.22515	1.852307	-0.78356	-0.06193	-2.90996	0.115744	-0.32288	-0.38728
Record 6	-2.1641	1.189428	0.852945	0.075487	0.681841	-0.5868	-1.1804	-0.01879
Record 7	0.469769	1.929283	0.812142	-1.47625	0.994879	-0.62837	-0.06811	0.085819
Record 8	0.661049	-1.93032	-0.09715	-0.90812	-0.29943	-1.59888	0.404845	-0.26835
Record 9	-0.46803	0.132706	0.021531	-1.96018	0.506949	0.759626	0.57081	-0.44338
Record 10	-1.03749	-0.25632	-2.00107	0.521806	-0.02076	-0.29226	0.323221	0.322351
Record 11	1.549026	-3.25469	0.954968	0.854991	-0.07518	0.358497	-0.60685	0.302978
Record 12	1.022203	1.58695	0.457787	-0.74948	0.016521	0.147105	0.120641	0.378133
Record 13	-0.88136	-0.64475	-2.79453	0.036814	0.486605	0.415582	-0.00381	0.049824
Record 14	-1.75141	-0.87198	1.198055	1.127915	-0.3573	0.825742	0.639212	0.038565
Record 15	1.420146	1.111869	1.002307	0.888121	0.102005	-1.00218	0.336121	-0.45045
Record 16	1.148568	-2.67019	-0.43791	-2.10617	-0.27087	-0.2002	-0.29805	-0.02607
Record 17	3.240586	0.733439	0.329943	0.976614	0.77662	1.659772	0.021123	-0.46544
Record 18	-0.44848	-0.16647	0.853495	-0.11825	0.330688	0.209125	0.255974	0.692308
Record 19	-1.93239	-0.73061	1.911675	0.797673	-0.03147	-0.52232	-0.31006	-0.11621
Record 20	-0.93865	0.514652	-1.29343	0.227153	1.146288	0.034353	-0.31192	-0.75316
Record 21	2.08224	1.323621	0.353972	-0.33871	-0.56149	-0.23258	-0.13256	0.408149
Record 22	0.609819	-0.55817	-0.81197	0.366143	-0.27271	0.649468	-0.77099	-0.08108

Minimum Variance Cutoff Example

The *Minimum Variance Cutoff* method allows the user to specify a percentage of the variance. When this method is selected, Analytic Solver Data Science will calculate the minimum number of principal components required to account for that percentage of the variance.

Click back to the Data sheet, then reopen the Principal Components Analysis dialog. Cells **x1 through x8** are already selected. Click **Next** on the dialog to advance to the Parameters tab.

This time, select **Minimum Variance Cutoff** and enter **50** for *%*. Keep **Use Correlation Matrix (Use Standardized Variables)** and **Show Data Transformation** selected, then click **Finish**.



Principal Components Analysis

PCA: Fitting

○ Use Covariance Matrix

● Use Correlation Matrix (Standardized Variables)

PCA: Model

○ # Components: 8

● Minimum Variance Cutoff (%): 50

PCA: Display

☑ Show Data Transformation

□ Show Q-Statistics

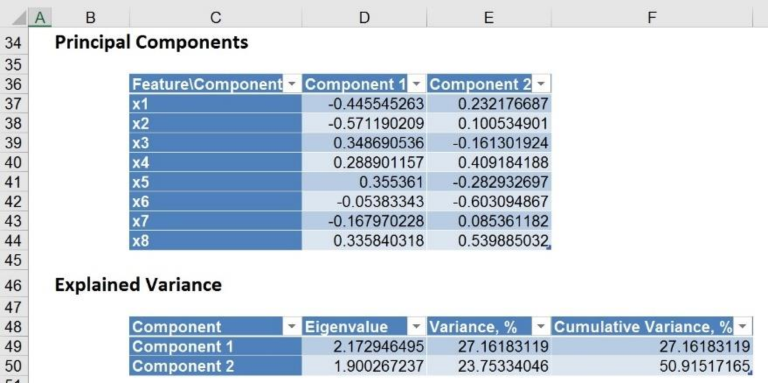
□ Show Hotteling's T-Squared Statistics

Specifies the percentage of variance.

Output Worksheets

Open *PCA_Output1* and scroll down to the Principal Components and Explained Variance tables. Notice that only the first two components are included in the output file since these two components account for over 50% of the variation.

Note: The Correlation Matrix is used only for internal calculations and is not included in the output results.



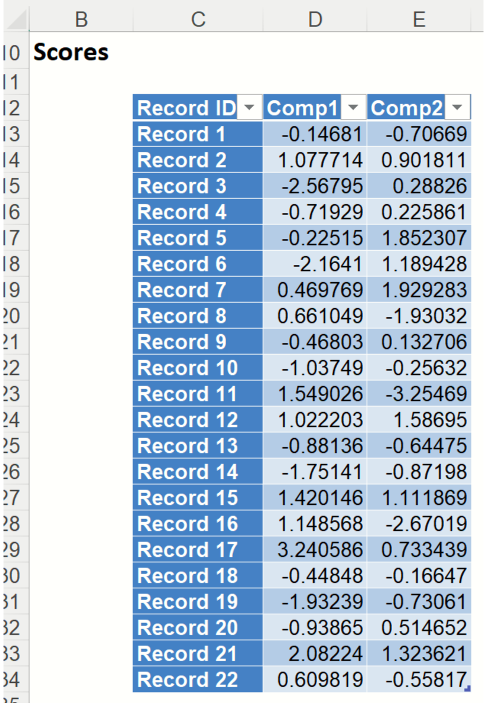
Principal Components

Feature\Component	Component 1	Component 2
x1	-0.445545263	0.232176687
x2	-0.571190209	0.100534901
x3	0.348690536	-0.161301924
x4	0.288901157	0.409184188
x5	0.355361	-0.282932697
x6	-0.05383343	-0.603094867
x7	-0.167970228	0.085361182
x8	0.335840318	0.539885032

Explained Variance

Component	Eigenvalue	Variance, %	Cumulative Variance, %
Component 1	2.172946495	27.16183119	27.16183119
Component 2	1.900267237	23.75334046	50.91517165

The output from *PCA_Scores1* is shown below. This table holds the weighted averages of the normalized variables (after each variable's mean is subtracted). (This matrix is described in the 2nd step of the PCA algorithm - see *Introduction* above.) Again, we are looking for the magnitude or absolute value of each figure in the table.



Scores

Record ID	Comp1	Comp2
Record 1	-0.14681	-0.70669
Record 2	1.077714	0.901811
Record 3	-2.56795	0.28826
Record 4	-0.71929	0.225861
Record 5	-0.22515	1.852307
Record 6	-2.1641	1.189428
Record 7	0.469769	1.929283
Record 8	0.661049	-1.93032
Record 9	-0.46803	0.132706
Record 10	-1.03749	-0.25632
Record 11	1.549026	-3.25469
Record 12	1.022203	1.58695
Record 13	-0.88136	-0.64475
Record 14	-1.75141	-0.87198
Record 15	1.420146	1.111869
Record 16	1.148568	-2.67019
Record 17	3.240586	0.733439
Record 18	-0.44848	-0.16647
Record 19	-1.93239	-0.73061
Record 20	-0.93865	0.514652
Record 21	2.08224	1.323621
Record 22	0.609819	-0.55817

After applying the Principal Components Analysis algorithm, users may proceed to analyze their dataset by applying additional data science algorithms featured in Analytic Solver Data Science.

Options for Principal Components Analysis

See below for an explanation of options on both tabs of the *Principal Components Analysis* (PCA) dialog*: Data* and *Parameters*.

The following options appear on all three tabs of the Principal Components Analysis dialog.

Help: Click the Help button to access documentation on all *Principal Components Analysis* options.

Cancel: Click the Cancel button to close the dialog without running *Principal Components Analysis.*

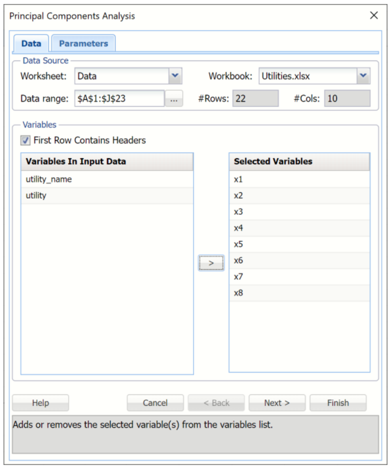
Next: Click the Next button to advance to the next tab.

Finish: Click Finish to accept all option settings on both dialogs, and run *Principal Components Analysis*.

Principal Components Analysis Data Tab

See below for documentation for all options appearing on the Data tab.

Principal Components Analysis Data Tab



Worksheet: Data

Workbook: Utilities.xlsx

Data range: $A$1:$J$23

#Rows: 22

#Cols: 10

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
utility_name	x1
utility	x2
	x3
	x4
	x5
	x6
	x7
	x8

Adds or removes the selected variable(s) from the variables list.

Data Source

Worksheet: Click the down arrow to select the desired worksheet where the dataset is contained.

Workbook: Click the down arrow to select the desired workbook where the dataset is contained.

Data range: Select or enter the desired data range within the dataset. This data range may either be a portion of the dataset or the complete dataset.

#Columns: Displays the number of columns in the data range. This option is read only.

#Rows: Displays the number of rows in the data range. This option is read only.

Variables

First Row Contains Headers: Select this checkbox if the first row in the dataset contains column headings.

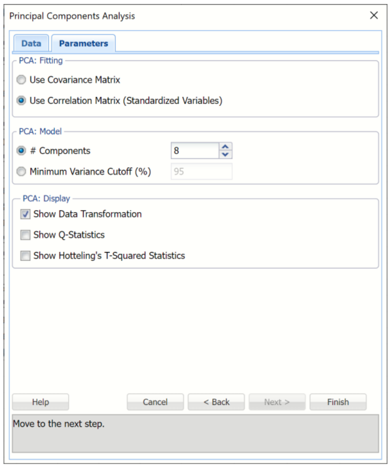
Variables In Input Data: This field contains the list of the variables, or features, included in the data range.

Selected Variables: This field contains the list of variables, or features, to be included in PCA.

- To include a variable in PCA, select the variable in the Variables In Input Data list, then click > to move the variable to the Selected Variables list.
- To remove a variable as a selected variable, click the variable in the Selected Variables list, then click < to move the variable back to the Variables In Input Data list.

Principal Components Analysis Parameters Tab

See below for documentation for all options appearing on the Principal Components Analysis Parameters tab.


Principal Components Analysis

PCA: Fitting

○ Use Covariance Matrix
● Use Correlation Matrix (Standardized Variables)

PCA: Model

● # Components: 8
○ Minimum Variance Cutoff (%): 95

PCA: Display

☑ Show Data Transformation
□ Show Q-Statistics
□ Show Hotteling's T-Squared Statistics

Move to the next step.

PCA: Fitting

To compute Principal Components the data is matrix multiplied by a transformation matrix. This option lets you specify the choice of calculating this transformation matrix.

Use Covariance matrix

The covariance matrix is a square, symmetric matrix of size n x n (number of variables by number of variables). The diagonal elements are variances and the off-diagonals are covariances. The eigenvalues and eigenvectors of the covariance matrix are computed and the transformation matrix is defined as the transpose of this eigenvector matrix. *If the covariance method is selected, the dataset should first be normalized.* One way to organize the data is to divide each variable by its standard deviation. Normalizing gives all variables equal importance in terms of variability.[3]

Use Correlation matrix (Standardized Variables)

An alternative method is to derive the transformation matrix on the eigenvectors of the correlation matrix instead of the covariance matrix. *The correlation matrix is equivalent to a covariance matrix for the data where each variable has been standardized to zero mean and unit variance.* This method tends to equalize the influence of each variable, inflating the influence of variables with relatively small variance and reducing the influence of variables with high variance. This option is selected by default.

PCA: Model

Select the number of principal components displayed in the output.

# components

Specify a fixed number of components by selecting this option and entering an integer value from 1 to n where n is the number of *Input variables* selected in the Data tab. This option is selected by default, the default value of n is equal to the number of input variables. This value can be decreased to 1.

[3] Shmueli, Galit, Nitin R. Patel, and Peter C. Bruce. Data Science for Business Intelligence. 2nd ed. New Jersey: Wiley, 2010

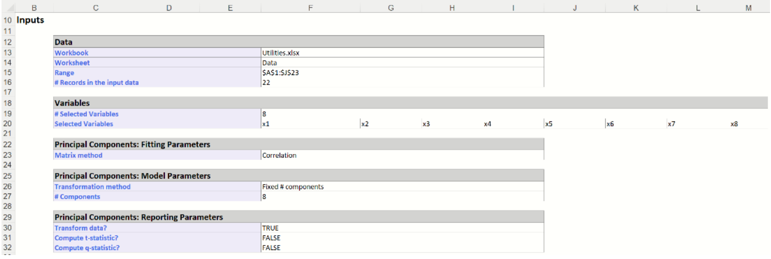
Minimum Variance Cutoff (%)

Select this option to specify a percentage. Analytic Solver Data Science will calculate the minimum number of principal components required to account for that percentage of variance.

PCA: Display

Select the type of output to be inserted into the workbook in this section of the Parameters tab. If no output is selected, by default, PC will output three tables: Inputs, Principal components and Explained Variance on the PCA_Output worksheet.

Inputs: This table displays the options selected on both tabs of the Principal Components Analysis dialog. *This table is accessible by clicking the Inputs link in the Output Navigator.*



Inputs

Data	
Workbook	Utilities.xlsx
Worksheet	Data
Range	$A$1:$J$23
# Records in the input data	22

Variables								
# Selected Variables	8							
Selected Variables	x1	x2	x3	x4	x5	x6	x7	x8

Principal Components: Fitting Parameters	
Matrix method	Correlation

Principal Components: Model Parameters	
Transformation method	Fixed # components
# Components	8

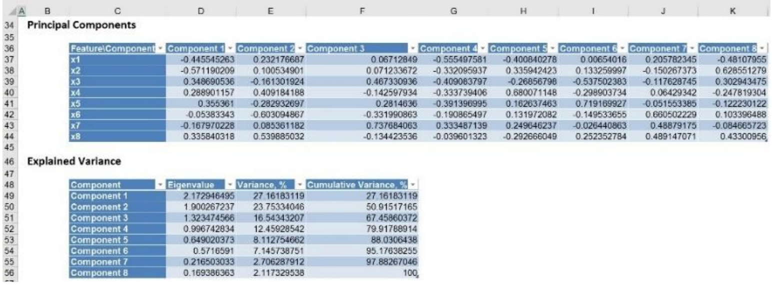
Principal Components: Reporting Parameters	
Transform data?	TRUE
Compute t-statistic?	FALSE
Compute q-statistic?	FALSE

Principal Components: This table displays how each variable affects each component.

In the example below, the maximum *magnitude* element for Component1 corresponds to x2 (-0.5712). This signifies that the first principal component is measuring the effect of x2 on the utility companies. Likewise, the second component appears to be measuring the effect of x6 on the utility companies (maximum magnitude = |-0.6031|). This table is accessible by clicking the Principal Components link in the Output Navigator.

Explained Variance: This table includes 3 columns: Eigenvalue, Variance, % and Cumulative Variance. *This table is accessible by clicking the Explained Variance link in the Output Navigator.*

- The Eigenvalue column displays the eigenvalues and eigenvectors computed from the covariance matrix for each variable. They are listed in order from largest to smallest. Larger eigenvalues denote that the variable should remain in the database. Variables with smaller eigenvalues will be removed according to the user's preference.
- The Variance, % column displays the variance attributed by each Component. In the example below, Component1 accounts for 27.16% of the variance while the second component accounts for 23.75%.
- The Cumulative Variance column displays the cumulative variance. In the example below, Components 1 and 2 account for more than 50% of the total variation. You can alternatively say the maximum magnitude element for component 1 corresponds to x2.



Principal Components

Feature\Component	Component 1	Component 2	Component 3	Component 4	Component 5	Component 6	Component 7	Component 8
x1	-0.445545263	0.232176687	0.06712849	-0.555497581	-0.400840278	0.00654016	0.205782345	-0.48107955
x2	-0.571190209	0.100534901	0.071233672	-0.332095937	0.335942423	0.133259997	-0.150267373	0.628551279
x3	0.348690536	-0.161301924	0.467330936	-0.409083797	-0.26856798	-0.537502383	-0.117628745	0.302943475
x4	0.288901157	0.409184188	-0.142597934	-0.333739406	0.680071148	-0.298903734	0.06429342	-0.247819304
x5	0.355361	-0.282932697	0.2814636	-0.391396995	0.162637463	0.719169927	-0.051553385	-0.122230122
x6	-0.05383343	-0.603094867	-0.331990863	-0.190865497	0.131972082	-0.149533655	0.660502229	0.103396488
x7	-0.167970228	0.085361182	0.737684063	0.333487139	0.249646237	-0.026440863	0.48879175	-0.084665723
x8	0.335840318	0.539885032	-0.134423536	-0.039601323	-0.292666049	0.252352784	0.489147071	0.43300956

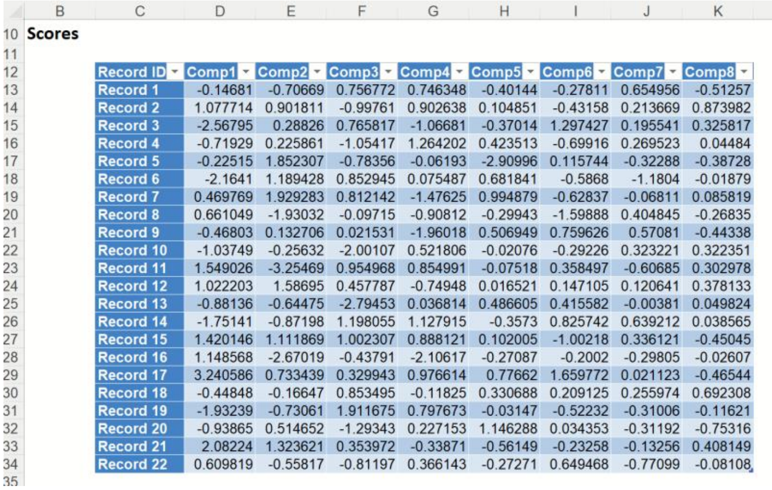
Explained Variance

Component	Eigenvalue	Variance, %	Cumulative Variance, %
Component 1	2.172946495	27.16183119	27.16183119
Component 2	1.900267237	23.75334046	50.91517165
Component 3	1.323474566	16.54343207	67.45860372
Component 4	0.996742834	12.45928542	79.91788914
Component 5	0.649020373	8.112754662	88.0306438
Component 6	0.5716591	7.145738751	95.17638255
Component 7	0.216503033	2.706287912	97.88267046
Component 8	0.169386363	2.117329538	100

Show Data Transformation

This option results in the display of a matrix, Scores, in which the columns are the principal components, the rows are the individual data records, and the value in each cell is the calculated score for that record on the relevant principal component. This option is selected by default.

This table holds the weighted averages of the normalized variables (after each variable's mean is subtracted). (This matrix is described in the 2nd step of the PCA algorithm **-** see *Introduction* above.) When reading the table, note the magnitude or absolute value of each value in the table. *This table is accessible by clicking the Scores link in the Output Navigator.*



Scores

Record ID	Comp1	Comp2	Comp3	Comp4	Comp5	Comp6	Comp7	Comp8
Record 1	-0.14681	-0.70669	0.756772	0.746348	-0.40144	-0.27811	0.654956	-0.51257
Record 2	1.077714	0.901811	-0.99761	0.902638	0.104851	-0.43158	0.213669	0.873982
Record 3	-2.56795	0.28826	0.765817	-1.06681	-0.37014	1.297427	0.195541	0.325817
Record 4	-0.71929	0.225861	-1.05417	1.264202	0.423513	-0.69916	0.269523	0.04484
Record 5	-0.22515	1.852307	-0.78356	-0.06193	-2.90996	0.115744	-0.32288	-0.38728
Record 6	-2.1641	1.189428	0.852945	0.075487	0.681841	-0.5868	-1.1804	-0.01879
Record 7	0.469769	1.929283	0.812142	-1.47625	0.994879	-0.62837	-0.06811	0.085819
Record 8	0.661049	-1.93032	-0.09715	-0.90812	-0.29943	-1.59888	0.404845	-0.26835
Record 9	-0.46803	0.132706	0.021531	-1.96018	0.506949	0.759626	0.57081	-0.44338
Record 10	-1.03749	-0.25632	-2.00107	0.521806	-0.02076	-0.29226	0.323221	0.322351
Record 11	1.549026	-3.25469	0.954968	0.854991	-0.07518	0.358497	-0.60685	0.302978
Record 12	1.022203	1.58695	0.457787	-0.74948	0.016521	0.147105	0.120641	0.378133
Record 13	-0.88136	-0.64475	-2.79453	0.036814	0.486605	0.415582	-0.00381	0.049824
Record 14	-1.75141	-0.87198	1.198055	1.127915	-0.3573	0.825742	0.639212	0.038565
Record 15	1.420146	1.111869	1.002307	0.888121	0.102005	-1.00218	0.336121	-0.45045
Record 16	1.148568	-2.67019	-0.43791	-2.10617	-0.27087	-0.2002	-0.29805	-0.02607
Record 17	3.240586	0.733439	0.329943	0.976614	0.77662	1.659772	0.021123	-0.46544
Record 18	-0.44848	-0.16647	0.853495	-0.11825	0.330688	0.209125	0.255974	0.692308
Record 19	-1.93239	-0.73061	1.911675	0.797673	-0.03147	-0.52232	-0.31006	-0.11621
Record 20	-0.93865	0.514652	-1.29343	0.227153	1.146288	0.034353	-0.31192	-0.75316
Record 21	2.08224	1.323621	0.353972	-0.33871	-0.56149	-0.23258	-0.13256	0.408149
Record 22	0.609819	-0.55817	-0.81197	0.366143	-0.27271	0.649468	-0.77099	-0.08108

Q Statistics and Hotteling's T-Squared Statistics

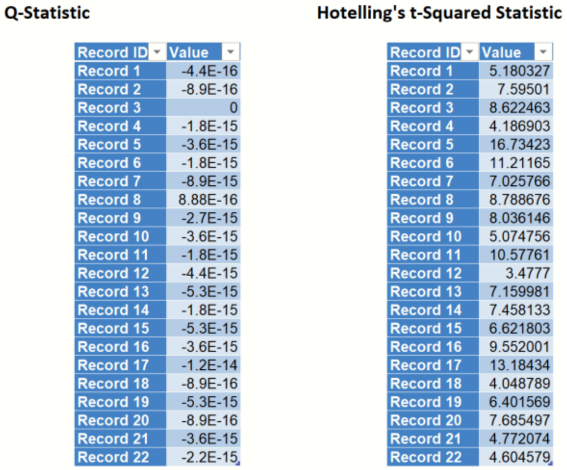
Q Statistics, or residuals, and Hotteling's T-Squared statistics are summary statistics which help explain how well a model fits the sample data and can also be used to detect any outliers in the data. A detailed explanation for each is beyond the scope of this guide. Please see the literature for more information on each of these statistics.

Show Q - Statistics

If this option is selected, Analytic Solver Data Science will include Q-Statistics in the output worksheet, PCA_Stats. Q statistics (or residuals) measure the difference between sample data and the projection of the model onto the sample data. These statistics an also be used to determine if any outliers exist in the data. Low values for Q statistics indicate a well fit model. *This table is also accessible by clicking the Q-Statistic link in the Output Navigator.*

Show Hotteling’s T-Squared Statistics

If this option is selected, Analytic Solver Data Science will include Hotteling’s T-Squared statistics in the output worksheet, PCA_Stats. T-Squared statistics measure the variation in the sample data within the mode and indicate how far the sample data is from the center of the model. These statistics can also be used to detect outliers in the sample data. Low T-Squared statistics indicate a well fit model. This table is accessible by clicking the *Hotteling’s t-Squared Statistic* link in the Output Navigator.



Q-Statistic

Record ID	Value
Record 1	-4.4E-16
Record 2	-8.9E-16
Record 3	0
Record 4	-1.8E-15
Record 5	-3.6E-15
Record 6	-1.8E-15
Record 7	-8.9E-15
Record 8	8.88E-16
Record 9	-2.7E-15
Record 10	-3.6E-15
Record 11	-1.8E-15
Record 12	-4.4E-15
Record 13	-5.3E-15
Record 14	-1.8E-15
Record 15	-5.3E-15
Record 16	-3.6E-15
Record 17	-1.2E-14
Record 18	-8.9E-16
Record 19	-5.3E-15
Record 20	-8.9E-16
Record 21	-3.6E-15
Record 22	-2.2E-15

Hotelling's t-Squared Statistic

Record ID	Value
Record 1	5.180327
Record 2	7.59501
Record 3	8.622463
Record 4	4.186903
Record 5	16.73423
Record 6	11.21165
Record 7	7.025766
Record 8	8.788676
Record 9	8.036146
Record 10	5.074756
Record 11	10.57761
Record 12	3.4777
Record 13	7.159981
Record 14	7.458133
Record 15	6.621803
Record 16	9.552001
Record 17	13.18434
Record 18	4.048789
Record 19	6.401569
Record 20	7.685497
Record 21	4.772074
Record 22	4.604579

k-Means Clustering

Introduction

Cluster Analysis, also called data segmentation, has a variety of goals which all relate to grouping or segmenting a collection of objects (also called observations, individuals, cases, or data rows) into subsets or "clusters". These “clusters” are grouped in such a way that the observations included in each cluster are more closely related to one another than objects assigned to different clusters. The most important goal of cluster analysis is the notion of the degree of similarity (or dissimilarity) between the individual objects being clustered. There are two major methods of clustering **--** hierarchical clustering and k-means clustering.

This chapter explains the k-Means Clustering algorithm. (See the Hierarchical Clustering chapter for information on this type of clustering analysis.) The goal of this process is to divide the data into a set number of clusters (k) and to assign each record to a cluster while minimizing the distribution within each cluster. A non-hierarchical approach to forming good clusters is to specify a desired number of clusters, say, k, then assign each case (object) to one of k clusters so as to minimize a measure of dispersion within the clusters. A very common measure is the sum of distances or sum of squared Euclidean distances from the mean of each cluster. The problem can be set up as an integer programming problem but because solving integer programs with a large number of variables is time consuming, clusters are often computed using a fast, heuristic method that generally produces good (but not necessarily optimal) solutions. The k-Means algorithm is one such method.

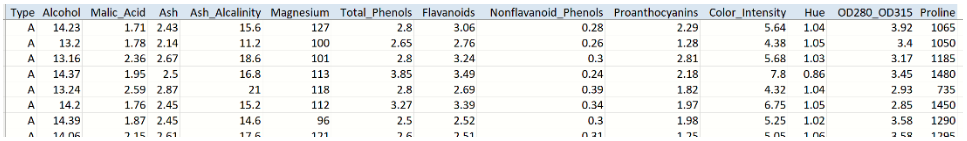
Example for k-Means Clustering

The example contained in this section uses the Wine.xlsx example file to demonstrate how to create a model using the k-Means Clustering algorithm.

Open this example file by clicking, **Help – Example Models,** then **Forecasting/Data Science Examples**.

As shown in the figure below, each row in this dataset represents a sample of wine taken from one of three wineries (A, B or C). In this example, the *Type* variable representing the winery is ignored and clustering is performed simply on the basis of the properties of the wine samples (the remaining variables).

Wine.xlsx dataset



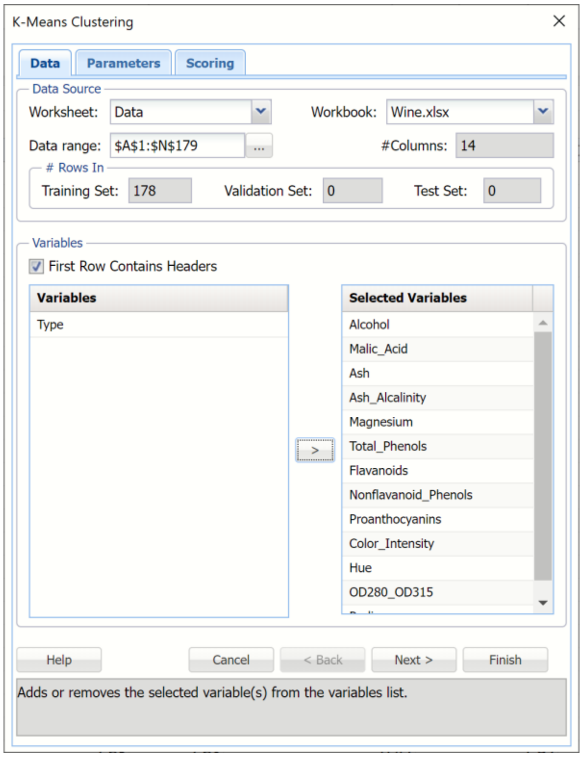
Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
A	14.23	1.71	2.43	15.6	127	2.8	3.06	0.28	2.29	5.64	1.04	3.92	1065
A	13.2	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.4	1050
A	13.16	2.36	2.67	18.6	101	2.8	3.24	0.3	2.81	5.68	1.03	3.17	1185
A	14.37	1.95	2.5	16.8	113	3.85	3.49	0.24	2.18	7.8	0.86	3.45	1480
A	13.24	2.59	2.87	21	118	2.8	2.69	0.39	1.82	4.32	1.04	2.93	735
A	14.2	1.76	2.45	15.2	112	3.27	3.39	0.34	1.97	6.75	1.05	2.85	1450
A	14.39	1.87	2.45	14.6	96	2.5	2.52	0.3	1.98	5.25	1.02	3.58	1290
A	14.06	2.15	2.61	17.6	121	2.6	2.51	0.31	1.25	5.05	1.06	3.58	1295

Inputs

Click **Data Science – Cluster – k-Means Clustering** to open the **k – Means Clustering** dialog.

Select all variables under *Variables* except *Type*, then click the > button to shift the selected variables to the *Selected Variables* field.

k-Means Clustering dialog, Data tab



Data Source

Worksheet: Data

Workbook: Wine.xlsx

Data range: $A$1:$N$179

#Columns: 14

Rows In

Training Set: 178

Validation Set: 0

Test Set: 0

Variables

☑ First Row Contains Headers

Variables	Selected Variables
Type	Alcohol
	Malic_Acid
	Ash
	Ash_Alcalinity
	Magnesium
	Total_Phenols
	Flavanoids
	Nonflavanoid_Phenols
	Proanthocyanins
	Color_Intensity
	Hue
	OD280_OD315

Adds or removes the selected variable(s) from the variables list.

Afterwards, click **Next** to advance to the next tab.

Enter **8** for *# Clusters* to instruct the k-Means Clustering algorithm to form 8 cohesive groups of observations in the Wine data. One can use the results of Hierarchical Clustering or several different values of k to understand the best setting for # Clusters.

Enter **10** for *# Iterations*. This option limits the number of iterations for the k-Means Clustering algorithm.

Random Seed is initialized to the default setting of "12345". This option initializes the random number generator that is used to assign the initial cluster centroids. Setting the random number seed to a positive value ensures the reproducibility of the analysis.

Increase *#Starts* to **5**. The final result of the k-Means Clustering algorithm depends on the initial choice on the cluster centroids. The best assignment (based on Sum of Squared Distances) is chosen as an initialization for further k-Means iterations.

Leave **Cluster Centers** selected under *Fitting* and **Fitting Metrics** selected under *Training Data*. For more information on the remaining output options, see the k-Means Clustering Options section immediately following this example.

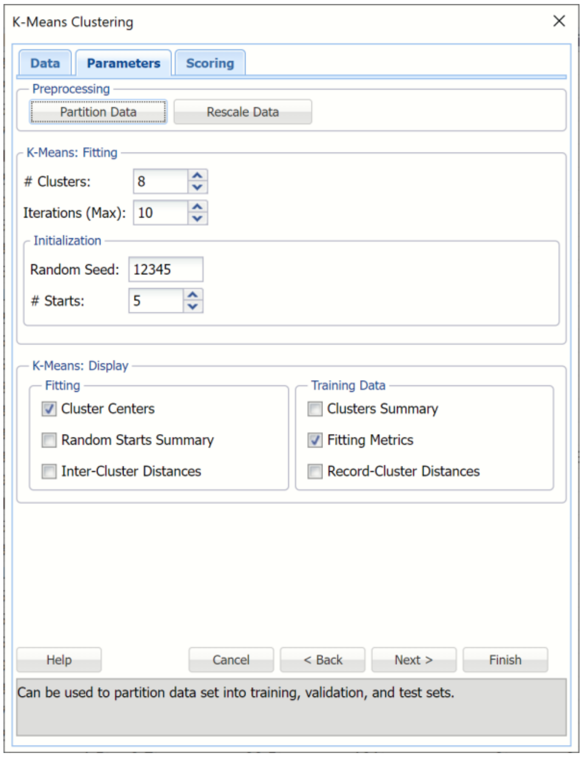
Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within the k-Means Clustering method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running k-Means Clustering. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning options, please see the Data Science Partitioning chapter that appears later in this guide. This example does not perform partitioning on the dataset.

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide. If rescaling has already been performed, this button will be disabled. This example does not utilize the Rescale Data feature.

k-Means Clustering dialog, Parameters tab



Clusters: 8

Iterations (Max): 10

Random Seed: 12345

Starts: 5

☑ Cluster Centers

☐ Random Starts Summary

☐ Inter-Cluster Distances

☐ Clusters Summary

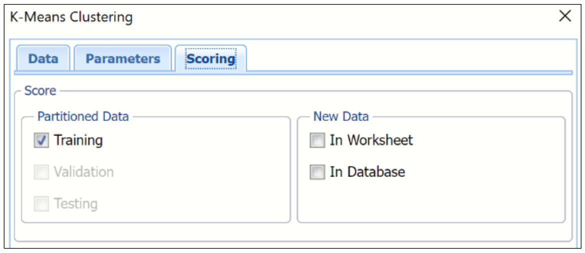
☑ Fitting Metrics

☐ Record-Cluster Distances

Can be used to partition data set into training, validation, and test sets.

Click **Next** to advance to the Scoring tab. Notice that Training, under Score Partitioned Data, is selected by default. Validation and Testing are disabled under Partitioned Data since partitioning was not performed on the dataset. Analytic Solver will score the training dataset. See the Scoring chapter in the Analytic Solver User Guide for information on scoring new data In worksheet or In Database.

k-Means Clustering dialog, Scoring tab



☑ Training

☐ Validation

☐ Testing

☐ In Worksheet

☐ In Database

Click **Finish**.

The k-Means Clustering method starts with k initial clusters. The algorithm proceeds by alternating between two steps: "assignment" – where each record is assigned to the cluster with the nearest centroid, and "update" – where new cluster centroids are recomputed based on the partitioning found in the "assignment" step.

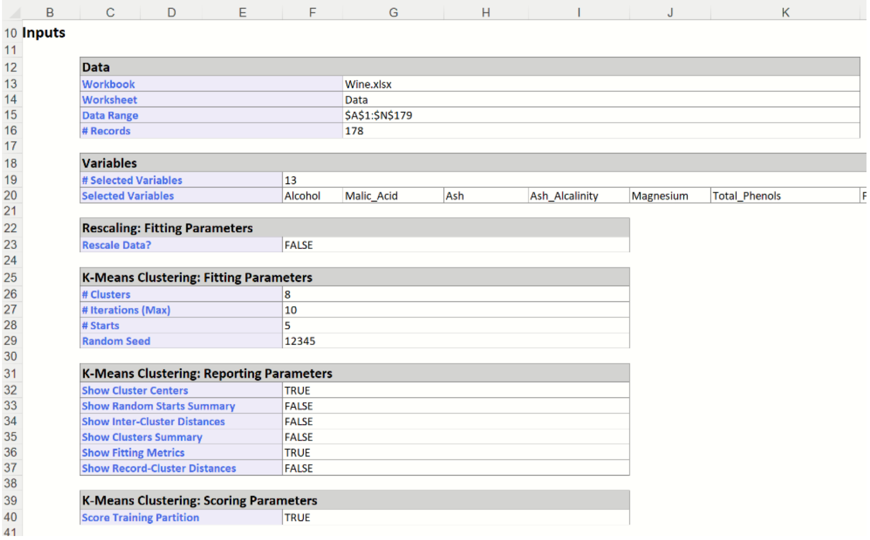
Results

The results of the clustering method, *KMC_Output* and *KMC_TrainingClusters*, are inserted to the right of the Data worksheet.

KMC_Output

In the top section of *KMC_Output*, the options that were selected in the Data, Parameters and Scoring tabs are listed.

k-Means Clustering Output: Inputs



Inputs

Data	
Workbook	Wine.xlsx
Worksheet	Data
Data Range	$A$1:$N$179
# Records	178

Variables							
# Selected Variables	13						
Selected Variables	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	F

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

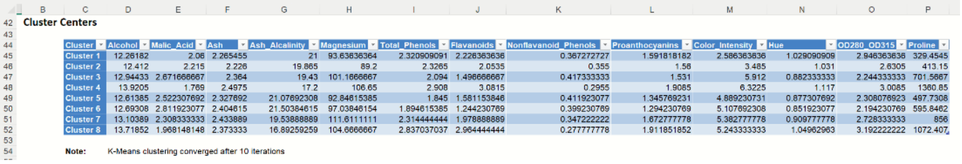
K-Means Clustering: Fitting Parameters	
# Clusters	8
# Iterations (Max)	10
# Starts	5
Random Seed	12345

K-Means Clustering: Reporting Parameters	
Show Cluster Centers	TRUE
Show Random Starts Summary	FALSE
Show Inter-Cluster Distances	FALSE
Show Clusters Summary	FALSE
Show Fitting Metrics	TRUE
Show Record-Cluster Distances	FALSE

K-Means Clustering: Scoring Parameters	
Score Training Partition	TRUE

Scroll down on the *KMC_Output* worksheet to the "Cluster Centers" tables (shown below – click the *Cluster Centers* link in the **Output Navigator** to view), displays detailed information about the clusters formed by the k-Means Clustering algorithm: the final centroids and inter-cluster distances. If the input data was normalized, k-Means Clustering operates on rescaled data. Results are not transformed back to their original scale.

k-Means Clustering Output: Cluster Centers



Cluster Centers

Cluster	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
Cluster 1	12.26182	2.06	2.265455	21	93.63636364	2.320909091	2.226363636	0.367272727	1.591818182	2.586363636	1.029090909	2.946363636	329.4545
Cluster 2	12.412	2.215	2.226	19.865	89.2	2.3265	2.0535	0.355	1.56	3.485	1.031	2.6305	413.15
Cluster 3	12.94433	2.671666667	2.364	19.43	101.1666667	2.094	1.496666667	0.417333333	1.531	5.912	0.882333333	2.244333333	701.5667
Cluster 4	13.9205	1.769	2.4975	17.2	106.65	2.908	3.0815	0.2955	1.9085	6.3225	1.117	3.0085	1360.85
Cluster 5	12.61385	2.522307692	2.327692	21.07692308	92.84615385	1.845	1.581153846	0.411923077	1.345769231	4.889230731	0.877307692	2.308076923	497.7308
Cluster 6	12.69308	2.811923077	2.404615	21.50384615	97.03846154	1.894615385	1.244230769	0.399230769	1.294230769	5.107692308	0.851923077	2.194230769	595.8462
Cluster 7	13.10389	2.308333333	2.433889	19.53888889	111.6111111	2.314444444	1.978888889	0.347222222	1.672777778	5.382777778	0.909777778	2.728333333	856
Cluster 8	13.71852	1.968148148	2.373333	16.89259259	104.6666667	2.837037037	2.964444444	0.277777778	1.911851852	5.243333333	1.04962963	3.192222222	1072.407

Note: K-Means clustering converged after 10 iterations

KMC_Training Clusters Output

Click the KMC_TrainingClusters worksheet to view the Fitting Metrics for the training dataset and Cluster assignments for each record.

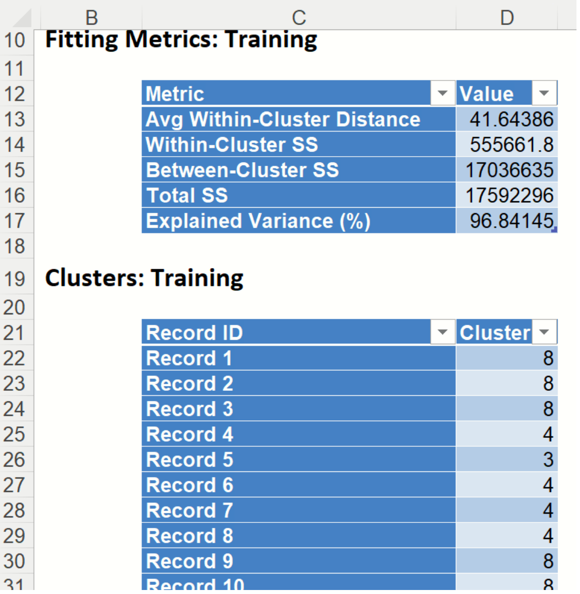
Fitting Metrics: This table lists several metrics computed from the training dataset. (Recall that our training dataset is the full dataset since we did not partition into training, validation or test partitions.)

- Avg Within-Cluster Distance – Average total distance from each record to the corresponding cluster center, for each cluster.
- Within Cluster SS – Sum of distances between the records and the corresponding cluster centers for each cluster. This statistic measures cluster compactness. (The algorithm is trying to minimize this measure.)
- Between Cluster SS – Sum of distances between the cluster centers and the total sample mean, divided by the number of records within each cluster. (Between Cluster SS measures cluster separation which the

algorithm is trying to maximize. This is equivalent to minimizing *Within Cluster SS*.)

- Total SS – Sum of distances from each record to the total sample mean, Total SS = Within Cluster SS + Between Cluster SS.
- Explained Variance (%) – Goodness of fit metrics, showing the degree of internal cohesion and external separation, Explained Variance % = Between Cluster SS / Total SS.

Clusters: Training - This table displays the final cluster assignment for each observation in the input data – the point is assigned to the "closest" cluster, i.e. the one with the nearest centroid.



Fitting Metrics: Training

Metric	Value
Avg Within-Cluster Distance	41.64386
Within-Cluster SS	555661.8
Between-Cluster SS	17036635
Total SS	17592296
Explained Variance (%)	96.84145

Clusters: Training

Record ID	Cluster
Record 1	8
Record 2	8
Record 3	8
Record 4	4
Record 5	3
Record 6	4
Record 7	4
Record 8	4
Record 9	8
Record 10	8

k-Means Clustering Options

See below for an explanation of options on all three tabs of the k-Means *Clustering* dialog*: Data, Parameters* and *Scoring* tabs.

The following options appear on all three tabs of the k-Means Clustering dialog.

Help: Click the Help button to access documentation on all k-Means Clustering options.

Cancel: Click the Cancel button to close the dialog without running k-Means Clustering.

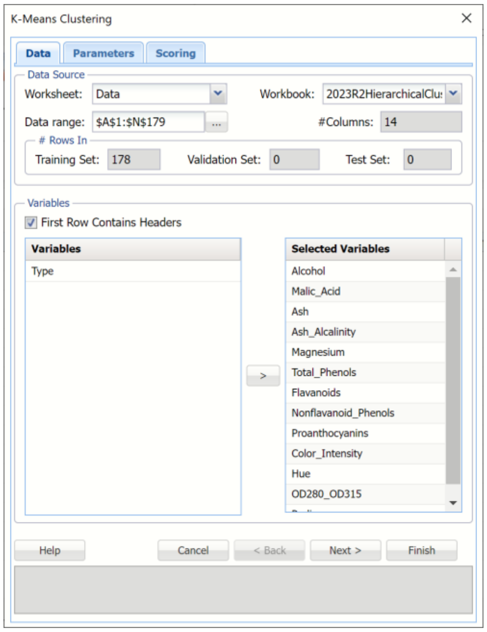
Next: Click the Next button to advance to the next tab.

Finish: Click Finish to accept all option settings on all three dialogs, and run k-Means Clustering.

k-Means Clustering Data Tab

See below for documentation for all options appearing on the k-Means Clustering Data tab.

k-Means Clustering Data tab



Data Source

Worksheet: Data

Workbook: 2023R2HierarchicalClu:

Data range: $A$1:$N$179

#Columns: 14

# Rows In

Training Set: 178

Validation Set: 0

Test Set: 0

Variables

☑ First Row Contains Headers

Variables
Type

Selected Variables
Alcohol
Malic_Acid
Ash
Ash_Alcalinity
Magnesium
Total_Phenols
Flavanoids
Nonflavanoid_Phenols
Proanthocyanins
Color_Intensity
Hue
OD280_OD315

Data Source

Worksheet: Click the down arrow to select the desired worksheet where the dataset is contained.

Workbook: Click the down arrow to select the desired workbook where the dataset is contained.

Data range: Select or enter the desired data range within the dataset. This data range may either be a portion of the dataset or the complete dataset.

#Columns: Displays the number of columns in the data range. This option is read only.

#Rows In Training Set, Validation Set and Test Set: Displays the number of columns in training, validation and/or test partitions, if they exist. This option is read only.

Variables

First Row Contains Headers: Select this checkbox if the first row in the dataset contains column headings.

Variables: This field contains the list of the variables, or features, included in the data range.

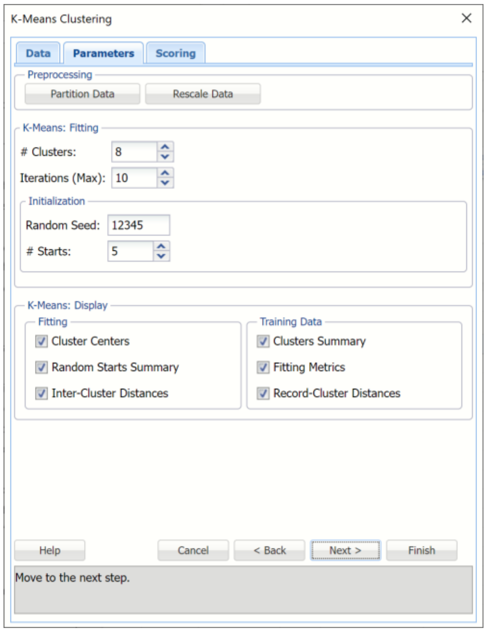
Selected Variables: This field contains the list of variables, or features, to be included in k-Means Clustering.

- To include a variable in k-Means Clustering, select the variable in the Variables list, then click > to move the variable to the Selected Variables list.
- To remove a variable as a selected variable, click the variable in the Selected Variables list, then click < to move the variable back to the Variables list.

k-Means Clustering Parameters Tab

See below for documentation for all options appearing on the k-Means Clustering Parameters tab.

k-Means Clustering Parameters tab



Clusters: 8

Iterations (Max): 10

Random Seed: 12345

Starts: 5

☑ Cluster Centers

☑ Random Starts Summary

☑ Inter-Cluster Distances

☑ Clusters Summary

☑ Fitting Metrics

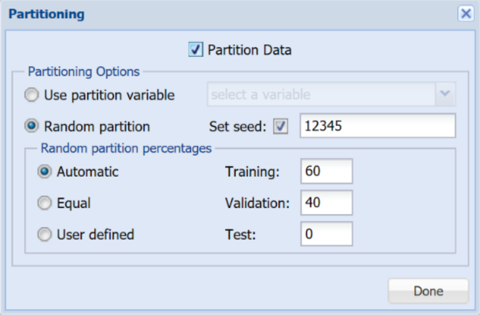
☑ Record-Cluster Distances

Move to the next step.

Preprocessing

Analytic Solver Data Science allows partitioning to be performed on the Parameters tab for k – Means Clustering, if the active data set is un-partitioned. If the active data set has already been partitioned, this button will be disabled. Clicking the Partition Data button opens the following dialog. Select Partition Data on the dialog to enable the partitioning options. See the Partitioning chapter for descriptions of each Partitioning option shown in the dialog below.

Partitioning "On-the-fly" dialog



☑ Partition Data

○ Use partition variable   select a variable

● Random partition   Set seed: ☑ 12345

● Automatic   Training: 60

○ Equal   Validation: 40

○ User defined   Test: 0

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. If the input data is normalized, k-Means

Clustering operates on rescaled data. Results are not transformed back to their original scale.

For more information on this feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Why use "on-the-fly" Partitioning and Rescaling?

If a data partition will be used to train and validate several different algorithms that will be compared for predictive power, it may be better to use the Ribbon Partition choices to create, rescale and/or partition the dataset. But if the rescaled data and/or data partition will be used with a single algorithm, or if it isn't crucial to compare algorithms on exactly the same data, "Partition-on-the-Fly" and "Rescale-on-the-fly" offers several advantages:

- User interface steps are saved, and the Analytic Solver task pane is not cluttered with partition and rescaling output.
- Partition-on-the-fly and Rescaling-on-the-fly is *much faster* than first rescaling the data, creating a standard partition and then running an algorithm.
- Partition-on-the-fly and Rescaling-on-the-fly can handle *larger* datasets without exhausting memory, since the intermediate partition results for the partitioned data is never created.

# Clusters

Enter the number of final cohesive groups of observations (k) to be formed here. The number of clusters should be at least 1 and at most the number of observations-1 in the data range. This value should be based on your knowledge of the data and the number of projected clusters. One can use the results of Hierarchical Clustering or several values of k to understand the best value for # Clusters. The default value for this option is 2.

# Iterations

This option limits the number of iterations for the k-Means Clustering algorithm. Even if the convergence criteria has not yet been met, the cluster adjustment will stop once the limit on *# Iterations* has been reached. The default value for this option is 10.

Random Seed

This option initializes the random number generator that is used to assign the initial cluster centroids. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the initial cluster centroids are calculated. The default value is "12345". The minimum value for this option is 1. To set the seed, type the number you want into the box. This option accepts positive integers with up to 9 digits.

# Starts

Enter a positive value greater than 1 for this option, to enter the number of desired starting points. The final result of the k-Means Clustering algorithm depends on the initial choice on the cluster centroids. The "random starts" option allows a better choice by trying several random assignments. The best assignment (based on Sum of Squared Distances) is chosen as an initialization for further k-Means iterations.

k-Means Display

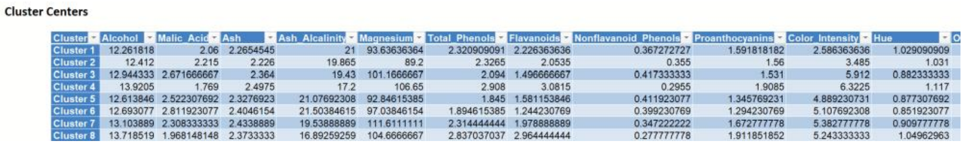
Use these options to display various output for k-Means Clustering. Output options under Fitting apply to the fitting of the model. Output options under Training are related to the application of the fitted model to the training partition.

- ***Fitting:*** *Output appears on the KMC_Output worksheet*
  - **Cluster Centers**

    The Cluster Centers table displays detailed information about the clusters formed by the k-Means Clustering algorithm. This table contains the coordinates of the cluster centers found by the algorithm.

    Accessible by clicking the Clusters Centers link on the Output Navigator.

Example of Cluster Centers output



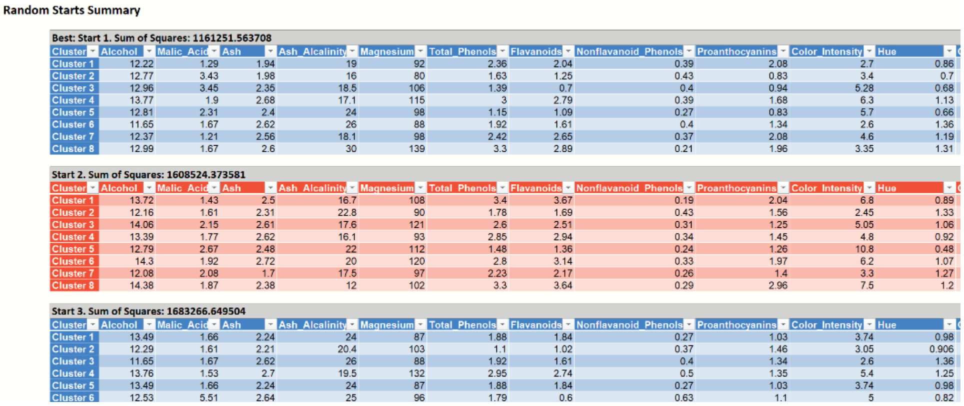
Cluster Centers

Cluster	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue
Cluster 1	12.261818	2.06	2.2654545	21	93.63636364	2.320909091	2.226363636	0.367272727	1.591818182	2.586363636	1.029090909
Cluster 2	12.412	2.215	2.226	19.865	89.2	2.3265	2.0535	0.355	1.56	3.485	1.031
Cluster 3	12.944333	2.671666667	2.364	19.43	101.1666667	2.094	1.496666667	0.417333333	1.531	5.912	0.882333333
Cluster 4	13.9205	1.769	2.4975	17.2	106.65	2.908	3.0815	0.2955	1.9085	6.3225	1.117
Cluster 5	12.613846	2.522307692	2.3276923	21.07692308	92.84615385	1.845	1.581153846	0.411923077	1.345769231	4.889230731	0.877307692
Cluster 6	12.693077	2.811923077	2.4046154	21.50384615	97.03846154	1.894615385	1.244230769	0.399230769	1.294230769	5.107692308	0.851923077
Cluster 7	13.103889	2.308333333	2.4338889	19.53888889	111.6111111	2.314444444	1.978888889	0.347222222	1.672777778	5.382777778	0.909777778
Cluster 8	13.718519	1.968148148	2.3733333	16.89259259	104.6666667	2.837037037	2.964444444	0.277777778	1.911851852	5.243333333	1.04962963

-
  - **Random Starts Summary**

    The table "Random Starts Summary" displays the information about the initial search for the best centroid assignment. The assignment marked by "Best Start" is used as the initial assignment of the centroids. Accessible by clicking the Random Starts Summary link on the Output Navigator.

Example of Random Starts Summary output



Random Starts Summary

Best: Start 1. Sum of Squares: 1161251.563708

Cluster	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue
Cluster 1	12.22	1.29	1.94	19	92	2.36	2.04	0.39	2.08	2.7	0.86
Cluster 2	12.77	3.43	1.98	16	80	1.63	1.25	0.43	0.83	3.4	0.7
Cluster 3	12.96	3.45	2.35	18.5	106	1.39	0.7	0.4	0.94	5.28	0.68
Cluster 4	13.77	1.9	2.68	17.1	115	3	2.79	0.39	1.68	6.3	1.13
Cluster 5	12.81	2.31	2.4	24	98	1.15	1.09	0.27	0.83	5.7	0.66
Cluster 6	11.65	1.67	2.62	26	88	1.92	1.61	0.4	1.34	2.6	1.36
Cluster 7	12.37	1.21	2.56	18.1	98	2.42	2.65	0.37	2.08	4.6	1.19
Cluster 8	12.99	1.67	2.6	30	139	3.3	2.89	0.21	1.96	3.35	1.31

Start 2. Sum of Squares: 1608524.373581

Cluster	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue
Cluster 1	13.72	1.43	2.5	16.7	108	3.4	3.67	0.19	2.04	6.8	0.89
Cluster 2	12.16	1.61	2.31	22.8	90	1.78	1.69	0.43	1.56	2.45	1.33
Cluster 3	14.06	2.15	2.61	17.6	121	2.6	2.51	0.31	1.25	5.05	1.06
Cluster 4	13.39	1.77	2.62	16.1	93	2.85	2.94	0.34	1.45	4.8	0.92
Cluster 5	12.79	2.67	2.48	22	112	1.48	1.36	0.24	1.26	10.8	0.48
Cluster 6	14.3	1.92	2.72	20	120	2.8	3.14	0.33	1.97	6.2	1.07
Cluster 7	12.08	2.08	1.7	17.5	97	2.23	2.17	0.26	1.4	3.3	1.27
Cluster 8	14.38	1.87	2.38	12	102	3.3	3.64	0.29	2.96	7.5	1.2

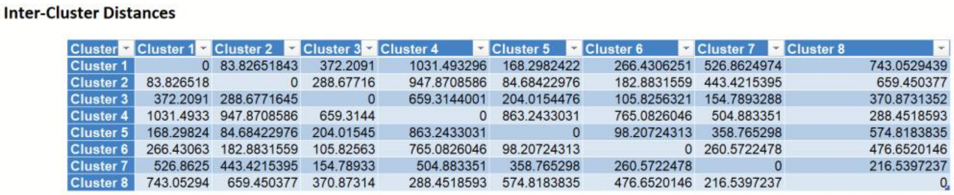
Start 3. Sum of Squares: 1683266.649504

Cluster	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue
Cluster 1	13.49	1.66	2.24	24	87	1.88	1.84	0.27	1.03	3.74	0.98
Cluster 2	12.29	1.61	2.21	20.4	103	1.1	1.02	0.37	1.46	3.05	0.906
Cluster 3	11.65	1.67	2.62	26	88	1.92	1.61	0.4	1.34	2.6	1.36
Cluster 4	13.76	1.53	2.7	19.5	132	2.95	2.74	0.5	1.35	5.4	1.25
Cluster 5	13.49	1.66	2.24	24	87	1.88	1.84	0.27	1.03	3.74	0.98
Cluster 6	12.53	5.51	2.64	25	96	1.79	0.6	0.63	1.1	5	0.82

-
  - **Inter-Cluster Distances**

    This table displays the distance between each cluster center. Accessible by clicking the Inter-Cluster Distances link on the Output Navigator.

Example of Inter-Cluster Distances output



Inter-Cluster Distances

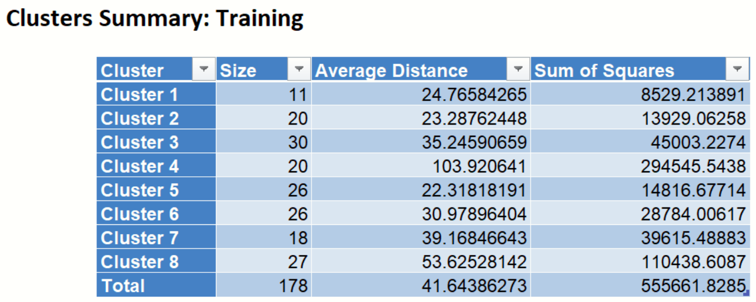
Cluster	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Cluster 6	Cluster 7	Cluster 8
Cluster 1	0	83.82651843	372.2091	1031.493296	168.2982422	266.4306251	526.8624974	743.0529439
Cluster 2	83.826518	0	288.67716	947.8708586	84.68422976	182.8831559	443.4215395	659.450377
Cluster 3	372.2091	288.6771645	0	659.3144001	204.0154476	105.8256321	154.7893288	370.8731352
Cluster 4	1031.4933	947.8708586	659.3144	0	863.2433031	765.0826046	504.883351	288.4518593
Cluster 5	168.29824	84.68422976	204.01545	863.2433031	0	98.20724313	358.765298	574.8183835
Cluster 6	266.43063	182.8831559	105.82563	765.0826046	98.20724313	0	260.5722478	476.6520146
Cluster 7	526.8625	443.4215395	154.78933	504.883351	358.765298	260.5722478	0	216.5397237
Cluster 8	743.05294	659.450377	370.87314	288.4518593	574.8183835	476.6520146	216.5397237	0

- ***Training Data*** Output appears on the KMC_TrainingClusters worksheet

  - **Clusters Summary**

    The "Cluster Summary" table displays the number of records (observations) included in each cluster, the within-cluster average distance and the total Sum of Squares. This information can be used to better understand how large and how sparse the resulting clusters are. Accessible by clicking the Clusters Summary link on the Output Navigator.

Example of Clusters Summary output



Clusters Summary: Training

Cluster	Size	Average Distance	Sum of Squares
Cluster 1	11	24.76584265	8529.213891
Cluster 2	20	23.28762448	13929.06258
Cluster 3	30	35.24590659	45003.2274
Cluster 4	20	103.920641	294545.5438
Cluster 5	26	22.31818191	14816.67714
Cluster 6	26	30.97896404	28784.00617
Cluster 7	18	39.16846643	39615.48883
Cluster 8	27	53.62528142	110438.6087
Total	178	41.64386273	555661.8285

  - **Fitting Metrics**

    Fitting Metrics: This table lists several metrics computed from the training dataset and is accessible by clicking the Fitting Metrics: Training link on the Output Navigator.

    (Recall that our training dataset is the full dataset since we did not partition into training, validation or test partitions.)

    - Avg Within-Cluster Distance – Average total distance from each record to the corresponding cluster center, for each cluster.

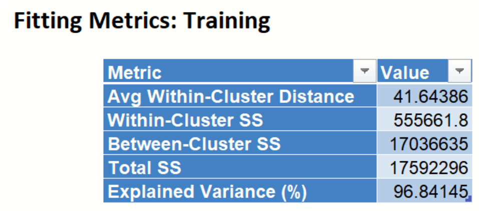
    - Within Cluster SS – Sum of distances between the records and the corresponding cluster centers for each cluster. This statistic measures cluster compactness. (The algorithm is trying to minimize this measure.)

    - Between Cluster SS – Sum of distances between the cluster centers and the total sample mean, divided by the number of records within each cluster. (Between Cluster SS measures cluster separation which the algorithm is trying to maximize. This is equivalent to minimizing *Within Cluster SS*.)

- Total SS – Sum of distances from each record to the total sample mean, Total SS = Within Cluster SS + Between Cluster SS.
- Explained Variance (%) – Goodness of fit metrics, showing the degree of internal cohesion and external separation, Explained Variance % = Between Cluster SS / Total SS.

Clusters: Training **-** This table displays the final cluster assignment for each observation in the input data – the point is assigned to the "closest" cluster, i.e. the one with the nearest centroid.

Example of Fitting Metrics output



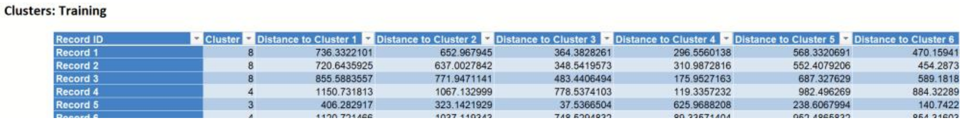
Fitting Metrics: Training

Metric	Value
Avg Within-Cluster Distance	41.64386
Within-Cluster SS	555661.8
Between-Cluster SS	17036635
Total SS	17592296
Explained Variance (%)	96.84145

- Record-Cluster Distances

Appends the distance of each record to each cluster in the Clusters: Training output table. Records are assigned to the "closest" cluster, i.e. the one with the nearest centroid. For example, in the 4th record, the final cluster assignment is 4 since the distance to that cluster centroid is the closest (119.34).

Example of Record-Cluster Distances output



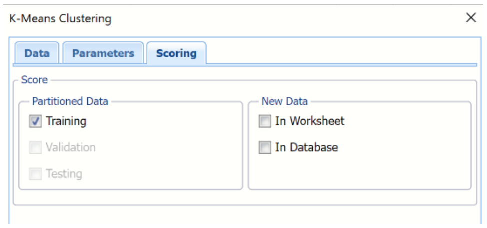
Clusters: Training

Record ID	Cluster	Distance to Cluster 1	Distance to Cluster 2	Distance to Cluster 3	Distance to Cluster 4	Distance to Cluster 5	Distance to Cluster 6
Record 1	8	736.3322101	652.967945	364.3828261	296.5560138	568.3320691	470.15941
Record 2	8	720.6435925	637.0027842	348.5419573	310.9872816	552.4079206	454.2873
Record 3	8	855.5883557	771.9471141	483.4406494	175.9527163	687.327629	589.1818
Record 4	4	1150.731813	1067.132999	778.5374103	119.3357232	982.496269	884.32289
Record 5	3	406.282917	323.1421929	37.5366504	625.9688208	238.6067994	140.7422
Record 6	4	1120.721466	1037.119343	748.6294832	80.33571404	952.4865832	854.31603

k-Means Clustering Scoring Tab

See below for documentation for all options appearing on the k-Means Clustering Scoring tab.

k-Means Clustering Scoring Tab



K-Means Clustering

Score

Partitioned Data

☑ Training

□ Validation

□ Testing

New Data

□ In Worksheet

□ In Database

Scoring Tab

Select the desired partition to apply k-Means Clustering, if the partition exists. If Validation and/or Testing partitions do not exist, then these two options will be disabled.

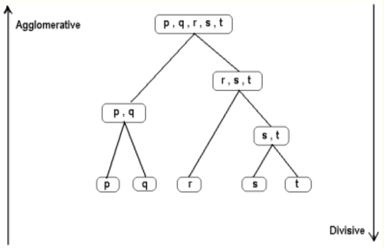
See the example above or the Scoring chapter at the end of the Analytic Solver Data Science User Guide for information on how to score new data In Worksheet or In Database.

Hierarchical Clustering

Introduction

Cluster Analysis, also called data segmentation, has a variety of goals. All relate to grouping or segmenting a collection of objects (also called observations, individuals, cases, or data rows) into subsets or "clusters", such that those within each cluster are more closely related to one another than objects assigned to different clusters. The most important goal of cluster analysis is the notion of degree of similarity (or dissimilarity) between the individual objects being clustered. There are two major methods of clustering -- hierarchical clustering and k-means clustering. (See the k-means clustering chapter for information on this type of clustering analysis.)

In hierarchical clustering the data are not partitioned into a particular cluster in a single step. Instead, a series of partitions takes place, which may run from a single cluster containing all objects to n clusters each containing a single object. Hierarchical Clustering is subdivided into *agglomerative* methods, which proceed by a series of fusions of the n objects into groups, and *divisive* methods, which separate n objects successively into finer groupings. The hierarchical clustering technique employed by Analytic Solver Data Science is an Agglomerative technique. Hierarchical clustering may be represented by a two dimensional diagram known as a dendrogram which illustrates the fusions or divisions made at each successive stage of analysis. An example of such a dendrogram is given below:



Agglomerative methods

An agglomerative hierarchical clustering procedure produces a series of partitions of the data, Pn, Pn-1, ... , P1. The first Pn consists of n single object 'clusters', the last P1, consists of a single group containing all n cases.

At each particular stage the method joins the two clusters which are closest together (most similar). (At the first stage, this amounts to joining together the two objects that are closest together, since at the initial stage each cluster has one object.)

Differences between methods arise because of the different methods of defining distance (or similarity) between clusters. Several agglomerative techniques are featured in Analytic Solver's Hierarchical Clustering. See the description of each occuring further down in this chapter.

Examples of Hierarchical Clustering

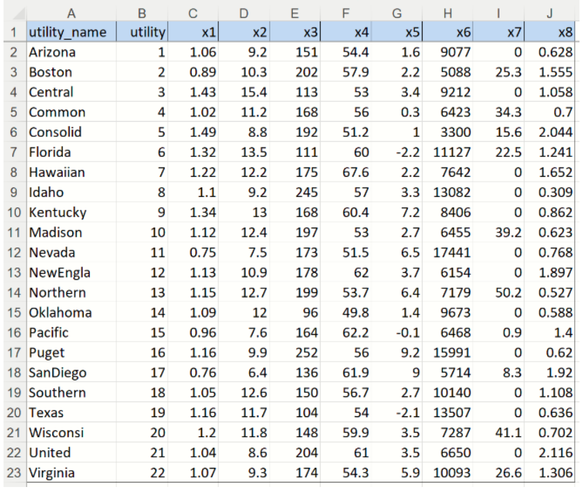
Two examples are used in this section to illustrate how to use Hierarchical Clustering. The first example uses Raw Data and the second example uses a distance matrix.

Hierarchical Cluster Using Raw Data

The utilities.xlsx example dataset (shown below) holds corporate data on 22 US public utilities. This example will illustrate how a user could use Analytic Solver Data Science to perform a cluster analysis using hierarchical clustering.

Open this example by clicking **Help – Example Models -- Forecasting/Data Science Examples – Utilities.**

Each record includes 8 observations. Before Hierarchical clustering is applied, the data will be “normalized”, or “standardized”. A popular method for normalizing continuous variables is to divide each variable by its standard deviation. After the variables are standardized, the distance can be computed between clusters using the Euclidean metric.



	A	B	C	D	E	F	G	H	I	J
1	utility_name	utility	x1	x2	x3	x4	x5	x6	x7	x8
2	Arizona	1	1.06	9.2	151	54.4	1.6	9077	0	0.628
3	Boston	2	0.89	10.3	202	57.9	2.2	5088	25.3	1.555
4	Central	3	1.43	15.4	113	53	3.4	9212	0	1.058
5	Common	4	1.02	11.2	168	56	0.3	6423	34.3	0.7
6	Consolid	5	1.49	8.8	192	51.2	1	3300	15.6	2.044
7	Florida	6	1.32	13.5	111	60	-2.2	11127	22.5	1.241
8	Hawaiian	7	1.22	12.2	175	67.6	2.2	7642	0	1.652
9	Idaho	8	1.1	9.2	245	57	3.3	13082	0	0.309
10	Kentucky	9	1.34	13	168	60.4	7.2	8406	0	0.862
11	Madison	10	1.12	12.4	197	53	2.7	6455	39.2	0.623
12	Nevada	11	0.75	7.5	173	51.5	6.5	17441	0	0.768
13	NewEngla	12	1.13	10.9	178	62	3.7	6154	0	1.897
14	Northern	13	1.15	12.7	199	53.7	6.4	7179	50.2	0.527
15	Oklahoma	14	1.09	12	96	49.8	1.4	9673	0	0.588
16	Pacific	15	0.96	7.6	164	62.2	-0.1	6468	0.9	1.4
17	Puget	16	1.16	9.9	252	56	9.2	15991	0	0.62
18	SanDiego	17	0.76	6.4	136	61.9	9	5714	8.3	1.92
19	Southern	18	1.05	12.6	150	56.7	2.7	10140	0	1.108
20	Texas	19	1.16	11.7	104	54	-2.1	13507	0	0.636
21	Wisconsi	20	1.2	11.8	148	59.9	3.5	7287	41.1	0.702
22	United	21	1.04	8.6	204	61	3.5	6650	0	2.116
23	Virginia	22	1.07	9.3	174	54.3	5.9	10093	26.6	1.306

An explanation of the variables is contained in the table below.

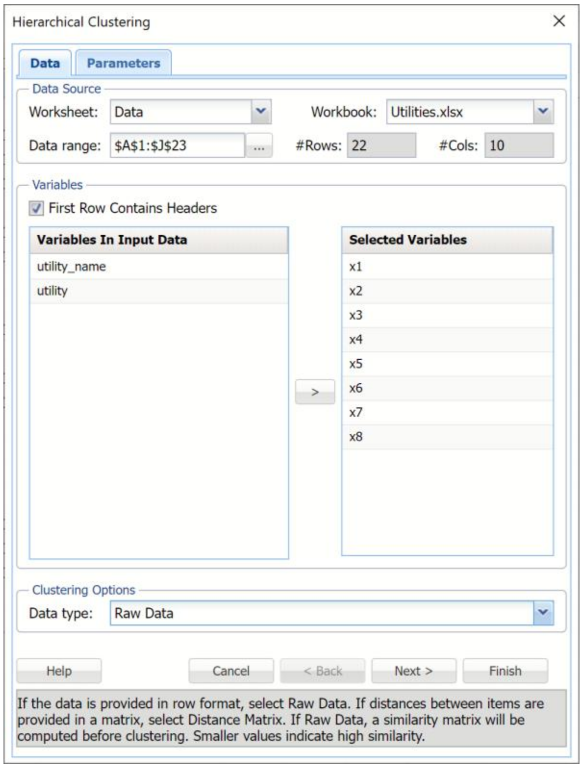
X1: Fixed-charge covering ratio (income/debt)
X2: Rate of return on capital
X3: Cost per KW capacity in place
X4: Annual Load Factor
X5: Peak KWH demand growth from 1974 to 1975
X6: Sales (KWH use per year)
X7: Percent Nuclear
X8: Total fuel costs (cents per KWH)

An economist analyzing this data might first begin her analysis by building a detailed cost model of the various utilities. However, to save a considerable amount of time and effort, she could instead cluster similar types of utilities, build a detailed cost model for just one ”typical” utility in each cluster, then from there, scale up from these models to estimate results for all utilities. This example will do just that.

Click **Cluster -- Hierarchical Clustering** to bring up the Hierarchical Clustering dialog.

On the Data tab, Select variables **x1 through x8** in the *Variables in Input Data* field, then click > to move the selected variables to the *Selected Variables* field.

Leave *Data Type at Raw Data* at the bottom of the dialog.


Hierarchical Clustering

Data | Parameters

Data Source

Worksheet: Data

Workbook: Utilities.xlsx

Data range: $A$1:$J$23

#Rows: 22

#Cols: 10

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
utility_name	x1
utility	x2
	x3
	x4
	x5
	x6
	x7
	x8

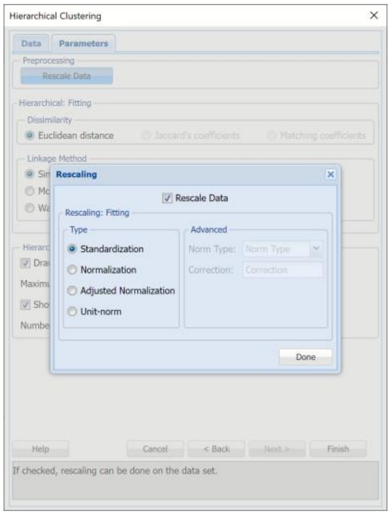
Clustering Options

Data type: Raw Data

If the data is provided in row format, select Raw Data. If distances between items are provided in a matrix, select Distance Matrix. If Raw Data, a similarity matrix will be computed before clustering. Smaller values indicate high similarity.

Then click **Next** to advance to the *Hierarchical Clustering*.

At the top of the dialog, select **Rescale data**. Use this dialog to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide. *For this example keep the default setting of Standardization*. Then click Done to close the dialog.


Hierarchical Clustering

Data | Parameters

Preprocessing

Hierarchical: Fitting

Dissimilarity

● Euclidean distance ○ Jaccard's coefficients ○ Matching coefficients

Linkage Method

Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

Advanced

Norm Type: Norm Type

Correction: Correction

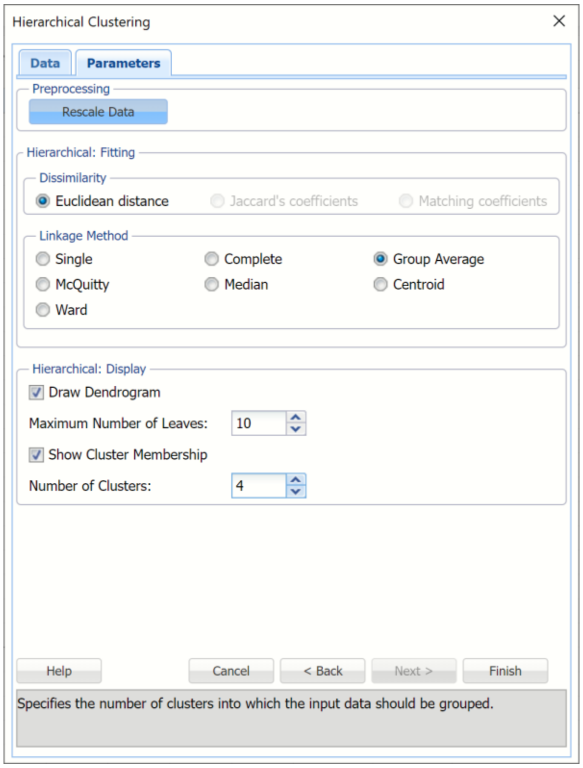
If checked, rescaling can be done on the data set.

Under *Dissimilarity*, **Euclidean distance** is selected by default. The Hierarchical clustering method uses the Euclidean Distance as the similarity measure for raw numeric data.

Note: When the data is binary the remaining two options, *Jaccard's coefficients* and *Matching coefficients* are enabled.

Under *Linkage Method*, select **Group average linkage**. Recall from the Introduction to this chapter, the group average linkage method calculates the average distance of all possible distances between each record in each cluster.

For purposes of assigning cases to clusters, we must specify the number of clusters in advance. Under Hierarchical: Display, increment *Number of Clusters* to 4. Keep the remaining options at their defaults as shown in the screenshot below. Then click **Finish**.



Dissimilarity

● Euclidean distance ○ Jaccard's coefficients ○ Matching coefficients

Linkage Method

○ Single ○ Complete ● Group Average
○ McQuitty ○ Median ○ Centroid
○ Ward

☑ Draw Dendrogram

Maximum Number of Leaves: 10

☑ Show Cluster Membership

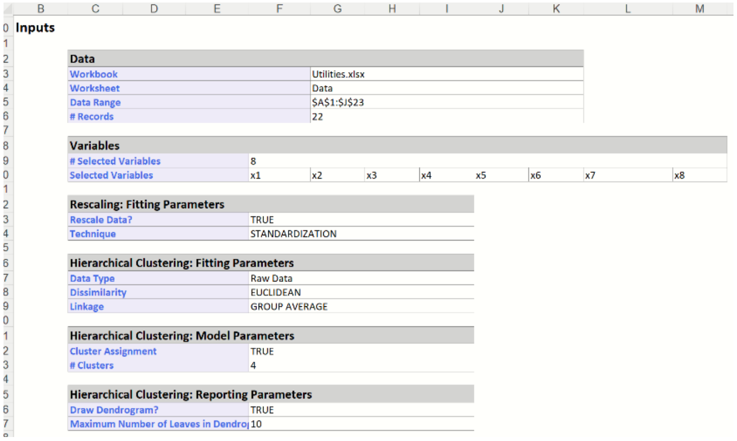
Number of Clusters: 4

Specifies the number of clusters into which the input data should be grouped.

Analytic Solver Data Science will create four clusters using the group average linkage method. The output *HC_Output*, *HC_Clusters* and *HC_Dendrogram* are inserted to the right of the Data worksheet.

HC_Output Worksheet

The top portion of the output simply displays the options selected on the Hierarchical Clustering dialog tabs.



Inputs

Data	
Workbook	Utilities.xlsx
Worksheet	Data
Data Range	$A$1:$J$23
# Records	22

Variables								
# Selected Variables	8							
Selected Variables	x1	x2	x3	x4	x5	x6	x7	x8

Rescaling: Fitting Parameters	
Rescale Data?	TRUE
Technique	STANDARDIZATION

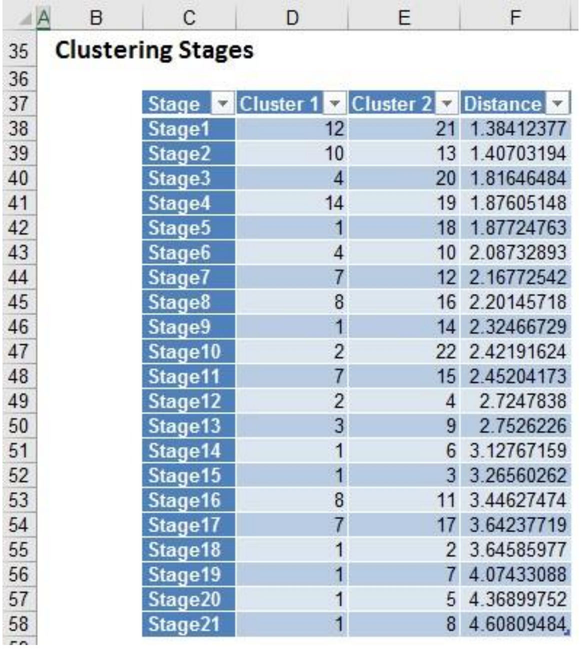
Hierarchical Clustering: Fitting Parameters	
Data Type	Raw Data
Dissimilarity	EUCLIDEAN
Linkage	GROUP AVERAGE

Hierarchical Clustering: Model Parameters	
Cluster Assignment	TRUE
# Clusters	4

Hierarchical Clustering: Reporting Parameters	
Draw Dendrogram?	TRUE
Maximum Number of Leaves in Dendro	10

Further down the HC_Output sheet is the Clustering Stages table. This table details the history of the cluster formation. Initially, each individual case is considered its own cluster (single member in each cluster). Analytic Solver Data Science begins the method with # clusters = # cases. At stage 1, below, clusters (i.e. cases) 12 and 21 were found to be closer together than any other two clusters (i.e. cases), so they are joined together in to cluster 12. At this point there is one cluster with two cases (cases 12 and 21), and 21 additional clusters that still have just one case in each. At stage 2, clusters 10 and 13 are found to be closer together than any other two clusters, so they are joined together into cluster 10.

This process continues until there is just one cluster. At various stages of the clustering process, there are different numbers of clusters. A graph called a dendrogram illustrates these steps.

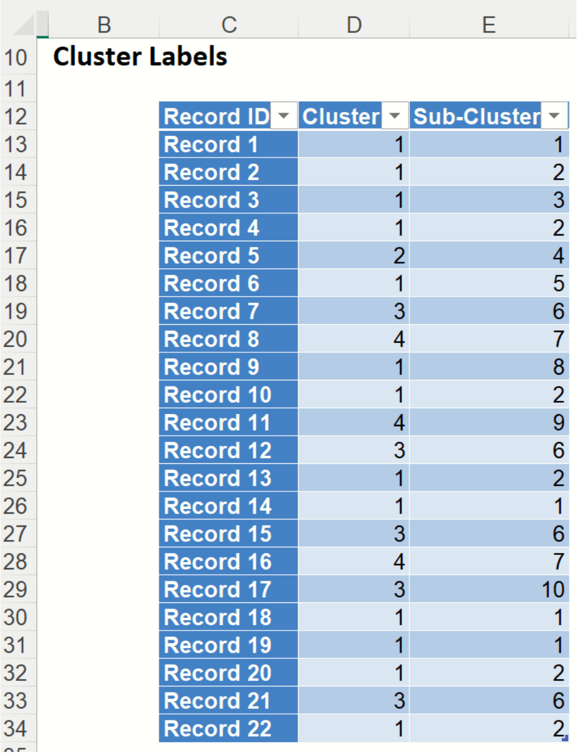


Clustering Stages

Stage	Cluster 1	Cluster 2	Distance
Stage1	12	21	1.38412377
Stage2	10	13	1.40703194
Stage3	4	20	1.81646484
Stage4	14	19	1.87605148
Stage5	1	18	1.87724763
Stage6	4	10	2.08732893
Stage7	7	12	2.16772542
Stage8	8	16	2.20145718
Stage9	1	14	2.32466729
Stage10	2	22	2.42191624
Stage11	7	15	2.45204173
Stage12	2	4	2.7247838
Stage13	3	9	2.7526226
Stage14	1	6	3.12767159
Stage15	1	3	3.26560262
Stage16	8	11	3.44627474
Stage17	7	17	3.64237719
Stage18	1	2	3.64585977
Stage19	1	7	4.07433088
Stage20	1	5	4.36899752
Stage21	1	8	4.60809484

HC_Clusters Worksheet

Click the HC_Clusters worksheet tab to view the Cluster Labels table. This table displays the assignment of each record to the four clusters and their sub-clusters.



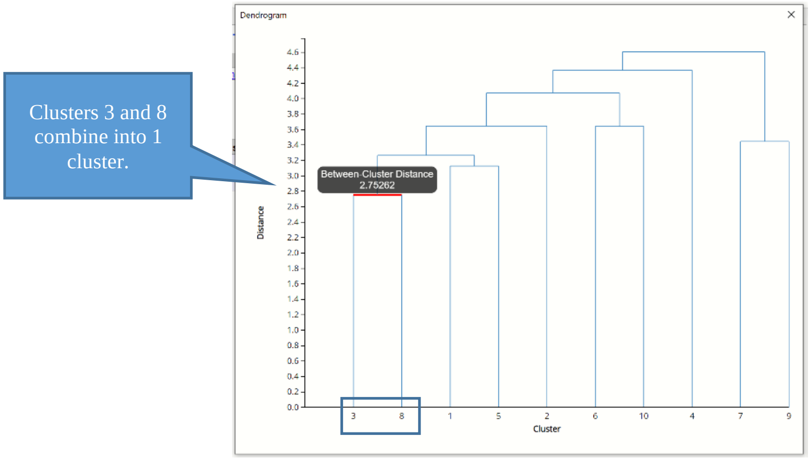
Cluster Labels

Record ID	Cluster	Sub-Cluster
Record 1	1	1
Record 2	1	2
Record 3	1	3
Record 4	1	2
Record 5	2	4
Record 6	1	5
Record 7	3	6
Record 8	4	7
Record 9	1	8
Record 10	1	2
Record 11	4	9
Record 12	3	6
Record 13	1	2
Record 14	1	1
Record 15	3	6
Record 16	4	7
Record 17	3	10
Record 18	1	1
Record 19	1	1
Record 20	1	2
Record 21	3	6
Record 22	1	2

HC_Dendrogram Output

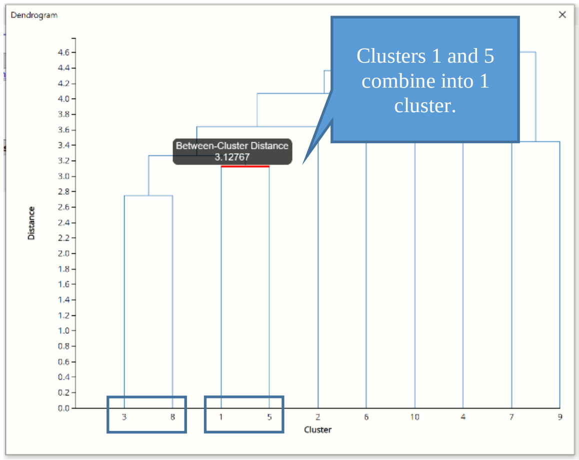
Click the HC_Dendrogram worksheet tab to view the clustering dendrogram. A dendrogram is a diagram that illustrates the hierarchical association between the clusters.

The Sub Cluster IDs are listed along the x-axis (in an order convenient for showing the cluster structure). The y-axis measures inter-cluster distance. Consider Cluster IDs 3 and 8-- they have an inter-cluster distance of 2.753. (Hover over the horizontal connecting line to see the Between-Cluster Distance.) No other cases have a smaller inter-cluster distance, so 3 and 8 are joined into one cluster, indicated by the horizontal line linking them.



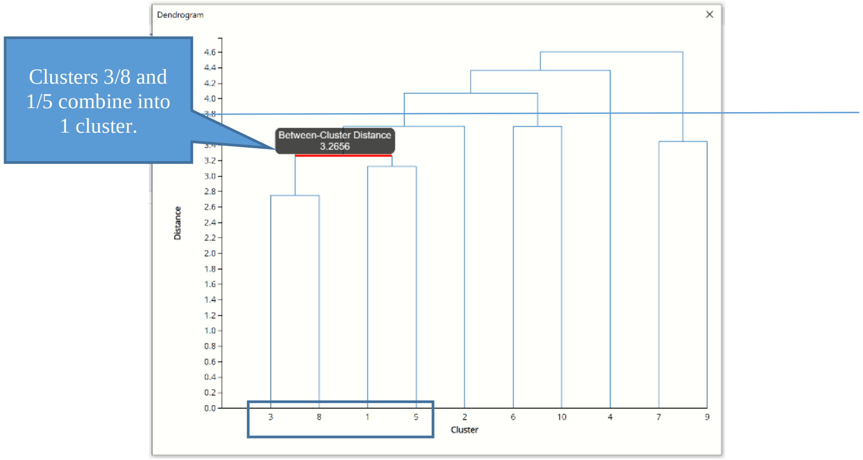
Clusters 3 and 8 combine into 1 cluster.

Next, we see that cases 1 and 5 have the next smallest inter-cluster distance, so they are joined into a 2nd cluster.



Clusters 1 and 5 combine into 1 cluster.

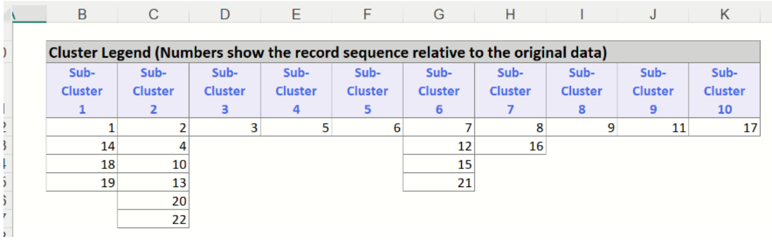
The next smallest inter-cluster distance is between the newly formed 3/8 and 1/5 clusters. This process repeats until all subclusters have been formed into 1 cluster.



Clusters 3/8 and 1/5 combine into 1 cluster.

If we draw a horizontal line through the diagram at any level on the y-axis (the distance measure), the vertical cluster lines that intersect the horizontal line indicate clusters whose members are at least that close to each other. If we draw a horizontal line at distance = 3.8, for example, we see that there are 4 clusters that have an inter-cluster distance of at least 3.8. In addition, we can see that a sub ID can belong to multiple clusters, depending on where we draw the line.

Click the ‘X’ in the upper right hand corner to close the dendrogram to view the Cluster Legend. This table shows the records that are assigned to each sub-cluster.



Cluster Legend (Numbers show the record sequence relative to the original data)

Sub-Cluster 1	Sub-Cluster 2	Sub-Cluster 3	Sub-Cluster 4	Sub-Cluster 5	Sub-Cluster 6	Sub-Cluster 7	Sub-Cluster 8	Sub-Cluster 9	Sub-Cluster 10
1	2	3	5	6	7	8	9	11	17
14	4				12	16			
18	10				15				
19	13				21				
	20								
	22								

Hierarchical Cluster Using a Distance Matrix

This next example illustrates Hierarchical Clustering when the data represents the distance between the ith and jth records. (When applied to raw data, Hierarchical clustering converts the data into the distance matrix format before proceeding with the clustering algorithm. Providing the distance measures in the data requires one less step for the Hierarchical clustering algorithm.)

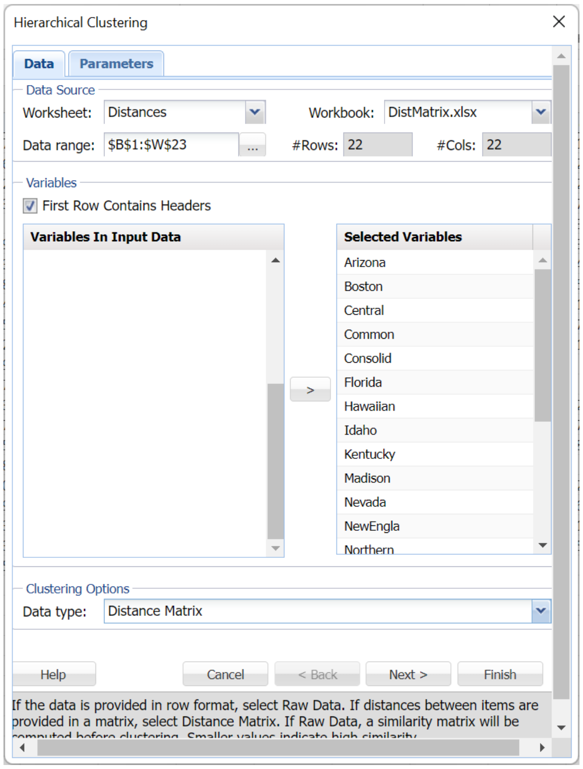
Open the DistMatrix example dataset by clicking **Help – Example Models – Forecasting / Data Science Examples**.

Open the Hierarchical Clustering dialog by clicking **Cluster – Hierarchical Clustering**.

At the top of the Data tab, **B1:W23** has been saved as the default for *Data range*.

All variables have been previously saved as Selected Variables and Distance Matrix has been selected by default.

Note: When Distance Matrix is selected, the distance matrix is validated to be a valid distance matrix (a square, symmetric matrix with 0’s on the diagonal).



Data Source

Worksheet: Distances

Workbook: DistMatrix.xlsx

Data range: $B$1:$W$23

#Rows: 22

#Cols: 22

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables

Arizona
Boston
Central
Common
Consolid
Florida
Hawaiian
Idaho
Kentucky
Madison
Nevada
NewEngla
Northern

Clustering Options

Data type: Distance Matrix

If the data is provided in row format, select Raw Data. If distances between items are provided in a matrix, select Distance Matrix. If Raw Data, a similarity matrix will be computed before clustering. Smaller values indicate high similarity.

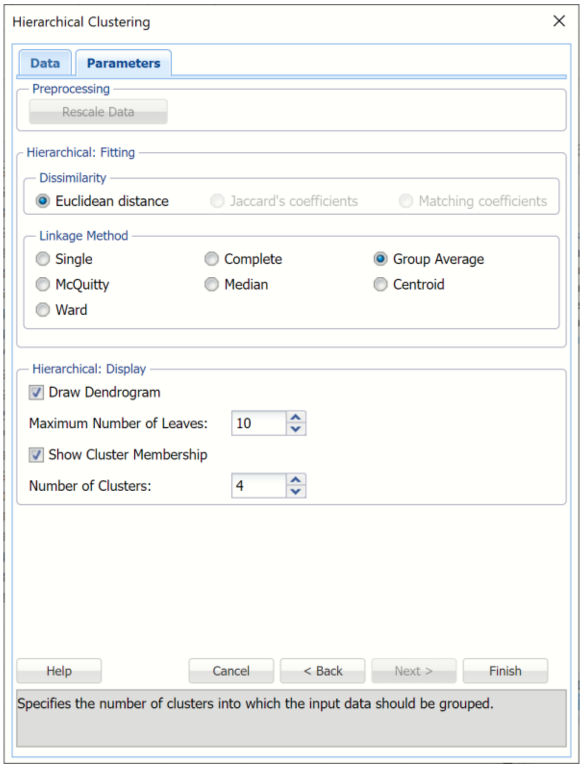
Click **Next**.

Rescale Data is disabled since the distance matrix is used.

Keep the default, *Euclidean distance*, selected for *Dissimalarity*. (See below for explanations for Jaccard's coefficients and Matching coefficients.)

Select **Group average linkage** as the Clustering Method.

Leave **Draw dendrogram**, **Maximum Number of Leaves.** Set *Number of Clusters* to **4**.



Hierarchical Clustering

Preprocessing

Hierarchical: Fitting

Dissimilarity

● Euclidean distance ○ Jaccard's coefficients ○ Matching coefficients

Linkage Method

○ Single ○ Complete ● Group Average
○ McQuitty ○ Median ○ Centroid
○ Ward

Hierarchical: Display

☑ Draw Dendrogram

Maximum Number of Leaves: 10

☑ Show Cluster Membership

Number of Clusters: 4

Specifies the number of clusters into which the input data should be grouped.

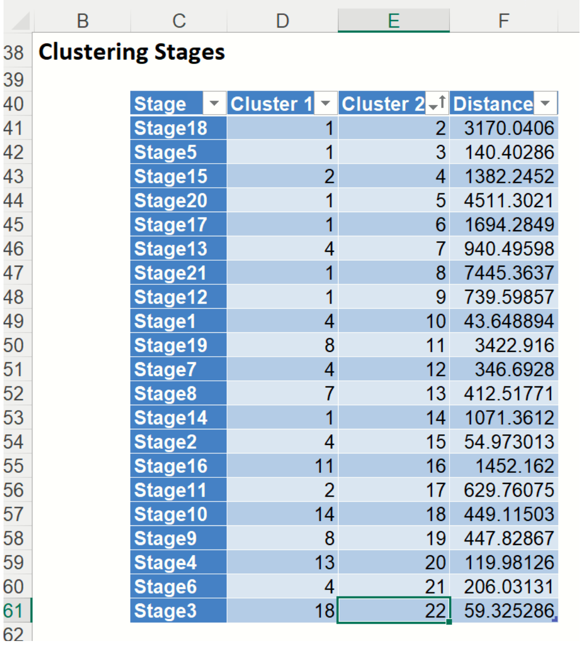
Then click **Finish**.

Output worksheets are inserted to the right of the Distances tab: *HC_Output*, *HC_Clusters*, and *HC_Dendrogram.* The contents of HC_Output and HC_Dendrogram are described below. See above for a description of the contents of HC_Clusters.

HC_Output Worksheet

As in the example above, the top of the HC_Output worksheet is the Inputs portion, which displays the choices selected on both tabs of the Hierarchical Clustering dialog.

Scroll down to the Clustering Stages table. As discussed above, this table details the history of the cluster formation. At the beginning, each individual case was considered its own cluster, # clusters = # cases. At stage 1, below, clusters (i.e. cases) 4 and 10 were found to be closer together than any other two clusters (i.e. cases), so 4 absorbed 10. At stage 2, clusters 4 and 15 are found to be closer together than any other two clusters, so 4 absorbed 15. At this point there is one cluster with three cases (cases 4, 10 and 15), and 20 additional clusters that still have just one case in each. This process continues until there is just one cluster at stage 21.

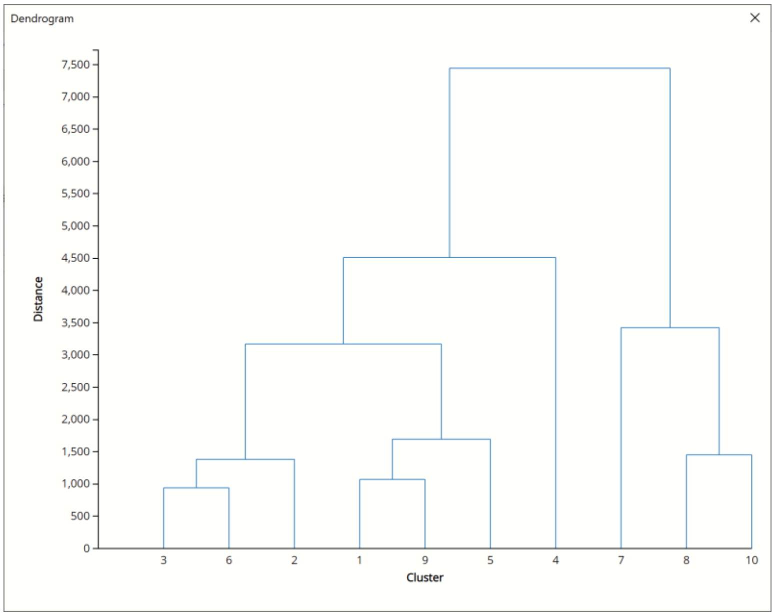


Clustering Stages

Stage	Cluster 1	Cluster 2	Distance
Stage18	1	2	3170.0406
Stage5	1	3	140.40286
Stage15	2	4	1382.2452
Stage20	1	5	4511.3021
Stage17	1	6	1694.2849
Stage13	4	7	940.49598
Stage21	1	8	7445.3637
Stage12	1	9	739.59857
Stage1	4	10	43.648894
Stage19	8	11	3422.916
Stage7	4	12	346.6928
Stage8	7	13	412.51771
Stage14	1	14	1071.3612
Stage2	4	15	54.973013
Stage16	11	16	1452.162
Stage11	2	17	629.76075
Stage10	14	18	449.11503
Stage9	8	19	447.82867
Stage4	13	20	119.98126
Stage6	4	21	206.03131
Stage3	18	22	59.325286

The Dendrogram output (included on *HC_Dendrogram*) is shown below.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select **HC_Dendrogram** for *Worksheet* and **Dendrogram for Hierarchical Clustering** for *Chart*.



One of the reasons why Hierarchical Clustering is so attractive to statisticians is because it's easy to understand and the clustering process can be easily illustrated with a dendrogram. However, there are a few limitations.

i. Hierarchical clustering requires computing and storing an n x n distance matrix. If using a large dataset, this requirement can be very slow and require large amounts of memory.

ii. Clusters created through Hierarchical clustering are not very stable. If records are eliminated, the results can be very different.

iii. Outliers in the data can impact the results negatively.

Options for Hierarchical Clustering

See below for an explanation of options on both tabs of the *Hierarchical Clustering* dialog*: Data and Parameters*.

The following options appear on all three tabs of the Hierarchical Clustering dialog.

Help: Click the Help button to access documentation on all Hierarchical Clustering options.

Cancel: Click the Cancel button to close the dialog without running Hierarchical Clustering.

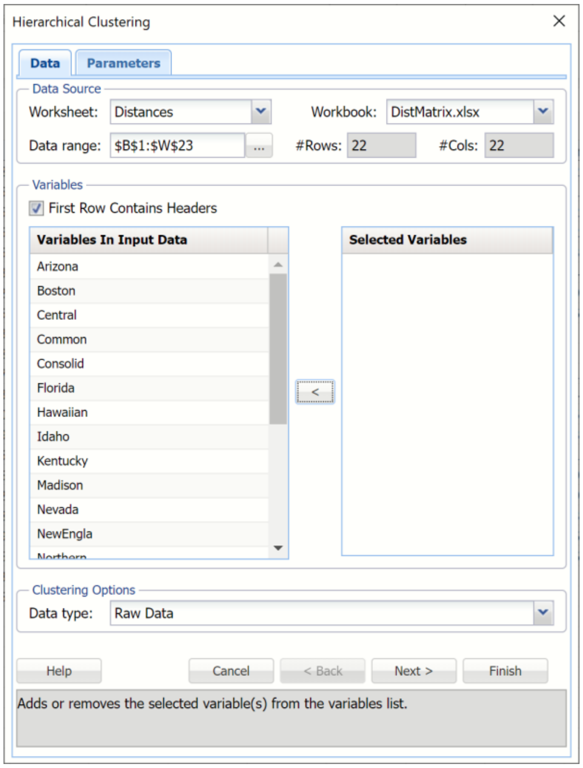
Next: Click the Next button to advance to the next tab.

Finish: Click Finish to accept all option settings on all three dialogs, and run Hierarchical Clustering.

Hierarchical Clustering Data tab

See below for documentation for all options appearing on the Hierarchical Clustering Data tab.

Hierarchical Clustering Data tab



Data Source

Worksheet: Distances

Workbook: DistMatrix.xlsx

Data range: $B$1:$W$23

#Rows: 22

#Cols: 22

Variables

☑ First Row Contains Headers

Variables In Input Data

Arizona
Boston
Central
Common
Consolid
Florida
Hawaiian
Idaho
Kentucky
Madison
Nevada
NewEngla
Northern

Selected Variables

Clustering Options

Data type: Raw Data

Adds or removes the selected variable(s) from the variables list.

Data Source

Worksheet: Click the down arrow to select the desired worksheet where the dataset is contained.

Workbook: Click the down arrow to select the desired workbook where the dataset is contained.

Data range: Select or enter the desired data range within the dataset. This data range may either be a portion of the dataset or the complete dataset.

#Rows: Displays the number of rows in the data range. This option is read only.

#Cols: Displays the number of columns in the data range. This option is read only.

Variables

First Row Contains Headers: Select this checkbox if the first row in the dataset contains column headings.

Variables: This field contains the list of the variables, or features, included in the data range.

Selected Variables: This field contains the list of variables, or features, to be included in Hierarchical Clustering.

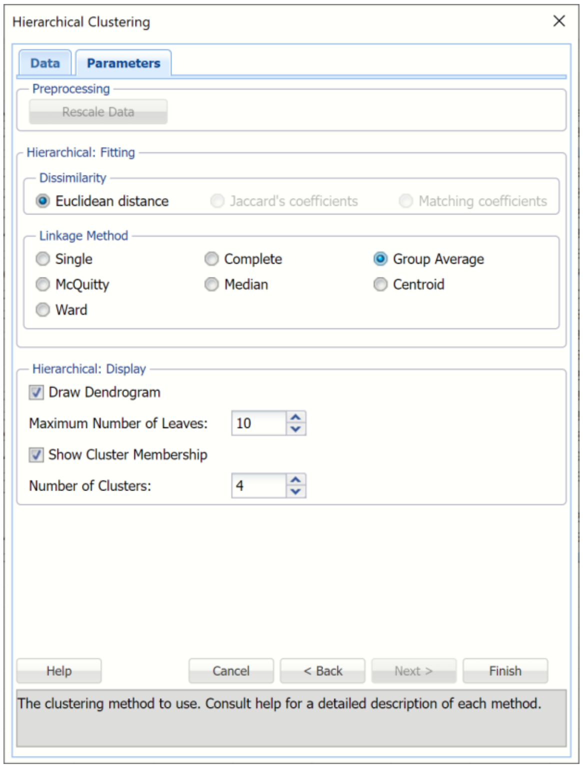
- To include a variable in Hierarchical Clustering, select the variable in the Variables list, then click > to move the variable to the Selected Variables list.
- To remove a variable as a selected variable, click the variable in the Selected Variables list, then click < to move the variable back to the Variables list.

Data Type

The Hierarchical clustering method can be used on raw data as well as the data in Distance Matrix format. Choose the appropriate option to fit your dataset. If *Raw Data* is chosen, Analytic Solver Data Science computes the similarity matrix before clustering.

Hierarchical Clustering Parameters tab

See below for documentation for all options appearing on the Hierarchical Clustering Parameters tab.



● Euclidean distance ○ Jaccard's coefficients ○ Matching coefficients

○ Single ○ Complete ● Group Average

○ McQuitty ○ Median ○ Centroid

○ Ward

☑ Draw Dendrogram

Maximum Number of Leaves: 10

☑ Show Cluster Membership

Number of Clusters: 4

The clustering method to use. Consult help for a detailed description of each method.

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Normalizing the data is important to ensure that the distance measure accords equal weight to each variable -- without normalization, the variable with the largest scale will dominate the measure. This option is disabled when Distance Matrix is selected for Data Type on the Data tab.

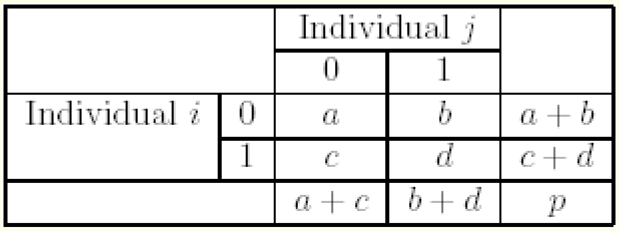
Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Dissimilarity Measures

Hierarchical clustering uses the Euclidean Distance as the similarity measure for working on raw numeric data.

When the data is binary, the remaining two options, Jaccard's coefficients and Matching coefficient are enabled.

Suppose we have binary values for all the x_{ij} 's. See the table below for individual *i's* and *j's.*



		Individual j		
		0	1	
Individual i	0	a	b	$a+b$
	1	c	d	$c+d$
		$a+c$	$b+d$	p

The most useful similarity measures in this situation are:

Jaccard’s coefficient = d/(b+c+d). This coefficient ignores zero matches.

The matching coefficient = (a + d)/p.

Clustering Method

Single linkage clustering

One of the simplest agglomerative hierarchical clustering methods is single linkage, also known as the nearest neighbor technique. The defining feature of this method is that distance between groups is defined as the distance between the closest pair of objects, where only pairs consisting of one object from each group are considered.

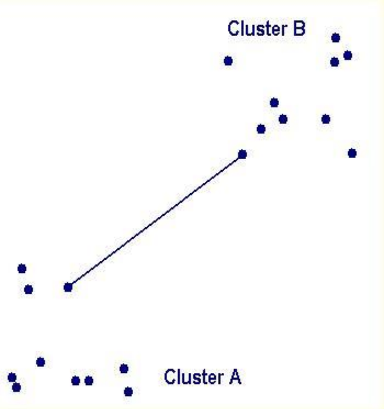
In the single linkage method, ***D(r,s)*** is computed as

D(r,s) = Min { *d(i,j)* : Where object *i* is in cluster ***r*** and object *j* is cluster ***s*** }

Here the distance between every possible object pair *(i,j)* is computed, where object *i* is in cluster ***r*** and object *j* is in cluster ***s***. The minimum value of these distances is said to be the distance between clusters ***r*** and ***s.*** In other words, the distance between two clusters is given by the value of the shortest link between the clusters.

At each stage of hierarchical clustering, the clusters ***r*** and ***s***, for which ***D(r,s)*** is minimum, are merged.

This measure of inter-group distance is illustrated in the figure below:



Complete linkage clustering

The complete linkage, also called farthest neighbor, clustering method is the opposite of single linkage. In this clustering method, the distance between groups is defined as the distance between the most distant pair of objects, one from each group.

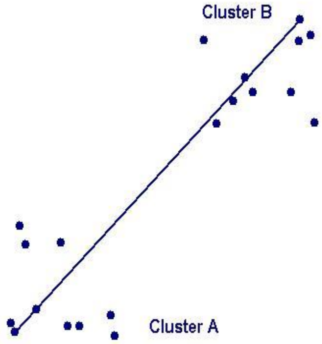
In the complete linkage method, ***D(r,s)*** is computed as

D(r,s) = Max { *d(i,j)* : Where object *i* is in cluster ***r*** and object *j* is cluster ***s*** }

Here the distance between every possible object pair *(i,j)* is computed, where object *i* is in cluster ***r*** and object *j* is in cluster ***s*** and the maximum value of these distances is said to be the distance between clusters ***r*** and ***s.*** In other words, the distance between two clusters is given by the value of the longest link between the clusters.

At each stage of hierarchical clustering, the clusters ***r*** and ***s***, for which ***D(r,s)*** is at the maximum, are merged.

The measure is illustrated in the figure below:



Average linkage clustering

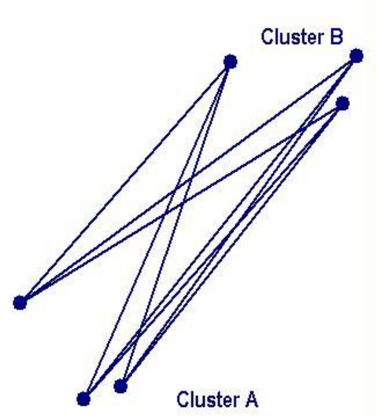
Here the distance between two clusters is defined as the average of distances between all pairs of objects, where each pair is made up of one object from each group.

In the average linkage method, ***D(r,s)*** is computed as

D(r,s) = $T_{rs} / (N_r * N_s)$

Where T_{rs} is the sum of all pairwise distances between cluster ***r*** and cluster ***s.*** N_r and N_s are the sizes of the clusters ***r*** and ***s*** respectively**.**

At each stage of hierarchical clustering, the clusters ***r*** and ***s***, for which ***D(r,s)*** is the minimum, are merged. The figure below illustrates average linkage clustering:



Centroid Method

With this method, groups once formed are represented by their mean values for each variable, that is, their mean vector, and inter-group distance is now defined in terms of distance between two such mean vectors.

In the group average linkage method, the two clusters **r** and **s** are merged such that, after merging, the average pairwise distance within the newly formed cluster, is minimized. Suppose we label the new cluster formed by merging clusters **r** and **s**, as **t**. Then **D(r,s)** , the distance between clusters **r** and **s** is computed as

D(r,s) = Average { d(i,j) : Where observations i and j are in cluster **t**, the cluster formed by merging clusters **r** and **s** }

At each stage of hierarchical clustering, the clusters **r** and **s**, for which **D(r,s)** is minimized, are merged. In this case, those two clusters are merged such that the newly formed cluster, on average, will have minimum pairwise distances between the points.

Ward's hierarchical clustering method

Ward (1963) proposed a clustering procedure seeking to form the partitions $P_{n,}$ $P_{n-1,........,}$ P_1 in a manner that minimizes the loss associated with each grouping, and to quantify that loss in a form that is readily interpretable. At each step in the analysis, the union of every possible cluster pair is considered and the two clusters whose fusion results in the minimum increase in 'information loss' are combined. Information loss is defined by Ward in terms of an error sum-of-squares criterion, ESS.

The rationale behind Ward's proposal can be illustrated most simply by considering univariate data. Suppose for example, 10 objects have scores (2, 6, 5, 6, 2, 2, 2, 2, 0, 0, 0) on some particular variable. The loss of information that would result from treating the ten scores as one group with a mean of 2.5 is represented by ESS given by,

$\text{ESS}_{\text{One group}} = (2 -2.5)^2 + (6 -2.5)^2 + ... + (0 -2.5)^2 = 50.5$

On the other hand, if the 10 objects are classified according to their scores into four sets,

{0,0,0}, {2,2,2,2}, {5}, {6,6}

The ESS can be evaluated as the sum of squares of four separate error sums of squares

$ESS_{\text{One group}} = ESS_{\text{group1}} + ESS_{\text{group2}} + ESS_{\text{group3}} + ESS_{\text{group4}} = 0.0$

Clustering the 10 scores into 4 clusters results in no loss of information.

McQuitty's Method

When this procedure is selected, at each step, when two clusters are to be joined, the distance of the new cluster to an existing cluster is computed as the average of the distances from the proposed cluster to the existing cluster.

Median Method

The Median Method also uses averaging when calculating the distance between two records or observations. However, this method uses the median instead of the mean.

Draw Dendrogram

Select this option to have Analytic Solver Data Science create a dendrogram to illustrate the clustering process.

Maximum Number of Leaves

If Draw Dendrogram is selected, this option is enabled. Use this option to define the maximum number of leaves in the dendrogram tree. The default setting is equal to 10 or if the number of records in the dataset is less than 10; then the default is the Minimum between the Number of Rows in the dataset and your current licensed limit.

Show cluster membership

Select this option to display the cluster number (ID) to which each record is assigned by the routine.

Number of Clusters

Recall that the agglomerative method of hierarchical clustering continues to form clusters until only one cluster is left. This option lets you stop the process at a given number of clusters.

Text Mining

Introduction

Text mining is the practice of automated analysis of one document or a collection of documents (corpus) and extracting non-trivial information from it. Also, Text Mining usually involves the process of transforming unstructured textual data into structured representation by analyzing the patterns derived from text. The results can be analyzed to discover interesting knowledge, some of which would only be found by a human carefully reading and analyzing the text. Typical widely-used tasks of Text Mining include but are not limited to Automatic Text Classification/Categorization, Topic Extraction, Concept Extraction, Documents/Terms Clustering, Sentiment Analysis, Frequency-based Analysis and many more. Some of these tasks could not be completed by a human, which makes Text Mining a particularly useful and applicable tool in modern Data Science. Analytic Solver Data Science takes an integrated approach to text mining as it does not totally separate analysis of unstructured data from traditional data science techniques applicable for structured information. While Analytic Solver Data Science is a very powerful tool for analyzing text only, it also offers automated treatment of mixed data, i.e. combination of multiple unstructured and structured fields. This is a particularly useful feature that has many real-world applications, such as analyzing maintenance reports, evaluation forms, insurance claims, etc. Analytic Solver Data Science uses the “bag of words” model – the simplified representation of text, where the precise grammatical structure of text and exact word order is disregarded. Instead, syntactic, frequency-based information is preserved and is used for text representation. Although such assumptions might be harmful for some specific applications of Natural Language Processing (NLP), it has been proven to work very well for applications such as Text Categorization, Concept Extraction and others, which are the particular areas addressed by Analytic Solver Data Science’s Text Mining capabilities. It has been shown in many theoretical/empirical studies that syntactic similarity often implies semantic similarity. One way to access syntactic relationships is to represent text in terms of Generalized Vector Space Model (GVSP). Advantage of such representation is a meaningful mapping of text to the numeric space, the disadvantage is that some semantic elements, e.g. order of words, are lost (recall the bag-of-words assumption).

Input to Text Miner (the Text Mining tool within Analytic Solver Data Science) could be of two main types – few relatively large documents (e.g. several books) or relatively large number of smaller documents (e.g. collection of emails, news articles, product reviews, comments, tweets, Facebook posts, etc.). While Analytic Solver Data Science is capable of analyzing large text documents, it is particularly effective for large corpuses of relatively small documents. Obviously, this functionality has limitless number of applications – for instance, email spam detection, topic extraction in articles, automatic rerouting of correspondence, sentiment analysis of product reviews and many more.

The input for text mining is a dataset on a worksheet, with at least one column that contains free-form text (or file paths to documents in a file system containing free-form text), and, optionally, other columns that contain traditional structured data. In the first tab of the Text Mining dialog, the user selects the text variable(s), and the other variable(s) to be processed.

The output for the text mining is a set of reports that contain general explorative information about the collection of documents and structured representations of text (free-form text columns are expanded to a set of new columns with numeric representation. The new columns will each correspond to either (i) a single term (word) found in the "corpus" of documents, or, if requested, (ii) a concept extracted from the corpus through Latent Semantic Indexing (LSI, also called LSA or Latent Semantic Analysis). Each concept represents an automatically derived complex combination of terms/words that have been identified to be related to a particular topic in the corpus of documents. The structural representation of text can serve as an input to any traditional data science techniques available in Analytic Solver Data Science – unsupervised/supervised, affinity, visualization techniques, etc. In addition, Analytic Solver Data Science also presents a visual representation of Text Mining results to allow the user to interactively explore the information, which otherwise would be extremely hard to analyze manually. Typical visualizations that aid in understanding of Text Mining outputs and that are produced by Analytic Solver Data Science are:

- Zipf plot – for visual/interactive exploration of frequency-based information extracted by Text Miner

- Scree Plot, Term-Concept and Document-Concept 2D scatter plots – for visual/interactive exploration of Concept Extraction results

If you are interested in visualizing specific parts of Text Mining analysis outputs, Analytic Solver Data Science provides rich capabilities for charting – the functionality that can be used to explore Text Mining results and supplement standard charts discussed above.

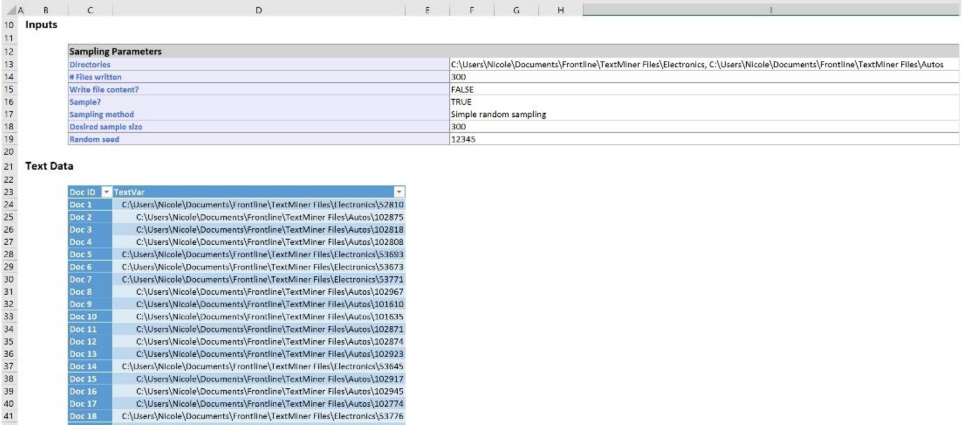
In the example below, you will learn how to use Text Miner in Analytic Solver Data Science to process/analyze approximately 1000 text files and use the results for automatic topic categorization. This will be achieved by using structured representation of text presented to Logistic Regression for building the model for classification.

Text Mining Example

This example uses the text files within the Text Mining Example Documents.zip archive file to illustrate how to use Analytic Solver Data Science's Text Mining tool. These documents were selected from the well-known text dataset (downloadable from http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/news20.html) which consists of 20,000 messages, collected from 20 different internet newsgroups. We selected about 1,200 of these messages that were posted to two interest groups, Autos and Electronics (about 500 documents from each).

Note: The Data Science Cloud app does not currently support importing from a File Folder.

The Text Mining Example Documents.zip archive file is located at C:\ProgramData\Frontline Systems\Datasets. See the section *Importing from a File Folder* within the **Sampling or Importing from a Database, Worksheet or File Folder** chapter for directions on extracting and importing the text files into Analytic Solver Data Science. We will pick up where this example leaves off, after the files have been imported and a sample created.



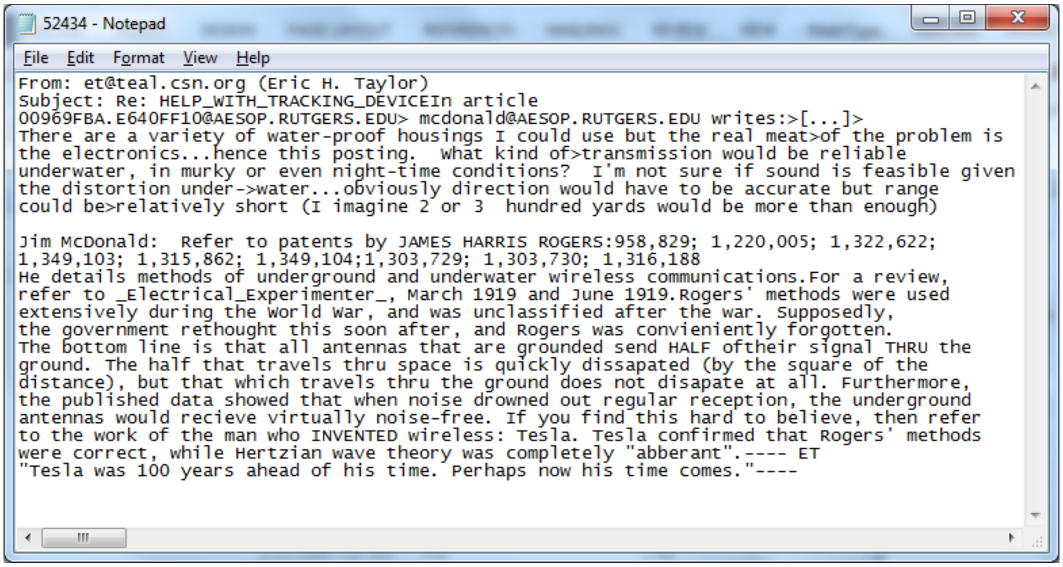
Inputs

Sampling Parameters	
Directories	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics, C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos
# Files written	300
Write file content?	FALSE
Sample?	TRUE
Sampling method	Simple random sampling
Desired sample size	300
Random seed	12345

Text Data

Doc ID	TextVar
Doc 1	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics\52810
Doc 2	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102875
Doc 3	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102818
Doc 4	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102808
Doc 5	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics\53693
Doc 6	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics\53673
Doc 7	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics\53771
Doc 8	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102967
Doc 9	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101610
Doc 10	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101635
Doc 11	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102871
Doc 12	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102874
Doc 13	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102923
Doc 14	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics\53645
Doc 15	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102917
Doc 16	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102945
Doc 17	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\102774
Doc 18	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Electronics\53776

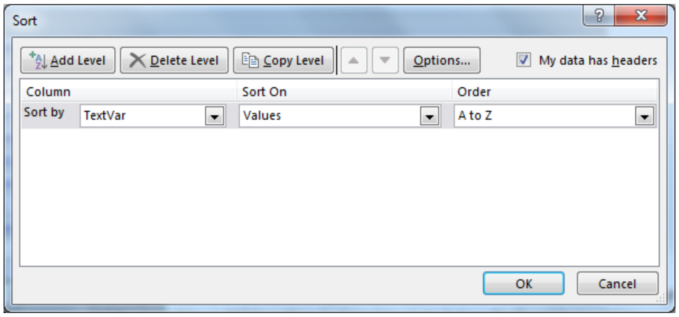
Here is an example of a document that appeared in the Electronics newsgroup. Note the appearance of email addresses, "From" and "Subject" lines. All three appear in each document.



```
From: et@teal.csn.org (Eric H. Taylor)
Subject: Re: HELP_WITH_TRACKING_DEVICEIn article
00969FBA.E640FF10@AESOP.RUTGERS.EDU> mcdonald@AESOP.RUTGERS.EDU writes:>[...]>
There are a variety of water-proof housings I could use but the real meat>of the problem is
the electronics...hence this posting.  What kind of>transmission would be reliable
underwater, in murky or even night-time conditions?  I'm not sure if sound is feasible given
the distortion under->water...obviously direction would have to be accurate but range
could be>relatively short (I imagine 2 or 3  hundred yards would be more than enough)

Jim McDonald:  Refer to patents by JAMES HARRIS ROGERS:958,829; 1,220,005; 1,322,622;
1,349,103; 1,315,862; 1,349,104;1,303,729; 1,303,730; 1,316,188
He details methods of underground and underwater wireless communications.For a review,
refer to _Electrical_Experimenter_, March 1919 and June 1919.Rogers' methods were used
extensively during the World War, and was unclassified after the war. Supposedly,
the government rethought this soon after, and Rogers was convieniently forgotten.
The bottom line is that all antennas that are grounded send HALF oftheir signal THRU the
ground. The half that travels thru space is quickly dissapated (by the square of the
distance), but that which travels thru the ground does not disapate at all. Furthermore,
the published data showed that when noise drowned out regular reception, the underground
antennas would recieve virtually noise-free. If you find this hard to believe, then refer
to the work of the man who INVENTED wireless: Tesla. Tesla confirmed that Rogers' methods
were correct, while Hertzian wave theory was completely "abberant".---- ET
"Tesla was 100 years ahead of his time. Perhaps now his time comes."----
```

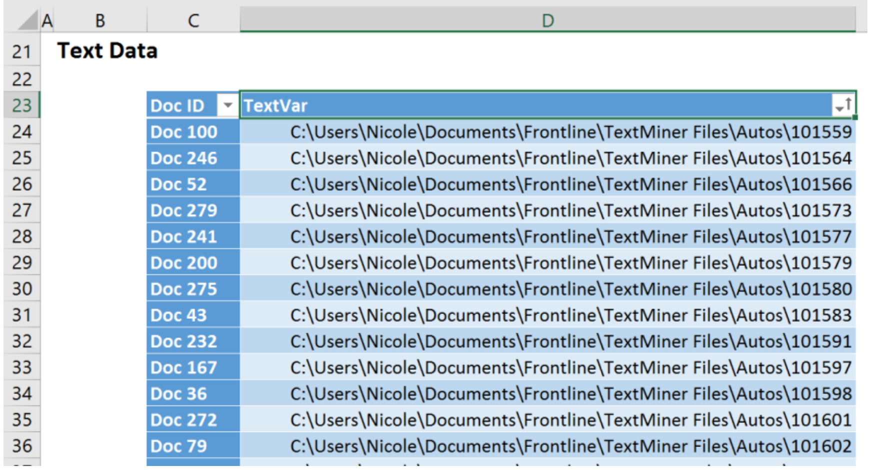
The selected file paths are now in random order, but we will need to categorize the "Autos" and "Electronics" files in order to be able to identify them later. To do this, we'll use Excel to sort the rows by the file path: Select columns C through D and rows 23 through 323, then choose **Sort** from the Data tab. In the Sort dialog, select column d, where the file paths are located, and click OK.



☑ My data has headers

	Column	Sort On	Order
Sort by	TextVar	Values	A to Z

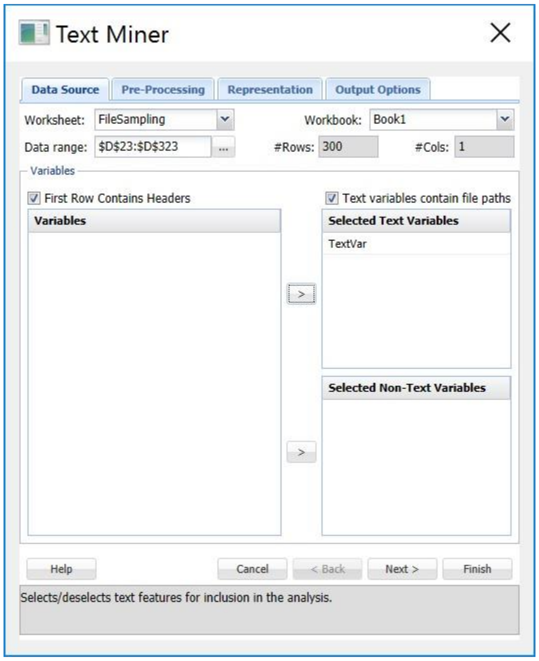
The file paths should now be sorted between *Electronics* and *Autos* files.



Text Data

Doc ID	TextVar
Doc 100	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101559
Doc 246	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101564
Doc 52	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101566
Doc 279	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101573
Doc 241	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101577
Doc 200	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101579
Doc 275	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101580
Doc 43	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101583
Doc 232	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101591
Doc 167	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101597
Doc 36	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101598
Doc 272	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101601
Doc 79	C:\Users\Nicole\Documents\Frontline\TextMiner Files\Autos\101602

On the Data Science Platform Ribbon tab, click the **Text** icon to bring up the *Text Miner* dialog. Select **TextVar** in the *Variables* list box, and click the upper > button to move it to the *Selected Text Variables* list box. By doing so, we are selecting the text in the documents as input to the Text Miner model. Ensure that "Text variables contain file paths" is **checked**.



Text Miner

Data Source | Pre-Processing | Representation | Output Options

Worksheet: FileSampling Workbook: Book1

Data range: $D$23:$D$323 #Rows: 300 #Cols: 1

Variables

☑ First Row Contains Headers

☑ Text variables contain file paths

Variables

Selected Text Variables: TextVar

Selected Non-Text Variables

Selects/deselects text features for inclusion in the analysis.

Click the **Next** button, or click the **Pre-Processing** tab at the top.

Leave the default setting for *Analyze all terms* selected under *Mode*. When this option is selected, Analytic Solver Data Science will examine all terms in the document. A "term" is defined as an individual entity in the text, which may or may not be an English word. A term can be a word, number, email, url, etc. terms are separated by all possible delimiting characters (i.e. \, ?, ', `, ~, |, \r, \n, \t, :, !, @, #, $, %, ^, &, *, (,), [,], {, }, <>,_, ;, =, -, +, \) with some exceptions related to stopwords, synonyms, exclusion terms and boilerplate normalization (URLs, emails, monetary amounts, etc.). Text Miner will not tokenize on these delimiters.

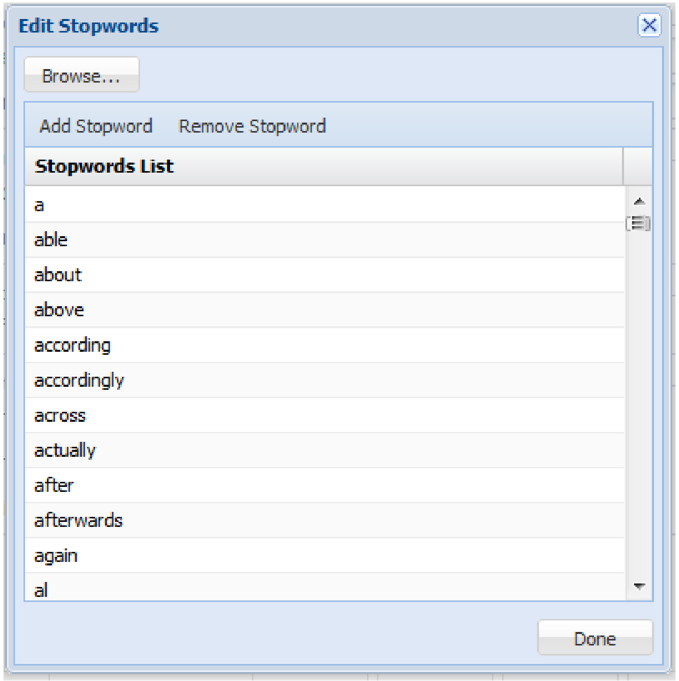
Note: Exceptions are related not to how terms are separated but as to whether they are split based on the delimiter. For example: URL's contain many characters such as "/", ";", etc. Text Miner will not tokenize on these characters in the URL but will consider the URL as a whole and will remove the URL if selected for removal. (See below for more information.)

If *Analyze specified terms only* is selected, the *Edit Terms* button will be enabled. If you click this button, the *Edit Exclusive Terms* dialog opens. Here you can add and remove terms to be considered for text mining. All other terms will be disregarded. For example, if we wanted to mine each document for a specific part name such as "alternator" we would click *Add Term* on *the Edit Exclusive Terms* dialog, then replace "New term" with "alternator" and click *Done* to return to the Pre-Processing dialog. During the text mining process, Analytic Solver Data Science would analyze each document for the term "alternator", excluding all other terms.

Leave both Start term/phrase and End term/phrase empty under Text Location. If this option is used, text appearing before the first occurrence of the Start Phrase will be disregarded and similarly, text appearing after End Phrase (if used) will be disregarded. For example, if text mining the transcripts from a Live Chat service, you would not be particularly interested in any text appearing before the heading "Chat Transcript" or after the heading "End of Chat Transcript". Thus you would enter "Chat Transcript" into the Start Phrase field and "End of Chat Transcript" into the End Phrase field.

Leave the default setting for *Stopword removal*. Click Edit to view a list of commonly used words that will be removed from the documents during pre-processing. To remove a word from the Stopword list, simply highlight the desired word, then click **Remove Stopword.** To add a new word to the list, click **Add Stopword**, a new term "stopword" will be added. Double click to edit.

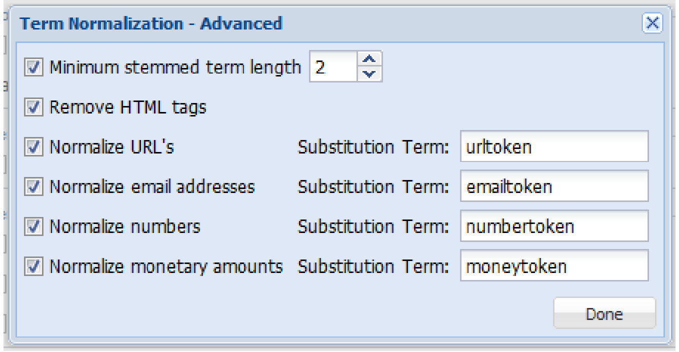
Analytic Solver Data Science also allows additional stopwords to be added or existing to be removed via a text document (*.txt) by using the Browse button to navigate to the file. Terms in the text document can be separated by a space, a comma, or both. If we were supplying our three terms in a text document, rather than in the *Edit Stopwords* dialog, the terms could be listed as: subject emailterm from or subject,emailterm,from or subject, emailterm, from. If we had a large list of additional stopwords, this would be the preferred way to enter the terms.



Stopwords List
a
able
about
above
according
accordingly
across
actually
after
afterwards
again
al

Click **Advanced** in the *Term Normalization* group to open the *Term Normalization – Advanced* dialog. **Select all options** as shown below. Then click **Done**. This dialog allows us to indicate to Analytic Solver Data Science, that

- If stemming reduced term length to 2 or less characters, disregard the term (*Minimum stemmed term length).*
- HTML tags, and the text enclosed, will be removed entirely. HTML tags and text contained inside these tags often contain technical, computer-generated information that is not typically relevant to the goal of the text mining application.
- URLs will be replaced with the term, "urltoken". Specific form of URLs do not normally add any meaning, but it is sometimes interesting to know how many URLs are included in a document.
- Email addresses will be replaced with the term, "emailtoken". Since the documents in our collection all contain a great many email addresses (and the distinction between the different emails often has little use in Text Mining), these email addresses will be replaced with the term "emailtoken".
- Numbers will be replaced with the term, "numbertoken".
- Monetary amounts will be substituted with the term, "moneytoken".



☑ Minimum stemmed term length 2

☑ Remove HTML tags

☑ Normalize URL's Substitution Term: urltoken

☑ Normalize email addresses Substitution Term: emailtoken

☑ Normalize numbers Substitution Term: numbertoken

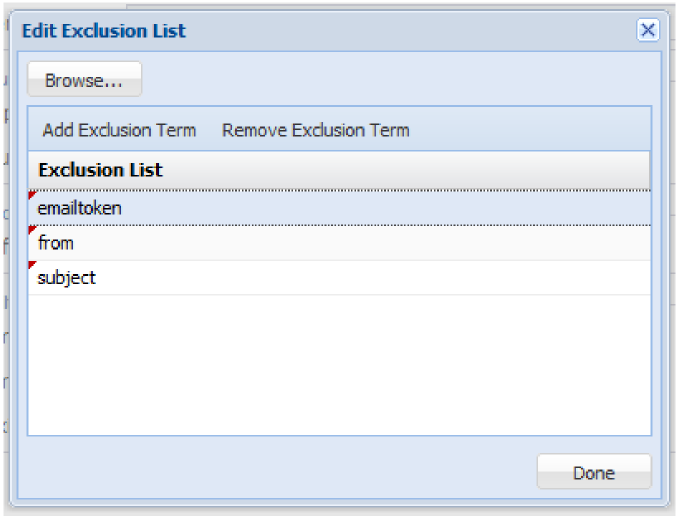
☑ Normalize monetary amounts Substitution Term: moneytoken

Recall that when we inspected an email from the document collection we saw several terms such as "subject", "from" and email addresses. Since all of our documents contain these terms, including them in the analysis will not provide any benefit and could bias the analysis. As a result, we will exclude these terms from all documents by selecting **Exclusion list** then clicking **Edit**. The *Edit Exclusion List* dialog opens. Click **Add Exclusion Term**. The label "exclusionterm" is added. Click to edit and change to "subject". Then repeat these same steps to add the term "from".

We can take the email issue one step further and completely remove the term "emailtoken" from the collection. Click **Add Exclusion Term** and edit "exclusionterm" to "emailtoken".

To remove a term from the exclusion list, highlight the term and click *Remove Exclusion Term*.

We could have also entered these terms into a text document (*.txt) and added the terms all at once by using the Browse button to navigate to the file and import the list. Terms in the text document can be separated by a space, a comma, or both. If, for example we were supplying excluded terms in a document rather than in the Edit Exclusion List dialog, we would enter the terms as: subject emailtoken from, or subject,emailtoken,from, or subject, emailtoken, from. If we had a large list of terms to be excluded, this would be the preferred way to enter the terms.



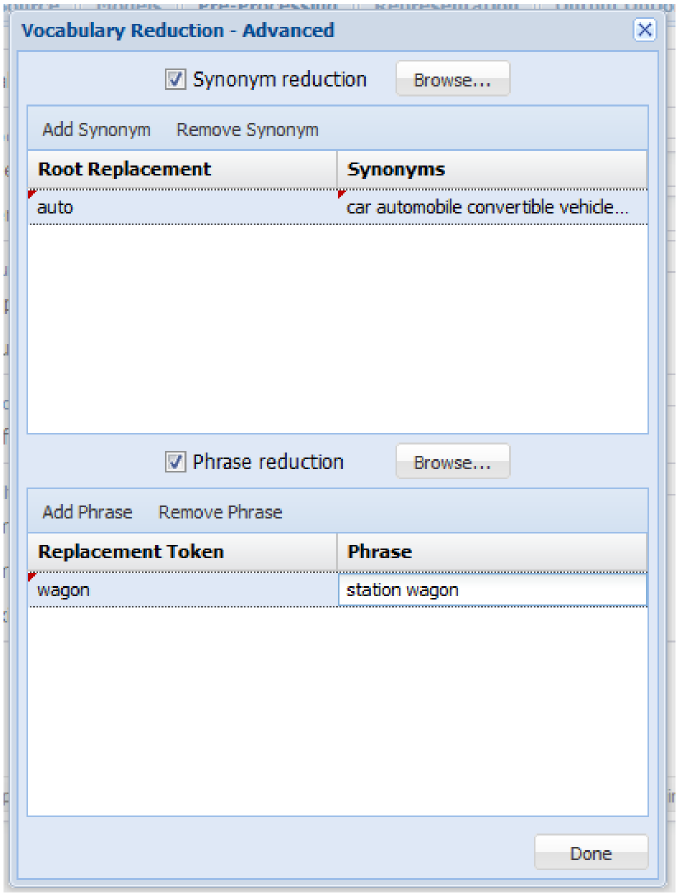
Exclusion List
emailtoken
from
subject

Click **Done** to close the dialog and return to Pre-Processing.

Analytic Solver Data Science also allows the combining of synonyms and full phrases by clicking **Advanced** within *Vocabulary Reduction*. Select **Synonym reduction** at the top of the dialog to replace synonyms such as "car", "automobile", "convertible", "vehicle", "sedan", "coupe", "subcompact", and "jeep" with "auto". Click **Add Synonym** and replace "rootterm" with "auto" then replace "synonym list" with "car, automobile, convertible, vehicle, sedan, coupe" (without the quotes). During pre-processing, Analytic Solver Data Science will replace the terms "car", "automobile", "convertible", "vehicle", "sedan", "coupe", "subcompact" and "jeep" with the term "auto". To remove a synonym from the list, highlight the term and click *Remove Synonym*.

If adding synonyms from a text file, each line must be of the form rootterm:synonymlist or using our example: auto:car automobile convertible vehicle sedan coup or auto:car,automobile,convertible,vehicle,sedan,coup. Note separation between the terms in the synonym list be either a space, a comma or both.   If we had a large list of synonyms, this would be the preferred way to enter the terms.

Analytic Solver Data Science also allows the combining of words into phrases that indicate a singular meaning such as “station wagon” which refers to a specific type of car rather than two distinct tokens – station and wagon. To add a phrase in the *Vocabulary Reduction – Advanced* dialog, select **Phrase reduction** and click **Add Phrase**. The term “phrasetoken” will be appear, click to edit and enter “wagon”. Click “phrase” to edit and enter “station wagon”. If supplying phrases through a text file (*.txt), each line of the file must be of the form phrasetoken:phrase or using our example, wagon:station wagon. If we had a large list of phrases, this would be the preferred way to enter the terms.



Vocabulary Reduction - Advanced

☑ Synonym reduction

Root Replacement	Synonyms
auto	car automobile convertible vehicle...

☑ Phrase reduction

Replacement Token	Phrase
wagon	station wagon

Enter **200** for *Maximum Vocabulary Size*. Analytic Solver Data Science will reduce the number of terms in the final vocabulary to the top 200 most frequently occurring in the collection.

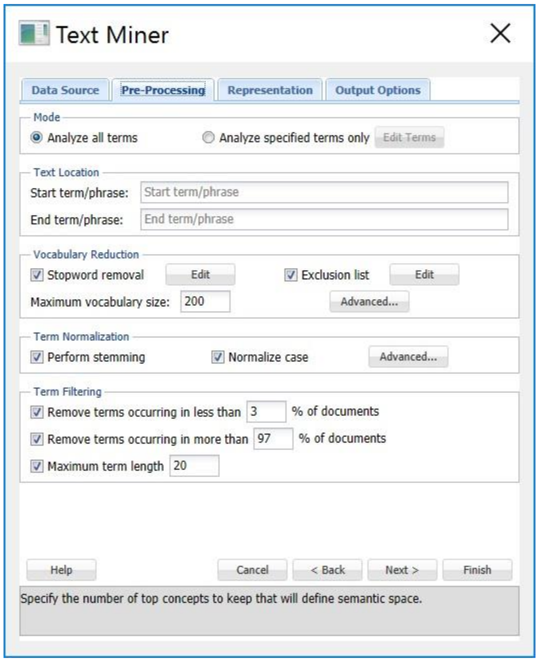
Leave *Perform stemming* at the selected default. Stemming is the practice of stripping words down to their “stems” or “roots”, for example, stemming terms such as “argue”, “argued”, “argues”, “arguing”, and “argus” would result in the stem “argu. However “argument” and “arguments” would stem to “argument”.

The stemming algorithm utilized in Analytic Solver Data Science is “smart” in the sense that while “running” would be stemmed to “run”, “runner” would not. . Analytic Solver Data Science uses the Porter Stemmer 2 algorithm for the English Language. For more information on this algorithm, please see the Webpage: http://tartarus.org/martin/PorterStemmer/

Leave the default selection for *Normalize case.* When this option is checked, Analytic Solver Data Science converts all text to a consistent (lower) case, so that Term, term, TERM, etc. are all normalized to a single token “term” before any processing, rather than creating three independent tokens with different case. This simple method can dramatically affect the frequency distributions of the corpus, leading to biased results.

Enter **3** for *Remove terms occurring in less than _% of documents* and **97** for *Remove terms occurring in more than _% of documents*. For many text mining applications, the goal is to identify terms that are useful for discriminating between documents. If a particular term occurs in all or almost all documents, it may not be possible to highlight the differences. If a term occurs in very few documents, it will often indicate great specificity of this term, which is not very useful for some Text Mining purposes.

Enter **20** for *Maximum term length.* Terms that contain more than 20 characters will be excluded from the text mining analysis and will not be present in the final reports. This option can be extremely useful for removing some parts of text which are not actual English words, for example, URLs or computer-generated tokens, or to exclude very rare terms such as Latin species or disease names, i.e. Pneumonoultramicroscopicsilicovolcanoconiosis.



Mode
● Analyze all terms ○ Analyze specified terms only

Text Location
Start term/phrase: Start term/phrase
End term/phrase: End term/phrase

Vocabulary Reduction
☑ Stopword removal ☑ Exclusion list
Maximum vocabulary size: 200

Term Normalization
☑ Perform stemming ☑ Normalize case

Term Filtering
☑ Remove terms occurring in less than 3 % of documents
☑ Remove terms occurring in more than 97 % of documents
☑ Maximum term length 20

Specify the number of top concepts to keep that will define semantic space.

Click **Next** to advance to *the Representation* tab or simply click **Representation** at the top.

Keep the default selection of *TF-IDF* (Term Frequency – Inverse Document Frequency) for *Term-Document Matrix Scheme*. A term-document matrix is a matrix that displays the frequency-based information of terms occurring in a document or collection of documents. Each column is assigned a term and each row a document. If a term appears in a document, a weight is placed in the corresponding column indicating the term's importance or contribution. Analytic Solver Data Science offers four different commonly used methods of weighting scheme used to represent each value in the matrix: Presence/Absence, Term Frequency, TF-IDF (the default) and Scaled term frequency. If *Presence/Absence* is selected, Analytic Solver Data Science will enter a 1 in the corresponding row/column if the term appears in the document and 0 otherwise. This matrix scheme does not take into account the number of times the term occurs in each document. If *Term Frequency* is selected, Analytic Solver Data Science will count the number of times the term appears in the document and enter this value into the corresponding row/column in the matrix. The default setting – Term Frequency – Inverse Document Frequency (*TF-IDF)* is the product of scaled term frequency and inverse document frequency. Inverse document frequency is calculated by taking the logarithm of the total number of documents divided by the number of documents that contain the term. A high value for TF-IDF indicates that a term that does not occur frequently in the collection of documents taken as a whole, appears quite frequently in the specified document. A TF-IDF value close to 0 indicates that the term appears frequently in the collection or rarely in a specific document. If *Scaled term frequency* is selected, Analytic Solver Data Science will normalize (bring to the same scale) the number of occurrences of a term in the documents (see the table below)..

It's also possible to create your own scheme by clicking the *Advanced* command button to open the *Term Document Matrix – Advanced* dialog. Here users can select their own choices for local weighting, global weighting, and normalization. Please see the table below for definitions regarding options for Term Frequency, Document Frequency and Normalization.

Local Weighting		**Global Weighting**		**Normalization**	
Binary	$lw_{td} = \begin{cases} 1, \text{if } tf_{td} > 0 \\ 0, \text{if } tf_{td} = 0 \end{cases}$	None	$gw_t = 1$	None	$n_d = 1$
Raw Frequency	$lw_{td} = tf_{td}$	Inverse	$gw_t = \log_2 \frac{N}{1 + df_t}$	Cosine	$n_d = \frac{1}{\lVert g_d \rVert_2}$
Logarithmic	$lw_{td} = \log(1 + tf_{td})$	Normal	$gw_t = \frac{1}{\sqrt{\sum_d tf_{td}^2}}$		
Augnorm	$lw_{td} = \frac{\left(\frac{tf_{td}}{\max_t tf_{td}}\right) + 1}{2}$	GF-IDF	$gw_t = \frac{cf_t}{df_t}$		
		Entropy	$gw_t = 1 + \sum_d \frac{p_{td} \log p_{td}}{\log N}$		
		IDF probability	$gw_t = \log_2 \frac{N}{1 + df_t}$		

Notations:

- tf_{td} – frequency of term t in a document d;
- df_t – document frequency of term t;
- lw_{td} – local weighting of term t in a document d;

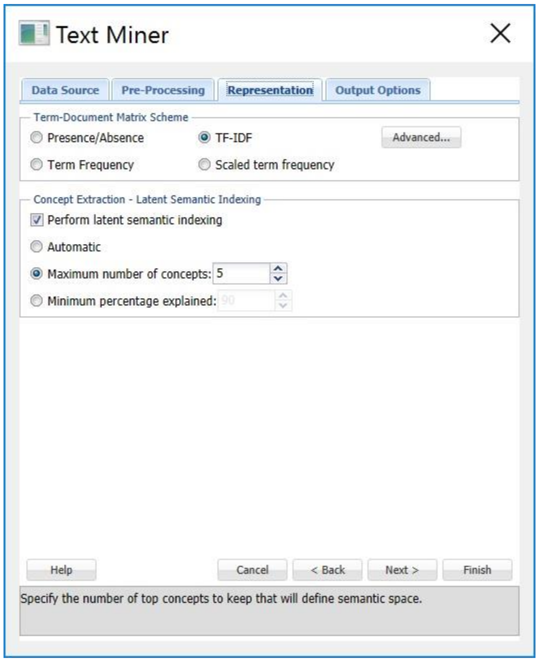
- gw_{td} – global weighting of term t in a document d;
- n_d – normalization of vector of terms representing the document d;
- N – total number of documents in the collection;
- cf_t – collection frequency of term t;
- p_{td} – estimated probability of term t to appear in a document d $\left(p_{td} = {}^{tf_{td}}/_{cf_t}\right)$;
- $\overline{g_d}$ – vector of terms representing the document d.

Finally, the element T_{td} of Term-Document Matrix is computed as $T_{td} = lw_{td} * gw_t * n_d, \forall t, d$

Leave *Perform latent semantic indexing* selected (the default). When this option is selected, Analytic Solver Data Science will use Latent Semantic Indexing (LSI) to detect patterns in the associations between terms and concepts to discover the meaning of the document.

The statistics produced and displayed in the Term-Document Matrix contain basic information on the frequency of terms appearing in the document collection. With this information we can "rank" the significance or importance of these terms relative to the collection and particular document. Latent Semantic Indexing, in comparison, uses singular value decomposition (SVD) to map the terms and documents into a common space to find patterns and relationships. For example: if we inspected our document collection, we might find that each time the term "alternator" appeared in an automobile document, the document also included the terms "battery" and "headlights". Or each time the term "brake" appeared in an automobile document, the terms "pads" and "squeaky" also appeared. However there is no detectable pattern regarding the use of the terms "alternator" and "brake". Documents including "alternator" might not include "brake" and documents including "brake" might not include "alternator". Our four terms, battery, headlights, pads, and squeaky describe two different automobile repair issues: failing brakes and a bad alternator. Latent Semantic Indexing will attempt to 1. Distinguish between these two different topics, 2. Identify the documents that deal with faulty brakes, alternator problems or both and 3. Map the terms into a common semantic space using singular value decomposition. SVD is a tool used by Text Miner to extract concepts that explain the main dimensions of meaning of the documents in the collection. The results of LSA are usually hard to examine because the construction of the concept representations will not be fully explained. Interpreting these results is actually more of an art, than a science. However, Analytic Solver Data Science provides several visualizations that simplify this process greatly.

Select **Maximum number of concepts** and increment the counter to **20**. Doing so will tell Analytic Solver Data Science to retain the top 20 of the most significant concepts. If *Automatic* is selected, Analytic Solver Data Science will calculate the importance of each concept, take the difference between each and report any concepts above the largest difference. For example if three concepts were identified (Concept1, Concept2, and Concept3) and given importance factors of 10, 8, and 2, respectively, Analytic Solver Data Science would keep Concept1 and Concept2 since the difference between Concept2 and Concept 3 (8-2=6) is larger than the difference between Concept1 and Concept2 (10-8=2). If *Minimum percentage explained* is selected, Analytic Solver Data Science will identify the concepts with singular values that, when taken together, sum to the minimum percentage explained, 90% is the default.



Term-Document Matrix Scheme

○ Presence/Absence ● TF-IDF

○ Term Frequency ○ Scaled term frequency

Concept Extraction - Latent Semantic Indexing

☑ Perform latent semantic indexing

○ Automatic

● Maximum number of concepts: 5

○ Minimum percentage explained: 90

Specify the number of top concepts to keep that will define semantic space.

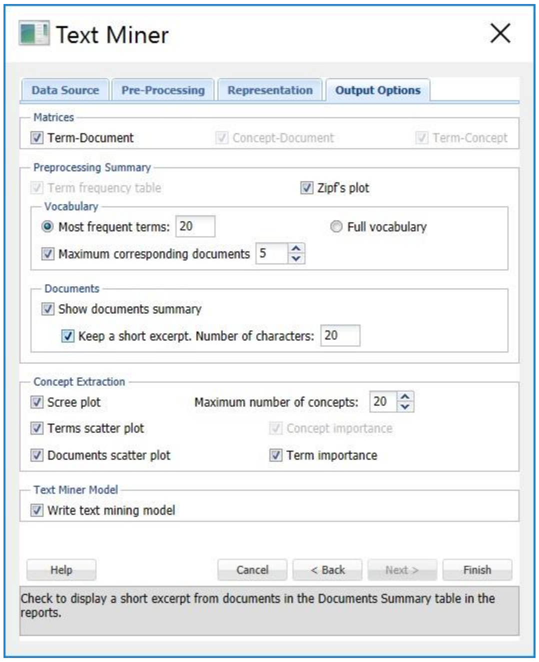
Click **Next** or the **Output Options** tab.

Keep *Term-Document* and *Concept-Document* selected under *Matrices* (the default) and select *Term-Concept* to print each matrix in the output. The *Term-Document* matrix displays the terms across the top of the matrix and the documents down the left side of the matrix. The *Concept – Document* and *Term – Concept* matrices are output from the *Perform latent semantic indexing* option that we selected on the *Representation* tab. In the first matrix, *Concept – Document*, 20 concepts will be listed across the top of the matrix and the documents will be listed down the left side of the matrix. The values in this matrix represent concept coordinates in the identified semantic space. In the *Term-Concept* matrix, the terms will be listed across the top of the matrix and the concepts will be listed down the left side of the matrix. The values in this matrix represent terms in the extracted semantic space.

Keep *Term frequency table* selected (the default) under *Preprocessing Summary* and select *Zipf's plot*. Increase the *Most frequent term*s to **20** and select *Maximum corresponding* documents. The Term frequency table will include the top 20 most frequently occurring terms. The first column, Collection Frequency, displays the number of times the term appears in the collection. The 2nd column, Document Frequency, displays the number of documents that include the term. The third column, Top Documents, displays the top 5 documents where the corresponding term appears the most frequently. The Zipf Plot graphs the document frequency against the term ranks in descending order of frequency.. Zipf's law states that the frequency of terms used in a free-form text drops exponentially, i.e. that people tend to use a relatively small number of words extremely frequently and use a large number of words very rarely.

Keep *Show documents summary* selected and check *Keep a short excerpt.* under *Documents*. Analytic Solver Data Science will produce a table displaying the document ID, length of the document, number of terms and 20 characters of the text of the document.

Select **all plots** under *Concept Extraction* to produce various plots in the output. Select *Write text mining model* under *Text Miner Model* to write the model to the output sheets.



Text Miner

Data Source | Pre-Processing | Representation | **Output Options**

Matrices
☑ Term-Document ☑ Concept-Document ☑ Term-Concept

Preprocessing Summary
☑ Term frequency table ☑ Zipf's plot

Vocabulary
● Most frequent terms: 20 ○ Full vocabulary
☑ Maximum corresponding documents 5

Documents
☑ Show documents summary
☑ Keep a short excerpt. Number of characters: 20

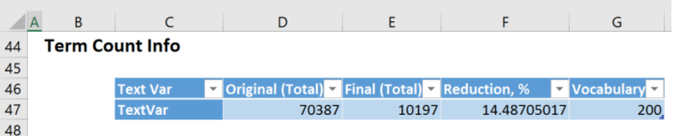
Concept Extraction
☑ Scree plot Maximum number of concepts: 20
☑ Terms scatter plot ☑ Concept importance
☑ Documents scatter plot ☑ Term importance

Text Miner Model
☑ Write text mining model

Check to display a short excerpt from documents in the Documents Summary table in the reports.

Click the **Finish** button to run the Text Mining analysis. Result worksheets are inserted to the right.

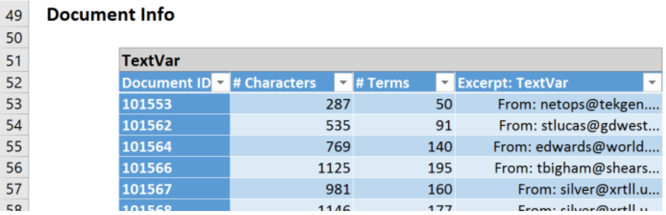
Select the *TM_Output* tab. The *Term Count* table shows that the original term count in the documents was reduced by 16.02% by the removal of stopwords, excluded terms, synonyms, phrase removal and other specified preprocessing procedures.



Term Count Info

Text Var	Original (Total)	Final (Total)	Reduction, %	Vocabulary
TextVar	70387	10197	14.48705017	200

Scroll down to the *Documents* table. This table lists each Document with its length, number of terms, and if *Keep a short excerpt* is selected on the *Output Options* tab and a value is present for *Number of characters*, then an excerpt from each document will be displayed.

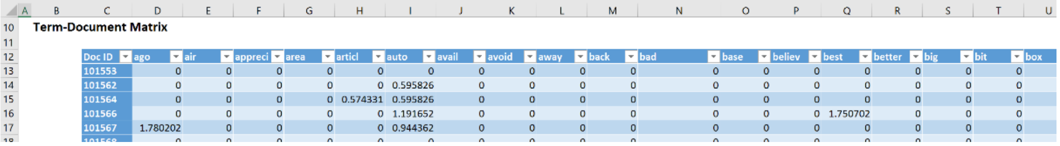


Document Info

TextVar

Document ID	# Characters	# Terms	Excerpt: TextVar
101553	287	50	From: netops@tekgen....
101562	535	91	From: stlucas@gdwest...
101564	769	140	From: edwards@world....
101566	1125	195	From: tbigham@shears...
101567	981	160	From: silver@xrtll.u...
101568	1146	177	From: silver@xrtll.u...

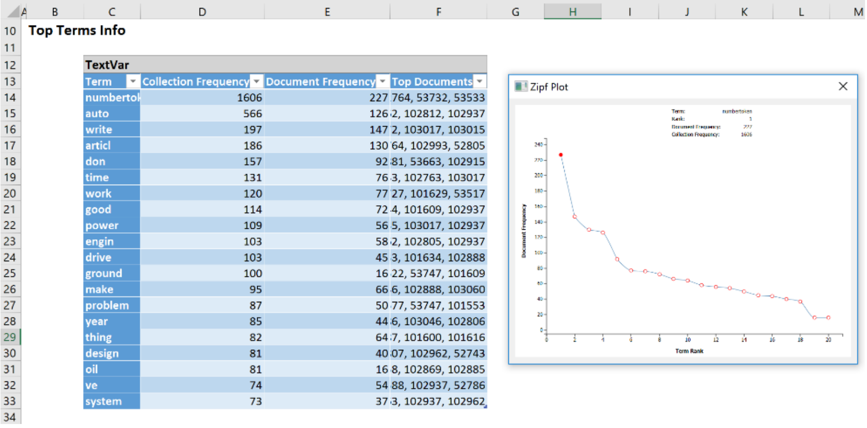
Double click T*M_TDM* to display the Term – Document Matrix. As discussed above, this matrix lists the 200 most frequently appearing terms across the top and the document IDs down the left. A portion of this table is shown below. If a term appears in a document, a weight is placed in the corresponding column indicating the importance of the term using our selection of TF-IDF on the Representation dialog.



Term-Document Matrix

Doc ID	ago	air	appreci	area	articl	auto	avail	avoid	away	back	bad	base	believ	best	better	big	bit	box
101553	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
101562	0	0	0	0	0	0.595826	0	0	0	0	0	0	0	0	0	0	0	
101564	0	0	0	0	0.574331	0.595826	0	0	0	0	0	0	0	0	0	0	0	
101566	0	0	0	0	0	1.191652	0	0	0	0	0	0	0	1.750702	0	0	0	
101567	1.780202	0	0	0	0	0.944362	0	0	0	0	0	0	0	0	0	0	0	

Click the TM_Vocabulary tab to view the Final List of Terms table. This table contains the top 20 terms occurring in the document collection, the number of documents that include the term and the top 5 document IDs where the corresponding term appears most frequently. In this list we see terms such as “car”, “power”, “engine”, “drive”, and “dealer” which suggests that many of the documents, even the documents from the electronic newsgroup, were related to autos.



Top Terms Info

TextVar

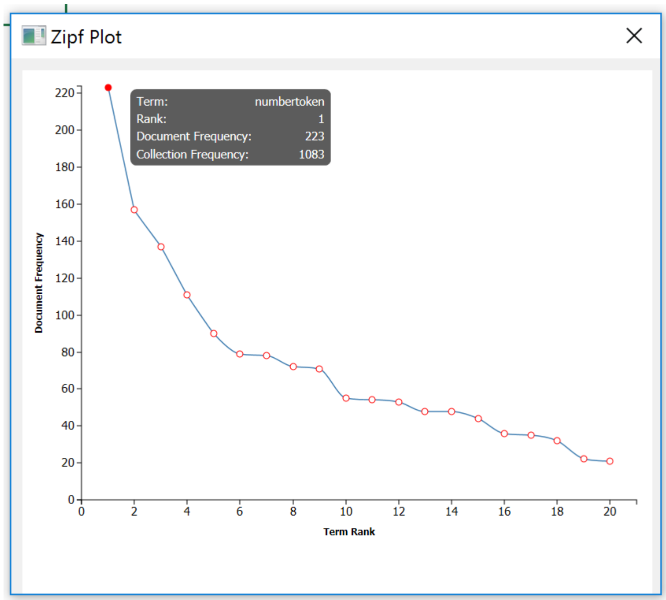
Term	Collection Frequency	Document Frequency	Top Documents
numbertok	1606	227	764, 53732, 53533
auto	566	126	2, 102812, 102937
write	197	147	2, 103017, 103015
articl	186	130	64, 102993, 52805
don	157	92	81, 53663, 102915
time	131	76	3, 102763, 103017
work	120	77	27, 101629, 53517
good	114	72	4, 101609, 102937
power	109	56	5, 103017, 102937
engin	103	58	2, 102805, 102937
drive	103	45	3, 101634, 102888
ground	100	16	22, 53747, 101609
make	95	66	6, 102888, 103060
problem	87	50	77, 53747, 101553
year	85	44	6, 103046, 102806
thing	82	64	7, 101600, 101616
design	81	40	07, 102962, 52743
oil	81	16	8, 102869, 102885
ve	74	54	88, 102937, 52786
system	73	37	3, 102937, 102962

Zipf Plot

Term: numbertoken
Rank: 1
Document Frequency: 227
Collection Frequency: 1606

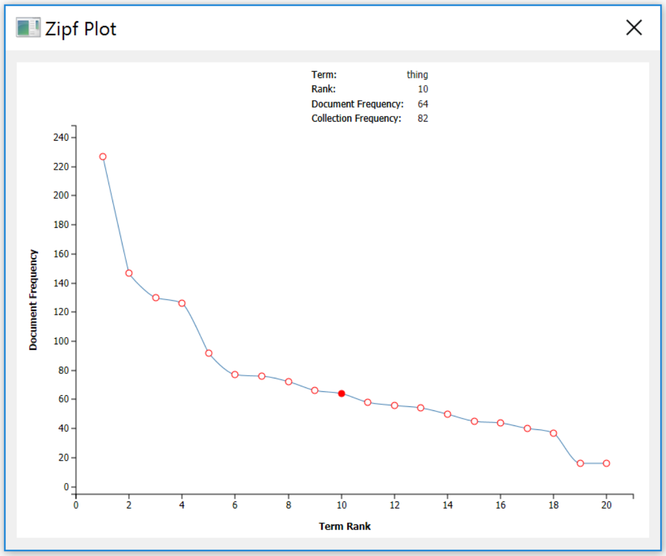
When you click on the TM_Vocabulary tab, the Zipf Plot opens. We see that our collection of documents obey the power law stated by Zipf (see above). As we move from left to right on the graph, the documents that contain the most frequently appearing terms (when ranked from most frequent to least frequent) drop quite steeply. Hover over each data point to see the detailed information about the term corresponding to this data point.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select the desired worksheet, in this case TM_Vocabulary, for *Worksheet* and the desired chart for *Chart*.



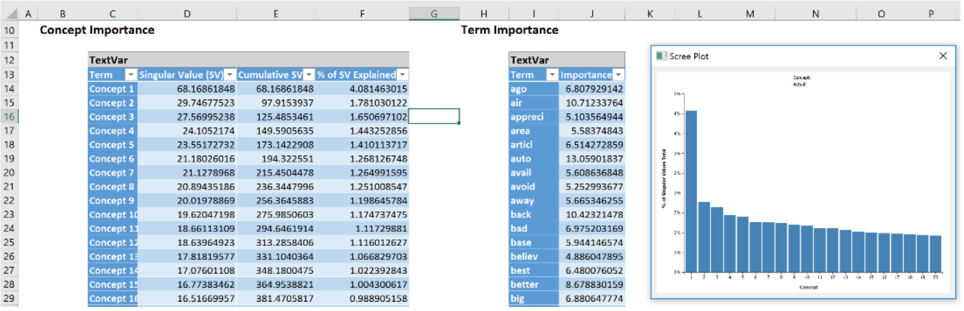
Term: numbertoken
Rank: 1
Document Frequency: 223
Collection Frequency: 1083

The term “numbertoken” is the most frequently occurring term in the document collection appearing in 223 documents (out of 300), 1,083 times total. Compare this to a less frequently occurring term such as "think" which appears in only 64 documents and only 82 times total.



Term: thing
Rank: 10
Document Frequency: 64
Collection Frequency: 82

Click the TM_LSASummary tab to view the Concept Importance and Term Importance tables. The first table, the Concept Importance table, lists each concept, it’s singular value, the cumulative singular value and the % singular value explained. (The number of concepts extracted is the minimum of the number of documents (985) and the number of terms (limited to 200).) These values are used to determine which concepts should be used in the Concept – Document Matrix, Concept – Term Matrix and the Scree Plot according to the Users selection on the Representation tab. In this example, we entered “20” for *Maximum number of concepts*.



Concept Importance

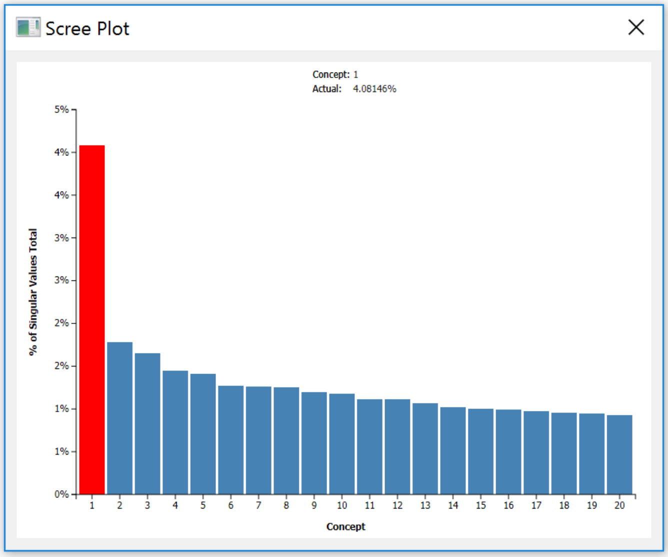
TextVar			
Term	Singular Value (SV)	Cumulative SV	% of SV Explained
Concept 1	68.16861848	68.16861848	4.081463015
Concept 2	29.74677523	97.9153937	1.781030122
Concept 3	27.56995238	125.4853461	1.650697102
Concept 4	24.1052174	149.5905635	1.443252856
Concept 5	23.55172732	173.1422908	1.410113717
Concept 6	21.18026016	194.322551	1.268126748
Concept 7	21.1278968	215.4504478	1.264991595
Concept 8	20.89435186	236.3447996	1.251008547
Concept 9	20.01978869	256.3645883	1.198645784
Concept 10	19.62047198	275.9850603	1.174737475
Concept 11	18.66113109	294.6461914	1.11729881
Concept 12	18.63964923	313.2858406	1.116012627
Concept 13	17.81819577	331.1040364	1.066829703
Concept 14	17.07601108	348.1800475	1.022392843
Concept 15	16.77383462	364.9538821	1.004300617
Concept 16	16.51669957	381.4705817	0.988905158

Term Importance

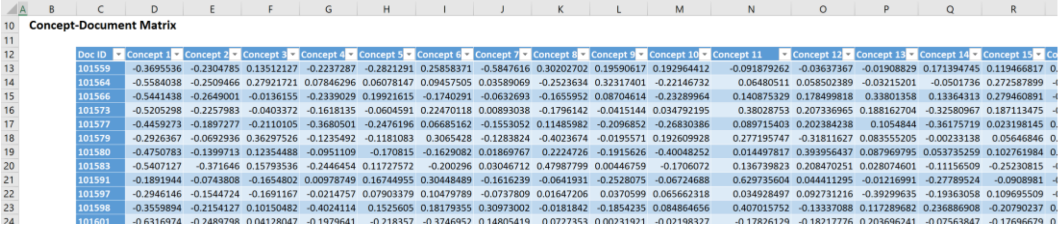
TextVar	
Term	Importance
ago	6.807929142
air	10.71233764
appreci	5.103564944
area	5.58374843
articl	6.514272859
auto	13.05901837
avail	5.608636848
avoid	5.252993677
away	5.665346255
back	10.42321478
bad	6.975203169
base	5.944146574
believ	4.886047895
best	6.480076052
better	8.678830159
big	6.880647774

The Term Importance table lists the 200 most important terms. (To increase the number of terms from 200, enter a larger value for Maximum Vocabulary on the Pre-processing tab of Text Miner.)

When you click the TM_LSASummary tab, the Scree Plot opens. This plot gives a graphical representation of the contribution or importance of each concept. The largest "drop" or "elbow" in the plot appears between the 1st and 2nd concept. This suggests that the first top concept explains the leading topic in our collection of documents. Any remaining concepts have significantly reduced importance. However, we can always select more than 1 concept to increase the accuracy of the analysis – it is advised to examine the Concept Importance table and the "Cumulative Singular Value" in particular to identify how many top concepts capture enough information for your application.



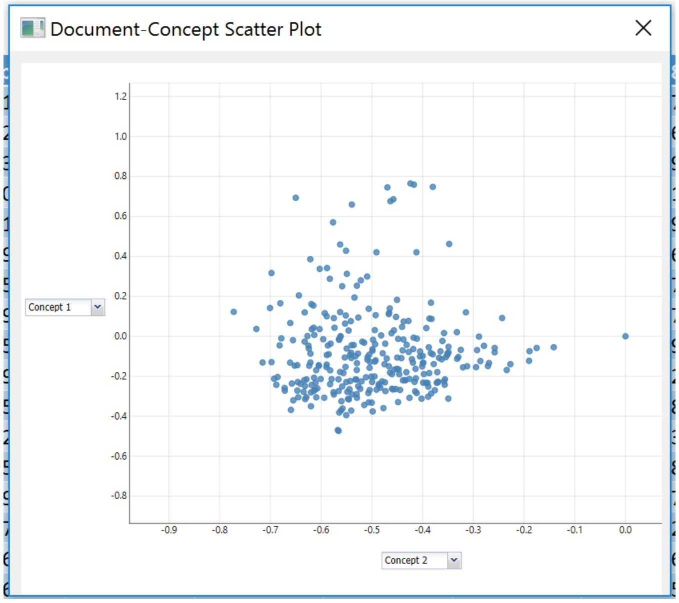
Click *TM_LSA_CDM* to display the Concept – Document Matrix. This matrix displays the top concepts (as selected on the Representation tab) along the top of the matrix and the documents down the left side of the matrix.



Concept-Document Matrix

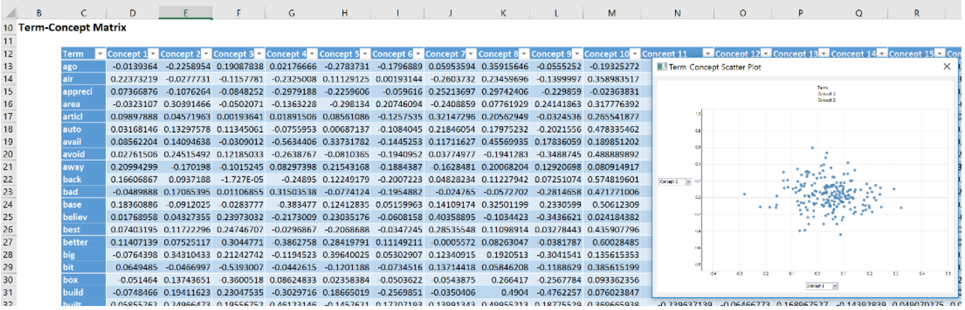
Doc ID	Concept 1	Concept 2	Concept 3	Concept 4	Concept 5	Concept 6	Concept 7	Concept 8	Concept 9	Concept 10	Concept 11	Concept 12	Concept 13	Concept 14	Concept 15
101559	-0.3695536	-0.2304785	0.13512127	-0.2237287	-0.2821291	0.25858371	-0.5847616	0.30202702	0.19590617	0.192964412	-0.091879262	-0.03637367	-0.01908829	0.171394745	0.119466817
101564	-0.5584038	-0.2509466	0.27921721	0.07846296	0.06078147	0.09457505	0.03589069	-0.2523634	0.32317401	-0.22146732	0.06480511	0.058502389	-0.03215201	-0.0501736	0.272587899
101566	-0.5441438	-0.2649001	-0.0136155	-0.2339029	0.19921615	-0.1740291	-0.0632693	-0.1655952	0.08704614	-0.23289964	0.140875329	0.178499818	0.33801358	0.13364313	0.279460891
101573	-0.5205298	-0.2257983	-0.0403372	-0.1618135	-0.0604591	0.22470118	0.00893038	-0.1796142	-0.0415144	0.034792195	0.38028753	0.207336965	0.188162704	-0.32580967	0.187113475
101577	-0.4459273	-0.1897277	-0.2110105	-0.3680501	-0.2476196	0.06685162	-0.1553052	0.11485982	-0.2096852	-0.26830386	0.089715403	0.202384238	0.1054844	-0.36175719	0.023198145
101579	-0.2926367	-0.0692936	0.36297526	-0.1235492	-0.1181083	0.3065428	-0.1283824	-0.4023674	-0.0195571	0.192609928	0.277195747	-0.31811627	0.083555205	-0.00233138	0.05646846
101580	-0.4750783	-0.1399713	0.12354488	-0.0951109	-0.170815	-0.1629082	0.01869767	0.2224726	-0.1915626	-0.40048252	0.014497817	0.393956437	0.087969795	0.053735259	0.102761984
101583	-0.5407127	-0.371646	0.15793536	-0.2446454	0.11727572	-0.200296	0.03046712	0.47987799	0.00446759	-0.1706072	0.136739823	0.208470251	0.028074601	-0.11156509	-0.25230815
101591	-0.1891944	-0.0743808	-0.1654802	0.00978749	0.16744955	0.30448489	-0.1616239	-0.0641931	-0.2528075	-0.06724688	0.629735604	0.044411295	-0.01216991	-0.27789524	-0.0908981
101597	-0.2946146	-0.1544724	-0.1691167	-0.0214757	0.07903379	0.10479789	-0.0737809	0.01647206	0.0370599	0.065662318	0.034928497	0.092731216	-0.39299635	-0.19363058	0.109695509
101598	-0.3559894	-0.2154127	0.10150482	-0.4024114	0.1525605	0.18179355	0.30973002	-0.0181842	-0.1854235	0.084864656	0.407015752	-0.13337088	0.117289682	0.236886908	-0.20790237
101601	-0.6316974	-0.2489798	0.04128047	-0.1979641	-0.218357	-0.3746952	0.14805419	0.0727353	0.00231921	-0.02198327	-0.17826129	-0.18217776	0.203696241	-0.07563847	-0.17696679

When you click on the TM_LSA_CDM tab, the Concept-Document Scatter Plot opens. This graph is a visual representation of the Concept – Document matrix. Note that Analytic Solver Data Science normalizes each document representation so it lies on a unit hypersphere. Documents that appear in the middle of the plot, with concept coordinates near 0 are not explained well by either of the shown concepts. The further the magnitude of coordinate from zero, the more effect that particular concept has for the corresponding document. In fact, two documents placed at extremes of a concept (one close to -1 and other to +1) indicates strong differentiation between these documents in terms of the extracted concept. This provides means for understanding actual meaning of the concept and investigating which concepts have the largest discriminative power, when used to represent the documents from the text collection.



You can examine all extracted concepts by changing the axes on a scatter plot - click the down pointing arrow next to Concept 1 or the concept on the Y axis by clicking the right pointing arrow next to Concept 2. Use your touchscreen or your mouse scroll wheel to zoom in and out.

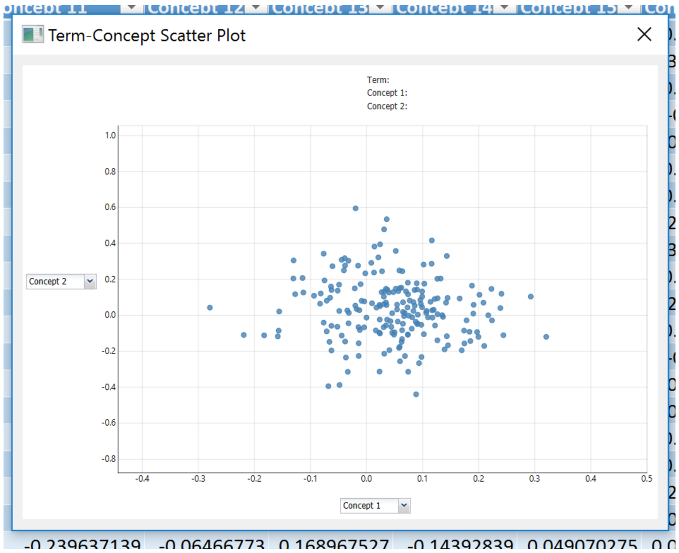
Double click *TM_LSA_CTM* to display the Concept – Term Matrix which lists the top 20 most important concepts along the top of the matrix and the top 200 most frequently appearing terms down the side of the matrix.



Term-Concept Matrix

Term	Concept 1	Concept 2	Concept 3	Concept 4	Concept 5	Concept 6	Concept 7	Concept 8	Concept 9	Concept 10	Concept 11	Concept 12	Concept 13	Concept 14	Concept 15
ago	-0.0139364	-0.2258954	0.19087838	0.02176666	-0.2783731	-0.1796889	0.05953594	0.35915646	-0.0555252	-0.19325272					
air	0.22373219	-0.0277731	-0.1157781	-0.2325008	0.11129125	0.00193144	-0.2603732	0.23459696	-0.1399997	0.358983517					
appreci	0.07366876	-0.1076264	-0.0848252	-0.2979188	-0.2259606	-0.059616	0.25213697	0.29742406	-0.229859	-0.02363831					
area	-0.0323107	0.30391466	-0.0502071	-0.1363228	-0.298134	0.20746094	-0.2408859	0.07761929	0.24141863	0.317776392					
articl	0.09897888	0.04571963	0.00193641	0.01891506	0.08561086	-0.1257535	0.32147296	0.20562949	-0.0324536	0.265541877					
auto	0.03168146	0.13297578	0.11345061	-0.0755953	0.00687137	-0.1084045	0.21846054	0.17975232	-0.2021556	0.478335462					
avail	0.08562204	0.14094638	-0.0309012	-0.5634406	0.33731782	-0.1445253	0.11711627	0.45569935	0.17836059	0.189851202					
avoid	0.02761506	0.24515492	0.12185033	-0.2638767	-0.0810365	-0.1940952	0.03774977	-0.1941283	-0.3488745	0.488889892					
away	0.20994299	-0.170198	-0.1015245	0.08297398	0.21543168	-0.1884387	-0.1628481	0.20068204	0.12920698	0.080914917					
back	0.16606867	0.0937188	-1.727E-05	-0.24895	0.12249179	-0.2007223	0.04828234	0.11227942	0.07251074	0.574819601					
bad	-0.0489888	0.17065395	0.01106855	0.31503538	-0.0774124	-0.1954882	-0.024765	-0.0572702	-0.2814658	0.471771006					
base	0.18360886	-0.0912025	-0.0283777	-0.383477	0.12412835	0.05159963	0.14109174	0.32501199	0.2330599	0.50612309					
believ	0.01768958	0.04327355	0.23973032	-0.2173009	0.23035176	-0.0608158	0.40358895	-0.1034423	-0.3436621	0.024184382					
best	0.07403195	0.11722296	0.24746707	-0.0296867	-0.2068688	-0.0347245	0.28535548	0.11098914	0.03278443	0.435907796					
better	0.11407139	0.07525117	0.3044771	-0.3862758	0.28419791	0.11149211	-0.0005572	0.08263047	-0.0381787	0.60028485					
big	-0.0764398	0.34310433	0.21242742	-0.1194523	0.39640025	0.05302907	0.12340915	0.1920513	-0.3041541	0.135615353					
bit	0.0649485	-0.0466997	-0.5393007	-0.0442615	-0.1201188	-0.0734516	0.13714418	0.05846208	-0.1188629	0.385615199					
box	-0.051464	0.13743651	-0.3600518	0.08624833	0.02358384	-0.0503622	-0.0543875	0.266417	-0.2567784	0.093362356					
build	-0.0748466	0.19411623	0.23047535	-0.3029716	0.18665019	-0.2569851	-0.0350406	0.4904	-0.4762257	0.076023847					
built	0.05855762	0.24966473	0.19556752	0.46123146	-0.1457621	0.17207193	0.13991343	0.49955213	0.18775529	0.369665938	-0.239637139	-0.06466773	0.168967527	-0.14392839	0.049070275

When you click on the TM_LSA-CTM tab, the Term-Concept Scatter Plot opens. This graph is a visual representation of the Concept – Term Matrix. It displays all terms from the final vocabulary in terms of two concepts. Similarly to the Concept-Document scatter plot, the Concept-Term scatter plot visualizes the distribution of vocabulary terms in the semantic space of meaning extracted with LSA. The coordinates are also normalized, so the range of axes is always [-1,1], where extreme values (close to +/-1) highlight the importance or “load” of each term to a particular concept. The terms appearing in a zero-neighborhood of concept range do not contribute much to a concept definition. In our example, if we identify a concept having a set of terms that can be divided into two groups: one related to “Autos” and other to “Electronics”, and these groups are distant from each other on the axis corresponding to this concept, this would definitely provide an evidence that this particular concept “caught” some pattern in the text collection that is capable of discriminating the topic of article. Therefore, Term-Concept scatter plot is an extremely valuable tool for examining and understanding the main topics in the collection of documents, finding similar words that indicate similar concept, or the terms explaining the concept from “opposite sides” (e.g. *term1* can be related to cheap affordable electronics and *term2* can be related to expensive luxury electronics)



Recall that if you want to examine different pair of concepts, click the down pointing arrow next to Concept 1 and the right pointing arrow next to Concept 2 to change the concepts on either axis. Use your touchscreen or mouse wheel to scroll in or out.

The *TFIDF_Stored and LSA_Stored* output sheets are used to process new documents using an existing text mining model. See the section below, Processing New Documents Based on an Existing Text Mining Model, to find out how to score new text documents using an existing Text Mining model. Note: When adding additional documents to an existing text mining model, Analytic Solver Data Science will not extract new terms or phrases from these new documents. Rather, Analytic Solver Data Science will first use the vocabulary from the model to build a Term-Document Matrix and then, if requested, will use transformation matrices to map documents in the new data onto the existing semantic space extracted from the “base” model. Please see below for an example explaining how to add additional documents to an existing Text Mining model.

From here, we can use any of the six classification algorithms to classify our documents according to some term or concept using the Term – Document matrix, Concept – Document matrix or Concept – Term matrix where each document becomes a "record" and each concept becomes a "variable". If wanting to classify documents based on a binary variable such as Auto email/non-Auto email, then we would use either the Term – Document or Concept – Document matrix. If wanting to cluster terms or classify terms, then we would use the Term-Concept matrix. We could even use the transpose of the Term – Document matrix where each term would become a "record" and each column would become a "feature". See the Analytic Solver Data Science User Guide for an example model that uses the Logistic Regression Classification method to create a classification model using the Concept Document matrix within *TM_LSA_CDM*.

This concludes our example on how to use Analytic Solver Data Science's new Text Miner feature. This example has illustrated how Analytic Solver Data Science provides powerful tools for importing a collection of documents for comprehensive text preprocessing, quantitation, and concept extraction, in order to create a model that can be used to process new documents - all performed without any manual intervention. When using Text Miner in conjunction with our classification algorithms, Analytic Solver Data Science can be used to classify customer reviews as satisfied/not satisfied, distinguish between which products garnered the least negative reviews, extract the topics of articles, cluster the documents/terms, etc. The applications for Text Miner are boundless.

Processing New Documents Based on an Existing Text Mining Model

In Analytic Solver Data Science, it's possible to process new text documents based on the existing Text Mining model(s) if the option Write Text Mining Model is selected on the Output Options tab. Two models (TFIDF_Stored, LSA_Stored) are created: for Term Frequency – Inverse Document Frequency Vectorization (TF-IDF) and, if Concept Extraction was performed, for Latent Semantic Analysis (LSA).

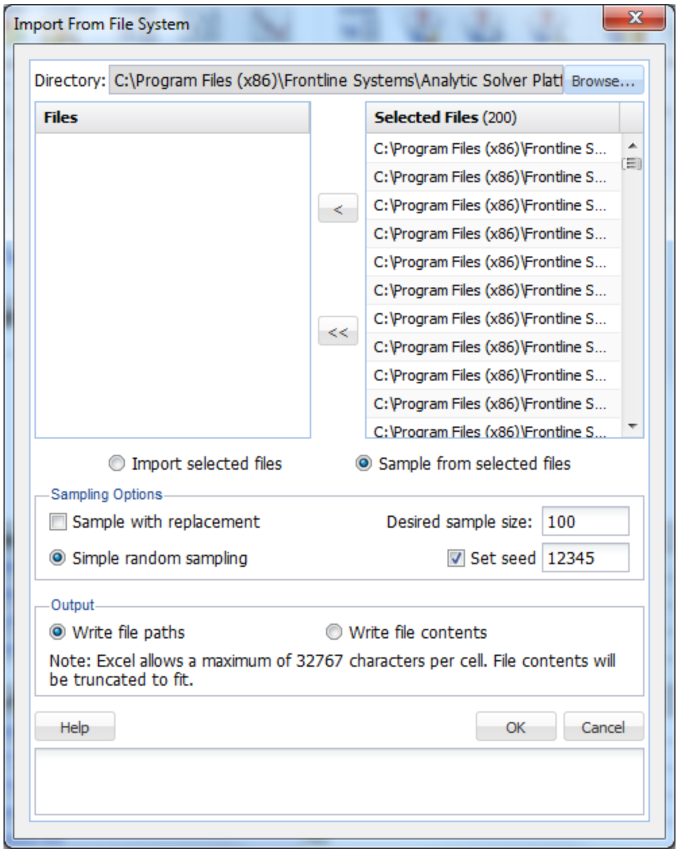
The TF-IDF model contains the information needed for processing the new text documents based on the vocabulary inferred from the training corpus. All preprocessing settings and stages will be applied to the text in new documents to ensure the proper mapping to the baseline vocabulary. The LSA model contains the information needed for "mapping" the Term-Document Matrices (TDM) representing the vectorized collection of new documents onto existing latent semantic space defined by the training documents. The below example illustrates how to vectorize the set of new documents and extract the concepts from them using the text models created in the previous section. Two hundred additional text documents (100 each for electronics and autos) have been extracted from the same Newsgroups dataset as used in the example above (complete dataset downloadable from http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/news20.html)

Note: The scoring of new data based on the TF-IDF model produces a Term-Document Matrix given a collection of new documents. The scoring based on the LSA model produces a Concept-Document Matrix (CDM) given a Term-Document Matrix.

Click **Help – Example Models** on the Data Science ribbon, then select **Examples** from the menu to open the Text Mining Example Documents.zip archive and extract these files to a desired location. See the section *Importing* *from a File Folder* within the **Sampling or Importing from a Database, Worksheet or File Folder** chapter for directions on extracting and importing the files into Analytic Solver Data Science.

Click **Sample – Import from File Folder** to open the *Import From File System* dialog. Click **Browse** and navigate to the location of the additional electronics text files. Set file type to **All Files**, (lower, right corner of Browse dialog) then select all 100 files in the folder. Click the >> button on the *Import From File System* dialog to move all files to *Selected Files.* Repeat these steps to load the additional auto documents in the *Additional autos* folder. You should now have 200 documents listed under *Selected Files*.

Select **Sample from selected files**, then enter **100** for *Desired Sample Size*. Keep *Write file paths* selected for *Output*, then click OK. Recall that when *Write file paths* is selected, pointers to the file locations are stored in *FileSampling*. If *Write file* contents is selected, the content of each text document will be written to a cell in *FileSampling*, up to a maximum of 32,767 characters.



Import From File System

Directory: C:\Program Files (x86)\Frontline Systems\Analytic Solver Platf

Files

Selected Files (200)

C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...
C:\Program Files (x86)\Frontline S...

○ Import selected files ● Sample from selected files

Sampling Options

☐ Sample with replacement Desired sample size: 100

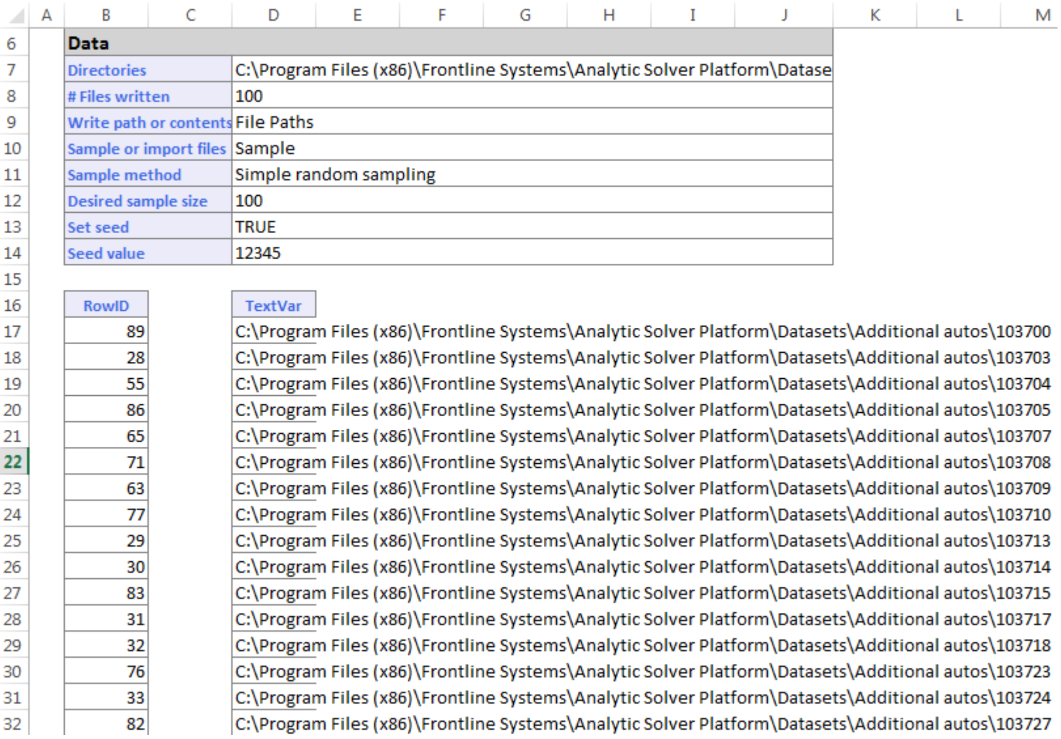
● Simple random sampling ☑ Set seed 12345

Output

● Write file paths ○ Write file contents

Note: Excel allows a maximum of 32767 characters per cell. File contents will be truncated to fit.

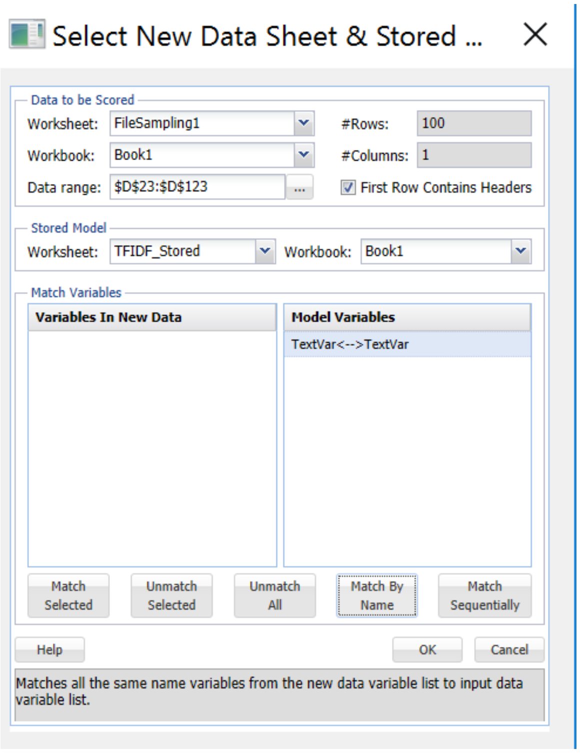
Click **OK**. FileSampling1 is inserted into the Solver Task Pane. We will again sort the documents by type (electronic or auto) by using Microsoft Excel's Sort functionality (on the Data menu).



Data	
Directories	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datase
# Files written	100
Write path or contents	File Paths
Sample or import files	Sample
Sample method	Simple random sampling
Desired sample size	100
Set seed	TRUE
Seed value	12345

RowID	TextVar
89	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103700
28	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103703
55	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103704
86	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103705
65	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103707
71	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103708
63	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103709
77	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103710
29	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103713
30	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103714
83	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103715
31	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103717
32	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103718
76	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103723
33	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103724
82	C:\Program Files (x86)\Frontline Systems\Analytic Solver Platform\Datasets\Additional autos\103727

Click **Score** on the Data Science ribbon to bring up the Select New Data Sheet & Stored Model Sheet dialog. Select Match By Name to match TextVar under Variables In New Data with TextVar under Model Variables.



Select New Data Sheet & Stored ...

Data to be Scored

Worksheet: FileSampling1 #Rows: 100

Workbook: Book1 #Columns: 1

Data range: $D$23:$D$123 ☑ First Row Contains Headers

Stored Model

Worksheet: TFIDF_Stored Workbook: Book1

Match Variables

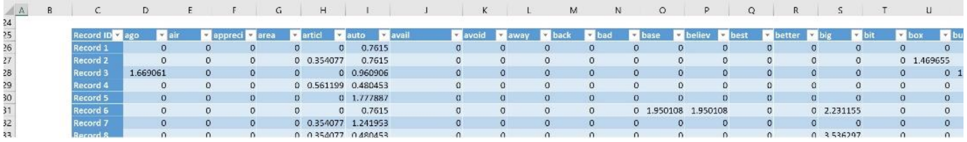
Variables In New Data	Model Variables
	TextVar<-->TextVar

Matches all the same name variables from the new data variable list to input data variable list.

Notice that FileSampling1 has been selected for Worksheet under Data to be Scored and TFIDF_Stored has been selected for Worksheet under Stored Model Model. This example will vectorize new data according to the TF-IDF model (i.e. product TDM – term-document matrix).

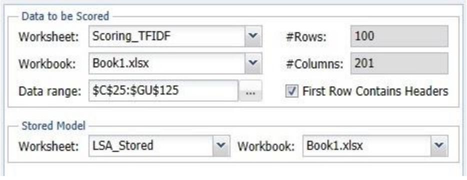
Alternatively, variables could be mapped by selecting *TextVar* under both *Selected Text Variables* and *Model Text Variables*, then click *Match Selected*. If *Match Sequentially* is used, Analytic Solver Data Science will match variables in the order that they appear. To unmatch a single pair of variables, highlight the desired variables in the *Model Text Variables* list box and select *Unmatch Selected*. To unmatch all variables, click *Unmatch All*.

Click OK to score the new documents using the existing model created in the above example.



Record ID	ago	air	appreci	area	articl	auto	avail	avoid	away	back	bad	base	believ	best	better	big	bit	box	bu
Record 1	0	0	0	0	0	0.7615	0	0	0	0	0	0	0	0	0	0	0	0	
Record 2	0	0	0	0	0.354077	0.7615	0	0	0	0	0	0	0	0	0	0	0	1.469655	
Record 3	1.669061	0	0	0	0	0.960906	0	0	0	0	0	0	0	0	0	0	0	0	1
Record 4	0	0	0	0	0.561199	0.480453	0	0	0	0	0	0	0	0	0	0	0	0	
Record 5	0	0	0	0	0	1.777887	0	0	0	0	0	0	0	0	0	0	0	0	
Record 6	0	0	0	0	0	0.7615	0	0	0	0	0	1.950108	1.950108	0	0	2.231155	0	0	
Record 7	0	0	0	0	0.354077	1.241953	0	0	0	0	0	0	0	0	0	0	0	0	
Record 8	0	0	0	0	0.354077	0.480453	0	0	0	0	0	0	0	0	0	3.536297	0	0	

To extract concepts for new data based on the LSA model (i.e. product CDM - Concept-Document matrix), we will score the term-document matrix. Click Score on the Data Science ribbon to bring up the Select New Data Sheet & Stored Model Sheet dialog.



Data to be Scored

Worksheet: Scoring_TFIDF

Workbook: Book1.xlsx

Data range: $C$25:$GU$125

#Rows: 100

#Columns: 201

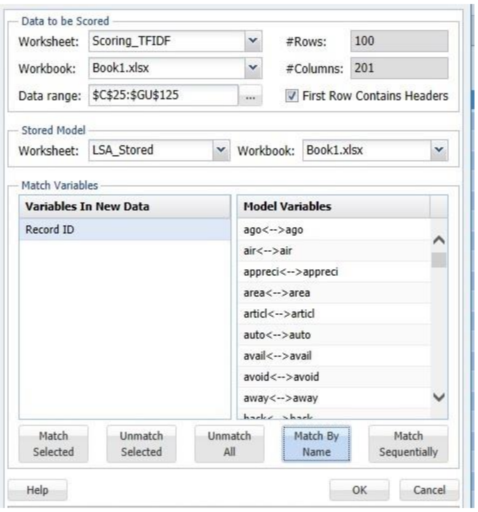
☑ First Row Contains Headers

Stored Model

Worksheet: LSA_Stored

Workbook: Book1.xlsx

Select LSA_Stored for Worksheet under Stored Model. Select Match By Name to match the terms from the Stored Model sheet (LSA_Stored) with the terms from the term document matrix.



Data to be Scored

Worksheet: Scoring_TFIDF

Workbook: Book1.xlsx

Data range: $C$25:$GU$125

#Rows: 100

#Columns: 201

☑ First Row Contains Headers

Stored Model

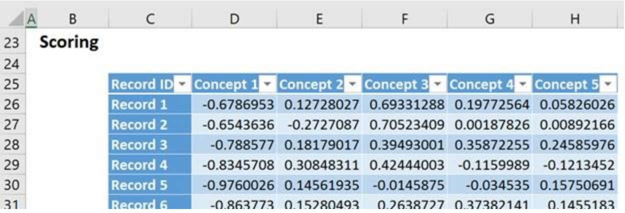
Worksheet: LSA_Stored

Workbook: Book1.xlsx

Match Variables

Variables In New Data	Model Variables
Record ID	ago<-->ago
	air<-->air
	appreci<-->appreci
	area<-->area
	articl<-->articl
	auto<-->auto
	avail<-->avail
	avoid<-->avoid
	away<-->away
	back<-->back

Click OK to score the term document matrix. The output is the Concept-Document matrix.

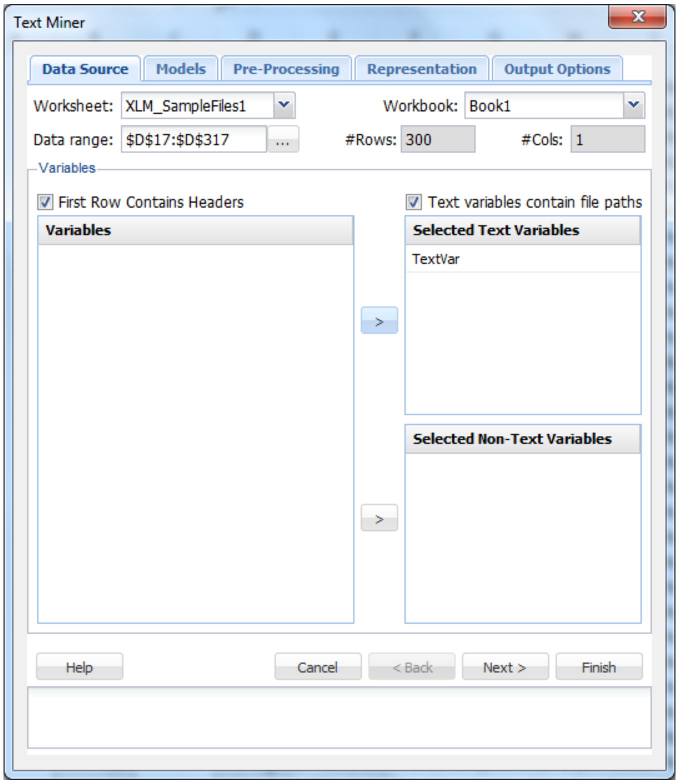


Scoring

Record ID	Concept 1	Concept 2	Concept 3	Concept 4	Concept 5
Record 1	-0.6786953	0.12728027	0.69331288	0.19772564	0.05826026
Record 2	-0.6543636	-0.2727087	0.70523409	0.00187826	0.00892166
Record 3	-0.788577	0.18179017	0.39493001	0.35872255	0.24585976
Record 4	-0.8345708	0.30848311	0.42444003	-0.1159989	-0.1213452
Record 5	-0.9760026	0.14561935	-0.0145875	-0.034535	0.15750691
Record 6	-0.863773	0.15280493	0.2638727	0.37382141	0.1455183

Text Mining Options

The following options appear on each of the five different Text Miner dialogs: Data Source, Models, Pre-Processing, Representation, and Output Options.



Worksheet: XLM_SampleFiles1

Workbook: Book1

Data range: $D$17:$D$317

#Rows: 300

#Cols: 1

☑ First Row Contains Headers

☑ Text variables contain file paths

TextVar

Variables

Variables contained in this listbox are text variables included within a dataset with at least one column that contains free-form text (or file paths to documents containing free-form text), and optionally other columns that contain traditional structured data.

First Row Contains Headers

Select this option if the first row of your dataset contains headers for your data. This option is selected by default.

Selected Text Variables

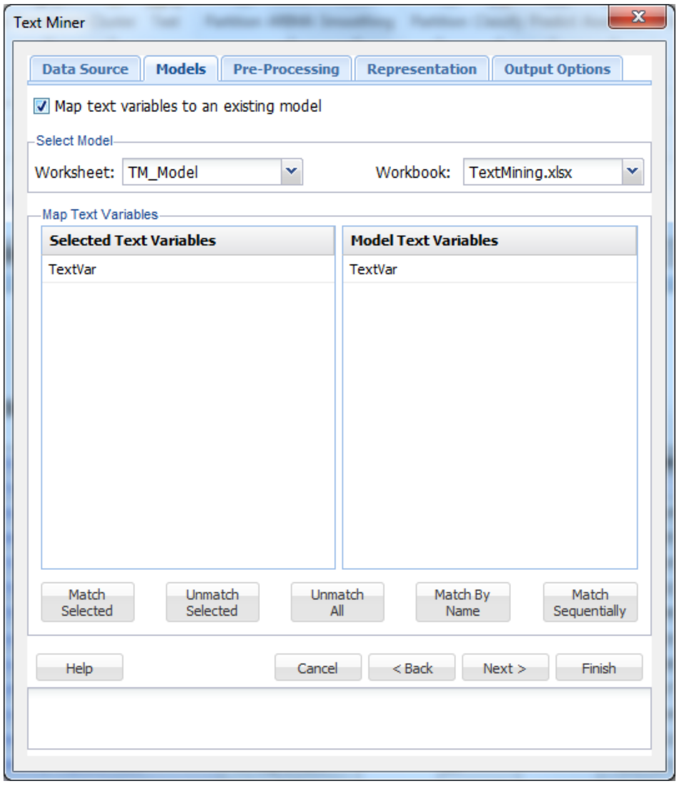
Variables contained in this listbox have been selected from the Variables listbox as inputs to Text Miner.

Text variables contain file paths

Select this option if the text variables within your dataset contain “pointers” or paths to a text document or collection of text documents. If your dataset contains “points”, this option will be selected by default.

Selected Non-Text Variables

Variables contained in this listbox have been selected from the Variables listbox as non-text inputs to Text Miner, i.e.numeric variables.



☑ Map text variables to an existing model

Select Model

Worksheet: TM_Model

Workbook: TextMining.xlsx

Map Text Variables

Selected Text Variables	Model Text Variables
TextVar	TextVar

Map text variables to an existing model

Select this option if processing new text documents based on an existing text model. Once this option is selected, options on the model dialog will be enabled, and options on both the Pre-Processing and Representation tabs will be disabled. Options used and defined in the existing “base” Text Miner model (created in the previous section) will be prefilled. Vocabulary for the new collection is defined in the existing base model and will be used to map the new documents to the existing space of terms. Some preprocessing options that affect vocabulary reduction and term normalization, are not applicable in this mode and are not prefilled. To change any of these options, you must create a new “baseline” Text Miner model.

Select Model Worksheet

The *TM_Model* output sheet should be automatically selected. If there are multiple *TM_Model* output sheets within the same Workbook, click the down arrow and select the desired one.

Select Model Workbook

The current workbook will be automatically prefilled. If multiple workbooks are opened, click the down arrow to select the workbook containing the desired one.

Selected Text Variables

Variables contained in this listbox are text variables included within a dataset, with at least one column that contains free-form text (or file paths to documents containing free-form text), and optionally other columns that contain traditional structured data.

Model Text Variables

Text variables included in this listbox are existing text variables already included in the existing or “base” Text Miner model.

Match Selected

Select one text variable from the Selected Text Variables and Model Text Variables listbox, then select Match Selected to manually map variables from the dataset to the existing model. The match will appear under Model Text Variables.

Unmatch Selected

Select a set of matched variables under Model Text Variables and click Unmatch Selected to unmatch the pair.

Unmatch All

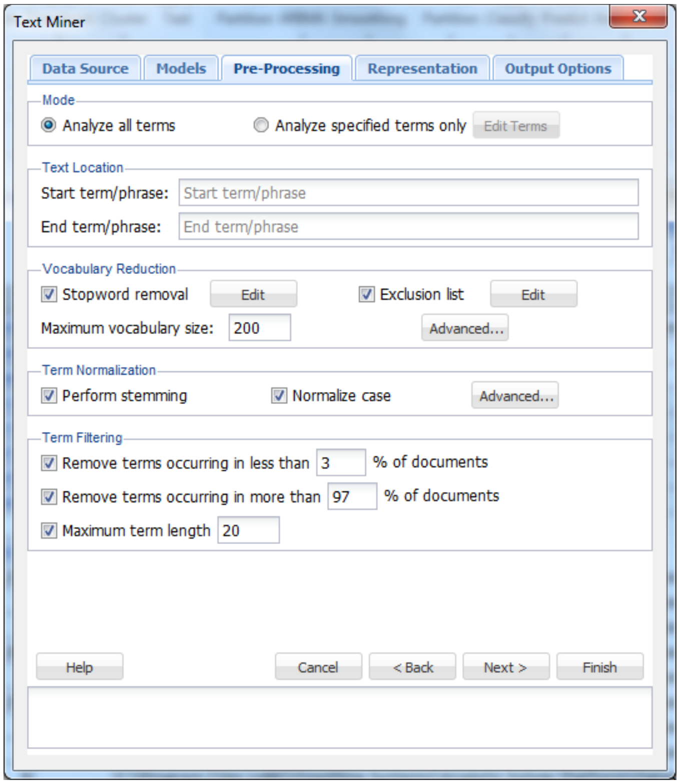
Click Unmatch All, to unmatch all previously matched variables under Model Text Variables.

Match By Name

Click Match By Name to match all variables in the Selected Text Variables listbox with variables of the same name in the Model Text Variables listbox.

Match Sequentially

Click Match Sequentially to match all variables, in order as listed, in the Selected Text Variables listbox with variables, in order as listed, in the Model Text Variables listbox.



Mode
● Analyze all terms ○ Analyze specified terms only

Text Location
Start term/phrase: Start term/phrase
End term/phrase: End term/phrase

Vocabulary Reduction
☑ Stopword removal ☑ Exclusion list
Maximum vocabulary size: 200

Term Normalization
☑ Perform stemming ☑ Normalize case

Term Filtering
☑ Remove terms occurring in less than 3 % of documents
☑ Remove terms occurring in more than 97 % of documents
☑ Maximum term length 20

Analyze All Terms

When this option is selected, Analytic Solver Data Science will examine all terms in the document. A “term” is defined as an individual entity in the text, which may or may not be an English word. A term can be a word, number, email, url, etc. This option is selected by default.

Analyze specified terms only

When this option is selected, Analytic Solver Data Science will examine all terms in the document. A “term” is defined as an individual entity in the text, which may or may not be an English word. A term can be a word, number, email, url, etc. If *Analyze specified terms only* is selected, the *Edit Terms* button will be enabled. If you click this button, the *Edit Exclusive Terms* dialog opens. Here you can add and remove terms to be considered for text mining. All other terms will be disregarded. For example, if we wanted to mine each document for a specific part name such as “alternator” we would click *Add Term* on *the Edit Exclusive Terms* dialog, then replace “New term” with “alternator” and click *Done* to return to the Pre-Processing dialog. During the text mining process, Analytic Solver Data Science would analyze each document for the term “alternator”, excluding all other terms.

Start term/phrase

If this option is used, text appearing before the first occurrence of the Start Phrase will be disregarded and similarly, text appearing after End Phrase (if used) will be disregarded. For example, if text mining the transcripts from a Live Chat service, you would not be particularly interested in any text appearing before the heading “Chat Transcript” or after the heading “End of Chat Transcript”. Thus you would enter “Chat Transcript” into the Start Phrase field and “End of Chat Transcript” into the End Phrase field.

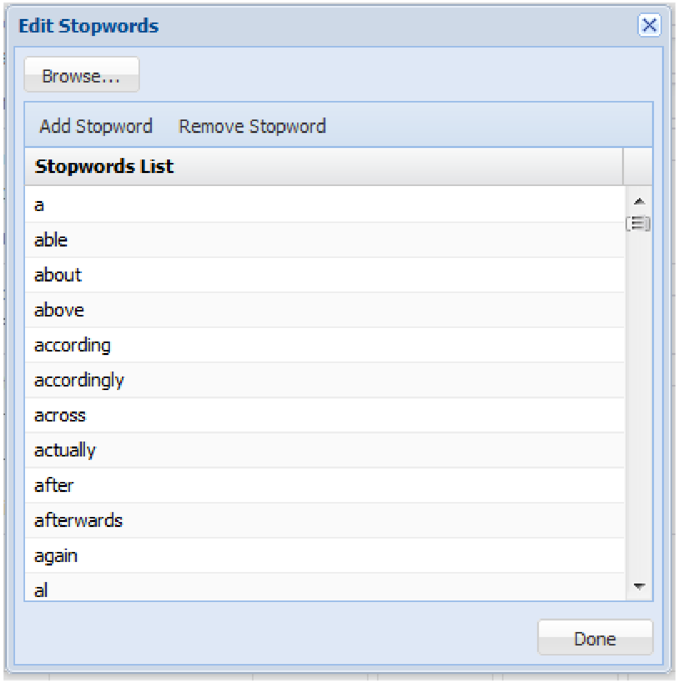
End term/phrase

If this option is selected, text appearing before the first occurrence of the Start Phrase will be disregarded and similarly, text appearing after End Phrase (if used) will be disregarded. For example, if text mining the transcripts from a Live Chat service, you would not be particularly interested in any text appearing before the heading “Chat Transcript” or after the heading “End of Chat Transcript”. Thus you would enter “Chat Transcript” into the Start Phrase field and “End of Chat Transcript” into the End Phrase field.

Stopword removal

If selected (the default), over 300 commonly used words/terms (such as a, to, the, and, etc.) will be removed from the document collection during preprocessing. Click the Edit command button to view the list of terms. To remove a word from the Stopword list, simply highlight the desired word, then click Remove Stopword. To add a new word to the list, click Add Stopword, a new term “stopword” will be added. Double click to edit.

Analytic Solver Data Science also allows additional stopwords to be added or existing to be removed via a text document (*.txt) by using the Browse button to navigate to the file. Terms in the text document can be separated by a space, a comma, or both. If we were supplying our three terms in a text document, rather than in the *Edit Stopwords* dialog, the terms could be listed as: subject emailterm from or subject,emailterm,from or subject, emailterm, from. If we had a large list of additional stopwords, this would be the preferred way to enter the terms.



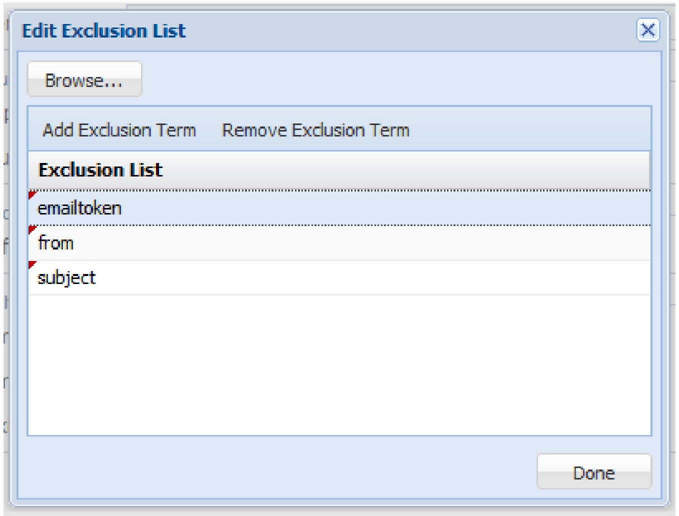
Stopwords List
a
able
about
above
according
accordingly
across
actually
after
afterwards
again
al

Click done to close the Edit Stopwords dialog and return to the Pre-Processing tab.

Exclusion list

If selected, terms entered into the Exclusion list will be removed from the document collection. This is beneficial if all or a large number of documents in the collection contain the same terms, for example, “from”, “to”, “subject” in a collection of emails. If all documents contain the same terms, including them in the analysis will not provide any benefit and could bias the analysis. Click **Edit** to enter the terms to be excluded. The *Edit Exclusion List* dialog opens. Click Add Exclusion Term. The label “exclusionterm” is added. Click to edit and enter the desired term. Analytic Solver Data Science will remove the terms listed in this dialog from the document collection during pre-processing. To remove a term from the exclusion list, highlight the term and click *Remove Exclusion Term*.

We could have also entered these terms into a text document (*.txt) and added the terms all at once by using the Browse button to navigate to the file and import the list. Terms in the text document can be separated by a space, a comma, or both. If, for example we were supplying excluded terms in a document rather than in the Edit Exclusion List dialog, we would enter the terms as: subject emailtoken from, or subject,emailtoken,from, or subject, emailtoken, from. If we had a large list of terms to be excluded, this would be the preferred way to enter the terms.



Exclusion List
emailtoken
from
subject

Click Done to close the dialog and return to Pre-Processing.

Vocabulary Reduction Advanced…

Analytic Solver Data Science also allows the combining of synonyms and full phrases by clicking Advanced within Vocabulary Reduction.

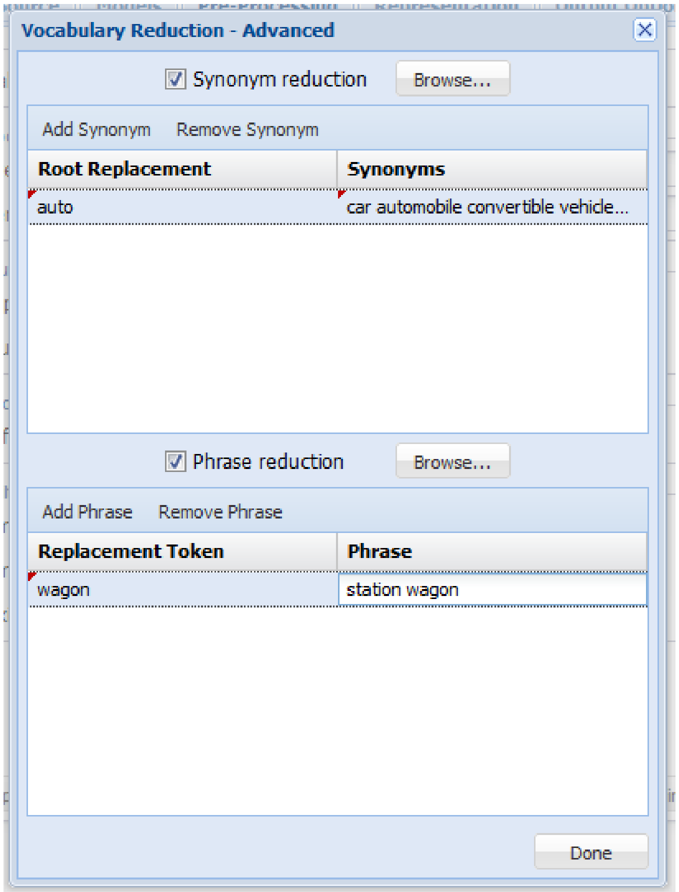
Synonym Reduction

Select Synonym reduction at the top of the dialog to replace synonyms such as “car”, “automobile”, “convertible”, “vehicle”, “sedan”, “coupe”, “subcompact”, and “jeep” with “auto”. Click *Add Synonym* and replace “rootterm” with the term to be substituted, then replace “synonym list” with the list of synonyms, i.e. :car, automobile, convertible, vehicle, sedan, coupe. During pre-processing, Analytic Solver Data Science will replace the terms “car”, “automobile”, “convertible”, “vehicle”, “sedan”, “coupe”, “subcompact” and “jeep” with the term “auto”. To remove a synonym from the list, highlight the term and click *Remove Synonym.*

If adding synonyms from a text file, each line must be of the form rootterm:synonymlist or using our example: auto:car automobile convertible vehicle sedan coup or auto:car,automobile,convertible,vehicle,sedan,coup. Note separation between the terms in the synonym list be either a space, a comma or both.   If we had a large list of synonyms, this would be the preferred way to enter the terms.

Phrase Reduction

Analytic Solver Data Science also allows the combining of words into phrases that indicate a singular meaning such as “station wagon” which refers to a specific type of car rather than two distinct tokens – station and wagon. To add a phrase in the *Vocabulary Reduction – Advanced* dialog, select *Phrase reduction* and click *Add Phrase*. The term “phrasetoken” will be appear. Click to edit and enter the term that will replace the phrase. i.e. wagon. Click “phrase” to edit and enter the phrase that will be substituted, i.e. “station wagon”. If supplying phrases through a text file (*.txt), each line of the file must be of the form phrasetoken:phrase or using our example, wagon:station wagon. If we had a large list of phrases, this would be the preferred way to enter the terms.



Vocabulary Reduction - Advanced

☑ Synonym reduction

Root Replacement	Synonyms
auto	car automobile convertible vehicle...

☑ Phrase reduction

Replacement Token	Phrase
wagon	station wagon

Click Done to return to the Pre-processing tab.

Maximum vocabulary size

Analytic Solver Data Science will reduce the number of terms in the final vocabulary to the most frequently occurring in the collection. The default is “1000”.

Perform stemming

Stemming is the practice of stripping words down to their “stems” or “roots”, for example, stemming terms such as “argue”, “argued”, “argues”, “arguing”, and “argus” would result in the stem “argu. However “argument” and “arguments” would stem to “argument”. The stemming algorithm utilized in Analytic Solver Data Science is “smart” in the sense that while “running” would be stemmed to “run”, “runner” would not. . Analytic Solver Data Science uses the Porter Stemmer 2 algorithm for the English Language. For more information on this algorithm, please see the Webpage: http://tartarus.org/martin/PorterStemmer/

Normalize case

When this option is checked, Analytic Solver Data Science converts all text to a consistent (lower) case, so that Term, term, TERM, etc. are all normalized to a single token “term” before any processing, rather than creating three independent tokens with different case. This simple method can dramatically affect the frequency distributions of the corpus, leading to biased results.

Term Normalization Advanced…

Click **Advanced** in the *Term Normalization* group to open the *Term Normalization – Advanced* dialog. This dialog allows us to replace or remove nonsensical terms such as HTML tags, URLs, Email addresses, etc. from the document collection. It’s possible to remove normalized terms completely by including the normalized term (for example, “emailtoken”) in the Exclusion list.

Minimum stemmed term length

If stemming reduced a term’s length to 2 or less characters, Text Miner will disregard the term. This option is selected by default.

Remove HTML tags

If selected, HTML tags will be removed from the document collection. HTML tags and text contained inside these tags contain technical, computer-generated information that is not typically relevant to the goal of the text mining application. This option is not selected by default.

Normalize URL’s

If selected, URLs appearing in the document collection will be replaced with the term, “urltoken”. URLs do not normally add any meaning, but it is sometimes interesting to know how many URLs are included in a document. This option is not selected by default.

Normalize email addresses

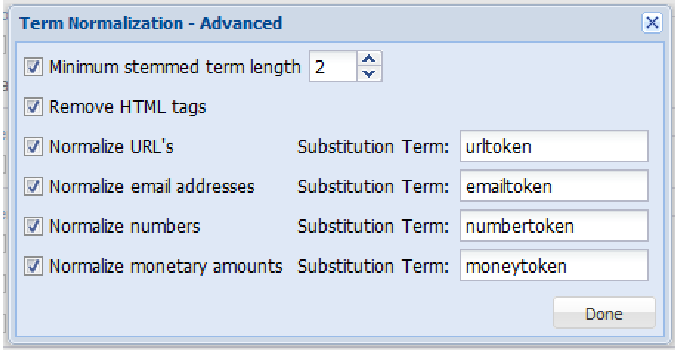
If selected, email addresses appearing in the document collection will be replaced with the term, “emailtoken”. This option is not selected by default.

Normalize numbers

If selected, numbers appearing in the document collection will be replaced with the term, “numbertoken”. This option is not selected by default.

Normalize monetary amounts

If selected, monetary amounts will be substituted with the term, “moneytoken”. This option is not selected by default.



Term Normalization - Advanced

☑ Minimum stemmed term length 2

☑ Remove HTML tags

☑ Normalize URL's Substitution Term: urltoken

☑ Normalize email addresses Substitution Term: emailtoken

☑ Normalize numbers Substitution Term: numbertoken

☑ Normalize monetary amounts Substitution Term: moneytoken

Remove terms occurring in less than __% of documents

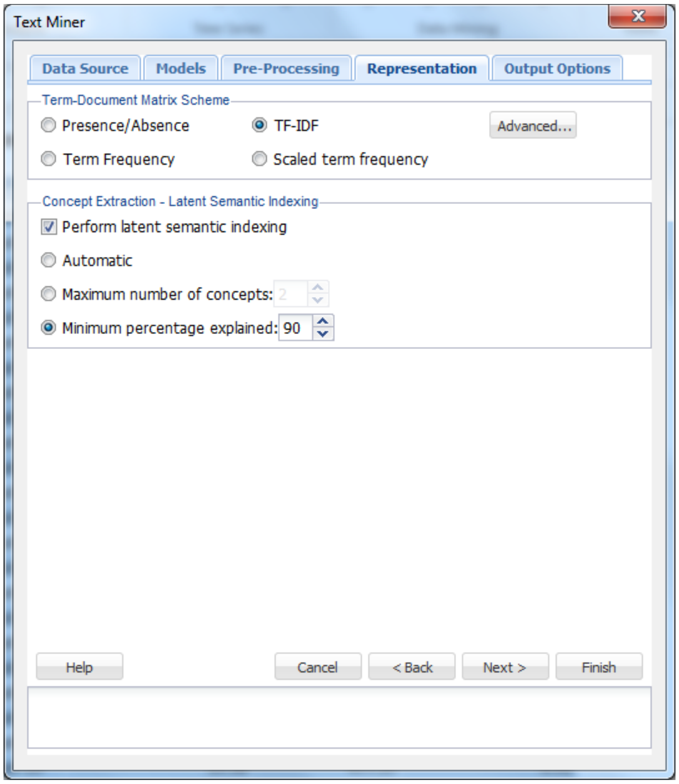
If selected, Text Miner will remove terms that appear in less than the percentage of documents specified. For most text mining applications, rarely occurring terms do not typically offer any added information or meaning to the document in relation to the collection. The default percentage is 2%.

Remove terms occurring in more than __% of documents

If selected, Text Miner will remove terms that appear in more than the percentage of documents specified. For many text mining applications, the goal is identifying terms that have discriminative power or terms that will differentiate between a number of documents. The default percentage is 98%.

Maximum term length

If selected, Text Miner will remove terms that contain a set number of characters. This option can be extremely useful for removing some parts of text which are not actual English words, for example, URLs or computer-generated tokens, or to exclude very rare terms such as Latin species or disease names, i.e. Pneumonoultramicroscopicsilicovolcanoconiosis.



Term-Document Matrix Scheme

○ Presence/Absence ● TF-IDF

○ Term Frequency ○ Scaled term frequency

Concept Extraction - Latent Semantic Indexing

☑ Perform latent semantic indexing

○ Automatic

○ Maximum number of concepts: 2

● Minimum percentage explained: 90

Term-Document Matrix Scheme

A term-document matrix is a matrix that displays the frequency-based information of terms occurring in a document or collection of documents. Each column is assigned a term and each row a document. If a term appears in a document, a weight is placed in the corresponding column indicating the term's importance or contribution. Analytic Solver Data Science offers four different commonly used methods of weighting scheme used to represent each value in the matrix: Presence/Absence, Term Frequency, TF-IDF (the default) and Scaled term frequency.

- If *Presence/Absence* is selected, Analytic Solver Data Science will enter a 1 in the corresponding row/column if the term appears in the document and 0 otherwise. This matrix scheme does not take into account the number of times the term occurs in each document.
- If *Term Frequency* is selected, Analytic Solver Data Science will count the number of times the term appears in the document and enter this value into the corresponding row/column in the matrix.
- The default setting – Term Frequency – Inverse Document Frequency (*TF-IDF)* is the product of scaled term frequency and inverse document frequency. Inverse document frequency is calculated by taking the logarithm of the total number of documents divided by the number of documents that contain the term. A high value for TF-IDF indicates that a term that does not occur frequently in the collection of documents taken as a whole, appears quite frequently in the specified document. A TF-IDF value close to 0 indicates that the term appears frequently in the collection or rarely in a specific document.

- If *Scaled term frequency* is selected, Analytic Solver Data Science will normalize (bring to the same scale) the number of occurrences of a term in the documents (see the table below).

It's also possible to create your own scheme by clicking the *Advanced* command button to open the *Term Document Matrix – Advanced* dialog. Here users can select their own choices for local weighting, global weighting, and normalization. Please see the table below for definitions regarding options for Term Frequency, Document Frequency and Normalization.

Local Weighting		Global Weighting		Normalization	
Binary	$lw_{td} = \begin{cases}1, \text{if } tf_{td} > 0\\ 0, \text{if } tf_{td} = 0\end{cases}$	None	$gw_t = 1$	None	$n_d = 1$
Raw Frequency	$lw_{td} = tf_{td}$	Inverse	$gw_t = \log_2 \frac{N}{1 + df_t}$	Cosine	$n_d = \frac{1}{\lVert \overline{g_d} \rVert_2}$
Logarithmic	$lw_{td} = \log(1 + tf_{td})$	Normal	$gw_t = \frac{1}{\sqrt{\sum_d tf_{td}^2}}$		
Augnorm	$lw_{td} = \frac{\left(\frac{tf_{td}}{\max_t tf_{td}}\right) + 1}{2}$	GF-IDF	$gw_t = \frac{cf_t}{df_t}$		
		Entropy	$gw_t = 1 + \sum_d \frac{p_{td} \log p_{td}}{\log N}$		
		IDF probability	$gw_t = \log_2 \frac{N}{1 + df_t}$		

Notations:

- tf_{td} – frequency of term t in a document d;
- df_t – document frequency of term t;
- lw_{td} – local weighting of term t in a document d;
- gw_{td} – global weighting of term t in a document d;
- n_d – normalization of vector of terms representing the document d;
- N – total number of documents in the collection;
- cf_t – collection frequency of term t;
- p_{td} – estimated probability of term t to appear in a document d $\left(p_{td} = {}^{tf_{td}}/_{cf_t}\right)$;
- $\overline{g_d}$ – vector of terms representing the document d.

Finally, the element T_{td} of Term-Document Matrix is computed as $T_{td} = lw_{td} * gw_t * n_d, \forall t, d$

Perform latent semantic indexing

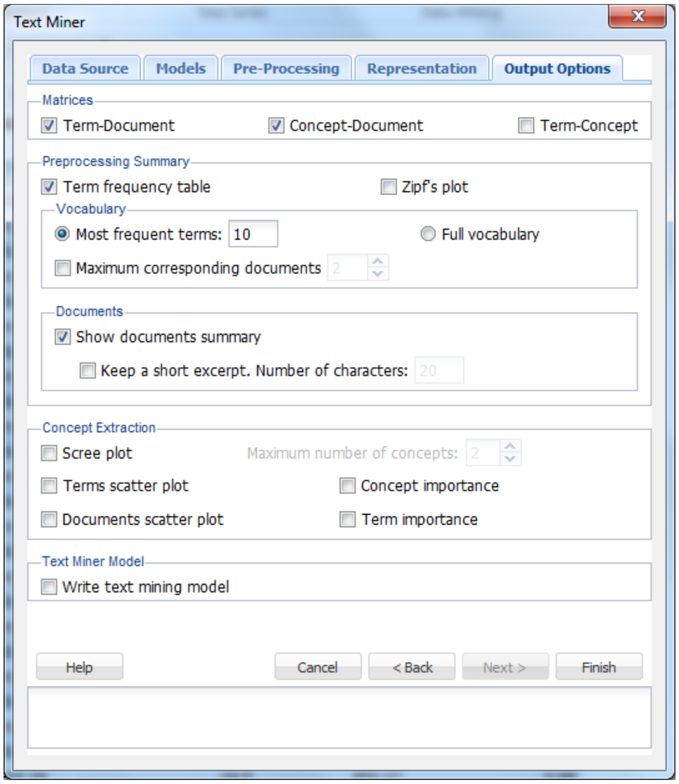
When this option is selected, Analytic Solver Data Science will use Latent Semantic Indexing (LSI) to detect patterns in the associations between terms and concepts to discover the meaning of the document.
The statistics produced and displayed in the Term-Document Matrix contain basic information on the frequency of terms appearing in the document collection. With this information we can "rank" the significance or importance

of these terms relative to the collection and particular document. Latent Semantic Indexing, in comparison, uses singular value decomposition (SVD) to map the terms and documents into a common space to find patterns and relationships. For example: if we inspected our document collection, we might find that each time the term “alternator” appeared in an automobile document, the document also included the terms “battery” and “headlights”. Or each time the term “brake” appeared in an automobile document, the terms “pads” and “squeaky” also appeared. However there is no detectable pattern regarding the use of the terms “alternator” and “brake”. Documents including “alternator” might not include “brake” and documents including “brake” might not include “alternator”. Our four terms, battery, headlights, pads, and squeaky describe two different automobile repair issues: failing brakes and a bad alternator. Latent Semantic Indexing will attempt to 1. Distinguish between these two different topics, 2. Identify the documents that deal with faulty brakes, alternator problems or both and 3. Map the terms into a common semantic space using singular value decomposition. SVD is a tool used by Text Miner to extract concepts that explain the main dimensions of meaning of the documents in the collection. The results of LSA are usually hard to examine because it can’t fully explain how the concept representation was constructed. It is more an art rather than science to make sense out of the results of LSA – Analytic Solver Data Science provides several visualizations that simplify this process greatly.

Concept Extraction – Latent Semantic Indexing

Select *Automatic, Maximum number of concepts* or *Minimum percentage explained.*

- If *Automatic* is selected, Analytic Solver Data Science will calculate the importance of each concept, take the difference between each and report any concepts above the largest difference. For example if three concepts were identified (Concept1, Concept2, and Concept3) and given importance factors of 10, 8, and 2, respectively, Analytic Solver Data Science would keep Concept1 and Concept2 since the difference between Concept2 and Concept 3 (8-2=6) is larger than the difference between Concept1 and Concept2 (10-8=2). If *Maximum number of concepts* is selected, Analytic Solver Data Science will identify the top number of concepts according to the value entered here. The default is 2 concepts.
- If *Minimum percentage explained* is selected, Analytic Solver Data Science will identify the concepts with singular values that, when taken together, sum to the minimum percentage explained, 90% is the default.
- If Maximum number of concepts is selected, Analytic Solver Data Science will retain the top significant concepts according to the value entered here.



Matrices

☑ Term-Document ☑ Concept-Document ☐ Term-Concept

Preprocessing Summary

☑ Term frequency table ☐ Zipf's plot

Vocabulary

● Most frequent terms: 10 ○ Full vocabulary

☐ Maximum corresponding documents 2

Documents

☑ Show documents summary

☐ Keep a short excerpt. Number of characters: 20

Concept Extraction

☐ Scree plot Maximum number of concepts: 2

☐ Terms scatter plot ☐ Concept importance

☐ Documents scatter plot ☐ Term importance

Text Miner Model

☐ Write text mining model

Term-Document Matrix

The term-document matrix is a matrix that displays the most frequently occurring terms across the top of the matrix and the document IDS down the left. If a term appears in a document, a weight is placed in the corresponding column indicating the importance of the term using our selection of TF-IDF on the Representation dialog. The number of terms contained in the matrix is controlled by the Maximum vocabulary size option on the Pre-Processing tab. The number of documents is equal to the number of documents in the sample. Analytic Solver Data Science offers four different commonly used methods for ranking the number of times a term appears in a document on the Pre-Processing tab: Presence/Absence, Term Frequency, TF-IDF (the default) and Scaled term frequency. This matrix is selected by default.

Concept-Document Matrix

The *Concept – Document* Matrix is enabled when *Perform latent semantic indexing* is selected on the *Representation* tab. The most important concepts will be listed across the top of the matrix and the documents will be listed down the left side of the matrix. The number of concepts is controlled by the setting for *Concept Extraction – Latent Semantic indexing* on the Representation tab: *Automatic*, *Maximum number of concepts*, or *Minimum percentage explained*. If a concept appears in a document, the singular value decomposition weight is placed in the corresponding column indicating the importance of the concept in the document. If Perform latent semantic indexing is selected, this option will also be selected by default.

Term-Concept Matrix

The *Term – Concept* matrix is enabled when *Perform latent semantic indexing* is selected on the *Representation* tab. The most important concepts will be listed across the top of the matrix and the most frequently occurring terms will be listed down the left. The number of most important concepts is controlled by the setting for *Concept Extraction – Latent Semantic indexing* option on the Representation tab: *Automatic*, *Maximum number of concepts*, or *Minimum percentage explained*. The number of terms in the matrix is controlled by the *Maximum vocabulary size* on the *Pre-Processing* tab. If a term appears in a concept, the singular value decomposition weight is placed in the corresponding column indicating the importance of the term in the concept. This option is not selected by default.

Term frequency table

The Term frequency table displays the most frequently occurring terms in the document collection according to the value entered for Most frequent terms. The first column of the table, Collection Frequency, displays the number of times the term appears in the collection. The 2nd column of the table, Document Frequency, displays the number of documents that include the term. The third column in the table, Top Documents, displays the top documents where the corresponding term appears the most frequently according to the Maximum corresponding documents setting (see below). This option is selected by default.

Most frequent terms

This option is enabled only when *Term frequency* table is selected. This option controls the number of terms displayed in the Term frequency table and Zipf's plot. This option is selected by default with a value of "10" terms.

Full vocabulary

This option is enabled only when *Term frequency* table is selected. If selected, the full vocabulary list will be displayed in the term frequency table.

Maximum corresponding documents

This option is enabled only when *Most frequent terms* is selected. This option controls the number of documents displayed in the third column of the Term frequency table. This option is not selected by default.

Zipf's plot

The Zipf Plot graphs the document frequency against the term ranks (or terms ranked in order of importance). Typically the number of terms in a document follow Zipf's law which states that the frequency of terms used in a free-form text drops exponentially. In other "words" (pun intended) when we speak we tend to use a few words a lot but most words very rarely. Hover over each point in the plot to see the most frequently occurring terms in the document collection. This option is not selected by default.

Show documents summary

If selected, Analytic Solver Data Science will produce a Documents table displaying the document ID, length of the document, and number of terms included in each document. This option is selected by default.

Keep a short except. Number of characters

If selected, Analytic Solver Data Science will produce a fourth column in the Documents table displaying the first N number of characters in the document. This option is not selected by default, but if selected, the default number of characters is “20”.

Scree Plot

This plot gives a graphical representation of the contribution or importance of each concept according to the setting for *Maximum number of concepts*. Find the largest “drop” or “elbow” in the plot to discover the leading topics in the document collection. When moving from left to right on the x-axis, the importance of each concept will diminish. This information may be used to limit the number of concepts (as variables) used as inputs into a classification model. This option is not selected by default.

Maximum number of concepts

If *Scree Plot* is enabled, *Maximum number of concepts* is enabled. Enter the number of concepts to be displayed in the Scree Plot here.

Terms scatter plot

This graph is a visual representation of the Concept – Term Matrix. It displays all terms from the final vocabulary in terms of two concepts. Similarly to Concept-Document scatter plot, Concept-Term scatter plot visualizes the distribution of vocabulary terms in the semantic space of meaning extracted with LSA. The coordinates are also normalized, so the range of axes is always [-1,1], where extreme values (close to +/-1) highlight the importance or “load” of each term to a particular concept. The terms appearing in a zero-neighborhood of concept range do not contribute much to a concept definition. In our example, if we would identify a concept having a set of terms that can be divided into two groups: one related to “Autos” and other to “Electronics”, and these groups would be distant from each other on the axis corresponding to this concept, this would definitely provide an evidence that this particular concept “caught” some pattern in the text collection that is capable of discriminating topic of article. Therefore, Term-Concept scatter plot is an extremely valuable tool for examining and understanding the main topics in the collection of documents, finding similar words that indicate similar concept, or the terms explaining the concept from “opposite sides” (e.g. *term1* can be related to cheap affordable electronics and *term2* can be related to expensive luxury electronics). This option is not selected by default.

Document scatter plot

This graph is a visual representation of the Document – Concept Matrix. This graph is a visual representation of the Concept – Document matrix. Note that Analytic Solver Data Science normalizes each document representation so it lies on a unit hypersphere. Documents that appear in the middle of the plot, with concept coordinates near 0 are not explained well by either shown concept. The further the magnitude of coordinate from zero, the more effect particular concept has for the corresponding document. In fact, two documents placed at extremes of a concept (one close to -1 and other to +1) indicates strong differentiation between these documents in terms of the extracted concept. This provides means for understanding actual meaning of the concept and investigating which concepts have the largest discriminative power, when used to represent the documents from text collection. This option is not selected by default.

Concept importance

This table displays the total number of concepts extracted, the Singular Value for each, the Cumulative Singular Value and the % of Singular Value explained which is used when *Minimum percentage explained* is selected for *Concept Extraction – Latent Semantic Indexing* on the *Representation* tab. This option is not selected by default.

Term Importance

This table display each term along with its Importance as calculated by singular value decomposition. This option is not selected by default.

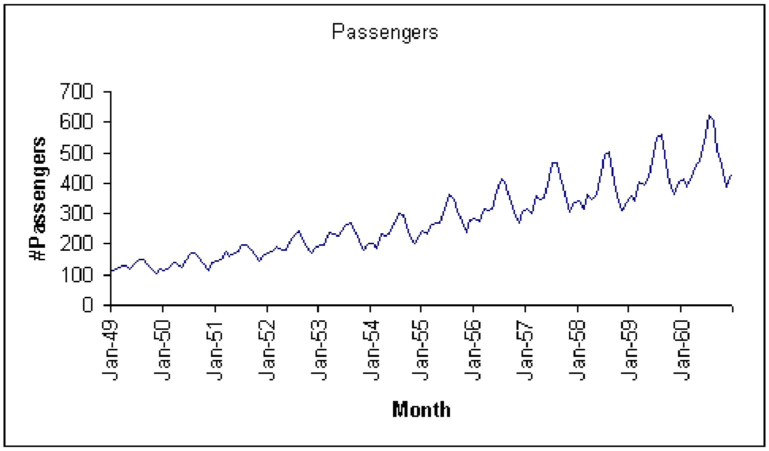
Write Text Mining Model

Select this option under *Text Miner Model* to write the base line or "base corpus" model to an output sheet. The base corpus model can be used to process new documents based on the existing text mining model. This option is not selected by default.

Exploring a Time Series Dataset

Introduction

Time series datasets contain a set of observations generated sequentially in time. Organizations of all types and sizes utilize time series datasets for analysis and forecasting for predicting next year's sales figures, raw material demand, monthly airline bookings, etc.   .



Example of a time series dataset: Monthly airline bookings.

A time series model is first used to obtain an understanding of the underlying forces and structure that produced the data and then secondly, to fit a model that will predict future behavior. In the first step, the analysis of the data, a model is created to uncover seasonal patterns or trends in the data, for example bathing suit sales in June. In the second step, forecasting, the model is used to predict the value of the data in the future, for example, next year's bathing suit sales. Separate modeling methods are required to create each type of model.

Analytic Solver Data Science features three techniques for exploring trends in a dataset, ACF (Autocorrelation function), ACVF (Autocovariance of data) and PACF (Partial autocorrelation function). These techniques help the user to explore various patterns in the data which can be used in the creation of the model. After the data is analyzed, a model can be fit to the data using Analytic Solver Data Science's ARIMA method.

Autocorrelation (ACF)

Autocorrelation (ACF) is the correlation between neighboring observations in a time series. When determining if an autocorrelation exists, the original time series is compared to the "lagged" series. This lagged series is simply the original series moved one time period forward (x_n vs $x_{n+1).}$ Suppose there are 5 time based observations: 10, 20, 30, 40, and 50. When lag = 1, the original series is moved forward one time period. When lag = 2, the original series is moved forward two time periods.

Day	Observed Value	Lag-1	Lag-2
1	10		
2	20	10	
3	30	20	10
4	40	30	20
5	50	40	30

The autocorrelation is computed according to the formula:

$$r_k = \frac{\sum_{i=k+1}^{n}(Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{i=1}^{n}(Y_t - \bar{Y})^2} \quad \text{where k = 0, 1, 2, …., n}$$

Where Y_t is the Observed Value at time t, $\bar{Y}$ is the mean of the Observed Values and Y_{t-k} is the value for Lag-k.

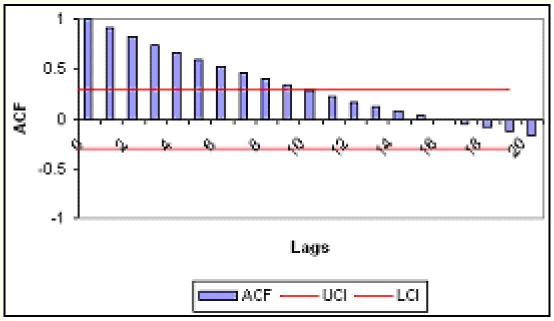
For example, using the values above, the autocorrelation for Lag-1 and Lag - 2 can be calculated as follows.

$\bar{Y}$ = (10 + 20 + 30 + 40 + 50) / 5 = 30

r_1 = ((20 – 30) * (10 - 30) + (30 - 30) * (20 - 30) + (40 - 30) * (30 - 30) + (50 – 30) * (40 – 30)) / ((10 – 30)^2 + (20 - 30)^2 + (30 – 30)^2 + (40 – 30)^2 + (50 – 30)^2) = 0.4

r_2 =((30 – 30) * (10 – 30) + (40 – 30) * (20 – 30) + (50 – 30) * (30 – 30)) / (((10 – 30)^2 + (20 - 30)^2 + (30 – 30)^2 + (40 – 30)^2 + (50 – 30)^2) = -0.1

The two red horizontal lines on the graph below delineate the Upper confidence level (UCL) and the Lower confidence level (LCL). If the data is random, then the plot should be within the UCL and LCL. If the plot exceeds either of these two levels, as seen in the plot above, then it can be presumed that some correlation exists in the data.



Partial Autocorrelation Function (PACF)

This technique is used to compute and plot the partial autocorrelations between the original series and the lags. However, PACF eliminates all linear dependence in the time series beyond the specified lag.

Autocovariance of Data (ACVF)

Autocovariance is the covariance of the time series with itself at pairs of time points.

ARIMA

An ARIMA (autoregressive integrated moving-average models) model is a regression-type model that includes autocorrelation. The basic assumption in estimating the ARIMA coefficients is that the data are stationary, that is, the trend or seasonality cannot affect the variance. This is generally not true. To achieve the stationary data, Analytic Solver Data Science will first apply "differencing": ordinary, seasonal or both.

After Analytic Solver Data Science fits the model, various results will be available. The quality of the model can be evaluated by comparing the time plot of the actual values with the forecasted values. If both curves are close, then it can be assumed that the model is a good fit. The model should expose any trends and seasonality, if any exist. If the residuals are random then the model can be assumed a good fit. However, if the residuals exhibit a trend, then the model should be refined. Fitting an ARIMA model with parameters (0,1,1) will give the same results as exponential smoothing. Fitting an ARIMA model with parameters (0,2,2) will give the same results as double exponential smoothing.

Partitioning

To avoid over fitting of the data and to be able to evaluate the predictive performance of the model on new data, we must first partition the data into training and validation sets using Analytic Solver Data Science's time series partitioning utility. After the data is partitioned, ACF, PACF, and ARIMA can be applied to the dataset.

Examples for Time Series Analysis

The examples below illustrate how Analytic Solver Data Science can be used to explore the Income.xlsx dataset to uncover trends and seasonalities in a dataset.

Click **Help – Example Models** on the Data Science ribbon, then **Forecasting/Data Science Examples**.

This dataset contains the average income of tax payers by state.

Typically the following steps are performed in a time series analysis.

1. The data is first partitioned into two sets with 60% of the data assigned to the training set and 40% of the data assigned to validation.
2. Exploratory techniques are applied to both the training and validation sets. If the results are in synch then the model can be fit. If the ACF and PACF plots are the same, then the same model can be used for both sets.
3. The model is fit using the ARIMA method.
4. When we fit a model using the ARIMA method, Analytic Solver displays the ACF and PACF plots for residuals. If these plots are in the band of UCL and LCL then it indicates that the residuals are random and the model is adequate.

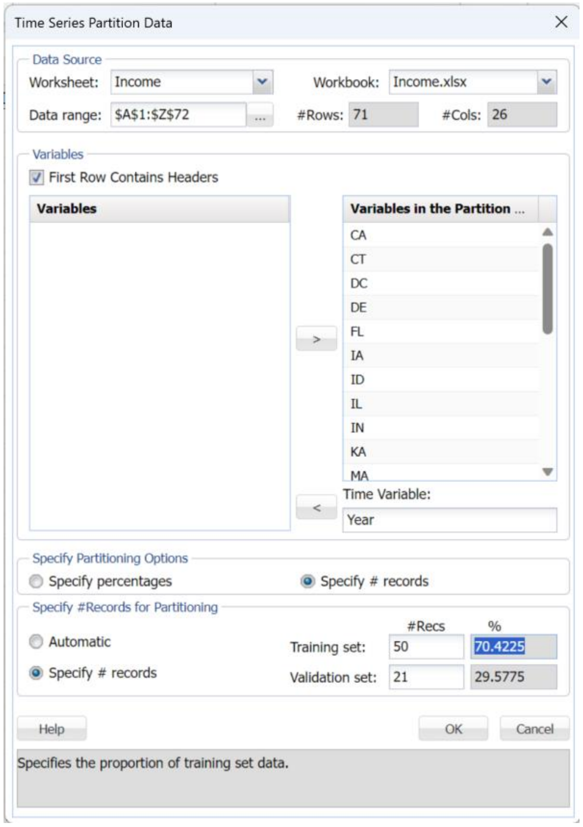
ii. If the residuals are not within the bands, then some correlation exists, and the model should be improved.

First we must perform a partition on the data. Click **Partition** within the Time Series group on the Data Science ribbon to open the following dialog.

Select **Year** under *Variables* and click > to define the variable as the *Time Variable*. Select the remaining variables under *Variables* and click > to include them in the partitioned data.

Select **Specify #Records** under *Specify Partitioning Options* to specify the number of records assigned to the training and validation sets. Then select **Specify #Records** under *Specify #Records for Partitioning*. Enter **50** for the number of *Training Set* records and **21** for the number of *Validation Set* records.

If **Specify Percentages** is selected under *Specify Partitioning Options*, Analytic Solver Data Science will assign a percentage of records to each set according to the values entered by the user or automatically entered under *Specify Percentages for Partitioning.*



Time Series Partition Data

Data Source

Worksheet: Income

Workbook: Income.xlsx

Data range: $A$1:$Z$72

#Rows: 71

#Cols: 26

Variables

☑ First Row Contains Headers

Variables

Variables in the Partition ...

CA
CT
DC
DE
FL
IA
ID
IL
IN
KA
MA

Time Variable: Year

Specify Partitioning Options

○ Specify percentages ● Specify # records

Specify #Records for Partitioning

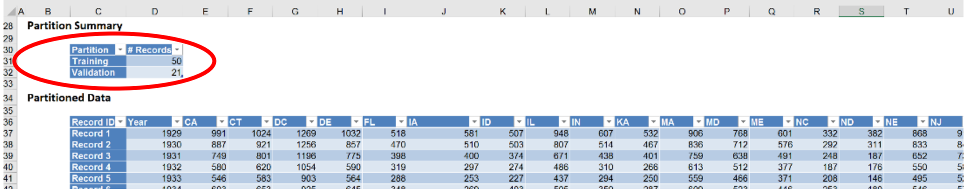
○ Automatic

● Specify # records

	#Recs	%
Training set:	50	70.4225
Validation set:	21	29.5775

Specifies the proportion of training set data.

Click **OK**. *TSPartition* is inserted to the right of the Income worksheet.



Partition Summary

Partition	# Records
Training	50
Validation	21

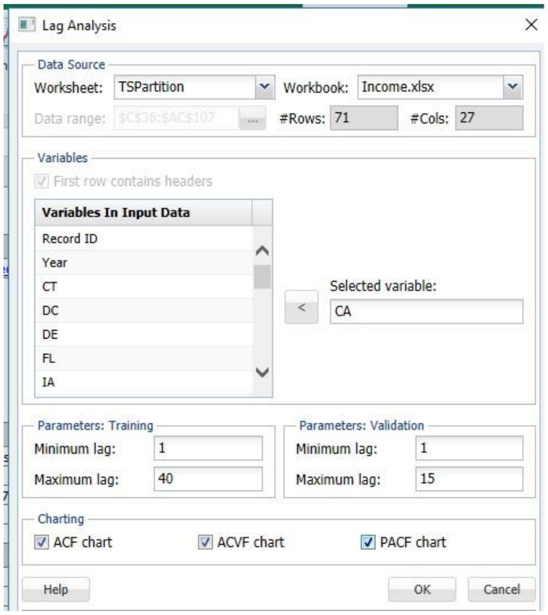
Partitioned Data

Record ID	Year	CA	CT	DC	DE	FL	IA	ID	IL	IN	KA	MA	MD	ME	NC	ND	NE	NJ
Record 1	1929	991	1024	1269	1032	518	581	507	948	607	532	906	768	601	332	382	868	9
Record 2	1930	887	921	1256	857	470	510	503	807	514	467	836	712	576	292	311	833	8
Record 3	1931	749	801	1196	775	398	400	374	671	438	401	759	638	491	248	187	652	7
Record 4	1932	580	620	1054	590	319	297	274	486	310	266	613	512	377	187	176	550	5
Record 5	1933	546	583	903	564	288	253	227	437	294	250	559	466	371	208	146	495	5

Note in the output above, the partitioning method is sequential (rather than random). The first 50 observations have been assigned to the training set and the remaining 21 observations have been assigned to the validation set.

Open the Lag Analysis dialog by clicking **ARIMA – Lag Analysis**. Select **CA** under *Variables in input data*, then click > to move the variable to *Selected* *variable*. Enter **1** for Minimum Lag and **40** for *Maximum Lag under Parameters: Training* and 1 for Minimum Lag and 15 for Maximum Lag under *Parameters: Validation.*

Under Charting, select ACF chart, ACVF chart, and PACF chart to include each chart in the output.



Lag Analysis

Data Source
Worksheet: TSPartition
Workbook: Income.xlsx
Data range: $C$36:$AC$107
#Rows: 71
#Cols: 27

Variables
☑ First row contains headers

Variables In Input Data
Record ID
Year
CT
DC
DE
FL
IA

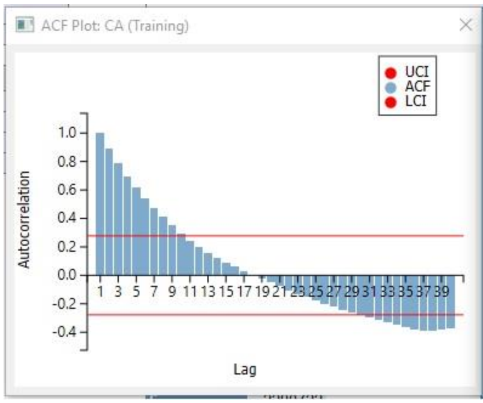
Selected variable: CA

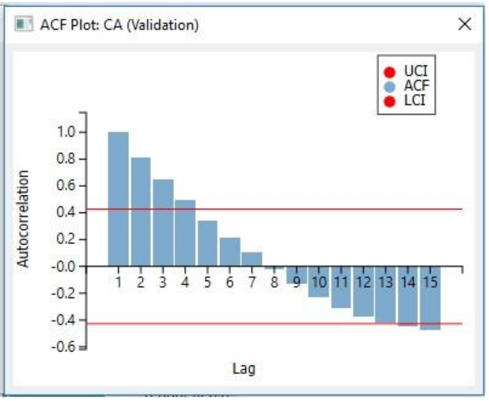
Parameters: Training
Minimum lag: 1
Maximum lag: 40

Parameters: Validation
Minimum lag: 1
Maximum lag: 15

Charting
☑ ACF chart ☑ ACVF chart ☑ PACF chart

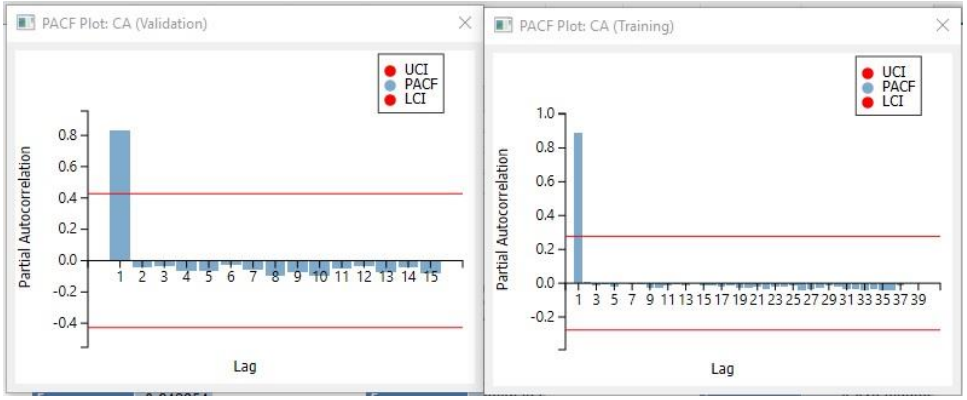
Click **OK**. *TS_Lags* is inserted right of the TSPartition worksheet.





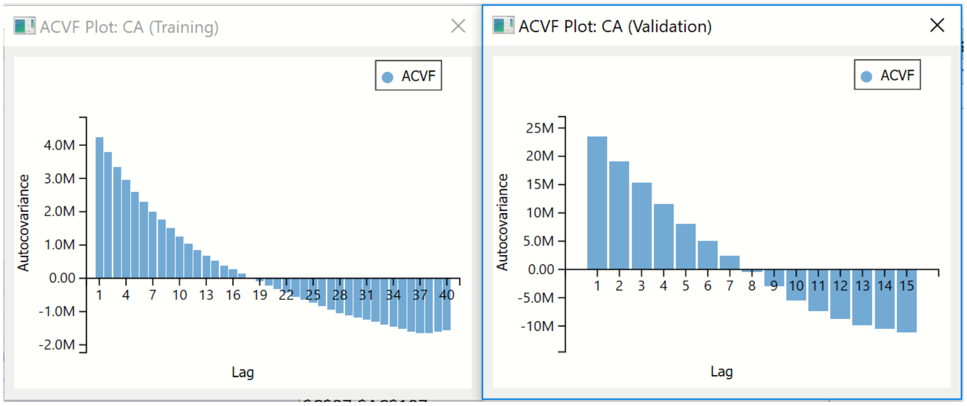
First, let's take a look at the ACF charts. Note on each chart, the autocorrelation decreases as the number of lags increase. This suggests that a definite pattern does exist in each partition. However, since the pattern does not repeat, it can be assumed that no seasonality is included in the data. In addition, both charts appear to exhibit a similar pattern.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select TS_Lags for *Worksheet* and **ACF/ACVF/PACF Training/Validation Data** for *Chart*.



The PACF functions show a definite pattern which means there is a trend in the data. However, since the pattern does not repeat, we can conclude that the data does not show any seasonality.

The screenshots below display the autocovariance values.



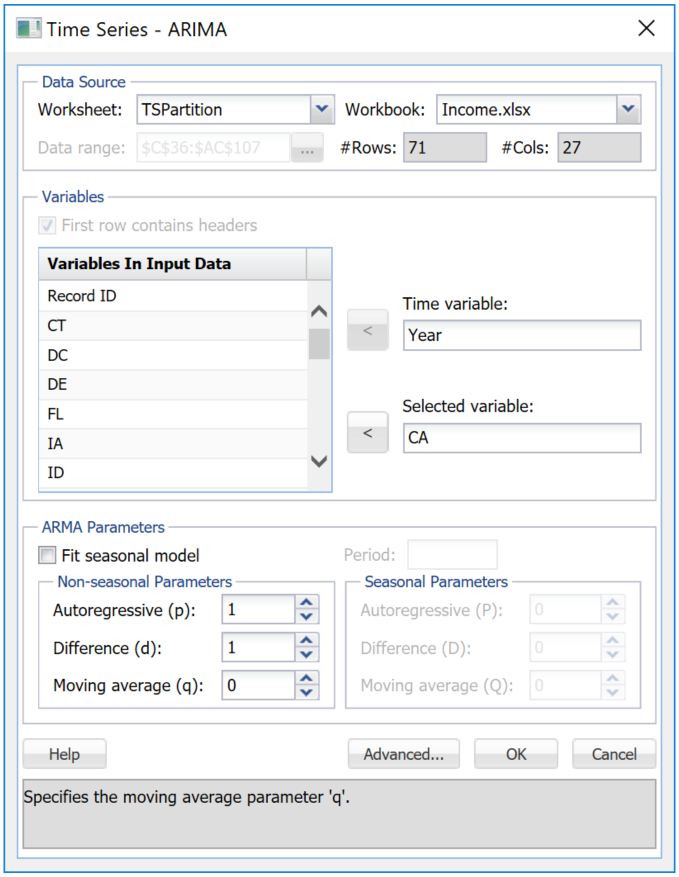
All three charts suggest that a definite pattern exists in the data, but no seasonality. In addition, both datasets exhibit the same behavior in both the training and validation sets which suggests that the same model could be appropriate for each. Now we are ready to fit the model.

The ARIMA model accepts three parameters: p – the number of autoregressive terms, d – the number of non-seasonal differences, and q – the number of lagged errors (moving averages).

Recall that the ACF plot showed no seasonality in the data which means that autocorrelation is almost static, decreasing with the number of lags increasing. This suggests setting q = 0 since there appears to be no lagged errors. The PACF plot displayed a large value for the first lag but minimal plots for successive lags. This suggest setting p =1. With most datasets, setting d =1 is sufficient or can at least be a starting point.

Click back to the TSPartition tab and then click **ARIMA – ARIMA Model** to bring up the *Time Series – ARIMA* dialog.

Select **CA** under *Variables in input data* then click > to move the variable to the *Selected Variable* field. Under *Nonseasonal Parameters* set **Autoregressive (p)** to **1**, **Difference (d)** to **1** and **Moving Average (q)** to **0**.



Time Series - ARIMA

Data Source

Worksheet: TSPartition

Workbook: Income.xlsx

Data range: $C$36:$AC$107

#Rows: 71

#Cols: 27

Variables

☑ First row contains headers

Variables In Input Data
Record ID
CT
DC
DE
FL
IA
ID

Time variable: Year

Selected variable: CA

ARMA Parameters

□ Fit seasonal model

Period:

Non-seasonal Parameters

Autoregressive (p): 1

Difference (d): 1

Moving average (q): 0

Seasonal Parameters

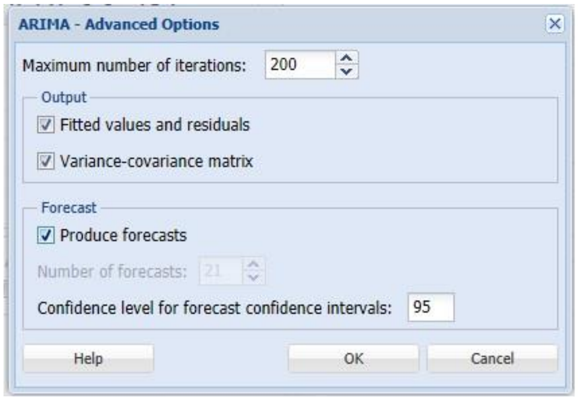
Autoregressive (P): 0

Difference (D): 0

Moving average (Q): 0

Specifies the moving average parameter 'q'.

Click **Advanced** to open the *ARIMA – Advanced Options* dialog. Select **Fitted Values and residuals**, **Produce forecasts**, and **Report Forecast Confidence Intervals**. The default *Confidence Level* setting of *95* is automatically entered. The option *Variance-covariance matrix* is selected by default.



ARIMA - Advanced Options

Maximum number of iterations: 200

Output

☑ Fitted values and residuals

☑ Variance-covariance matrix

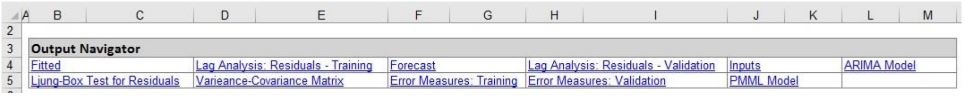
Forecast

☑ Produce forecasts

Number of forecasts: 21

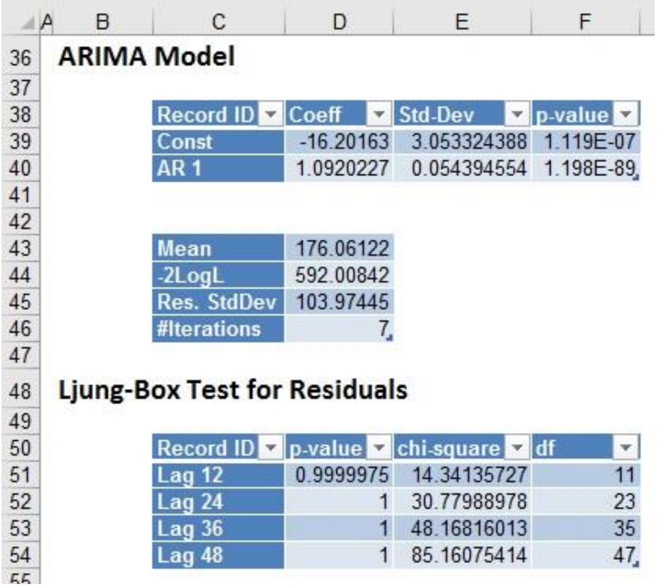
Confidence level for forecast confidence intervals: 95

Click **OK** on the *ARIMA-Advanced Options* dialog and again on the *Time Series – ARIMA* dialog. Analytic Solver Data Science calculates and displays various parameters and charts in four output sheets, *Arima_Output*, *Arima_Fitted*, *Arima_Forecast* and *Arima_Stored*. Click the *Arima_Output* tab to view the Output Navigator.



Output Navigator					
Fitted	Lag Analysis: Residuals - Training	Forecast	Lag Analysis: Residuals - Validation	Inputs	ARIMA Model
Ljung-Box Test for Residuals	Varieance-Covariance Matrix	Error Measures: Training	Error Measures: Validation	PMML Model	

Click the ARIMA Model link on the Output Navigator to move to display the ARIMA Model and Ljung-Box Test Results on Residuals.



ARIMA Model

Record ID	Coeff	Std-Dev	p-value
Const	-16.20163	3.053324388	1.119E-07
AR 1	1.0920227	0.054394554	1.198E-89

Mean	176.06122
-2LogL	592.00842
Res. StdDev	103.97445
#Iterations	7

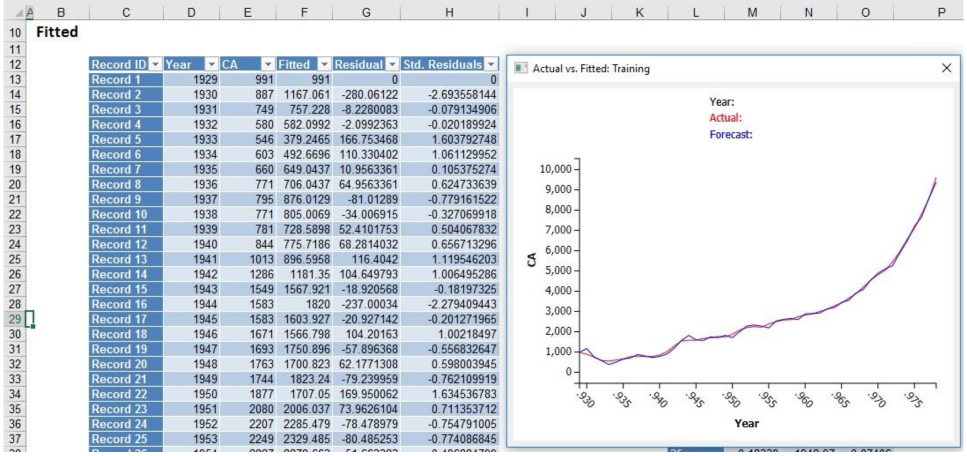
Ljung-Box Test for Residuals

Record ID	p-value	chi-square	df
Lag 12	0.9999975	14.34135727	11
Lag 24	1	30.77988978	23
Lag 36	1	48.16816013	35
Lag 48	1	85.16075414	47

Analytic Solver has calculated the constant term and the AR1 term for our model, as seen above. These are the constant and f1 terms of our forecasting equation. See the following output of the Chi - square test.

The very small p-values for the constant term (1.119E-7) and AR1 term (1.19e-89) suggest that the model is a good fit to our data.

Click the Fitted link on the Output Navigator. This table plots the actual and fitted values and the resulting residuals for the training partition. As shown in the graph below, the Actual and Forecasted values match up fairly well. The usefulness of the model in forecasting will depend upon how close the actual and forecasted values are in the Forecast, which we will inspect later.

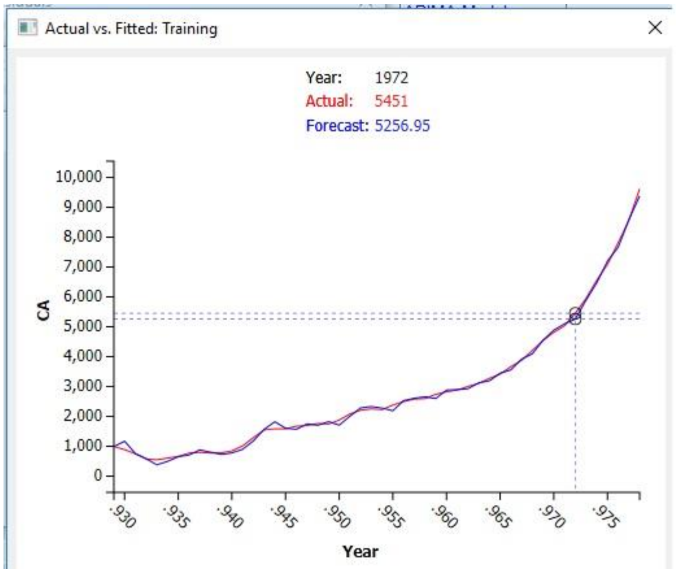


Fitted

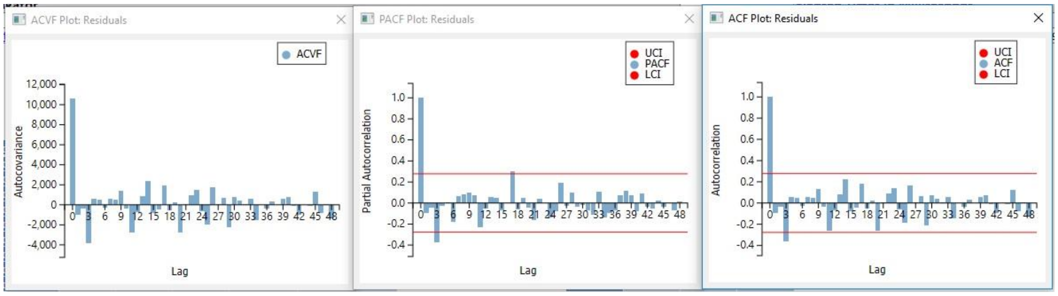
Record ID	Year	CA	Fitted	Residual	Std. Residuals
Record 1	1929	991	991	0	0
Record 2	1930	887	1167.061	-280.06122	-2.693558144
Record 3	1931	749	757.228	-8.2280083	-0.079134906
Record 4	1932	580	582.0992	-2.0992363	-0.020189924
Record 5	1933	546	379.2465	166.753468	1.603792748
Record 6	1934	603	492.6696	110.330402	1.061129952
Record 7	1935	660	649.0437	10.9563361	0.105375274
Record 8	1936	771	706.0437	64.9563361	0.624733639
Record 9	1937	795	876.0129	-81.01289	-0.779161522
Record 10	1938	771	805.0069	-34.006915	-0.327069918
Record 11	1939	781	728.5898	52.4101753	0.504067832
Record 12	1940	844	775.7186	68.2814032	0.656713296
Record 13	1941	1013	896.5958	116.4042	1.119546203
Record 14	1942	1286	1181.35	104.649793	1.006495286
Record 15	1943	1549	1567.921	-18.920568	-0.18197325
Record 16	1944	1583	1820	-237.00034	-2.279409443
Record 17	1945	1583	1603.927	-20.927142	-0.201271965
Record 18	1946	1671	1566.798	104.20163	1.00218497
Record 19	1947	1693	1750.896	-57.896368	-0.556832647
Record 20	1948	1763	1700.823	62.1771308	0.598003945
Record 21	1949	1744	1823.24	-79.239959	-0.762109919
Record 22	1950	1877	1707.05	169.950062	1.634536783
Record 23	1951	2080	2006.037	73.9626104	0.711353712
Record 24	1952	2207	2285.479	-78.478979	-0.754791005
Record 25	1953	2249	2329.485	-80.485253	-0.774086845

Use your mouse to select a point on the graph to compare the Actual value to the Forecasted value.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select **Arima_Fitted** for *Worksheet* and **ACF/ACVF/PACF Training/Validation Data** for *Chart*.

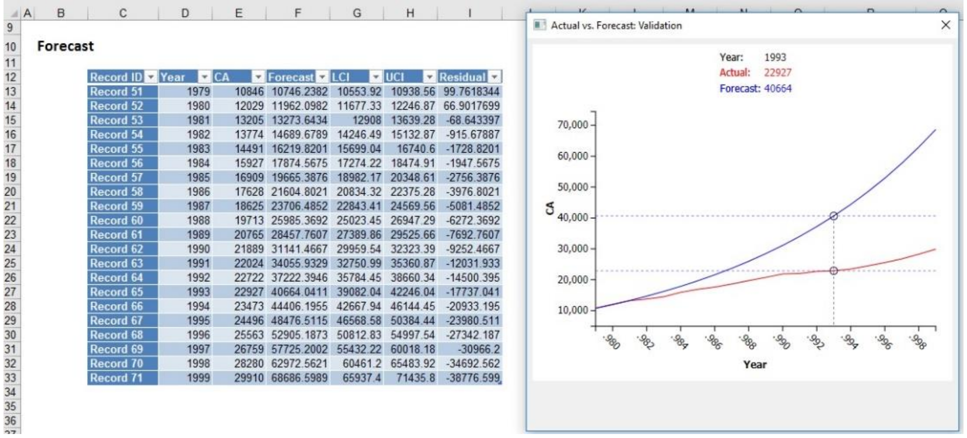


Take a look at the ACF and PACF plots for Errors found at the bottom of *ARIMA_Output*. Analytic Solver contains one more additional chart, the ACVF Plot for the Residuals.



With the exception of Lag1, the majority of the lags in the PACF and ACF charts are either clearly within the UCL and LCL bands or just outside of these bands. This suggests that the residuals are random and are not correlated.

Click the Forecast link on the Output Navigator to display the Forecast Data table and charts.



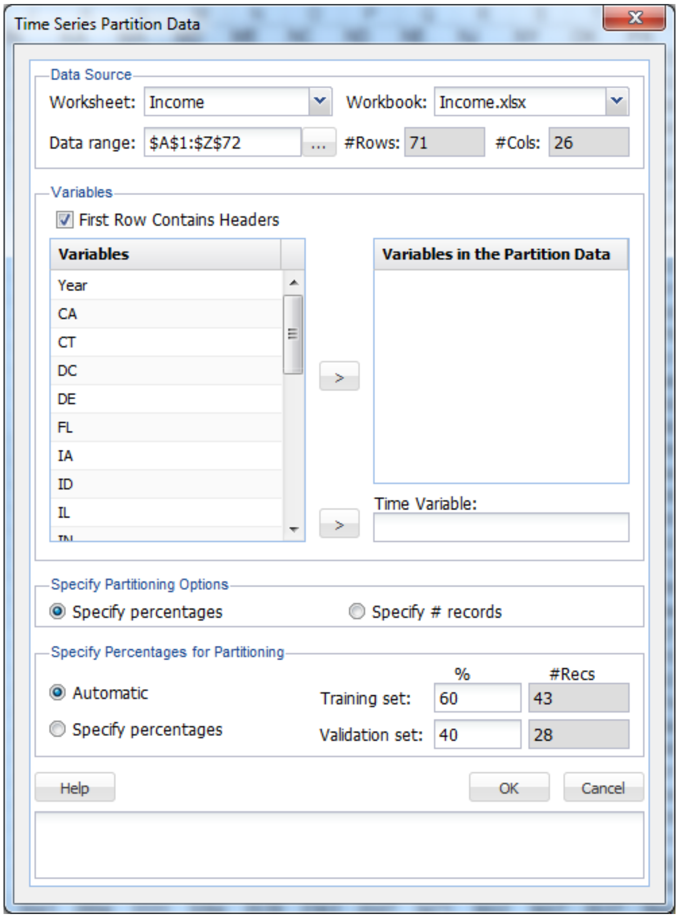
Forecast

Record ID	Year	CA	Forecast	LCI	UCI	Residual
Record 51	1979	10846	10746.2382	10553.92	10938.56	99.7618344
Record 52	1980	12029	11962.0982	11677.33	12246.87	66.9017699
Record 53	1981	13205	13273.6434	12908	13639.28	-68.643397
Record 54	1982	13774	14689.6789	14246.49	15132.87	-915.67887
Record 55	1983	14491	16219.8201	15699.04	16740.6	-1728.8201
Record 56	1984	15927	17874.5675	17274.22	18474.91	-1947.5675
Record 57	1985	16909	19665.3876	18982.17	20348.61	-2756.3876
Record 58	1986	17628	21604.8021	20834.32	22375.28	-3976.8021
Record 59	1987	18625	23706.4852	22843.41	24569.56	-5081.4852
Record 60	1988	19713	25985.3692	25023.45	26947.29	-6272.3692
Record 61	1989	20765	28457.7607	27389.86	29525.66	-7692.7607
Record 62	1990	21889	31141.4667	29959.54	32323.39	-9252.4667
Record 63	1991	22024	34055.9329	32750.99	35360.87	-12031.933
Record 64	1992	22722	37222.3946	35784.45	38660.34	-14500.395
Record 65	1993	22927	40664.0411	39082.04	42246.04	-17737.041
Record 66	1994	23473	44406.1955	42667.94	46144.45	-20933.195
Record 67	1995	24496	48476.5115	46568.58	50384.44	-23980.511
Record 68	1996	25563	52905.1873	50812.83	54997.54	-27342.187
Record 69	1997	26759	57725.2002	55432.22	60018.18	-30966.2
Record 70	1998	28280	62972.5621	60461.2	65483.92	-34692.562
Record 71	1999	29910	68686.5989	65937.4	71435.8	-38776.599

The table shows the actual and forecasted values along with LCI (Lower Confidence Interval), UCI (Upper Confidence Interval) and Residual values. The "Lower" and "Upper" values represent the lower and upper bounds of the confidence interval. There is a 95% chance that the forecasted value will fall into this range. The graph to the right plots the Actual values for CA against the Forecasted values. Again, click any point on either curve to compare the Actual against the Forecasted values.

Options for Exploring Time Series Datasets

The options described below appear on one of the 3 Time Series dialogs.



Time Series Partition Data

Data Source

Worksheet: Income
Workbook: Income.xlsx
Data range: $A$1:$Z$72
#Rows: 71
#Cols: 26

Variables

☑ First Row Contains Headers

Variables
Year
CA
CT
DC
DE
FL
IA
ID
IL
IN

Variables in the Partition Data

Time Variable:

Specify Partitioning Options

● Specify percentages ○ Specify # records

Specify Percentages for Partitioning

● Automatic
○ Specify percentages

	%	#Recs
Training set:	60	43
Validation set:	40	28

The options below appear on the *Time Series Partition Data* tab.

Time variable

Select a time variable from the available variables and click the > button. If a Time Variable is not selected, Analytic Solver will assign one to the partitioned data.

Variables in the Partition Data

Select one or more variables from the *Variables* field by clicking on the corresponding selection button.

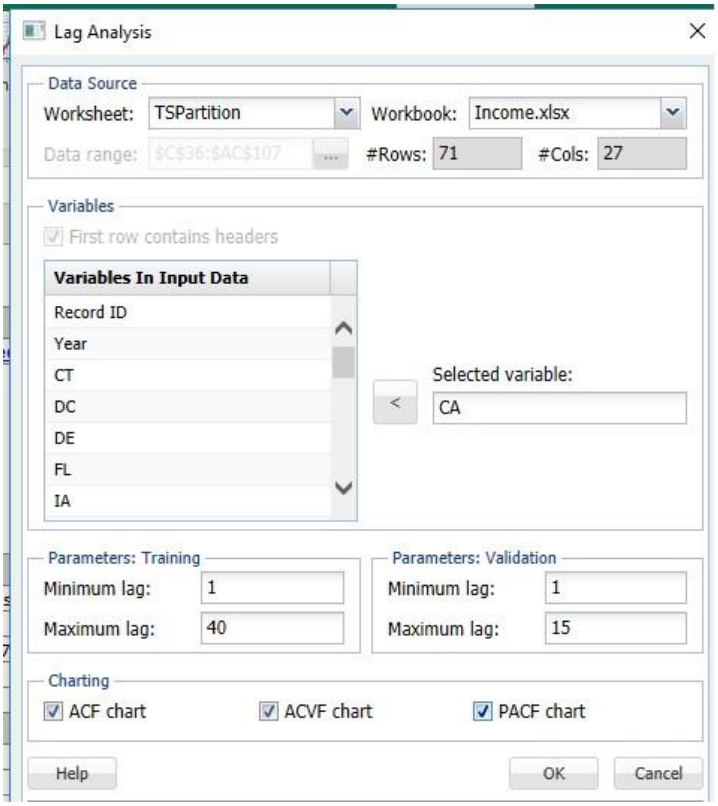
Specify Partitioning Options

Select **Specify Percentages** to specify the percentage of the total number of records desired in the Validation and Training sets. Select **Specify # Records** to enter the desired number of records in the Validation and Training sets.

Specify Percentages for Partitioning

Select **Automatic** to have Analytic Solver automatically use 60% of the records in the Training set and 40% of the records in the Validation set. Select **Specify # Records** under *Specify Partitioning Options*, to manually select the number of records to include in the Validation and Training sets. If **Specify Percentages** is selected under *Specify Partitioning Options*, then select **Specify Percentages** to specify the percentage of the total number of records to be included in the Validation and Training sets.

The options below appear on the *Lag Analysis* dialog.



Lag Analysis

Data Source

Worksheet: TSPartition

Workbook: Income.xlsx

Data range: $C$36:$AC$107

#Rows: 71

#Cols: 27

Variables

☑ First row contains headers

Variables In Input Data
Record ID
Year
CT
DC
DE
FL
IA

Selected variable: CA

Parameters: Training

Minimum lag: 1

Maximum lag: 40

Parameters: Validation

Minimum lag: 1

Maximum lag: 15

Charting

☑ ACF chart ☑ ACVF chart ☑ PACF chart

Variables in the input data

Select one or more variables from the *Variables* field by clicking on the corresponding selection button.

Selected variable

The selected variable appears here.

Parameters: Training

Enter the minimum and maximum lags for the Training Data here. The # lags for the Training set should be >= 1 and < N where N is the number of records in the Training dataset.

Parameters: Validation

Enter the minimum and maximum lags for the Validation Data here. The # lags for the Validation Data set should be >= 1 and < N where N is the number of records in the Validation dataset.

Plot ACF Chart

If this option is selected, Analytic Solver plots the autocorrelations for the selected variable.

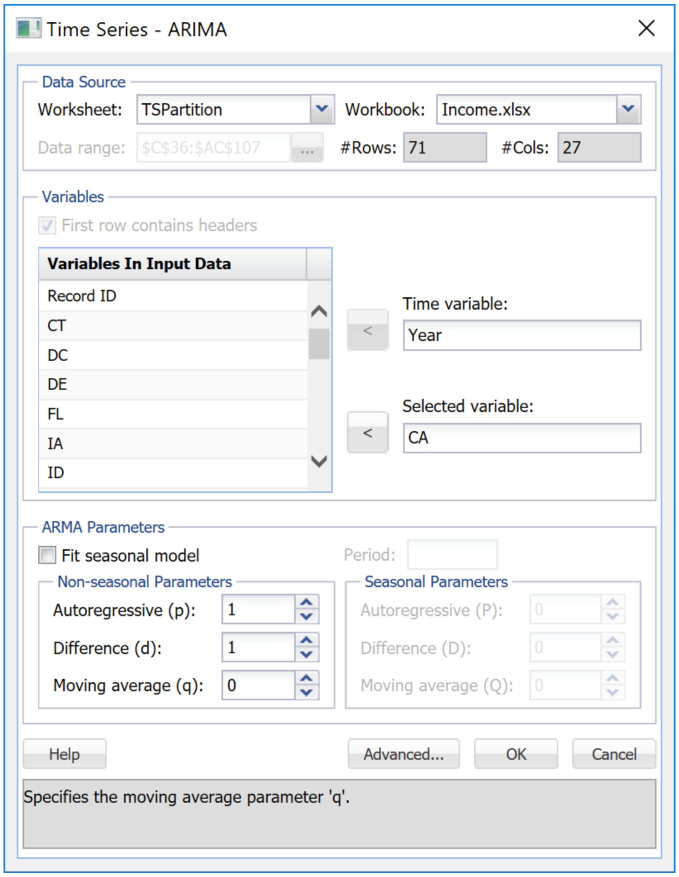
Plot PACF Chart

If this option is selected, Analytic Solver plots the partial autocorrelations for the selected variable.

Plot ACVF Chart

If this option is selected, Analytic Solver plots the Autocovariance of Data for the selected variable.

The options below appear on the *Time Series – ARIMA* dialog.



Time Series - ARIMA

Data Source

Worksheet: TSPartition

Workbook: Income.xlsx

Data range: $C$36:$AC$107

#Rows: 71

#Cols: 27

Variables

☑ First row contains headers

Variables In Input Data
Record ID
CT
DC
DE
FL
IA
ID

Time variable: Year

Selected variable: CA

ARMA Parameters

□ Fit seasonal model

Period:

Non-seasonal Parameters

Autoregressive (p): 1

Difference (d): 1

Moving average (q): 0

Seasonal Parameters

Autoregressive (P): 0

Difference (D): 0

Moving average (Q): 0

Specifies the moving average parameter 'q'.

Time Variable

The Time variable is automatically selected when using a partitioned dataset. When using an unpartitioned dataset, select the desired Time variable by clicking the > button.

Selected Variable

Select the desired variable to be included in the ARIMA model by clicking the > button.

Fit seasonal model

Select this option to specify a seasonal model. The seasonal parameters are enabled when this option is selected.

Period

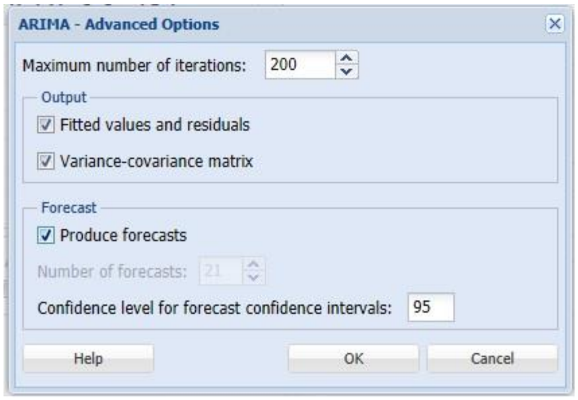
If Fit seasonal model is selected, this option is enabled. Seasonality in a dataset appears as patterns at specific periods in the time series.

Nonseasonal Parameters

Enter the nonseasonal parameters here for Autoregressive (p), Difference (d), and Moving Average (q).

Seasonal Parameters

Enter the Seasonal parameters here for Autoregressive (P), Difference (D), and Moving Average (Q).



ARIMA - Advanced Options

Maximum number of iterations: 200

Output

☑ Fitted values and residuals

☑ Variance-covariance matrix

Forecast

☑ Produce forecasts

Number of forecasts: 21

Confidence level for forecast confidence intervals: 95

The options below appear on the *ARIMA – Advanced Options* dialog.

Maximum number of iterations

Enter the maximum number of iterations here. The default is 200 iterations.

Fitted Values and residuals

Analytic Solver Data Science will include the fitted values and residuals in the output if this option is selected.

Variance-covariance matrix

Analytic Solver Data Science will include the variance-covariance matrix in the output if this option is selected. This option is selected by default.

Produce forecasts

If this option is selected, Analytic Solver Data Science will display the desired number of forecasts. If the data has been partitioned, Analytic Solver will display the forecasts on the validation data.

Number of forecasts

If *Produce forecasts* is selected and a non-partitioned dataset is being used, this option is enabled. The maximum number of forecasts is 100.

Confidence level for forecast confidence intervals

If this option is selected, enter the desired confidence level here. (The default level is 95%.) The Lower and Upper values of the computed confidence levels will be included in the output. The forecasted value will be guaranteed to fall within this range for the specified confidence level.

Smoothing Techniques

Introduction

Data collected over time is likely to show some form of random variation. "Smoothing techniques" can be used to reduce or cancel the effect of these variations. These techniques, when properly applied, will "smooth" out the random variation in the time series data to reveal any underlying trends that may exist.

Analytic Solver Data Science features four different smoothing techniques: Exponential, Moving Average, Double Exponential, and Holt Winters. The first two techniques, Exponential and Moving Average, are relatively simple smoothing techniques and should not be performed on datasets involving seasonality. The last two techniques are more advanced techniques which can be used on datasets involving seasonality.

Exponential smoothing

Exponential smoothing is one of the more popular smoothing techniques due to its flexibility, ease in calculation and good performance. As in Moving Average Smoothing, a simple average calculation is used. Exponential Smoothing, however, assigns exponentially decreasing weights starting with the most recent observations. In other words, new observations are given relatively more weight in the average calculation than older observations. Analytic Solver Data Science utilizes the formulas below in the Exponential Smoothing tool.

$S_0 = x_0$
$S_t = \alpha x_{t-1} + (1-\alpha)st-1, t > 0$

where

- original observations are denoted by $\{x_t\}$ starting at $t = 0$
- α is the smoothing factor which lies between 0 and 1

As with Moving Average Smoothing, Exponential Smoothing should only be used when the dataset contains no seasonality. The forecast will be a constant value which is the smoothed value of the last observation.

Moving Average Smoothing

In this simple technique each observation is assigned an equal weight. Additional observations are forecasted by using the average of the previous observations. If we have the time series $X_1, X_2, X_3, \ldots.., X_{t,}$ then this technique will predict X_{t+k} as follows :
$S_t = \text{Average } (x_{t-k+1}, x_{t-k+2}, \ldots.., x_t)$, t= k, k+1, k+2, ...N
where k is the smoothing parameter. Analytic Solver Data Science allows a parameter value between 2 and t-1 where t is the number of observations in the dataset. Care should be taken when choosing this parameter as a large parameter value will oversmooth the data while a small parameter value will undersmooth the data. Using the past three observations are enough to predict the next observations. As with Exponential Smoothing, this technique should not be applied when seasonality is present in the dataset.

Double exponential smoothing

Double exponential smoothing can be defined as “Exponential smoothing of Exponential smoothing”. As stated above, Exponential smoothing should not be used when the data includes seasonality. However, Double Exponential smoothing introduces a 2nd equation which includes a trend parameter. Therefore, this technique can and should be used when a trend is inherent in the dataset, but not used when seasonality is present. Double exponential smoothing is defined in the following manner:

$S_t = A_t + B_t$, t = 1,2,3,..., N

Where, $A_t = aX_t + (1- a) S_{t-1}$ $0< a <= 1$

$B_t = b (A_t - A_{t-1}) + (1 - b) B_{t-1}$ $0< b <= 1$

The forecast equation is: $X_{t+k} = A_t + K B_t$, K = 1, 2, 3, ...

where a denotes the Alpha parameter and b denotes the Trend parameters. Analytic Solver Data Science allows these two parameters to be entered manually. In addition, Analytic Solver includes an optimize feature which will chose the best values for Alpha and Trend based on the Forecasting Mean Squared Error. If the trend parameter is 0, then this technique is equivalent to the Exponential Smoothing technique. (However, results may not be identical due to different initialization methods for these two techniques.)

Holt Winters Smoothing

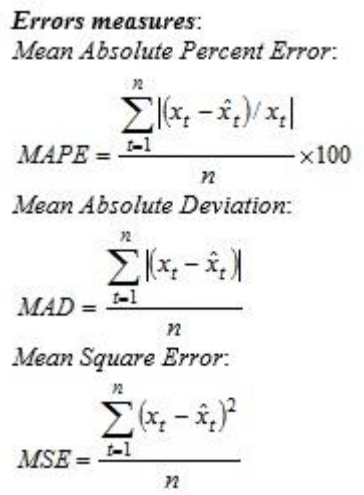
What happens if the data exhibits trends as well as seasonality? We now introduce a third parameter, g to account for seasonality (sometimes called periodicity) in a dataset. The resulting set of equations is called the Holt-Winters method after the names of the inventors. The Holt Winters method can be used on datasets involving trend and seasonality (a, b , g). Values for all three parameters can range between 0 and 1.

There are three models associated with this method:

Multiplicative: $X_t = (A_t+ B_t)* S_t +e_t$ A_t and B_t are previously calculated initial estimates. S_t is the average seasonal factor for the t^{th} season.

Additive: $X_t = (A_t+ B_t) +SN_t + e_t$

No Trend: b = 0, so, $X_t = A_t * SN_t +e_t$



***Errors measures*:**

Mean Absolute Percent Error:

$$MAPE = \frac{\sum_{t=1}^{n} \left|(x_t - \hat{x}_t)/x_t\right|}{n} \times 100$$

Mean Absolute Deviation:

$$MAD = \frac{\sum_{t=1}^{n} \left|(x_t - \hat{x}_t)\right|}{n}$$

Mean Square Error:

$$MSE = \frac{\sum_{t=1}^{n} (x_t - \hat{x}_t)^2}{n}$$

Holt Winters smoothing is similar to exponential smoothing if b and g = 0 and is similar to double exponential smoothing if g = 0.

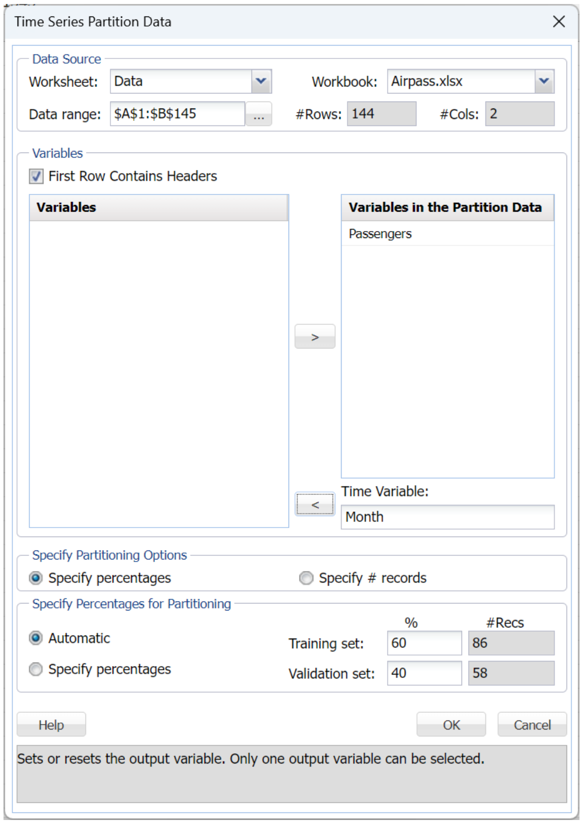
Exponential Smoothing Example

This example illustrates how to use Analytic Solver's Exponential Smoothing technique to uncover trends in the example time series, Airpass.xlsx and Income.xlsx. To open both files, click **Help – Example Models -- Forecasting/Data Science Examples**

Airpass.xlsx contains the monthly totals of international airline passengers from 1949 - 1960. Income.xlsx contains the average income of tax payers by state.

Click **Partition** in the *Time Series* group on the Data Science ribbon to open the *Time Series Partition* dialog, as shown below.

Select **Month** as the *Time Variable*. Select **Passengers** as the *Variables in the Partition Data*.



Time Series Partition Data

Data Source

Worksheet: Data

Workbook: Airpass.xlsx

Data range: $A$1:$B$145

#Rows: 144

#Cols: 2

Variables

☑ First Row Contains Headers

Variables	Variables in the Partition Data
	Passengers

Time Variable: Month

Specify Partitioning Options

● Specify percentages ○ Specify # records

Specify Percentages for Partitioning

● Automatic

○ Specify percentages

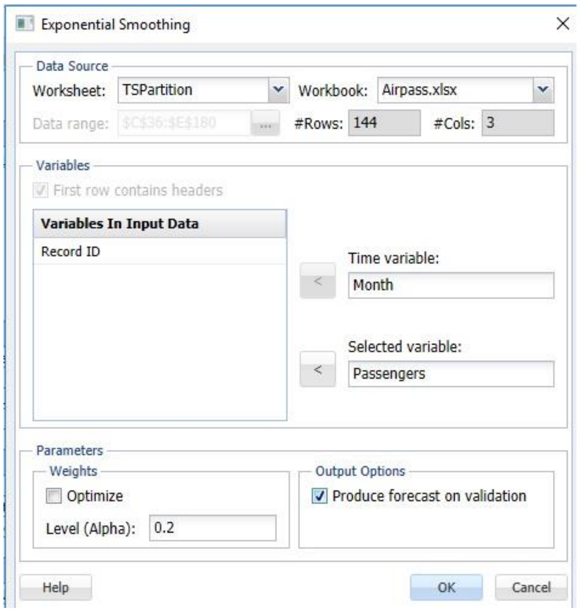
	%	#Recs
Training set:	60	86
Validation set:	40	58

Sets or resets the output variable. Only one output variable can be selected.

Then click **OK** to partition the data into training and validation sets. (Partitioning is optional. Smoothing techniques may be run on full unpartitioned datasets.) The result of the partition, TSPartition, is inserted right of the Airpass worksheet.

Click **Smoothing – Exponential** to open the *Exponential Smoothing* dialog.

Select **Month** as the *Time Variable*, unless already selected. Select **Passengers** as the *Selected variable* and also **Produce Forecast on validation**.



Exponential Smoothing

Data Source

Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data
Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Weights

☐ Optimize

Level (Alpha): 0.2

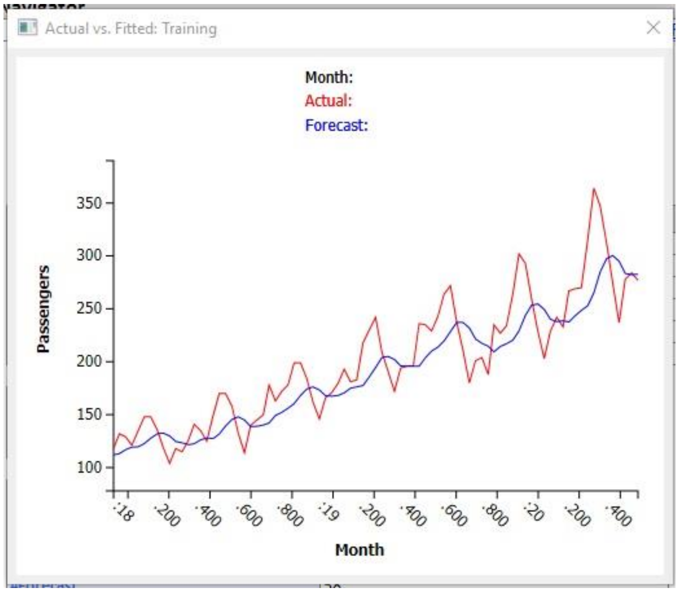
Output Options

☑ Produce forecast on validation

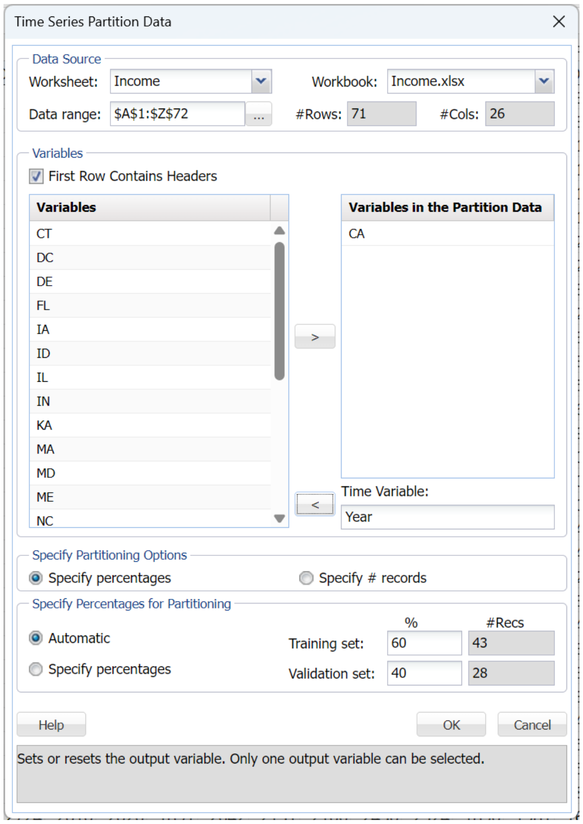
Click **OK** to apply the smoothing technique. *Expo* and *Expo_Stored* will be inserted right of the Data worksheet. See the “Scoring New Data” chapter for information on the *Expo_Stored* sheet.

The Actual Vs Fitted: Training chart shows that the Exponential smoothing technique does not result in a good fit as the model does not effectively capture the seasonality in the dataset. As a result, the summer months where the number of airline passengers are typically high appear to be under forecasted (i.e. too low) and the forecasts for months with low passenger numbers are too high. Consequently, *an exponential smoothing forecast should never be used when the dataset includes seasonality*. An alternative would be to perform a regression on the model and then apply this technique to the residuals.

*Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select **Expo** for Worksheet and **Time Series Training Data** or **Time Series Validation Data** for Chart.*



Now let's take a look at an example that does not include seasonality. Click Partition within the Time Series group on the Data Science ribbon to open the Time Series Partition dialog. First partition the dataset into training and validation sets using **Year** as the *Time Variable* and **CA** as the *Variables in the partition data*.


Time Series Partition Data

Data Source

Worksheet: Income

Workbook: Income.xlsx

Data range: $A$1:$Z$72

#Rows: 71

#Cols: 26

Variables

☑ First Row Contains Headers

Variables: CT, DC, DE, FL, IA, ID, IL, IN, KA, MA, MD, ME, NC

Variables in the Partition Data: CA

Time Variable: Year

Specify Partitioning Options

● Specify percentages ○ Specify # records

Specify Percentages for Partitioning

● Automatic

○ Specify percentages

	%	#Recs
Training set:	60	43
Validation set:	40	28

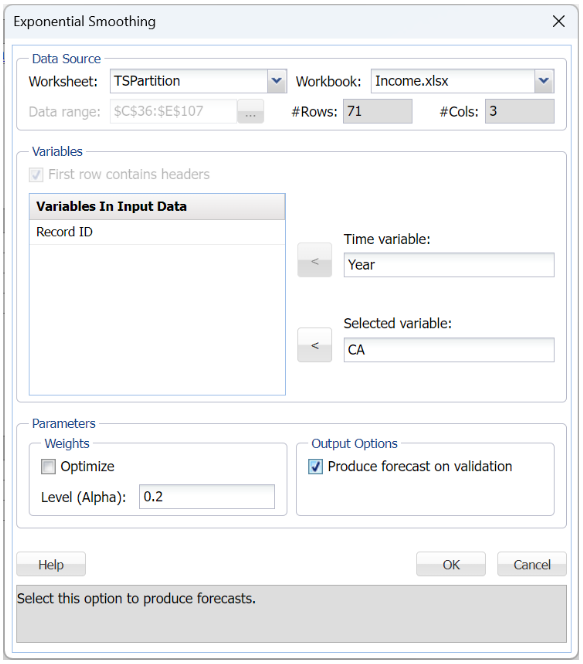
Sets or resets the output variable. Only one output variable can be selected.

Click **OK** to accept the partitioning defaults and create the two sets (Training and Validation). *TSPartition* is inserted right of the Income worksheet. Click **Smoothing – Exponential** from the Data Science ribbon to open the *Exponential Smoothing* dialog.

Select **Year** for *Time Variable* if it has not already been selected. Select **CA** as the *Selected Variable* and **Produce forecast on validation**.

The smoothing parameter (Alpha) determines the magnitude of weights assigned to the observations. For example, a value close to 1 would result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights. A value close to 0 would result in the earliest observations being assigned the largest weights and the latest observations being assigned the smallest weights. As a result, the value of Alpha depends on how much influence the most recent observations should have on the model.

Analytic Solver includes the Optimize feature that will choose the Alpha parameter value that results in the minimum residual mean squared error. It is recommended that this feature be used carefully as it can often lead to a model that is overfit to the training set. An overfit model rarely exhibits high predictive accuracy in the validation set.



Exponential Smoothing

Data Source

Worksheet: TSPartition Workbook: Income.xlsx

Data range: $C$36:$E$107 #Rows: 71 #Cols: 3

Variables

☑ First row contains headers

Variables In Input Data
Record ID

Time variable: Year

Selected variable: CA

Parameters

Weights

□ Optimize

Level (Alpha): 0.2

Output Options

☑ Produce forecast on validation

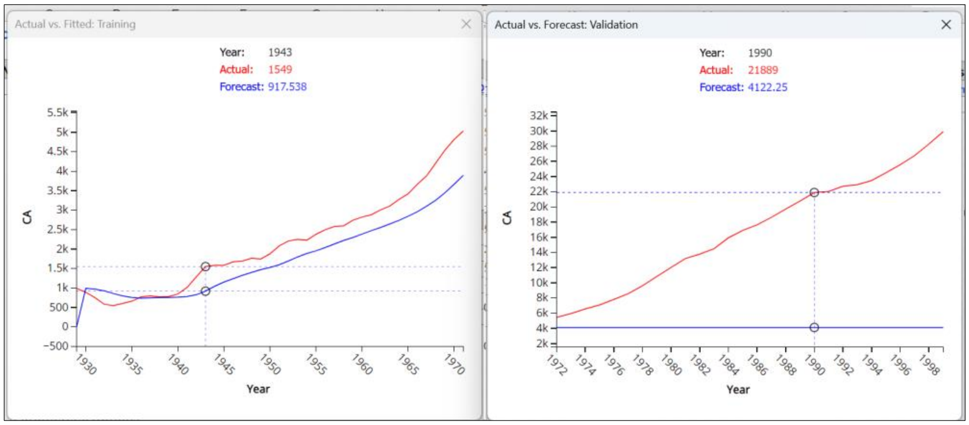
Help OK Cancel

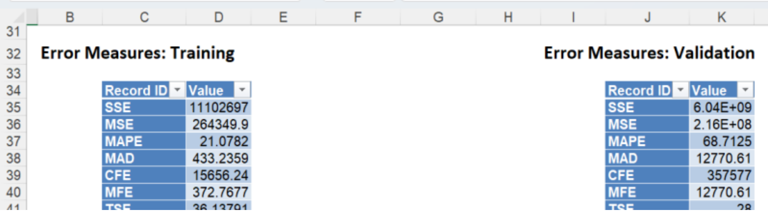
Select this option to produce forecasts.

If we click **OK** to accept the default Alpha value of 0.2. Two output sheets, *Expo* and *Expo_Stored*, will be inserted right of the Data worksheet. For more information on the *Expo_Stored* worksheet, see the chapter "Scoring New Data" in the Analytic Solver Data Science User Guide.

The *Training and Validation Error Measures* tables show a fitted model with a MSE of 258,202.3 for the Training set and a MSE of 2.16E08 for the Validation set. These are fairly large numbers and indicate that the model is not well-fit.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select Expo for Worksheet and ***Time Series Training Data*** *or* ***Time Series Validation Data*** *for Chart.*





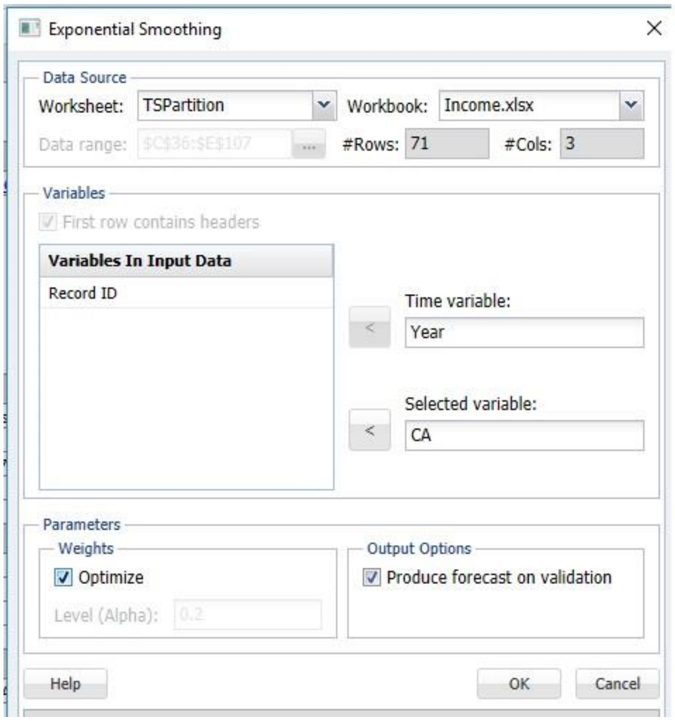
Error Measures: Training

Record ID	Value
SSE	11102697
MSE	264349.9
MAPE	21.0782
MAD	433.2359
CFE	15656.24
MFE	372.7677
TSE	36.13791

Error Measures: Validation

Record ID	Value
SSE	6.04E+09
MSE	2.16E+08
MAPE	68.7125
MAD	12770.61
CFE	357577
MFE	12770.61
TSE	28

Click **Smoothing – Exponential Smoothing** to run the technique a second time. Again select **CA** as the *Selected Variable* and **Produce forecast on validation**. However, this time, select **Optimize**, then click **OK**.



Exponential Smoothing

Data Source

Worksheet: TSPartition

Workbook: Income.xlsx

Data range: $C$36:$E$107

#Rows: 71

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data
Record ID

Time variable: Year

Selected variable: CA

Parameters

Weights

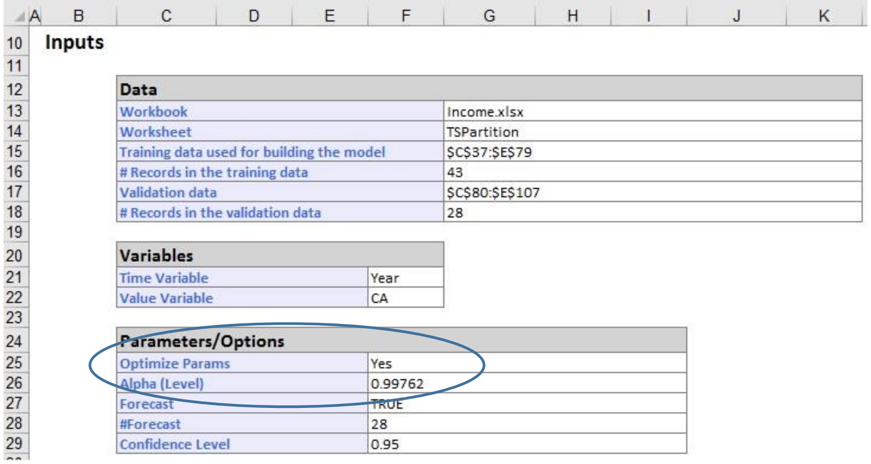
☑ Optimize

Level (Alpha): 0.2

Output Options

☑ Produce forecast on validation

Expo1 is inserted right of the Expo worksheet. Analytic Solver used an Alpha = 0.9976…



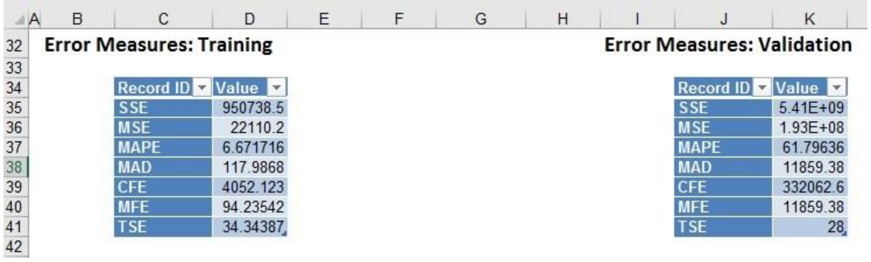
Inputs

Data	
Workbook	Income.xlsx
Worksheet	TSPartition
Training data used for building the model	$C$37:$E$79
# Records in the training data	43
Validation data	$C$80:$E$107
# Records in the validation data	28

Variables	
Time Variable	Year
Value Variable	CA

Parameters/Options	
Optimize Params	Yes
Alpha (Level)	0.99762
Forecast	TRUE
#Forecast	28
Confidence Level	0.95

which results in a MSE of 22,110.2 for the Training Set and a MSE of 1.93E08 for the Validation Set. Although an alpha of .9976 did result in lower values, the MSE in both the training and validation sets indicates the model is still not a good fit.

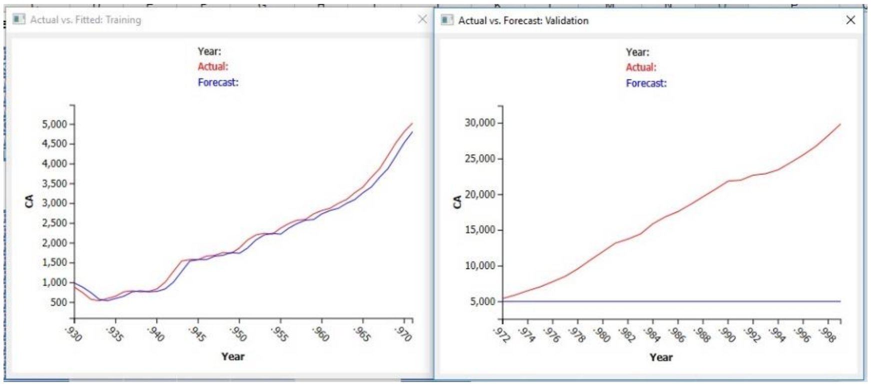


Error Measures: Training

Record ID	Value
SSE	950738.5
MSE	22110.2
MAPE	6.671716
MAD	117.9868
CFE	4052.123
MFE	94.23542
TSE	34.34387

Error Measures: Validation

Record ID	Value
SSE	5.41E+09
MSE	1.93E+08
MAPE	61.79636
MAD	11859.38
CFE	332062.6
MFE	11859.38
TSE	28

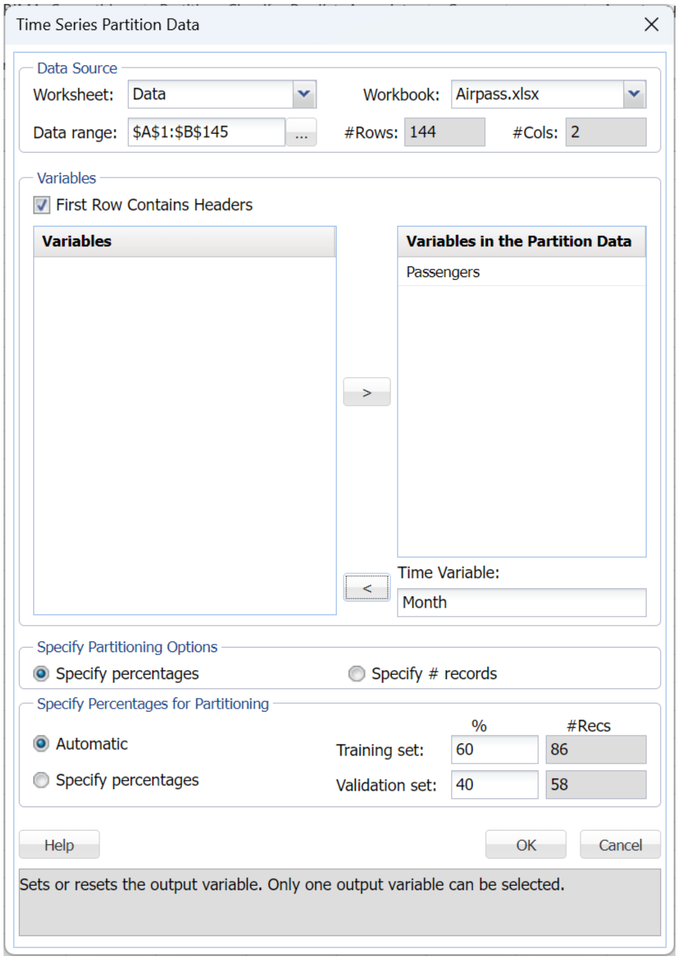


Note: Click the Charts icon on the Data Science Cloud Ribbon to view the charts shown above.

Moving Average Smoothing Example

This example illustrates how to use Analytic Solver's Moving Average Smoothing technique to uncover trends in the Airpass.xlsx time series dataset. Click **Help – Example Models --Forecasting/Data Science Examples** to open the dataset. Airpass.xlsx contains monthly totals of international airline passengers from 1949 - 1960.

Click **Partition** in the *Time Series* group on the Data Science ribbon to open the *Time Series Partition* dialog. *Select* **Month** as the *Time Variable.* Select **Passengers** as the *Variables in the partition data*. Then click **OK** to partition the data into training and validation sets. (Partitioning is optional. Smoothing techniques may be run on full unpartitioned datasets.)



Time Series Partition Data

Data Source

Worksheet: Data

Workbook: Airpass.xlsx

Data range: $A$1:$B$145

#Rows: 144

#Cols: 2

Variables

☑ First Row Contains Headers

Variables	Variables in the Partition Data
	Passengers

Time Variable: Month

Specify Partitioning Options

● Specify percentages ○ Specify # records

Specify Percentages for Partitioning

● Automatic

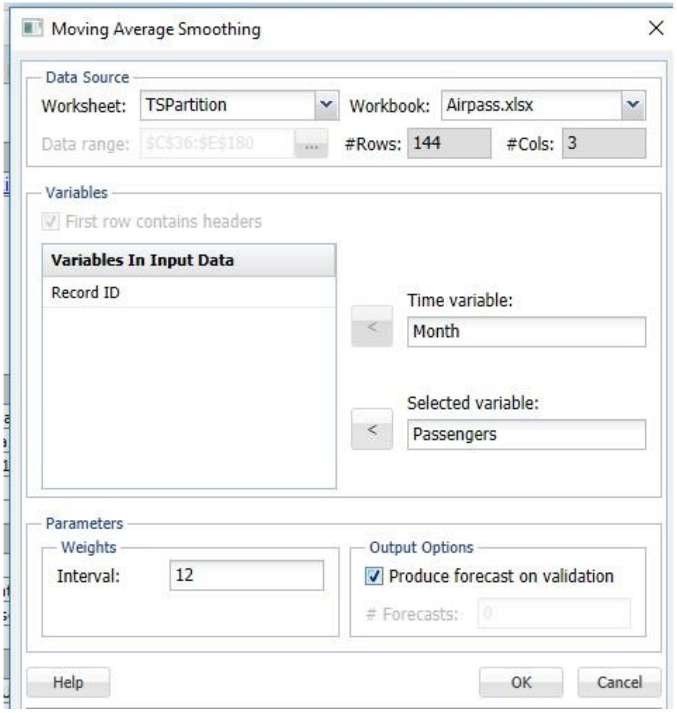
○ Specify percentages

	%	#Recs
Training set:	60	86
Validation set:	40	58

Sets or resets the output variable. Only one output variable can be selected.

The output sheet, *TSPartition*, will be inserted directly right of the Airpass sheet. Click **Smoothing – Moving Average** to open the *Moving Average Smoothing* dialog.

Select **Month** for *Time Variable* if not already selected. Select **Passengers** as the *Selected variable*. Since this dataset is expected to include some seasonality (i.e. airline passenger numbers increase during the holidays and summer months), the value for the *Interval* parameter should be the length of one seasonal cycle, i.e. 12 months. As a result, enter **12** for Interval. Select **Produce forecast on validation**.



Moving Average Smoothing

Data Source

Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data

Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Weights

Interval: 12

Output Options

☑ Produce forecast on validation

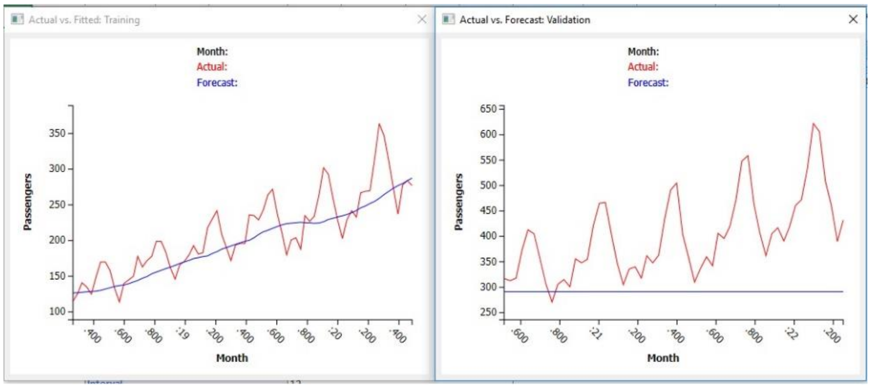
Forecasts: 0

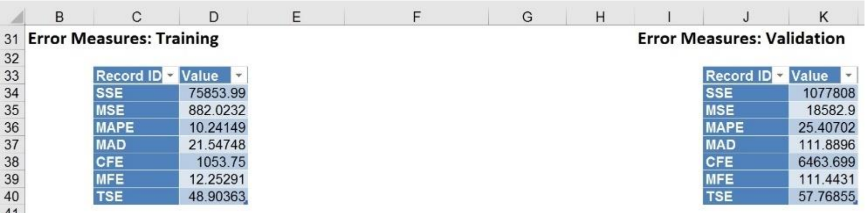
Afterwards, click **OK** to apply the smoothing technique to the partitioned dataset.

The report, *MovingAvg*, will be inserted directly right of TSPartition.

The Actual Vs. Fitted: Training and The Actual Vs. Forecast: Validation charts show that the moving average smoothing technique does not result in a good fit as the model does not effectively capture the seasonality in the dataset. The summer months where the number of airline passengers are typically high, appear to be under forecasted and the months where the number of airline passengers are low, the model results in a forecast that is too high. *A moving average forecast should never be used when the dataset includes seasonality*. An alternative would be to perform a regression on the model and then apply this technique to the residuals.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select MovingAvg for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.





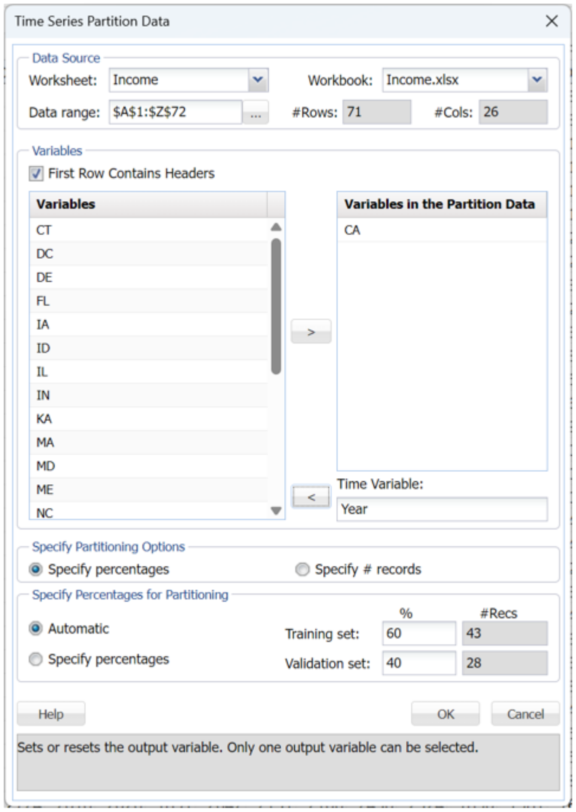
Error Measures: Training

Record ID	Value
SSE	75853.99
MSE	882.0232
MAPE	10.24149
MAD	21.54748
CFE	1053.75
MFE	12.25291
TSE	48.90363

Error Measures: Validation

Record ID	Value
SSE	1077808
MSE	18582.9
MAPE	25.40702
MAD	111.8896
CFE	6463.699
MFE	111.4431
TSE	57.76855

Now let’s take a look at an example that does not include seasonality. Open the example dataset **Income.xlsx**. This dataset contains the average income of tax payers by state. First partition the dataset into training and validation sets using **Year** as the *Time Variable* and **CA** as the *Variables in the partition data.*



Time Series Partition Data

Data Source

Worksheet: Income

Workbook: Income.xlsx

Data range: $A$1:$Z$72

#Rows: 71

#Cols: 26

Variables

☑ First Row Contains Headers

Variables
CT
DC
DE
FL
IA
ID
IL
IN
KA
MA
MD
ME
NC

Variables in the Partition Data
CA

Time Variable: Year

Specify Partitioning Options

● Specify percentages ○ Specify # records

Specify Percentages for Partitioning

● Automatic

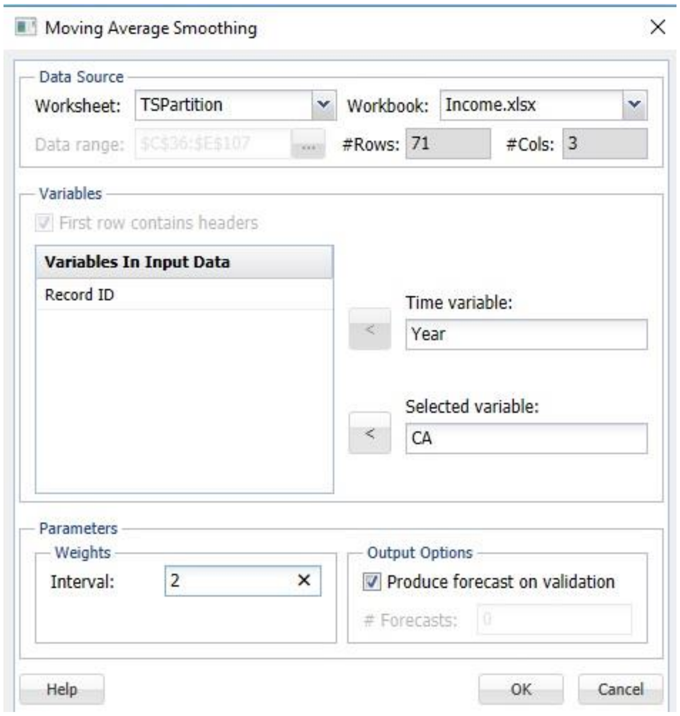
○ Specify percentages

	%	#Recs
Training set:	60	43
Validation set:	40	28

Sets or resets the output variable. Only one output variable can be selected.

Then click **OK** to accept the partitioning defaults and create the two partitions (Training and Validation). The output, *TSPartition*, will be inserted right of the Income sheet..

Click **Smoothing – Moving Average** from the Data Science ribbon to open the *Moving Average Smoothing* dialog. **Year** is already selected for *Time Variable.* Select **CA** as the *Selected variable* and *Produce forecast on validation* checkbox.



Moving Average Smoothing

Data Source

Worksheet: TSPartition Workbook: Income.xlsx

Data range: $C$36:$E$107 #Rows: 71 #Cols: 3

Variables

☑ First row contains headers

Variables In Input Data: Record ID

Time variable: Year

Selected variable: CA

Parameters

Weights

Interval: 2

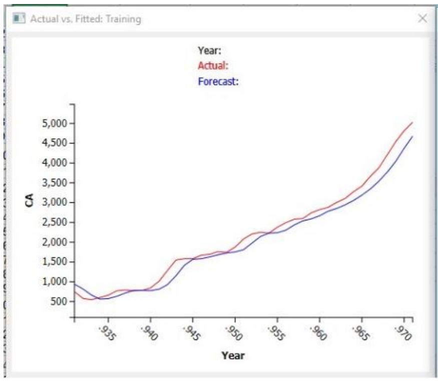
Output Options

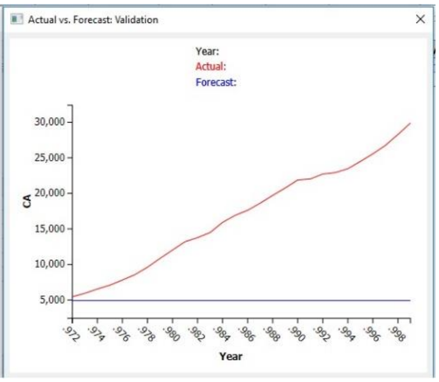
☑ Produce forecast on validation

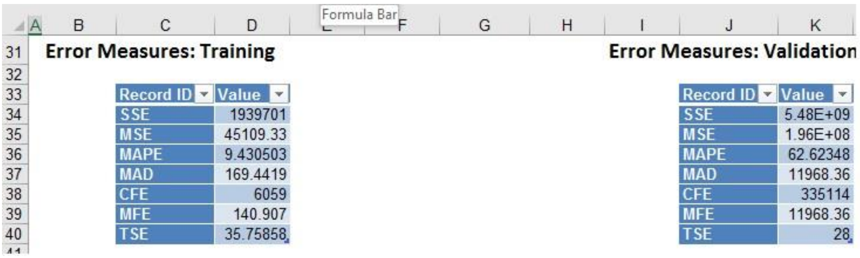
Forecasts: 0

Click **OK** to run the Moving Average Smoothing technique.

Click *MovingAvg*, inserted right of the TSPartition worksheet, to view the Actual vs Fitted: Training and Actual vs. Forecast: Validation charts and Error Measures.







Error Measures: Training

Record ID	Value
SSE	1939701
MSE	45109.33
MAPE	9.430503
MAD	169.4419
CFE	6059
MFE	140.907
TSE	35.75858

Error Measures: Validation

Record ID	Value
SSE	5.48E+09
MSE	1.96E+08
MAPE	62.62348
MAD	11968.36
CFE	335114
MFE	11968.36
TSE	28

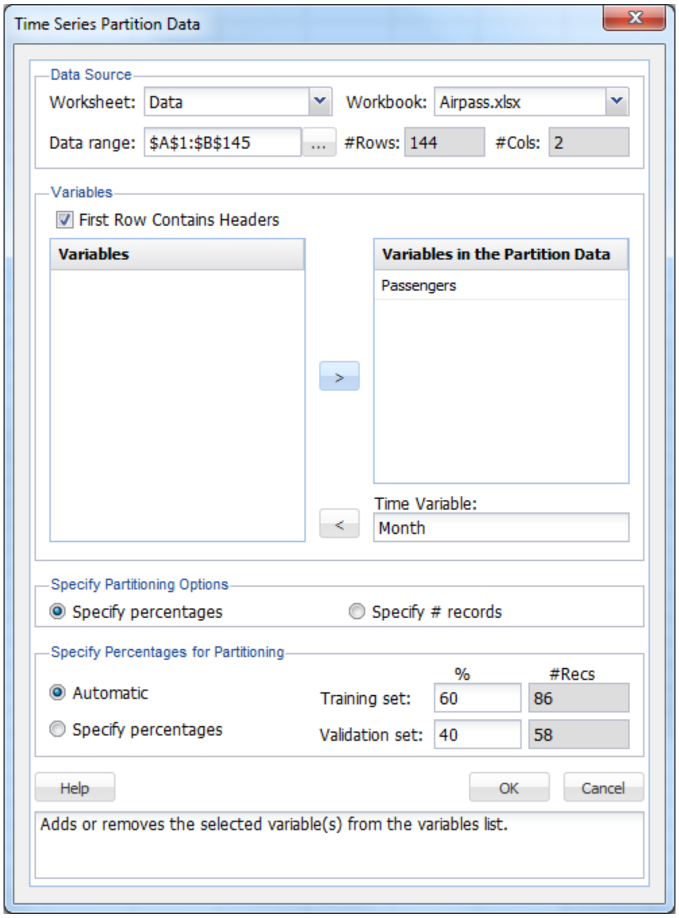
Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select MovingAvg for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.

MovingAvg_Stored is available for scoring new data. Please see the "Scoring New Data" chapter within the Analytic Solver Data Science User Guide for more information on scoring new data using a stored model sheet.

Double Exponential Smoothing Example

This example illustrates how to use Analytic Solver's Double Exponential Smoothing technique to uncover seasonality trends in the Airpass.xlsx time series dataset. (See the examples above for instructions on how to open this example.)

Click **Partition** in the *Time Series* group on the Data Science ribbon to open the *Time Series Partition* dialog. Select **Month** as the *Time Variable.* Select **Passengers** as the *Variables in the Partition Data.*



Time Series Partition Data

Data Source

Worksheet: Data
Workbook: Airpass.xlsx
Data range: $A$1:$B$145
#Rows: 144
#Cols: 2

Variables

☑ First Row Contains Headers

Variables	Variables in the Partition Data
	Passengers

Time Variable: Month

Specify Partitioning Options

● Specify percentages ○ Specify # records

Specify Percentages for Partitioning

● Automatic
○ Specify percentages

	%	#Recs
Training set:	60	86
Validation set:	40	58

Adds or removes the selected variable(s) from the variables list.

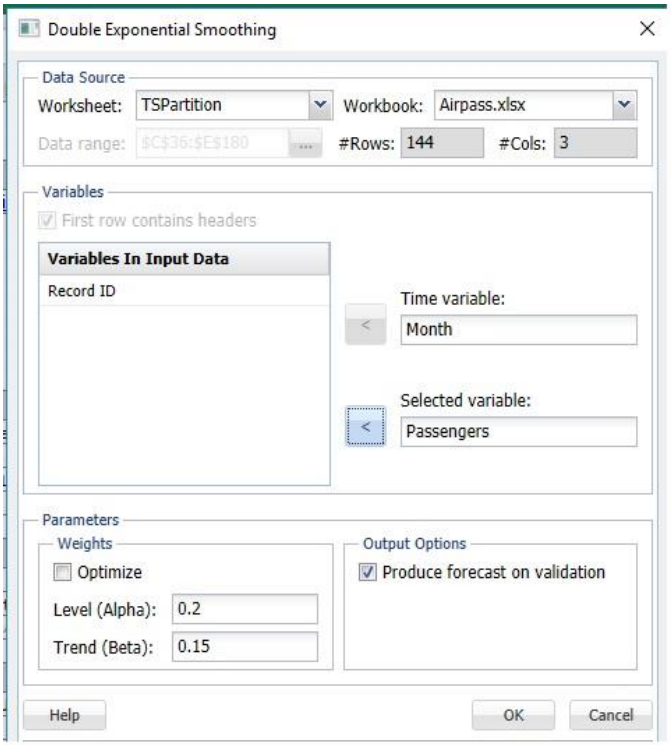
Then click **OK** to partition the data into training and validation sets. *TSPartition* will be inserted right of the Airpass worksheet.

Click **Smoothing – Double Exponential** to open the *Double Exponential Smoothing* dialog.

Select **Month** as the *Time Variable,* if not already selected. Select **Passengers** as the *Selected variable*, then check **Produce Forecast on validation** to test the forecast on the validation set.

This example uses the defaults for both the Alpha and Trend parameters. However, Analytic Solver Data Science includes a feature that will choose the

Alpha and Trend parameter values that result in the minimum residual mean squared error. It is recommended that this feature be used carefully as this feature most often leads to a model that is overfit to the training set. An overfit model rarely exhibits high predictive accuracy in the validation set.



Double Exponential Smoothing

Data Source

Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data

Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Weights

□ Optimize

Level (Alpha): 0.2

Trend (Beta): 0.15

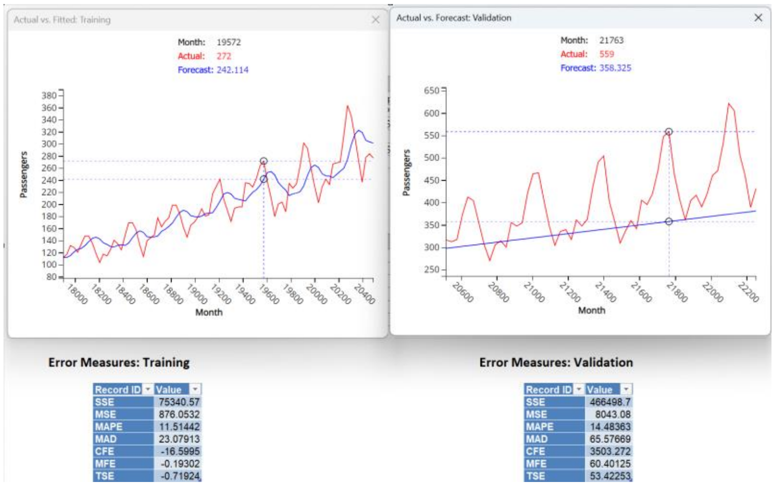
Output Options

☑ Produce forecast on validation

Click **OK** to run the Double Exponential Smoothing algorithm. The output, *DoubleExpo* and *DoubleExpo_Stored,* will be inserted right of the TSPartition worksheet.

Click on the *DoubleExpo* tab to view the results of the smoothing. Click on any point on either graph to see the Actual vs. Forecast results at the top of the chart.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select **DoubleExp** for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart.*



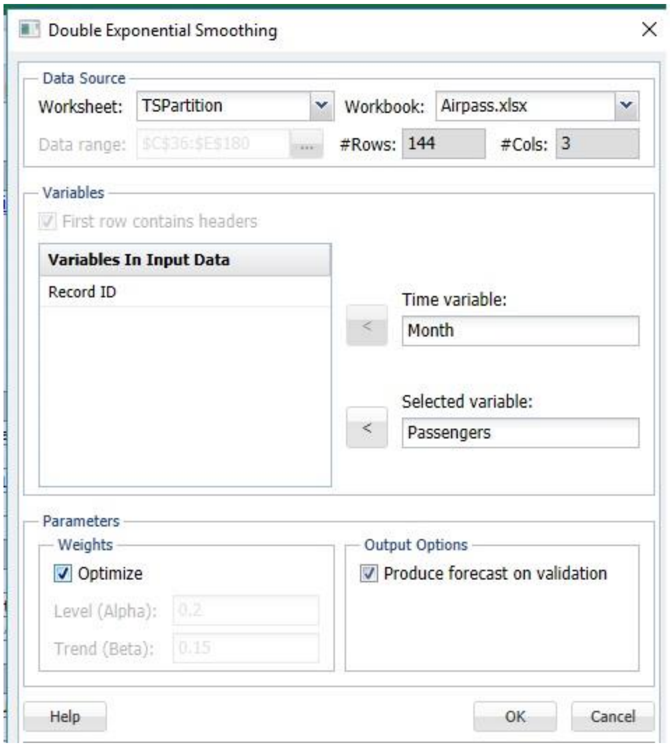
Error Measures: Training

Record ID	Value
SSE	75340.57
MSE	876.0532
MAPE	11.51442
MAD	23.07913
CFE	-16.5995
MFE	-0.19302
TSE	-0.71924

Error Measures: Validation

Record ID	Value
SSE	466498.7
MSE	8043.08
MAPE	14.48363
MAD	65.57669
CFE	3503.272
MFE	60.40125
TSE	53.42253

If instead, the *Optimize* feature is used ….



Double Exponential Smoothing

Data Source

Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data
Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Weights

☑ Optimize

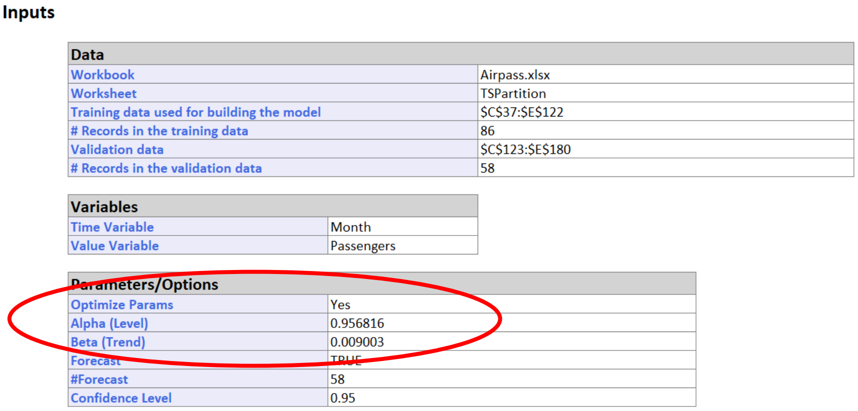
Level (Alpha): 0.2

Trend (Beta): 0.15

Output Options

☑ Produce forecast on validation

... an Alpha of 0.9568 is chosen along with a Trend of 0.009.



Inputs

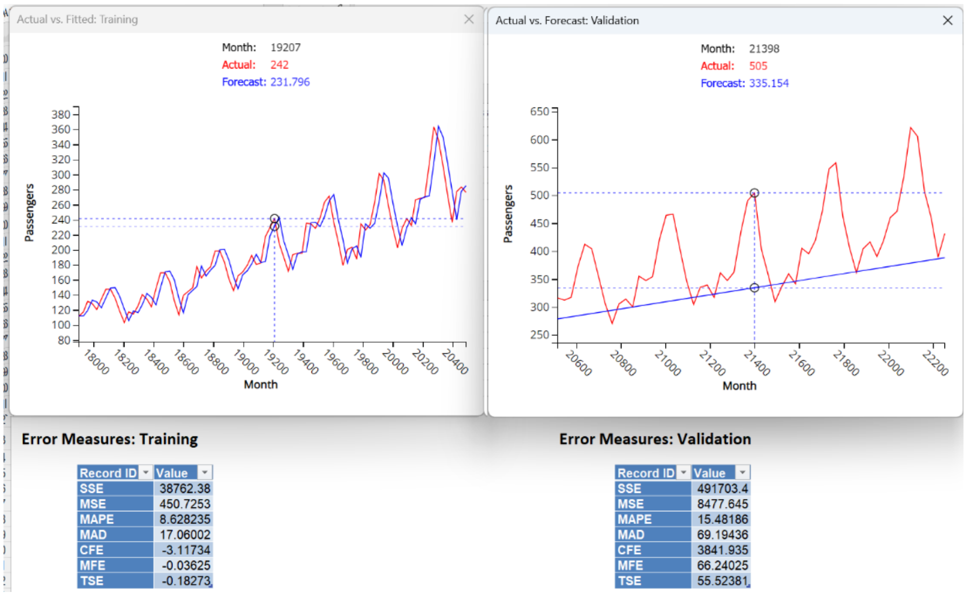
Data	
Workbook	Airpass.xlsx
Worksheet	TSPartition
Training data used for building the model	$C$37:$E$122
# Records in the training data	86
Validation data	$C$123:$E$180
# Records in the validation data	58

Variables	
Time Variable	Month
Value Variable	Passengers

Parameters/Options	
Optimize Params	Yes
Alpha (Level)	0.956816
Beta (Trend)	0.009003
Forecast	TRUE
#Forecast	58
Confidence Level	0.95

These parameters result in a MSE of 450.7 for the Training set and a MSE of 8477.64 for the Validation Set. Again the model created with the parameters from the Optimize algorithm appear to result in a model with a better fit than a model created with the default parameters.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select **DoubleExp1** for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.



Error Measures: Training

Record ID	Value
SSE	38762.38
MSE	450.7253
MAPE	8.628235
MAD	17.06002
CFE	-3.11734
MFE	-0.03625
TSE	-0.18273

Error Measures: Validation

Record ID	Value
SSE	491703.4
MSE	8477.645
MAPE	15.48186
MAD	69.19436
CFE	3841.935
MFE	66.24025
TSE	55.52381

Holt Winters Smoothing Example

This example illustrates how to use Analytic Solver's Holt Winters Smoothing technique to uncover trends in the time series dataset Airpass.xlsx. This example will create three different forecasts, one for each Holt Winters model type, beginning with Multiplicative.

Click back to the TSPartition worksheet and click **Smoothing – Holt Winters** – **Multiplicative** to open the *Holt Winters Smoothing (Multiplicative Model)* dialog.

Select Month for the Time variable, if not already selected, and Passengers for Selected variable.

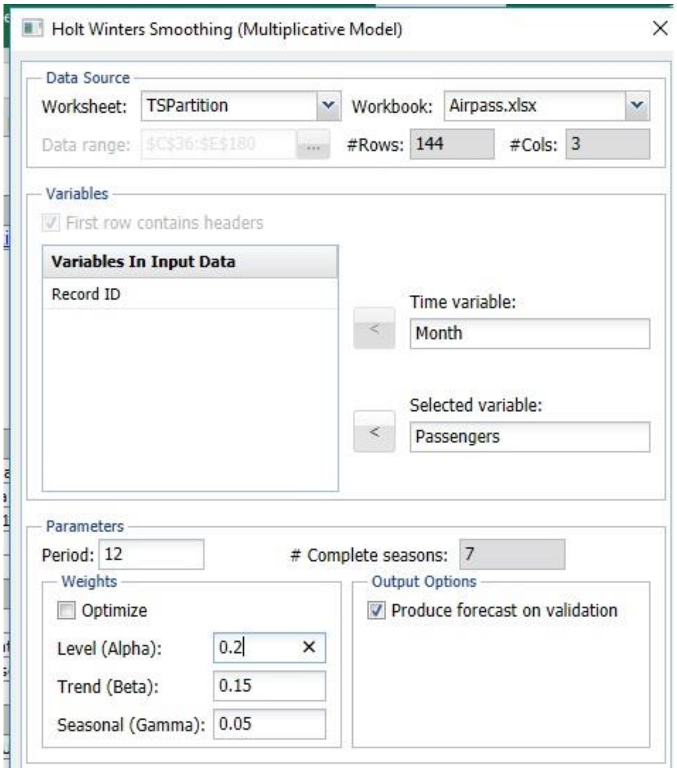
Since our dataset contains airline passengers, we can assume some seasonality exists in the data since most passengers fly during the summer and holiday months (i.e. December).

It takes a full 12 months to complete the seasonality cycle so enter **12** for *Period*, *# Complete seasons* is automatically entered with the number 7. This example will use the defaults for the three parameters: *Alpha*, *Beta*, and *Gamma*.

Values between 0 and 1 can be entered for each parameter. As with Exponential Smoothing, values close to 1 will result in the most recent observations being weighted more than earlier observations.

In the Multiplicative model, it is assumed that the values for the different seasons differ by percentage amounts.

Produce Forecast on validation is selected by default.



Holt Winters Smoothing (Multiplicative Model)

Data Source

Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data

Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Period: 12

Complete seasons: 7

Weights

☐ Optimize

Level (Alpha): 0.2

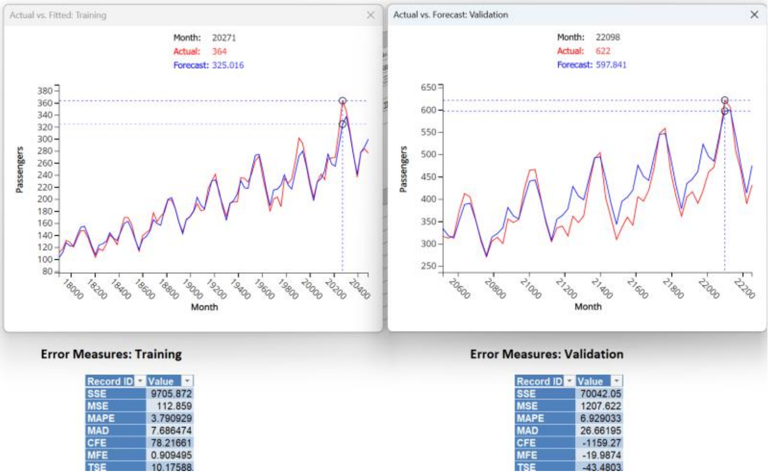
Trend (Beta): 0.15

Seasonal (Gamma): 0.05

Output Options

☑ Produce forecast on validation

Click **OK** to run the smoothing technique. The results of the smoothing technique, *MulHoltWinters* and *MulHoltWinters_Stored*, are inserted to the right of the TSPartition worksheet. For more information on *MulHoltWinters_Stored*, see the chapter “Scoring New Data” within the Analytic Solver User Guide.



Error Measures: Training

Record ID	Value
SSE	9705.872
MSE	112.859
MAPE	3.790929
MAD	7.686474
CFE	78.21661
MFE	0.909495
TSE	10.17588

Error Measures: Validation

Record ID	Value
SSE	70042.05
MSE	1207.622
MAPE	6.929033
MAD	26.66195
CFE	-1159.27
MFE	-19.9874
TSE	-43.4803

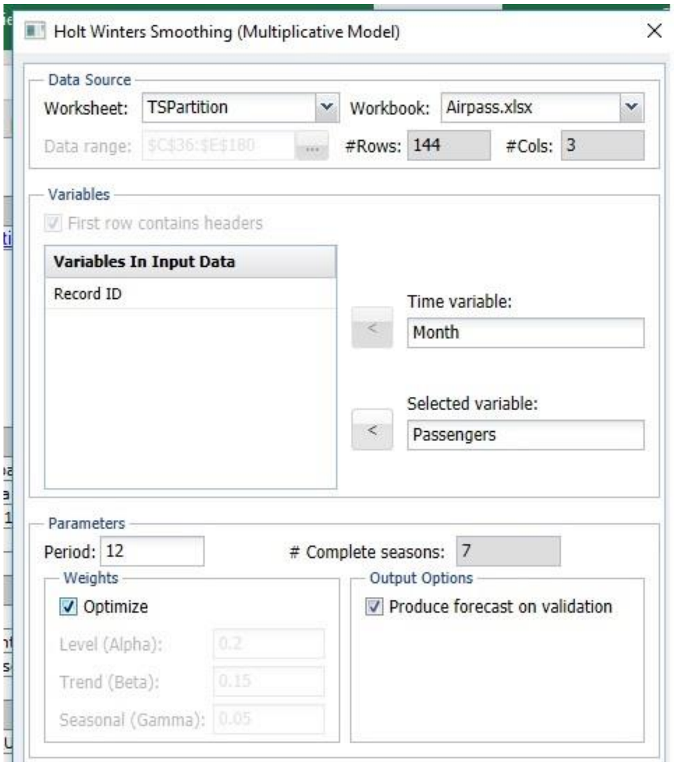
Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select MulHoltWinters for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.

If you inspect the MSE (Mean Squared Error) term in the Error Measures (Validation) table, you’ll see that this value is fairly high. In addition, the peaks for the Forecast data appear to lag behind the peaks in the Validation data. This suggests that our *Trend (Beta)* parameter is too large.

Let’s go back and try the Multiplicative method one more time using the Optimize parameter. This parameter will choose the best values for the Alpha, Trend, and Seasonal parameters based on the Forecasting Mean Squared Error. It is recommended that this feature be used carefully as this option can lead to overfitting. An overfit model rarely exhibits high predictive accuracy in the validation set.

Click **Smoothing – Holt-Winters – Multiplicative** on the Data Science ribbon.

We will again select **Passengers** for *Selected Variable* (if not already selected) and **12** for *Period.* **Produce Forecast on Validation** is selected by default. This time we'll also select the Optimize parameter.



Holt Winters Smoothing (Multiplicative Model)

Data Source

Worksheet: TSPartition   Workbook: Airpass.xlsx

Data range: $C$36:$E$180   #Rows: 144   #Cols: 3

Variables

☑ First row contains headers

Variables In Input Data: Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Period: 12   # Complete seasons: 7

Weights

☑ Optimize

Level (Alpha): 0.2

Trend (Beta): 0.15

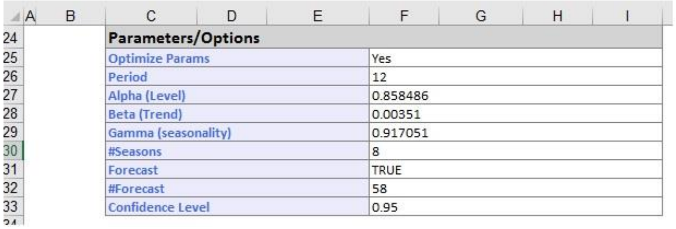
Seasonal (Gamma): 0.05

Output Options

☑ Produce forecast on validation

Afterwards, click **OK** to proceed with the smoothing technique.

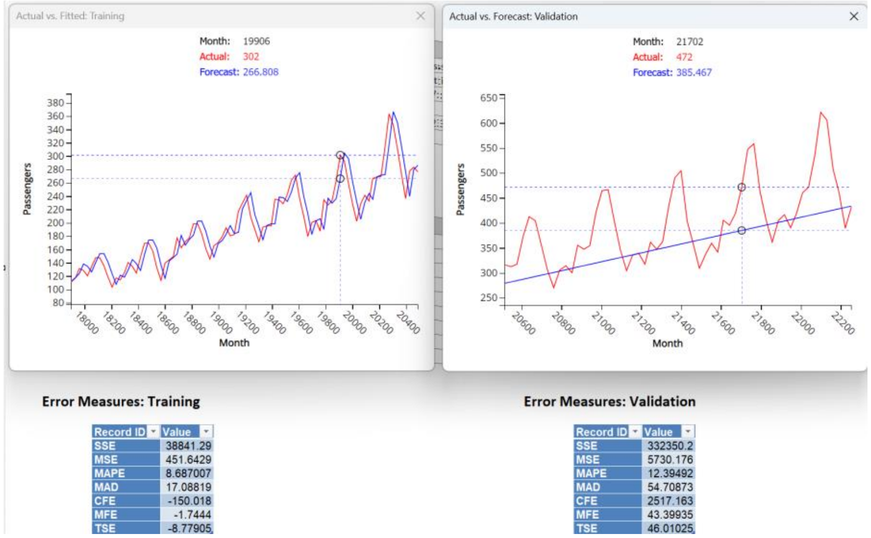
MulHoltWinters1 is inserted to the right.



Parameters/Options	
Optimize Params	Yes
Period	12
Alpha (Level)	0.858486
Beta (Trend)	0.00351
Gamma (seasonality)	0.917051
#Seasons	8
Forecast	TRUE
#Forecast	58
Confidence Level	0.95

The Parameters/Options table gives us the parameter settings as chosen by the Optimize feature for Alpha (0.858), Beta (0.003) and Gamma (0.917). (Recall that the default settings as 0.20 (Alpha), 0.15 (Beta) and 0.05 (Seasonal).) Scroll down to find the Training and Validation Error Measures.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select MulHoltWinters1 for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.



Error Measures: Training

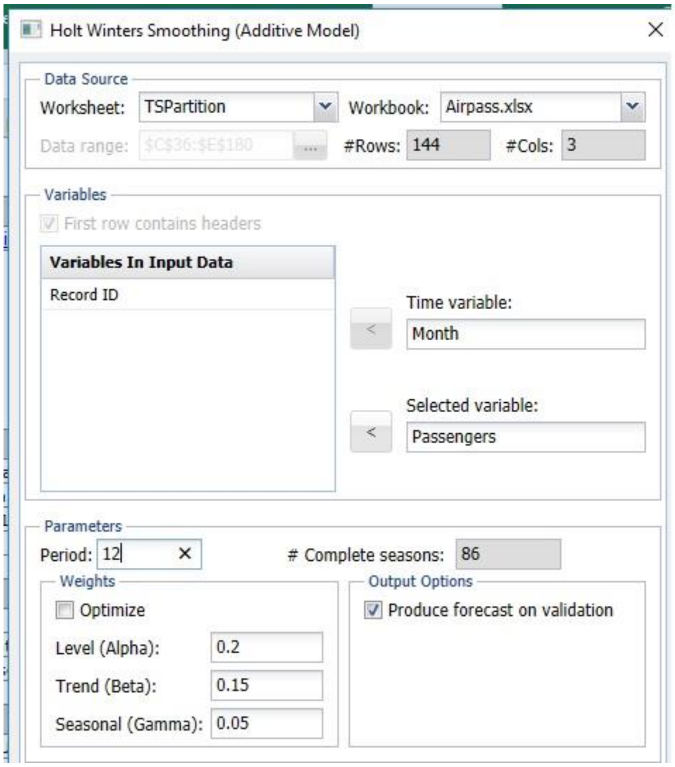
Record ID	Value
SSE	38841.29
MSE	451.6429
MAPE	8.687007
MAD	17.08819
CFE	-150.018
MFE	-1.7444
TSE	-8.77905

Error Measures: Validation

Record ID	Value
SSE	332350.2
MSE	5730.176
MAPE	12.39492
MAD	54.70873
CFE	2517.163
MFE	43.39935
TSE	46.01025

Now let's create a new model using the Additive model. This technique assumes the values for the different seasons differ by a constant amount. Click back to the *TSPartition* sheet, then click **Smoothing – Holt Winters – Additive** to open the *Holt Winters Smoothing (Additive Model)* dialog.

Select *Month* for Time Variable, if not already selected, and **Passengers** for *Selected variable*. Again, enter **12** for *Period*. **Produce Forecast on validation** is selected by default.



Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

☑ First row contains headers

Record ID

Time variable: Month

Selected variable: Passengers

Period: 12

Complete seasons: 86

☐ Optimize

Level (Alpha): 0.2

Trend (Beta): 0.15

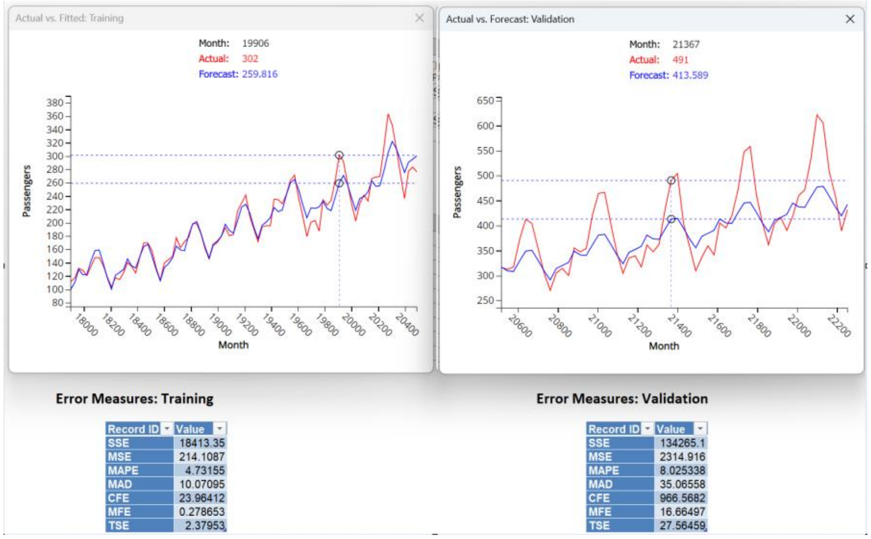
Seasonal (Gamma): 0.05

☑ Produce forecast on validation

Click **OK** to run the smoothing technique. *AddHoltWinters* and *AddHoltWinters_Stored* will be inserted right of the TSPartition worksheet. For more information on *AddHoltWinters_Stored*, see the chapter, "Scoring New Data" within the Analytic Solver User Guide.

Click a point anywhere on the function to compare Actual vs. Forecasted at the top of the graph.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select AddHoltWinters for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.



Error Measures: Training

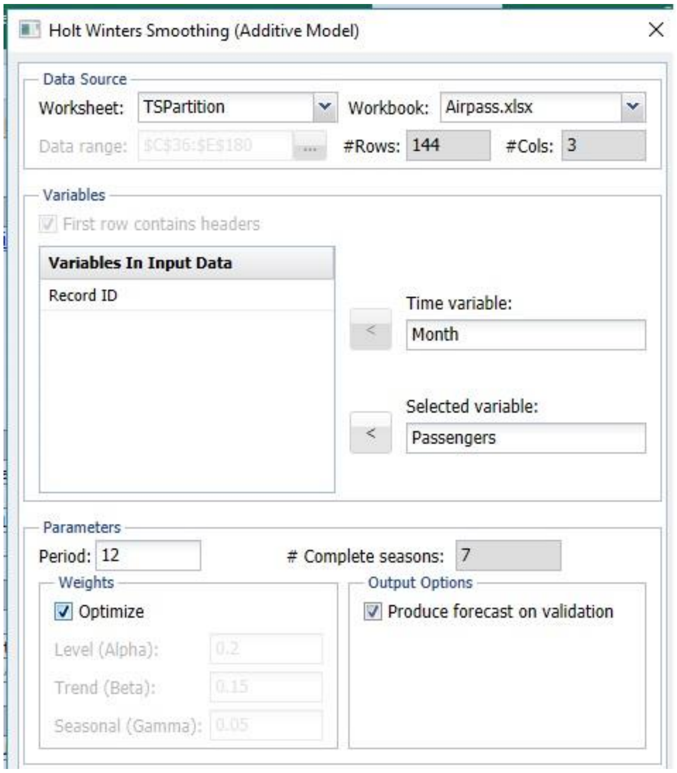
Record ID	Value
SSE	18413.35
MSE	214.1087
MAPE	4.73155
MAD	10.07095
CFE	23.96412
MFE	0.278653
TSE	2.37953

Error Measures: Validation

Record ID	Value
SSE	134265.1
MSE	2314.916
MAPE	8.025338
MAD	35.06558
CFE	966.5682
MFE	16.66497
TSE	27.56459

Let's try the Additive model again using the Optimize feature. Click back to *TSPartition* and then click **Smoothing – Holt-Winters – Additive** on the Data Science ribbon.

Select **Month** for *Time variable* and Select **Passengers** for *Selected variable,* then **12** for *Period.* **Produce Forecast on Validation** is selected by default. Select **Optimize** to run the Optimize algorithm which will pick the best values for the three parameters, Alpha, Beta, and Gamma.



Holt Winters Smoothing (Additive Model)

Data Source

Worksheet: TSPartition Workbook: Airpass.xlsx

Data range: $C$36:$E$180 #Rows: 144 #Cols: 3

Variables

☑ First row contains headers

Variables In Input Data

Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Period: 12 # Complete seasons: 7

Weights

☑ Optimize

Level (Alpha): 0.2

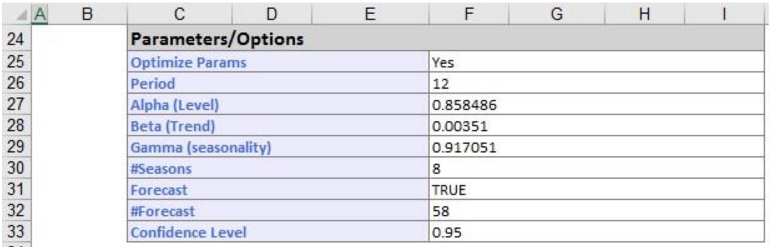
Trend (Beta): 0.15

Seasonal (Gamma): 0.05

Output Options

☑ Produce forecast on validation

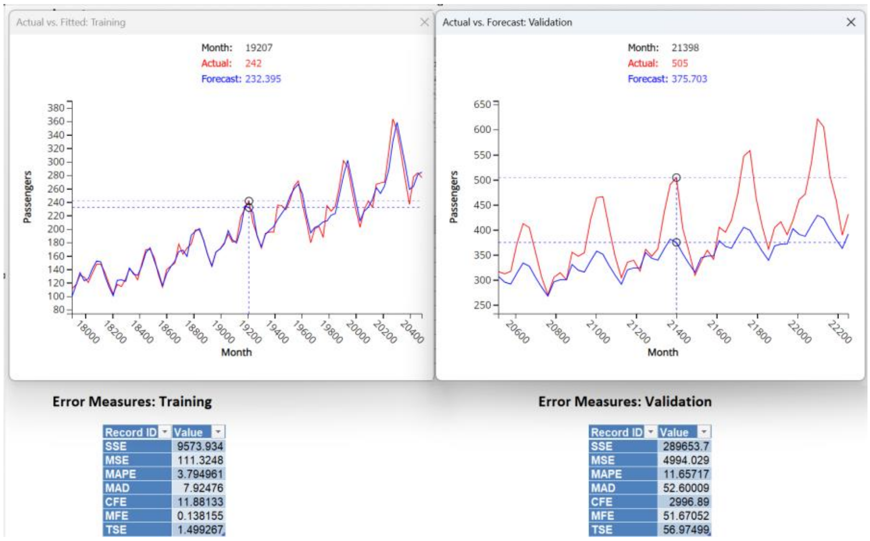
Click **OK**, then click the, *AddHoltWinters1*, tab.



Parameters/Options	
Optimize Params	Yes
Period	12
Alpha (Level)	0.858486
Beta (Trend)	0.00351
Gamma (seasonality)	0.917051
#Seasons	8
Forecast	TRUE
#Forecast	58
Confidence Level	0.95

Notice the parameter values chosen by the Optimize algorithm were 0.858 for Alpha, .00351 for Beta, and 0.917 for Gamma. Scroll down to view the results of the model fitting.

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select AddHoltWinters1 for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.



Error Measures: Training

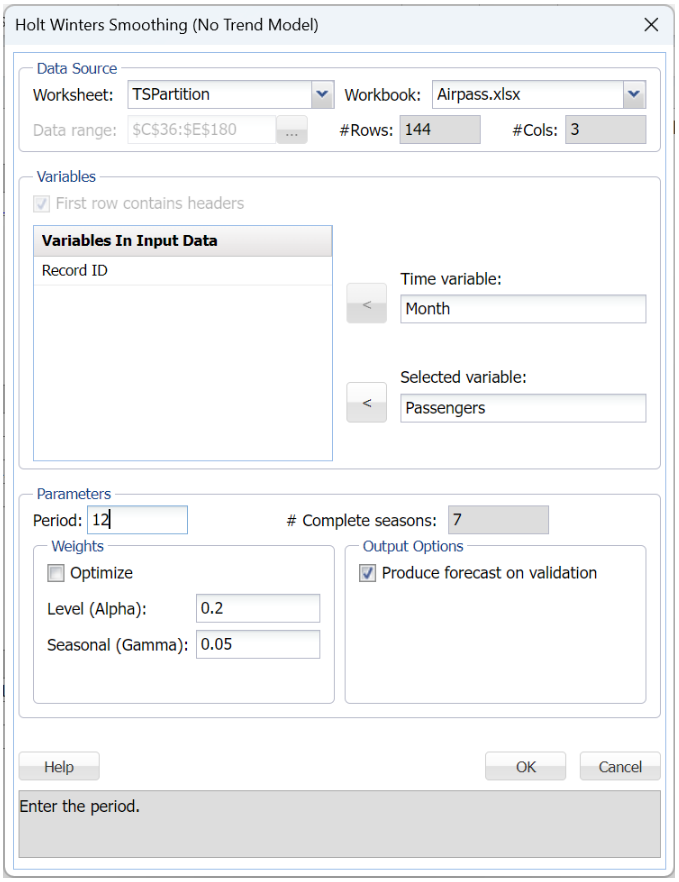
Record ID	Value
SSE	9573.934
MSE	111.3248
MAPE	3.794961
MAD	7.92476
CFE	11.88133
MFE	0.138155
TSE	1.499267

Error Measures: Validation

Record ID	Value
SSE	289653.7
MSE	4994.029
MAPE	11.65717
MAD	52.60009
CFE	2996.89
MFE	51.67052
TSE	56.97499

The last Holt Winters model should be used with time series that contain seasonality, but no trends. Click back to *TSPartition*, then click **Smoothing – Holt Winters – No Trend** to open the *Holt Winters Smoothing (No trend Model)* dialog.

Select *Month* for *Time Variable* unless already selected. Select **Passengers** as the *Selected variable*. Enter **12** for *Period*. **Produce Forecast on validation** is selected by default. Notice that the trend parameter is missing. Values for *Alpha* and *Gamma* can range from 0 to 1. A value of 1 for each parameter will assign higher weights to the most recent observations and lower weights to the earlier observations. This example will accept the default values.



Holt Winters Smoothing (No Trend Model)

Data Source

Worksheet: TSPartition

Workbook: Airpass.xlsx

Data range: $C$36:$E$180

#Rows: 144

#Cols: 3

Variables

☑ First row contains headers

Variables In Input Data
Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Period: 12

Complete seasons: 7

Weights

□ Optimize

Level (Alpha): 0.2

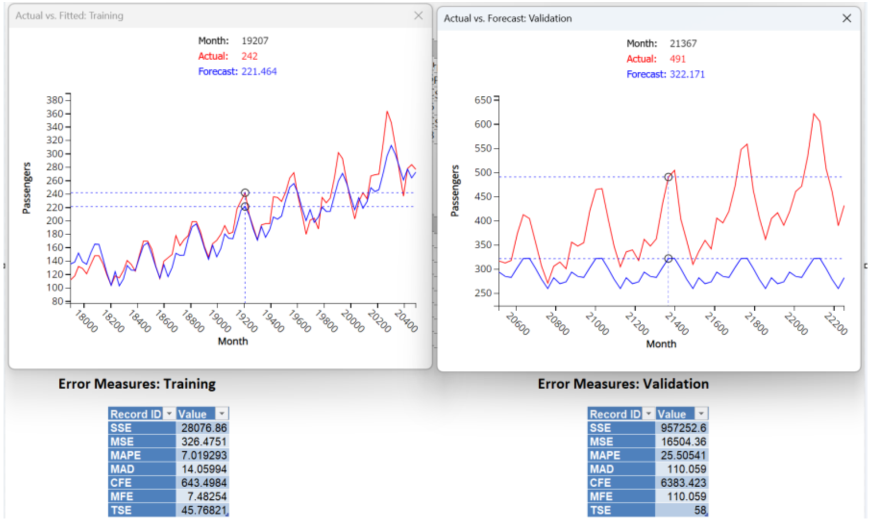
Seasonal (Gamma): 0.05

Output Options

☑ Produce forecast on validation

Enter the period.

Click **OK** to run the smoothing technique. *NoTrendHoltWinters* and *NoTrendHoltWinters_Stored* are right of the TSPartition worksheet.



Error Measures: Training

Record ID	Value
SSE	28076.86
MSE	326.4751
MAPE	7.019293
MAD	14.05994
CFE	643.4984
MFE	7.48254
TSE	45.76821

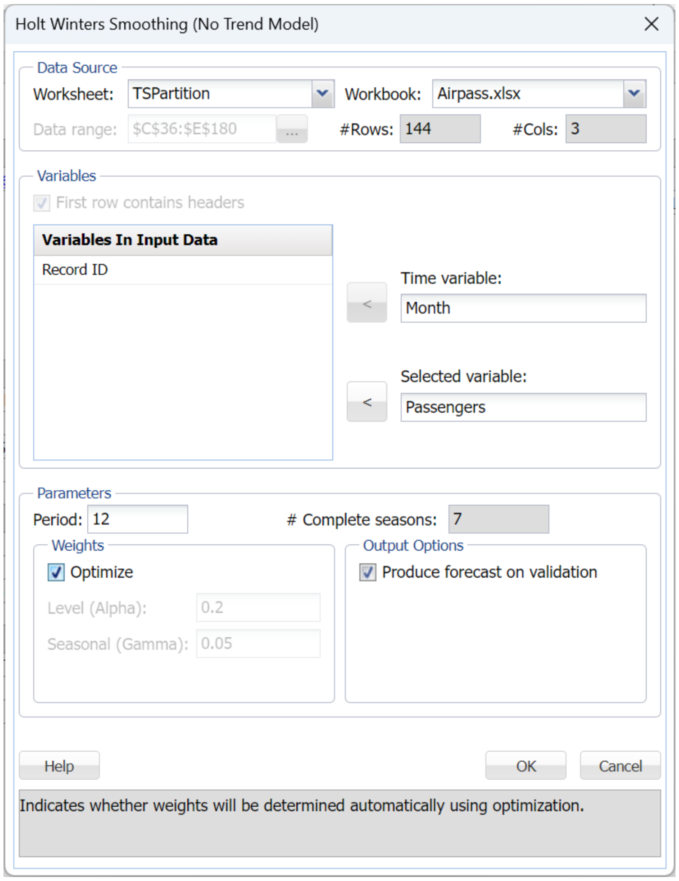
Error Measures: Validation

Record ID	Value
SSE	957252.6
MSE	16504.36
MAPE	25.50541
MAD	110.059
CFE	6383.423
MFE	110.059
TSE	58

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select NoTrendHoltWinters for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.

Let's try the No Trend model again using the Optimize feature. Click back to *TSPartition*, then click **Smoothing – Holt-Winters – No Trend** on the Data Science ribbon.

Select **Month** for *Time variable*, unless already selected, and **Passengers** for *Selected Variable,* **12** for *Period.* **Produce Forecast on Validation** is selected by default. Select **Optimize** to run the Optimize algorithm which will pick the best values for the two parameters, Alpha and Gamma.



Holt Winters Smoothing (No Trend Model)

Data Source

Worksheet: TSPartition Workbook: Airpass.xlsx

Data range: $C$36:$E$180 #Rows: 144 #Cols: 3

Variables

☑ First row contains headers

Variables In Input Data
Record ID

Time variable: Month

Selected variable: Passengers

Parameters

Period: 12 # Complete seasons: 7

Weights

☑ Optimize

Level (Alpha): 0.2

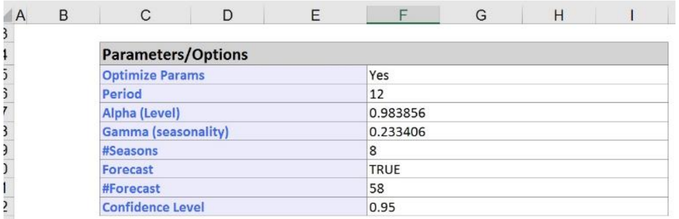
Seasonal (Gamma): 0.05

Output Options

☑ Produce forecast on validation

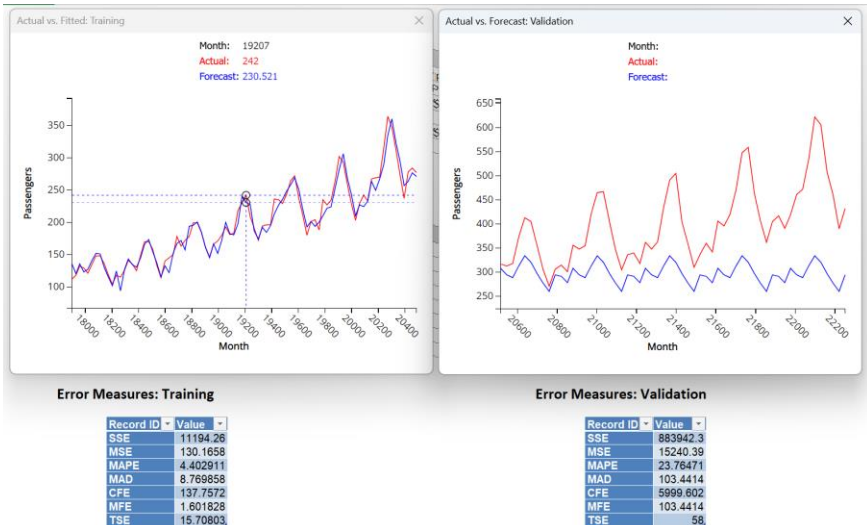
Indicates whether weights will be determined automatically using optimization.

Click **OK**. *NoTrendHoltWinters1* is inserted to the right.



Parameters/Options	
Optimize Params	Yes
Period	12
Alpha (Level)	0.983856
Gamma (seasonality)	0.233406
#Seasons	8
Forecast	TRUE
#Forecast	58
Confidence Level	0.95

Notice the parameter values chosen by the Optimize algorithm were 0.984 for Alpha and 0.233 for Gamma. Scroll down to view the results of the model fitting.



Error Measures: Training

Record ID	Value
SSE	11194.26
MSE	130.1658
MAPE	4.402911
MAD	8.769858
CFE	137.7572
MFE	1.601828
TSE	15.70803

Error Measures: Validation

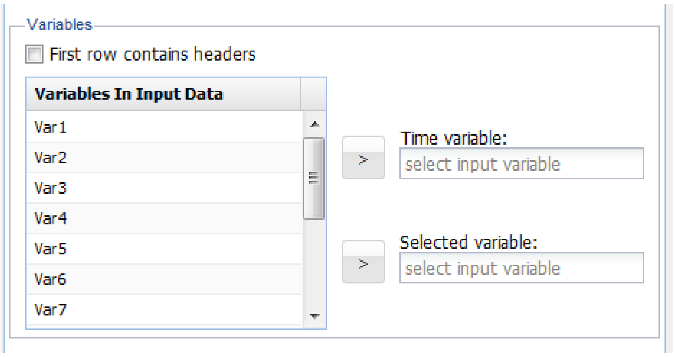
Record ID	Value
SSE	883942.3
MSE	15240.39
MAPE	23.76471
MAD	103.4414
CFE	5999.602
MFE	103.4414
TSE	58

Note: To view these two charts in the Cloud app, click the Charts icon on the Ribbon, select NoTrendHoltWinters for *Worksheet* and **Time Series Training Data** or **Time Series Validation Data** for *Chart*.

Common Smoothing Options

Common Options

The following options are common to each of the Smoothing techniques.



First row contains headers

When this option is selected, variables will be listed in the *Variables in input data* list box according to the first row in the dataset. If this option is not checked, variables will appear as *VarX* where X = 1,2,3,4, etc.

Variables in Input Data

All variables in the dataset will be listed here.

Time variable

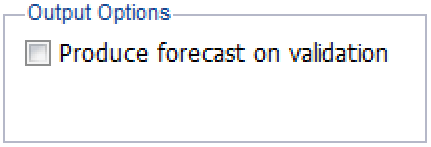
Select a variable associated with time from the Variables in input data list box.

Selected variable

Select a variable to apply the smoothing technique.

Output Options

If applying this smoothing technique to partitioned data, the option *Produce forecast on validation* will appear. Otherwise, the option *Produce forecast* will appear. If selected, Analytic Solver Data Science will include a forecast on the output results.

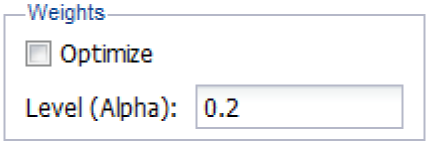


Output Options

☐ Produce forecast on validation

Exponential Smoothing Options

This section explains the options included in the *Weights* section on the *Exponential Smoothing* dialog.



Weights

☐ Optimize

Level (Alpha): 0.2

Optimize

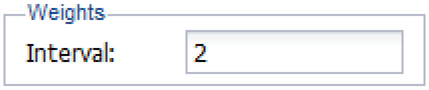
Select this option if you want to select the Alpha Level that minimizes the residual mean squared errors in the training and validation sets. Take care when using this feature as this option can result in an over fitted model. This option is not selected by default.

Level (Alpha)

Enter the smoothing parameter here. This parameter is used in the weighted average calculation and can be from 0 to 1. A value of 1 or close to 1 will result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights. A value of 0 or close to 0 will result in the most recent observations being assigned the smallest weights and the earliest observations being assigned the largest weights. The default value is 0.2.

Moving Average Smoothing Options

The section describes the options included in the Weights section of the Moving Average Smoothing dialog.



Weights

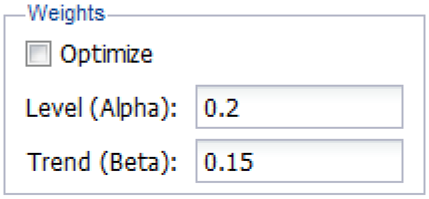
Interval: 2

Interval

Enter the window width of the moving average here. This parameter accepts a value of 1 up to N -1(where N is the number of observations in the dataset). If a value of 5 is entered for the Interval, then Analytic Solver Data Science will use the average of the last five observations for the last smoothed point or $F_{t = (}Y_t + Y_{t-1} + Y_{t-2} + Y_{t-3} + Y_{t-4}) / 5$. The default value is 2.

Double Exponential Smoothing Options

This section describes the options appearing in the Weights section on the Double Exponential Smoothing dialog.



Weights

□ Optimize

Level (Alpha): 0.2

Trend (Beta): 0.15

Optimize

Select this option to select the Alpha and Beta values that minimize the residual mean squared errors in the training and validation sets. Take care when using this feature as this option can result in an over fitted model. This option is not selected by default.

Level (Alpha)

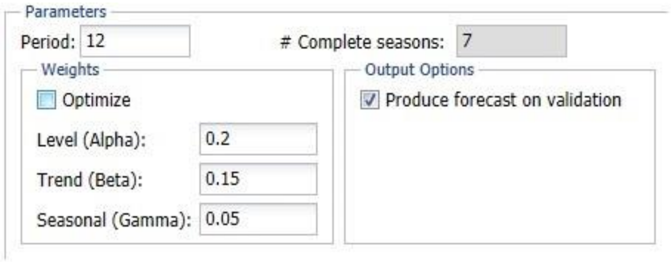
Enter the smoothing parameter here. This parameter is used in the weighted average calculation and can be from 0 to 1. A value of 1 or close to 1 will result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights in the weighted average calculation. A value of 0 or close to 0 will result in the most recent observations being assigned the smallest weights and the earliest observations being assigned the largest weights in the weighted average calculation. The default is 0.2.

Trend (Beta)

The Double Exponential Smoothing technique includes an additional parameter, Beta, to contend with trends in the data. This parameter is also used in the weighted average calculation and can be from 0 to 1. A value of 1 or close to 1 will result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights in the weighted average calculation. A value of 0 or close to 0 will result in the most recent observations being assigned the smallest weights and the earliest observations being assigned the largest weights in the weighted average calculation. The default is 0.15.

Holt Winter Smoothing Options

The options in this section appear in the Parameters and Weights section of the Holt Winters Smoothing dialogs.



Parameters

Period: 12

Complete seasons: 7

Weights

☐ Optimize

Level (Alpha): 0.2

Trend (Beta): 0.15

Seasonal (Gamma): 0.05

Output Options

☑ Produce forecast on validation

Period

Enter the number of periods that make up one season. The value for *# Complete seasons* will be automatically filled.

Optimize

Select this option to select the Alpha, Beta, and Gamma values that minimize the residual mean squared errors in the training and validation sets. Take care when using this feature as this option can result in an over fitted model. This option is not selected by default.

Level (Alpha)

Enter the smoothing parameter here. This parameter is used in the weighted average calculation and can be from 0 to 1. A value of 1 or close to 1 will result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights in the weighted average calculation. A value of 0 or close to 0 will result in the most recent observations being assigned the smallest weights and the earliest observations being assigned the largest weights in the weighted average calculation. The default is 0.2.

Trend (Beta)

The Holt Winters Smoothing also utilizes the Trend parameter, Beta, to contend with trends in the data. This parameter is also used in the weighted average calculation and can be from 0 to 1. A value of 1 or close to 1 will result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights in the weighted average calculation. A value of 0 or close to 0 will result in the most recent observations being assigned the smallest weights and the earliest observations being assigned the largest weights in the weighted average calculation. The default is 0.15.

This option is not included on the No Trend Model dialog.

Seasonal (Gamma)

The Holt Winters Smoothing technique utilizes an additional seasonal parameter, Gamma, to manage the presence of seasonality in the data. This parameter is also used in the weighted average calculation and can be from 0 to 1. A value of 1 or close to 1 will result in the most recent observations being assigned the largest weights and the earliest observations being assigned the smallest weights in the weighted average calculation. A value of 0 or close to 0 will result in the most recent observations being assigned the smallest weights and the earliest observations being assigned the largest weights in the weighted average calculation. The default is 0.05.

Produce Forecast

If this option is selected, Analytic Solver Data Science will include a forecast on the output results.

If running the Holt Winters Smoothing technique on an unpartitioned dataset, the following two options are enabled.

Update estimate each time

If applying this smoothing technique to an unpartitioned dataset, this option is enabled. Select this option to create an updated forecast each time that a forecast is generated.

# Forecasts

If applying this smoothing technique to an unpartitioned dataset, this option is enabled. Enter the desired number of forecasts here.

.

Data Science Partitioning

Introduction

One very important issue when fitting a model is how well the newly created model will behave when applied to new data.   To address this issue, the dataset can be divided into multiple partitions: a training partition used to create the model, a validation partition to test the performance of the model and, if desired, a third test partition. Partitioning is performed randomly, to protect against a biased partition, according to proportions specified by the user or according to rules concerning the dataset type. For example, when creating a time series forecast, data is partitioned by chronological order.

Training Set

The training dataset is used to train or build a model. For example, in a linear regression, the training dataset is used to fit the linear regression model, i.e. to compute the regression coefficients. In a neural network model, the training dataset is used to obtain the network weights. After fitting the model on the training dataset, the performance of the model should be tested on the validation dataset.

Validation Set

Once a model is built using the training dataset, the performance of the model must be validated using new data. If the training data itself was utilized to compute the accuracy of the model fit, the result would be an overly optimistic estimate of the accuracy of the model. This is because the training or model fitting process ensures that the accuracy of the model for the training data is as high as possible -- the model is specifically suited to the training data. To obtain a more realistic estimate of how the model would perform with unseen data, we must set aside a part of the original data and not include this set in the training process. This dataset is known as the *validation dataset*.

To validate the performance of the model, Analytic Solver Data Science measures the discrepancy between the actual observed values and the predicted value of the observation. This discrepancy is known as the error in prediction and is used to measure the overall accuracy of the model.

Test Set

The validation dataset is often used to fine-tune models. For example, you might try out neural network models with various architectures and test the accuracy of each on the validation dataset to choose the best performer among the competing architectures. In such a case, when a model is finally chosen, its accuracy with the validation dataset is still an optimistic estimate of how it would perform with unseen data. This is because the final model has come out as the winner among the competing models based on the fact that its accuracy with the validation dataset is highest. As a result, it is a good idea to set aside yet another portion of data which is used neither in training nor in validation. This set is known as the *test dataset*. The accuracy of the model on the test data gives a realistic estimate of the performance of the model on completely unseen data.

Analytic Solver Data Science provides two methods of partitioning: Standard Partitioning and Partitioning with Oversampling. Analytic Solver Data Science provides two approaches to standard partitioning: random partitioning and user-defined partitioning.

Random Partitioning

In simple random sampling, every observation in the main dataset has equal probability of being selected for the partition dataset. For example, if you specify 60% for the training dataset, then 60% of the total observations are randomly selected for the training dataset. In other words, each observation has a 60% chance of being selected.

Random partitioning uses the system clock as a default to initialize the random number seed. Alternatively, the random seed can be manually set which will result in the same observations being chosen for the training/validation/test sets each time a standard partition is created.

User – defined Partitioning

In user-defined partitioning, the partition variable specified is used to partition the dataset. This is useful when you have already predetermined the observations to be used in the training, validation, and/or test sets. This partition variable takes the value: "t" for training, "v" for validation and "s" for test. Rows with any other values in the Partition Variable column are ignored. The partition variable serves as a flag for writing each observation to the appropriate partition(s).

Partition with Oversampling

This method of partitioning is used when the percentage of successes in the output variable is very low, e.g. callers who “opt in” to a short survey at the end of a customer service call. Typically, the number of successes, in this case, the number of people who finish the survey, is very low so information connected with these callers is minimal. As a result, it would be almost impossible to formulate a model based on these callers. In these types of cases, we must use Oversampling (also called weighted sampling). Oversampling can be used when there are only two classes, one of much greater importance than the other, i.e. callers who finish the survey as compared to callers who simply hang up.

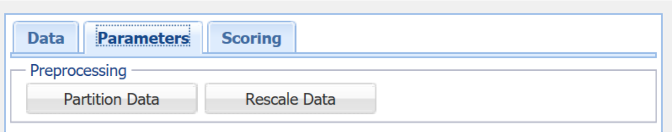
Analytic Solver Data Science takes the following steps when partitioning with oversampling.

1. The data is partitioned by randomly allocating 50% of the success values for the output variable to the training set. The output variable must be limited to two classes which can either be numbers or strings.
2. Analytic Solver Data Science maintains the **% success in training set** specified by the user in the training set by randomly selecting the required records with failures.
3. The remaining 50% of successes are randomly allocated to the validation set.

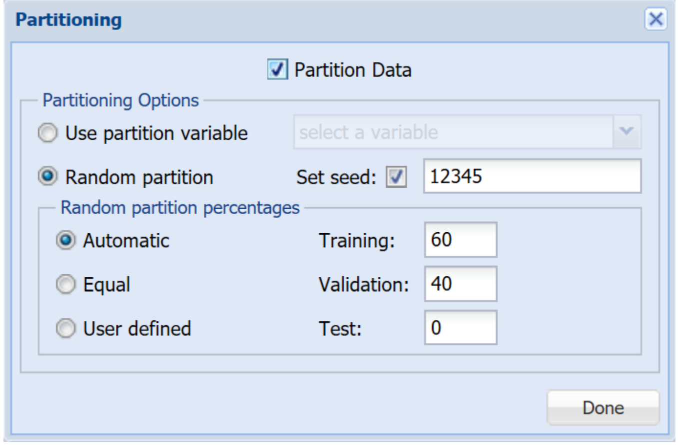
4. If **% validation data to be taken away as test data** is selected, then Analytic Solver Data Science will create an appropriate test set from the validation set.

Partition Options

It is no longer always necessary to partition a dataset before running a classification or regression algorithm. Rather, you can now perform partitioning on the Parameters tab for each classification or regression method.



If the active data set is un-partitioned, the Partition Data command button, will be enabled. If the active data set has already been partitioned, this button will be disabled. Clicking the Partition Data button opens the following dialog. Select Partition Data on the dialog to enable the partitioning options.



☑ Partition Data

Partitioning Options

○ Use partition variable: select a variable

● Random partition — Set seed: ☑ 12345

Random partition percentages

● Automatic — Training: 60

○ Equal — Validation: 40

○ User defined — Test: 0

If a data partition will be used to train and validate several different classification or regression algorithms that will be compared for predictive power, it may be better to use the Ribbon Partition choices to create a partitioned dataset. But if the data partition will be used with a single algorithm, or if it isn't crucial to compare algorithms on exactly the same partitioned data, "Partition-on-the-Fly" offers several advantages:

- User interface steps are saved, and the Analytic Solver task pane is not cluttered with partition output.
- Partition-on-the-fly is *much faster* than creating a standard partition and then running an algorithm.
- Partition-on-the-fly can handle *larger* datasets without exhausting memory, since the intermediate partition results for the partitioned data is never created.

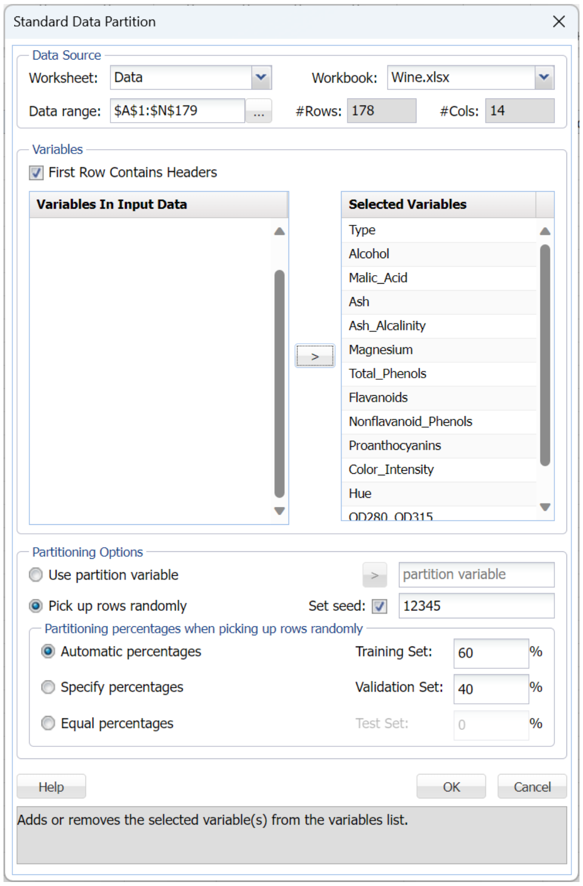
Standard Data Partition Example

The example in this section illustrates how to use Analytic Solver Data Science's partition utility to partition the example dataset, Wine.xlsx. Click **Help – Example Models -- Forecasting/Data Science Examples** to open.

Click **Partition – Standard Partition** on the Data Science Ribbon. The *Standard Data Partition* dialog opens.

Highlight all variables in the *Variables In Input Data* list box, then click > to include them in the partitioned data. Then click **OK** to accept the remainder of the default settings. Sixty percent of the observations will be assigned to the Training set and forty percent of the observations will be assigned to the Validation set.

Standard Data Partition dialog



Standard Data Partition

Data Source

Worksheet: Data

Workbook: Wine.xlsx

Data range: $A$1:$N$179

#Rows: 178

#Cols: 14

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables

- Type
- Alcohol
- Malic_Acid
- Ash
- Ash_Alcalinity
- Magnesium
- Total_Phenols
- Flavanoids
- Nonflavanoid_Phenols
- Proanthocyanins
- Color_Intensity
- Hue
- OD280_OD315

Partitioning Options

○ Use partition variable

partition variable

● Pick up rows randomly

Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages

Training Set: 60 %

○ Specify percentages

Validation Set: 40 %

○ Equal percentages

Test Set: 0 %

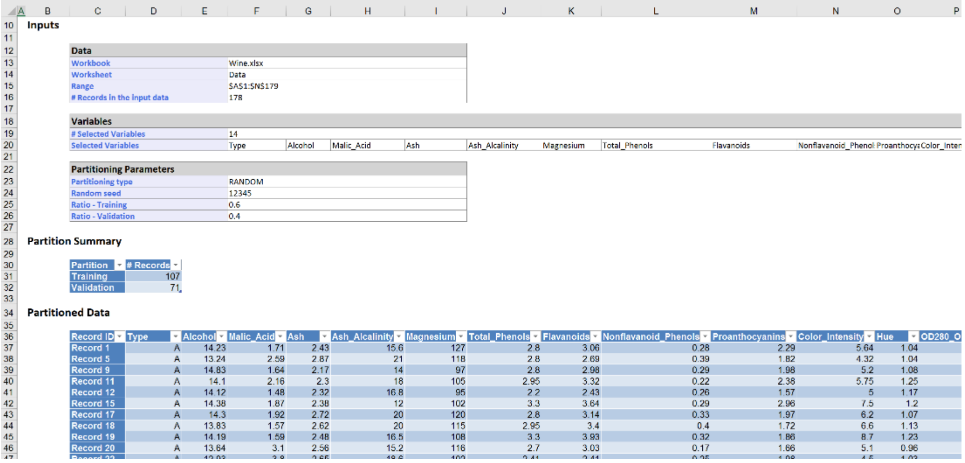
Adds or removes the selected variable(s) from the variables list.

STDPartition inserted right of the Data worksheet.

107 observations were assigned to the training set and 71 observations were assigned to the validation set, or roughly 60% and 40% of the observations, respectively.

Under Partitioned Data, the first 107 records (rows 37 thru 143) are the records assigned to the training partition and the remaining 71 records (rows 144 thru 214) are the records assigned to the validation partition.

Results for Standard Data Partitioning



Inputs

Data	
Workbook	Wine.xlsx
Worksheet	Data
Range	$A$1:$N$179
# Records in the input data	178

Variables											
# Selected Variables	14										
Selected Variables	Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenol	Proanthocya	Color_Inter

Partitioning Parameters	
Partitioning type	RANDOM
Random seed	12345
Ratio - Training	0.6
Ratio - Validation	0.4

Partition Summary

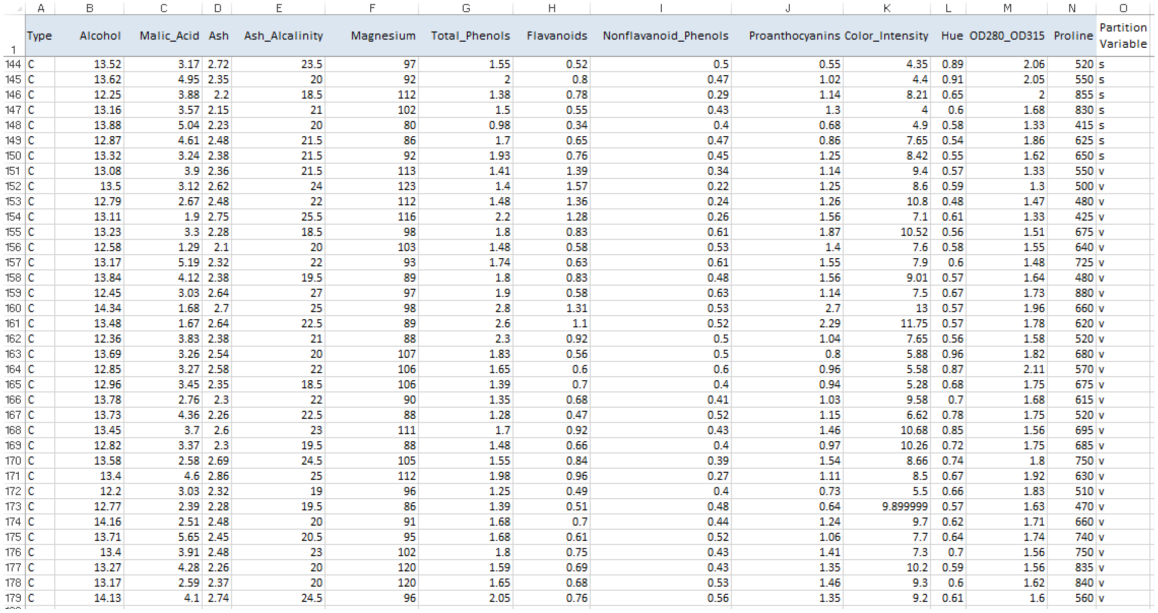
Partition	# Records
Training	107
Validation	71

Partitioned Data

Record ID	Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_O
Record 1	A	14.23	1.71	2.43	15.6	127	2.8	3.06	0.28	2.29	5.64	1.04	
Record 5	A	13.24	2.59	2.87	21	118	2.8	2.69	0.39	1.82	4.32	1.04	
Record 9	A	14.83	1.64	2.17	14	97	2.8	2.98	0.29	1.98	5.2	1.08	
Record 11	A	14.1	2.16	2.3	18	105	2.95	3.32	0.22	2.38	5.75	1.25	
Record 12	A	14.12	1.48	2.32	16.8	95	2.2	2.43	0.26	1.57	5	1.17	
Record 15	A	14.38	1.87	2.38	12	102	3.3	3.64	0.29	2.96	7.5	1.2	
Record 17	A	14.3	1.92	2.72	20	120	2.8	3.14	0.33	1.97	6.2	1.07	
Record 18	A	13.83	1.57	2.62	20	115	2.95	3.4	0.4	1.72	6.6	1.13	
Record 19	A	14.19	1.59	2.48	16.5	108	3.3	3.93	0.32	1.86	8.7	1.23	
Record 20	A	13.64	3.1	2.56	15.2	116	2.7	3.03	0.17	1.66	5.1	0.96	

It is also possible for the user to specify which sets each observation should be assigned. In column O, enter a “t”, “v” or “s” to indicate the assignment of each record to either the training dataset (t), the validation dataset (v), or the test dataset (s), as shown in the screenshot below.

Wine Dataset with Partition Variable

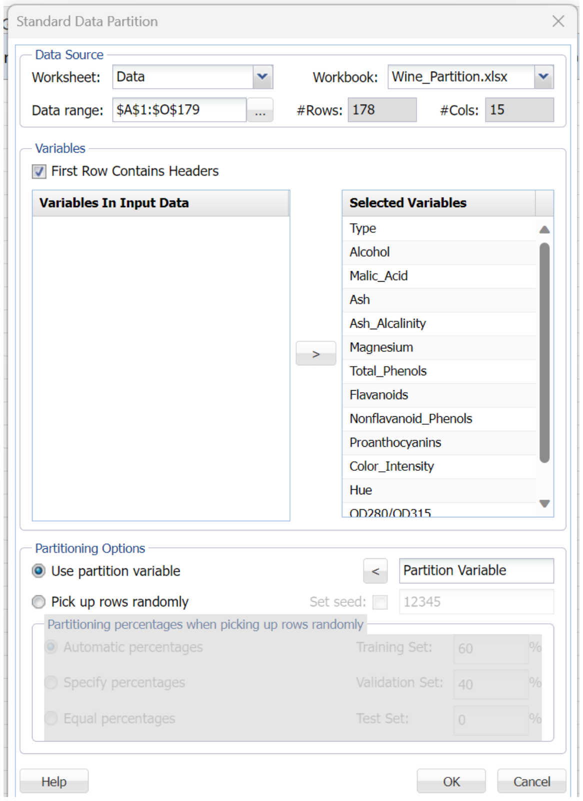


	Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline	Partition Variable
144	C	13.52	3.17	2.72	23.5	97	1.55	0.52	0.5	0.55	4.35	0.89	2.06	520	s
145	C	13.62	4.95	2.35	20	92	2	0.8	0.47	1.02	4.4	0.91	2.05	550	s
146	C	12.25	3.88	2.2	18.5	112	1.38	0.78	0.29	1.14	8.21	0.65	2	855	s
147	C	13.16	3.57	2.15	21	102	1.5	0.55	0.43	1.3	4	0.6	1.68	830	s
148	C	13.88	5.04	2.23	20	80	0.98	0.34	0.4	0.68	4.9	0.58	1.33	415	s
149	C	12.87	4.61	2.48	21.5	86	1.7	0.65	0.47	0.86	7.65	0.54	1.86	625	s
150	C	13.32	3.24	2.38	21.5	92	1.93	0.76	0.45	1.25	8.42	0.55	1.62	650	s
151	C	13.08	3.9	2.36	21.5	113	1.41	1.39	0.34	1.14	9.4	0.57	1.33	550	v
152	C	13.5	3.12	2.62	24	123	1.4	1.57	0.22	1.25	8.6	0.59	1.3	500	v
153	C	12.79	2.67	2.48	22	112	1.48	1.36	0.24	1.26	10.8	0.48	1.47	480	v
154	C	13.11	1.9	2.75	25.5	116	2.2	1.28	0.26	1.56	7.1	0.61	1.33	425	v
155	C	13.23	3.3	2.28	18.5	98	1.8	0.83	0.61	1.87	10.52	0.56	1.51	675	v
156	C	12.58	1.29	2.1	20	103	1.48	0.58	0.53	1.4	7.6	0.58	1.55	640	v
157	C	13.17	5.19	2.32	22	93	1.74	0.63	0.61	1.55	7.9	0.6	1.48	725	v
158	C	13.84	4.12	2.38	19.5	89	1.8	0.83	0.48	1.56	9.01	0.57	1.64	480	v
159	C	12.45	3.03	2.64	27	97	1.9	0.58	0.63	1.14	7.5	0.67	1.73	880	v
160	C	14.34	1.68	2.7	25	98	2.8	1.31	0.53	2.7	13	0.57	1.96	660	v
161	C	13.48	1.67	2.64	22.5	89	2.6	1.1	0.52	2.29	11.75	0.57	1.78	620	v
162	C	12.36	3.83	2.38	21	88	2.3	0.92	0.5	1.04	7.65	0.56	1.58	520	v
163	C	13.69	3.26	2.54	20	107	1.83	0.56	0.5	0.8	5.88	0.96	1.82	680	v
164	C	12.85	3.27	2.58	22	106	1.65	0.6	0.6	0.96	5.58	0.87	2.11	570	v
165	C	12.96	3.45	2.35	18.5	106	1.39	0.7	0.4	0.94	5.28	0.68	1.75	675	v
166	C	13.78	2.76	2.3	22	90	1.35	0.68	0.41	1.03	9.58	0.7	1.68	615	v
167	C	13.73	4.36	2.26	22.5	88	1.28	0.47	0.52	1.15	6.62	0.78	1.75	520	v
168	C	13.45	3.7	2.6	23	111	1.7	0.92	0.43	1.46	10.68	0.85	1.56	695	v
169	C	12.82	3.37	2.3	19.5	88	1.48	0.66	0.4	0.97	10.26	0.72	1.75	685	v
170	C	13.58	2.58	2.69	24.5	105	1.55	0.84	0.39	1.54	8.66	0.74	1.8	750	v
171	C	13.4	4.6	2.86	25	112	1.98	0.96	0.27	1.11	8.5	0.67	1.92	630	v
172	C	12.2	3.03	2.32	19	96	1.25	0.49	0.4	0.73	5.5	0.66	1.83	510	v
173	C	12.77	2.39	2.28	19.5	86	1.39	0.51	0.48	0.64	9.899999	0.57	1.63	470	v
174	C	14.16	2.51	2.48	20	91	1.68	0.7	0.44	1.24	9.7	0.62	1.71	660	v
175	C	13.71	5.65	2.45	20.5	95	1.68	0.61	0.52	1.06	7.7	0.64	1.74	740	v
176	C	13.4	3.91	2.48	23	102	1.8	0.75	0.43	1.41	7.3	0.7	1.56	750	v
177	C	13.27	4.28	2.26	20	120	1.59	0.69	0.43	1.35	10.2	0.59	1.56	835	v
178	C	13.17	2.59	2.37	20	120	1.65	0.68	0.53	1.46	9.3	0.6	1.62	840	v
179	C	14.13	4.1	2.74	24.5	96	2.05	0.76	0.56	1.35	9.2	0.61	1.6	560	v

Click **Partition – Standard Partition** on the Data Science ribbon to open the Standard Data Partition dialog.

Select **Use Partition Variable** in the Partitioning options section, select **Partition Variable** in the Variables list box, then click > next to **Use Partition Variable**. Analytic Solver Data Science will use the values in the Partition Variable column to create the training, validation, and test sets. Records with a “t” in the O column will be designated as training records. Records with a “v” in the O column will be designated as validating records and records with an “s” in this column will be designated as testing records. Now highlight all remaining variables in the list box and click > to include them in the partitioned data.

Standard Data Partition with Partition Variable



Standard Data Partition

Data Source

Worksheet: Data   Workbook: Wine_Partition.xlsx

Data range: $A$1:$O$179   #Rows: 178   #Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data		Selected Variables
	>	Type Alcohol Malic_Acid Ash Ash_Alcalinity Magnesium Total_Phenols Flavanoids Nonflavanoid_Phenols Proanthocyanins Color_Intensity Hue OD280/OD315

Partitioning Options

● Use partition variable   < Partition Variable

○ Pick up rows randomly   Set seed: □ 12345

Partitioning percentages when picking up rows randomly

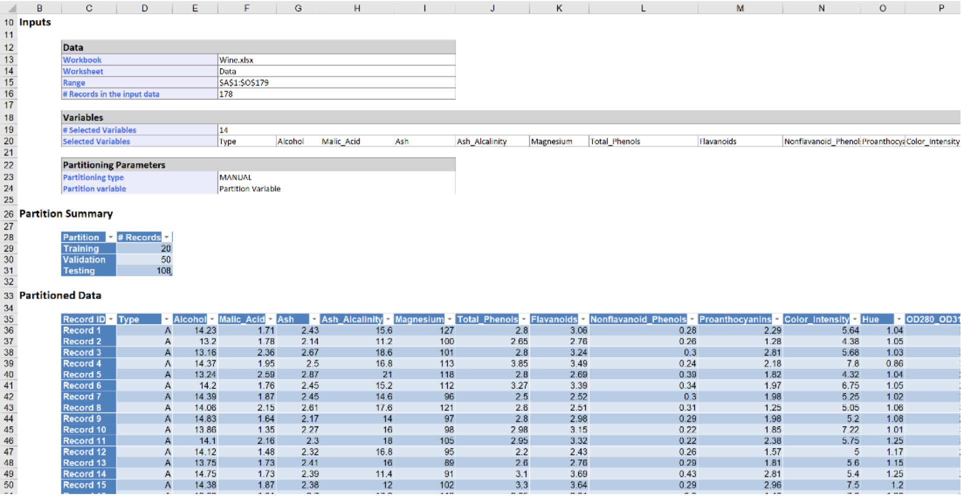
○ Automatic percentages   Training Set: 60 %

○ Specify percentages   Validation Set: 40 %

○ Equal percentages   Test Set: 0 %

Click **OK** to create the partitions. *STDPartition1* is inserted right. If you inspect the results, you will find that all records assigned a “t” now belong to the training set, all records assigned a “v” now belong to the validation set, and all records assigned an “s” now belong to the test set.

Results for Standard Data Partition with Partition Variable



Inputs

Data	
Workbook	Wine.xlsx
Worksheet	Data
Range	$A$1:$O$179
# Records in the input data	178

Variables											
# Selected Variables	14										
Selected Variables	Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenol	Proanthocya	Color_Intensity

Partitioning Parameters	
Partitioning type	MANUAL
Partition variable	Partition Variable

Partition Summary

Partition	# Records
Training	20
Validation	50
Testing	108

Partitioned Data

Record ID	Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD31
Record 1	A	14.23	1.71	2.43	15.6	127	2.8	3.06	0.28	2.29	5.64	1.04	
Record 2	A	13.2	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	
Record 3	A	13.16	2.36	2.67	18.6	101	2.8	3.24	0.3	2.81	5.68	1.03	
Record 4	A	14.37	1.95	2.5	16.8	113	3.85	3.49	0.24	2.18	7.8	0.86	
Record 5	A	13.24	2.59	2.87	21	118	2.8	2.69	0.39	1.82	4.32	1.04	
Record 6	A	14.2	1.76	2.45	15.2	112	3.27	3.39	0.34	1.97	6.75	1.05	
Record 7	A	14.39	1.87	2.45	14.6	96	2.5	2.52	0.3	1.98	5.25	1.02	
Record 8	A	14.06	2.15	2.61	17.6	121	2.6	2.51	0.31	1.25	5.05	1.06	
Record 9	A	14.83	1.64	2.17	14	97	2.8	2.98	0.29	1.98	5.2	1.08	
Record 10	A	13.86	1.35	2.27	16	98	2.98	3.15	0.22	1.85	7.22	1.01	
Record 11	A	14.1	2.16	2.3	18	105	2.95	3.32	0.22	2.38	5.75	1.25	
Record 12	A	14.12	1.48	2.32	16.8	95	2.2	2.43	0.26	1.57	5	1.17	
Record 13	A	13.75	1.73	2.41	16	89	2.6	2.76	0.29	1.81	5.6	1.15	
Record 14	A	14.75	1.73	2.39	11.4	91	3.1	3.69	0.43	2.81	5.4	1.25	
Record 15	A	14.38	1.87	2.38	12	102	3.3	3.64	0.29	2.96	7.5	1.2	

Partition with Oversampling Example

This example illustrates the use of partitioning with oversampling using Analytic Solver Data Science. Click **Help – Example Models** on the Data

Science ribbon, then **Forecasting/Data Science Examples** to open the example model, **Catalog_multi.xlsx**.

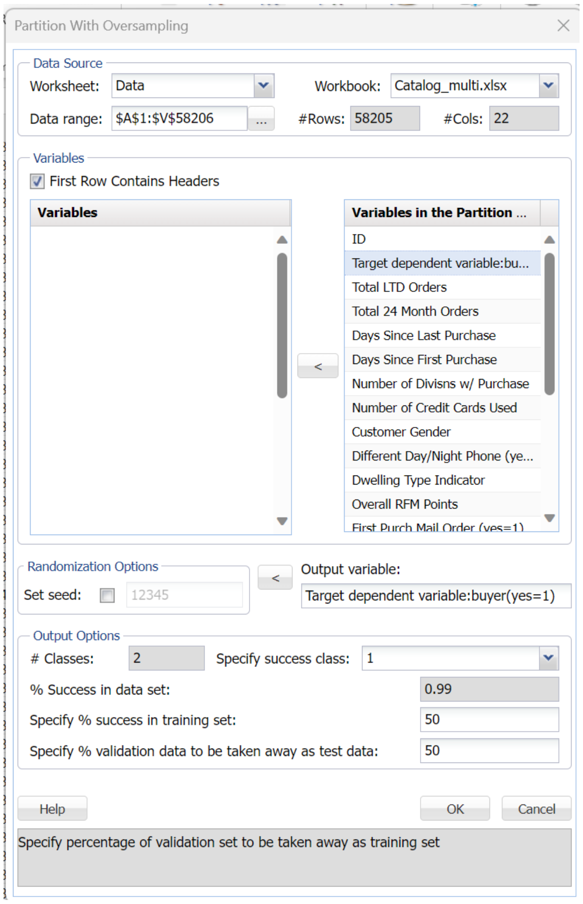
This sample dataset contains information associated with the response of a direct mail offer, published by DMEF, the Direct Marketing Educational Foundation. The output variable is Target dependent variable:buyer(yes=1). Since the success rate for the target variable (*Target dependent variable:buyer(yes=1)*) is less than 1%, the data will be "trained" with a 50% success rate using Analytic Solver Data Science's oversampling utility.

Click **Partition – Partition with Oversampling** (in the Data Science section of the Data Science ribbon) to open the *Partition with Oversampling* dialog.

First confirm that Data Range at the top of the dialog is displayed as **$A$1:$V$58206**. If not, simply click in the Data Range field and type the correct range.

Select all variables in the **Variables** list box then click > to move all variables to the **Variables in the Partition Data** list box. Afterwards, highlight **Target dependent variable: buyer(yes = 1)** in the *Variables in the Partition Data* list box then click the > immediately right of *Output variable* to designate this variable as the output variable. Reminder: this output variable is limited to two classes, e.g. 0/1 or "yes"/"no".

Enter **50** for *Specify % validation data to be taken away as test data*.



Partition With Oversampling

Data Source

Worksheet: Data

Workbook: Catalog_multi.xlsx

Data range: $A$1:$V$58206

#Rows: 58205

#Cols: 22

Variables

☑ First Row Contains Headers

Variables	Variables in the Partition ...
	ID
	Target dependent variable:bu...
	Total LTD Orders
	Total 24 Month Orders
	Days Since Last Purchase
	Days Since First Purchase
	Number of Divisns w/ Purchase
	Number of Credit Cards Used
	Customer Gender
	Different Day/Night Phone (ye...
	Dwelling Type Indicator
	Overall RFM Points
	First Purch Mail Order (yes=1)

Randomization Options

Set seed: □ 12345

Output variable: Target dependent variable:buyer(yes=1)

Output Options

Classes: 2

Specify success class: 1

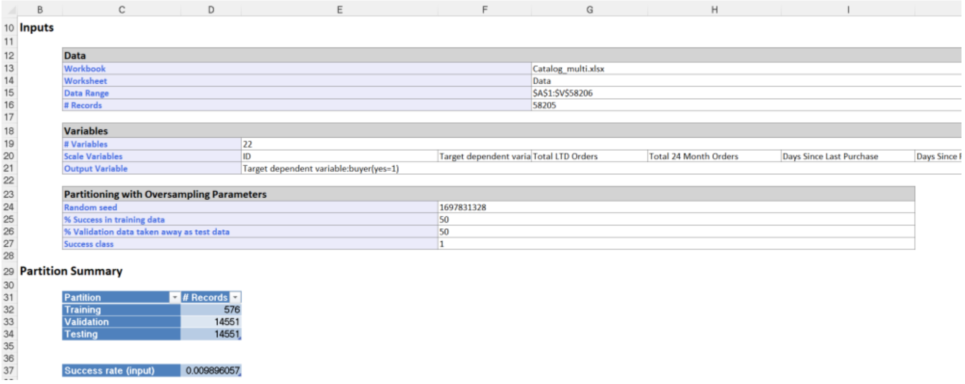
% Success in data set: 0.99

Specify % success in training set: 50

Specify % validation data to be taken away as test data: 50

Specify percentage of validation set to be taken away as training set

Click **OK** to partition the data. OSPartition is inserted right.



Inputs

Data	
Workbook	Catalog_multi.xlsx
Worksheet	Data
Data Range	$A$1:$V$58206
# Records	58205

Variables						
# Variables	22					
Scale Variables	ID	Target dependent varia	Total LTD Orders	Total 24 Month Orders	Days Since Last Purchase	Days Since F
Output Variable	Target dependent variable:buyer(yes=1)					

Partitioning with Oversampling Parameters	
Random seed	1697831328
% Success in training data	50
% Validation data taken away as test data	50
Success class	1

Partition Summary

Partition	# Records
Training	576
Validation	14551
Testing	14551

Success rate (input)	0.009896057

The percentage of success records in the original data set is 0.009896 or 576/58204 (number of successes / number of total rows in original dataset). 50% was specified for both *Specify % success in training set* and *Specify % validation data to be taken away as test data* in the Partition with Oversampling dialog. As a result, Analytic Solver Data Science has randomly allocated 50% of the successes (the 1's) to the training set and the remaining 50% to the validation set. This means that there are 288 successes in the training set and 288 successes in the validation set. To complete the training set, Analytic Solver Data Science randomly selected 288 non successes (0's). The training set has 576 rows (288 1's + 288 0's).

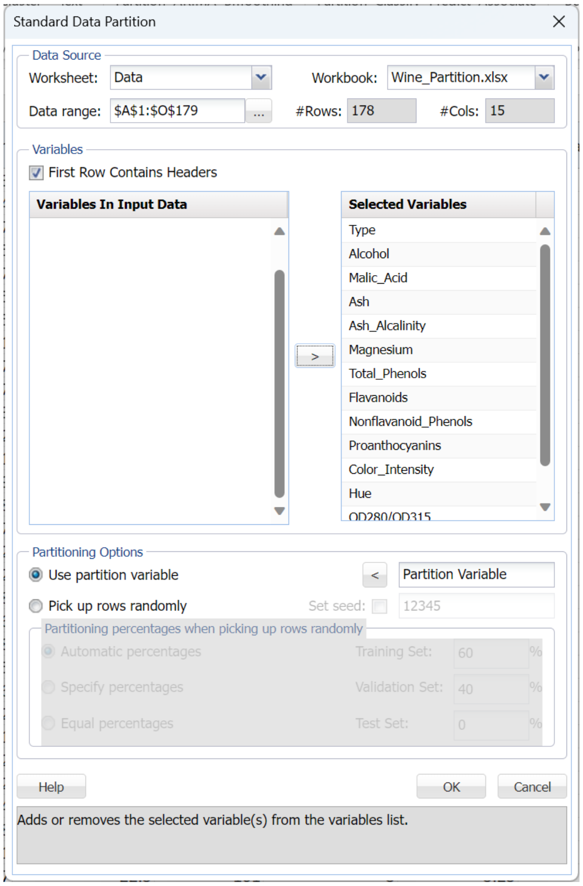
The output above shows that the % Success in original data set is .9896. Analytic Solver Data Science will maintain this percentage in the validation set as well by allocating as many 0's as needed. Since 288 successes (1's) have already been allocated to the validation set, 14,551 non successes (0's) must be added to the validation set to maintain the .98% ratio.

Since we specified 50% of validation data should be taken as test data, Analytic Solver Data Science has allocated 50% of the validation records to the test set. Each set contains 14,551 rows.

Standard Partitioning Options

The options below appear on the *Standard Data Partition* dialog, shown below.

Standard Data Partitioning Dialog



Standard Data Partition

Data Source

Worksheet: Data

Workbook: Wine_Partition.xlsx

Data range: $A$1:$O$179

#Rows: 178

#Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables:
Type
Alcohol
Malic_Acid
Ash
Ash_Alcalinity
Magnesium
Total_Phenols
Flavanoids
Nonflavanoid_Phenols
Proanthocyanins
Color_Intensity
Hue
OD280/OD315

Partitioning Options

● Use partition variable — Partition Variable

○ Pick up rows randomly — Set seed: □ 12345

Partitioning percentages when picking up rows randomly

○ Automatic percentages — Training Set: 60 %

○ Specify percentages — Validation Set: 40 %

○ Equal percentages — Test Set: 0 %

Adds or removes the selected variable(s) from the variables list.

Use partition variable

Select this option when assigning each record to a specific set using an added variable in the dataset. Each observation should be assigned a “t”, “v” or “s” to delineate “training”, “validation” or “test”, respectively.

Select this variable from the *Variables in Input Data* list box then click >, to the right of the *Use partition variable* radio button, to add the appropriate variable as the partition variable.

Set Seed

Random partitioning uses the system clock as a default to initialize the random number seed. By default this option is selected to specify a seed for random number generation for the partitioning. Setting this option will result in the same records being assigned to the same set on successive runs. The default seed entry is 12345.

Pick up rows randomly

When this option is selected, Analytic Solver Data Science will randomly select observations to be included in the training, validation, and test sets.

Automatic percentages

If *Pick up rows randomly* is selected under *Partitioning options*, this option will be enabled. Select this option to accept the defaults of 60% and 40% for the percentages of records to be included in the training and validation sets. This is the default selection.

Specify percentages

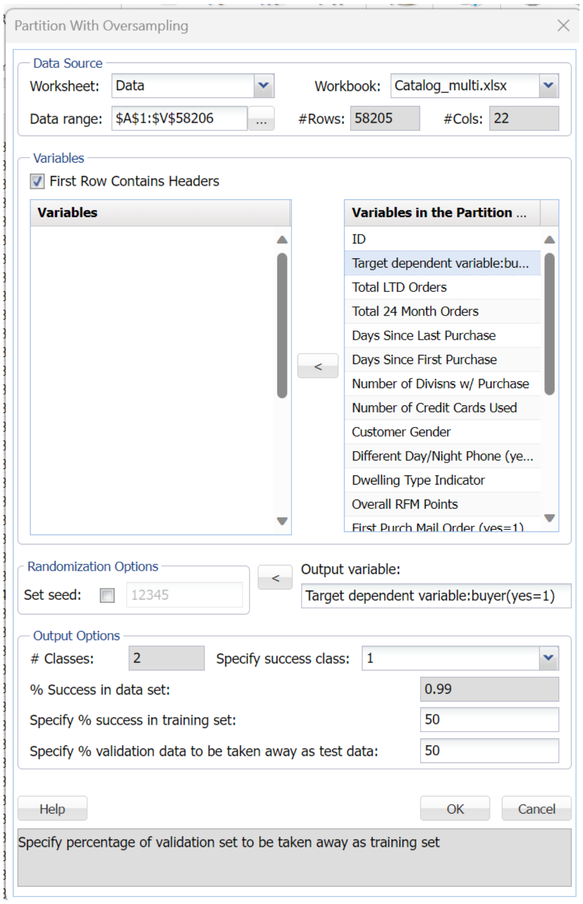
If *Pick up rows randomly* is selected under *Partitioning options*, this option will be enabled. Select this option to manually enter percentages for training set, validation set and test sets. Records will be randomly allocated to each set according to these percentages.

Equal percentages

If *Pick up rows randomly* is selected under *Partitioning options*, this option will be enabled. If this option is selected, Analytic Solver Data Science will allocate 33.33% of the records in the database to each set: training, validation, and test.

Partitioning with Oversampling Options

The following options appear on the *Partitioning with Oversampling* dialog, as shown below.



Partition With Oversampling

Data Source

Worksheet: Data

Workbook: Catalog_multi.xlsx

Data range: $A$1:$V$58206

#Rows: 58205

#Cols: 22

Variables

☑ First Row Contains Headers

Variables

Variables in the Partition ...

ID

Target dependent variable:bu...

Total LTD Orders

Total 24 Month Orders

Days Since Last Purchase

Days Since First Purchase

Number of Divisns w/ Purchase

Number of Credit Cards Used

Customer Gender

Different Day/Night Phone (ye...

Dwelling Type Indicator

Overall RFM Points

First Purch Mail Order (yes=1)

Randomization Options

Set seed: □ 12345

Output variable: Target dependent variable:buyer(yes=1)

Output Options

Classes: 2

Specify success class: 1

% Success in data set: 0.99

Specify % success in training set: 50

Specify % validation data to be taken away as test data: 50

Specify percentage of validation set to be taken away as training set

Set seed

Random partitioning uses the system clock as a default to initialize the random number seed. This option is not selected by default. Setting this option will result in the same records being assigned to the same set on successive runs. The default seed entry is 12345.

Output variable

Select the output variable from the variables listed in the *Variables in the partition data* list box.

#Classes

After the output variable is chosen, the number of classes (distinct values) for the output variable will be displayed here. Analytic Solver Data Science supports a class size of 2.

Specify Success class

After the output variable is chosen, you can select the success value for the output variable here (i.e. 0 or 1 or “yes” or “no”).

% of success in data set

After the output variable is selected, the percentage of the number of successes in the dataset is listed here.

Specify % success in training set

Enter the percentage of successes to be assigned to the training set here. The default is 50%. With this setting, 50% of the successes will be assigned to the training set and 50% will be assigned to the validation set.

Specify % validation data to be taken away as test data

If a test set is desired, specify the percentage of the validation set that should be allocated to the test set here.

Find Best Model for Classification and Regression

Introduction

Analytic Solver Data Science includes comprehensive, powerful support for data science and machine learning. Using these tools, you can *train* or fit your data to a wide range of statistical and machine learning models: Classification and regression trees, neural networks, linear and logistic regression, discriminant analysis, naïve Bayes, k-nearest neighbors and more. But the task of choosing and comparing these models, and selecting parameters for each one was up to you.

With the new Find Best Model options, you can automate this work as well! Find Best Model uses methods similar to those in (expensive high-end) tools like DataRobot and RapidMiner, to automatically choose types of ML models and their parameters, validate and compare them according to criteria that you choose, and deliver the model that best fits your data.

See the Analytic Solver User Guide to find a complete walk-through of this feature. Each classification learner may be ran independently. The rest of this chapter contains explanations of each option contained on the Find Best Model dialogs: Data, Parameter and Scoring.

Find Best Model Parameter Options

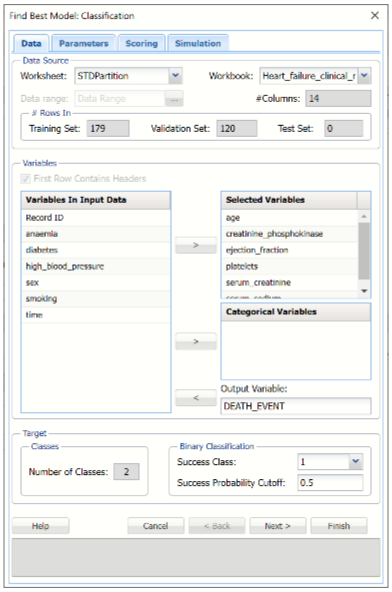
All options contained on the Find Best Model dialogs are described below.

Data Tab

The Data tab is where the data source is listed, the input and output variables are selected and the Success Class and Probability are set.

Data Source	
Workbook	Click the down arrow to select the workbook where the Find Best Model: Classification method will be applied.
Worksheet	Click the down arrow to select the worksheet where the Find Best Model: Classification method will be applied.
Data range	Select the range where the data appears on the selected worksheet.
#Columns	(Read-only) The number of columns in the data range.
# Rows In: Training Set	(Read-only) The number of rows in the training partition.
# Rows In: Validation Set	(Read-only) The number of rows in the validation partition.
# Rows In: Test Set	(Read-only) The number of rows in the test partition.

Find Best Model: Classification Data Tab



Data Source

Worksheet: STDPartition

Workbook: Heart_failure_clinical_r

Data range: Data Range

#Columns: 14

Rows In

Training Set: 179 Validation Set: 120 Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
anaemia
diabetes
high_blood_pressure
sex
smoking
time

Selected Variables
age
creatinine_phosphokinase
ejection_fraction
platelets
serum_creatinine
serum_sodium

Categorical Variables

Output Variable: DEATH_EVENT

Target

Classes

Number of Classes: 2

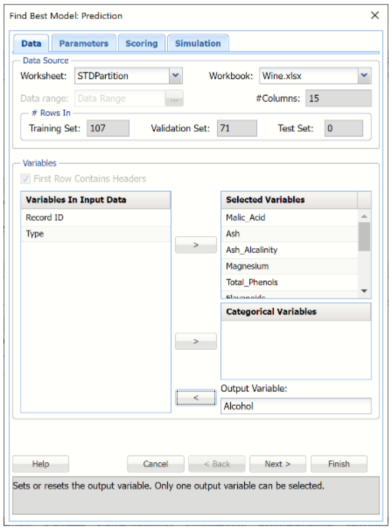
Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Variables	
First Row Contains Headers	Select this option if the first row of the dataset contains column headings.
Variables in Input Data	Variables contained in the dataset.
Selected Variables	Variables appearing under Selected Variables will be treated as continuous.
Categorical Variables	Variables appearing under Categorical Variables will be treated as categorical.
Output Variable	Select the output variable, or the variable to be classified, here.

Find Best Model: Prediction Data Tab



Data Source

Worksheet: STDPartition

Workbook: Wine.xlsx

Data range: Data Range

#Columns: 15

Rows In

Training Set: 107 Validation Set: 71 Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
Type

Selected Variables
Malic_Acid
Ash
Ash_Alcalinity
Magnesium
Total_Phenols
Flavonoids

Categorical Variables

Output Variable: Alcohol

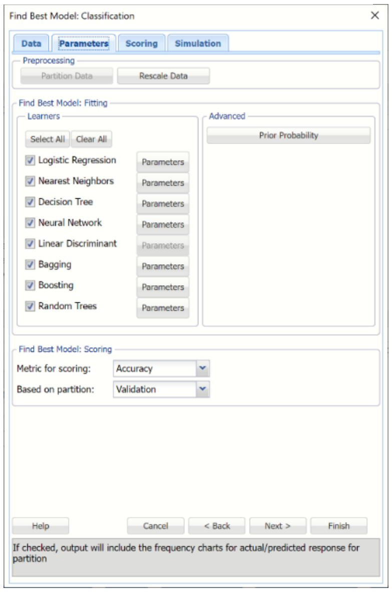
Sets or resets the output variable. Only one output variable can be selected.

Target – Find Best Model Classification Only	
Classes: Number of Classes	(Read-only) The number of classes that exist in the output variable.
Binary Classification: Success Class	This option is selected by default. Select the class to be considered a “success” or the significant class. This option is enabled when the number of classes in the output variable is equal to 2.
Binary Classification: Success Probability Cutoff	Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a “success” (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a “non-success” (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

Parameters tab

Select Find Best Model parameters here such as the learners to be applied, and their options, and the metric and partition to be used to measure the performance of each learner.

Find Best Model: Classification Parameters Tab

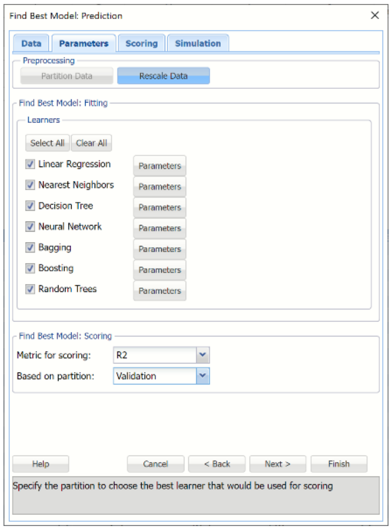


☑ Logistic Regression
☑ Nearest Neighbors
☑ Decision Tree
☑ Neural Network
☑ Linear Discriminant
☑ Bagging
☑ Boosting
☑ Random Trees

Metric for scoring: Accuracy
Based on partition: Validation

If checked, output will include the frequency charts for actual/predicted response for partition

Find Best Model: Prediction Parameters Tab



☑ Linear Regression
☑ Nearest Neighbors
☑ Decision Tree
☑ Neural Network
☑ Bagging
☑ Boosting
☑ Random Trees

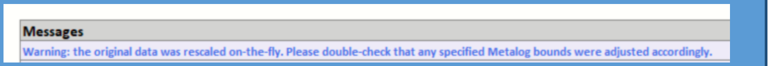
Metric for scoring: R2
Based on partition: Validation

Specify the partition to choose the best learner that would be used for scoring

<table>
<tr><th colspan="2">Preprocessing</th></tr>
<tr><td>Partition Data</td><td>Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.</td></tr>
<tr><td>Rescale Data</td><td>Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.</td></tr>
<tr><td></td><td>

Notes on Rescaling and the Simulation functionality

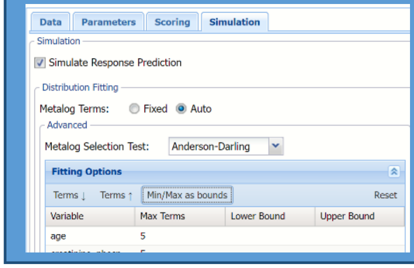
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot to the left, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the CFBM_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction
Metalog Terms: ○ Fixed ● Auto
Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound
age	5		

</td></tr>
</table>

Find Best Model Fitting Learners	
Select All	Click to select all available learners.
Clear All	Click to deselect all previously selected learners.
Learners	Select to run the desired Learner. Click the Parameters button to change a learner-specific option.

Find Best Model: Scoring - The model with the "best" metric in the selected partition, will be used in scoring.	
Metric for Scoring	Select the down arrow to select the metric to determine the best fit model. See the tables below for a description of each metric.
Based on partition	Select the down arrow to select the metric in the desired partition to determine the best fit model.

Find Best Model: Classification Learner Parameters	
Logistic Regression	
Fit Intercept	When this option is selected, the default setting, Analytic Solver Data Science will fit the Logistic Regression intercept. If this option is not selected, Analytic Solver Data Science will force the intercept term to 0.
Iterations (Max)	Estimating the coefficients in the Logistic Regression algorithm requires an iterative non-linear maximization procedure. You can specify a maximum number of iterations to prevent the program from getting lost in very lengthy iterative loops. This value must be an integer greater than 0 or less than or equal to 100 (1< value <= 100).
K-Nearest Neighbors	For more information on each parameter, see the K-Nearest Neighbors Classification chapter within the Analytic Solver Reference Guide.
# Neighbors (k)	Enter a value for the parameter K in the Nearest Neighbor algorithm.
Classification Tree	For more information on each parameter, see the Classification Tree chapter within the Analytic Solver Reference Guide.
Tree Growth Levels, Nodes, Splits, Tree Records in Terminal Nodes	In the *Tree Growth* section, select Levels, Nodes, Splits, and Records in Terminal Nodes. Values entered for these options limit tree growth, i.e. if 10 is entered for Levels, the tree will be limited to 10 levels.
Prune	If a validation partition exists, this option is enabled. When this option is selected, Analytic Solver Data Science will prune the tree using the validation set. Pruning the tree using the validation set reduces the error from over-fitting the tree to the training data. Click Tree for Scoring to click the Tree type used for scoring: Fully Grown, Best Pruned, Minimum Error, User Specified or Number of Decision Nodes.
Neural Network	For more information on each parameter, see the Neural Network Classification chapter within the Analytic Solver Reference Guide.
Architecture	Click *Add Layer* to add a hidden layer. To delete a layer, click *Remove Layer*. Once the layer is added, enter the desired Neurons.
Hidden Layer	Nodes in the hidden layer receive input from the input layer. The output of the hidden nodes is a weighted sum of the input values. This weighted sum is computed with weights that are initially set at random values. As the network “learns”, these weights are adjusted. This weighted sum is used to compute the hidden node’s output using a *transfer function*. The default selection is *Sigmoid*.
Output Layer	As in the hidden layer output calculation (explained in the above paragraph), the output layer is also computed using the same transfer function as described for *Activation: Hidden Layer*. The default selection is *Sigmoid.*
Training Parameters	Click Training Parameters to open the Training Parameters dialog to specify parameters related to the training of the Neural Network algorithm.
Stopping Rules	Click Stopping Rules to open the Stopping Rules dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error.

Linear Discriminant	For more information on this learner, see the Discriminant Analysis Classification chapter within the Analytic Solver Reference Guide.
Bagging Ensemble Method	For more information on each parameter, see the Ensemble Methods Classification chapter within the Analytic Solver Reference Guide.
Number of Weak Learners	This option controls the number of "weak" classification models that will be created. The ensemble method will stop when the number or classification models created reaches the value set for this option. The algorithm will then compute the weighted sum of votes for each class and assign the "winning" classification to each record.
Weak Learner	Under Ensemble: Classification click the down arrow beneath Weak Leaner to select one of the six featured classifiers: Discriminant Analysis, Logistic Regression, k-NN, Naïve Bayes, Neural Networks, or Decision Trees. The command button to the right will be enabled. Click this command button to control various option settings for the weak leaner.
Random Seed for Bootstrapping	Enter an integer value to specify the seed for random resampling of the training data for each weak learner. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is "12345". If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.
Boosting Ensemble Method	For more information on each parameter, see the Boosting Classification Ensemble Method chapter within the Analytic Solver Reference Guide.
Number of Weak Learners Weak Learner	See description above.
Adaboost Variant	In AdaBoost.M1 (Freund), the constant is calculated as: α_b= ln((1-e_b)/e_b) In AdaBoost.M1 (Breiman), the constant is calculated as: α_b= 1/2ln((1-e_b)/e_b) In SAMME, the constant is calculated as: α_b= 1/2ln((1-e_b)/e_b + ln(k-1) where k is the number of classes
Random Seed for Resampling	Enter an integer value to specify the seed for random resampling of the training data for each weak learner. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is "12345".
Random Trees Ensemble Method	For more information on each parameter, see the Boosting Classification Ensemble Method chapter within the Analytic Solver Reference Guide.
Number of Weak Learners Random Seed for Bootstrapping Weak Learner	See description above.

Number of Randomly Selected Features	The Random Trees ensemble method works by training multiple “weak” classification trees using a fixed number of randomly selected features then taking the mode of each class to create a “strong” classifier. The option *Number of randomly selected features* controls the fixed number of randomly selected features in the algorithm. The default setting is **3**.
Feature Selection Random Seed	If an integer value appears for *Feature Selection Random seed*, Analytic Solver Data Science will use this value to set the feature selection random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Find Best Model: Prediction Learner Parameters	
Linear Regression	For more information on each parameter, see the Linear Regression Method chapter within the Analytic Solver Reference Guide.
Fit Intercept	If this option is selected, a constant term will be included in the model. Otherwise, a constant term will not be included in the equation. This option is selected by default.
K-Nearest Neighbors	For more information on each parameter, see the K-Nearest Neighbors Regression Method chapter within the Analytic Solver Reference Guide.
# Neighbors (k)	Enter a value for the parameter K in the Nearest Neighbor algorithm.
DecisionTree	For more information on each parameter, see the Regression Tree Method chapter within the Analytic Solver Reference Guide.
Tree Growth Levels, Nodes, Splits, Tree Records in Terminal Nodes	In the *Tree Growth* section, select Levels, Nodes, Splits, and Records in Terminal Nodes. Values entered for these options limit tree growth, i.e. if 10 is entered for Levels, the tree will be limited to 10 levels.
Prune	If a validation partition exists, this option is enabled. When this option is selected, Analytic Solver Data Science will prune the tree using the validation set. Pruning the tree using the validation set reduces the error from over-fitting the tree to the training data. Click Tree for Scoring to click the Tree type used for scoring: Fully Grown, Best Pruned, Minimum Error, User Specified or Number of Decision Nodes.
Neural Network	For more information on each parameter, see the Neural Network Regression Method chapter within the Analytic Solver Reference Guide.
Architecture	Click *Add Layer* to add a hidden layer. To delete a layer, click *Remove Layer*. Once the layer is added, enter the desired Neurons.
Hidden Layer	Nodes in the hidden layer receive input from the input layer. The output of the hidden nodes is a weighted sum of the input values. This weighted sum is computed with weights that are initially set at random values. As the network “learns”, these weights are adjusted. This weighted sum is used to compute the hidden node’s output using a *transfer function*. The default selection is *Sigmoid*.

Output Layer	As in the hidden layer output calculation (explained in the above paragraph), the output layer is also computed using the same transfer function as described for *Activation: Hidden Layer*. The default selection is *Sigmoid*.
Training Parameters	Click Training Parameters to open the Training Parameters dialog to specify parameters related to the training of the Neural Network algorithm.
Stopping Rules	Click Stopping Rules to open the Stopping Rules dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error.
Bagging Ensemble Method	For more information on each parameter, see the Ensemble Methods chapter within the Analytic Solver Reference Guide.
Number of Weak Learners	This option controls the number of "weak" regression models that will be created. The ensemble method will stop when the number of regression models created reaches the value set for this option. The algorithm will then compute the weighted sum of votes for each class and assign the "winning" value to each record.
Weak Learner	Under Ensemble: Common click the down arrow beneath Weak Leaner to select one of the four featured classifiers: Linear Regression, k-NN, Neural Networks or Decision Tree. The command button to the right will be enabled. Click this command button to control various option settings for the weak leaner.
Random Seed for Bootstrapping	Enter an integer value to specify the seed for random resampling of the training data for each weak learner. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is "12345". If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.
Boosting Ensemble Method	For more information on each parameter, see the Boosting Regression Ensemble Method chapter within the Analytic Solver Reference Guide.
Number of Weak Learners Weak Learner	See description above.
Step Size	The Adaboost algorithm minimizes a loss function using the gradient descent method. The Step size option is used to ensure that the algorithm does not descend too far when moving to the next step. It is recommended to leave this option at the default of 0.3, but any number between 0 and 1 is acceptable. A Step size setting closer to 0 results in the algorithm taking smaller steps to the next point, while a setting closer to 1 results in the algorithm taking larger steps towards the next point.
Random Trees Ensemble Method	For more information on each parameter, see the Boosting Classification Ensemble Method chapter within the Analytic Solver Reference Guide.
Number of Weak Learners Random Seed for Bootstrapping Weak Learner	See description above.
Number of Randomly Selected Features	The Random Trees ensemble method works by training multiple "weak" classification trees using a fixed number of randomly selected features then taking the mode of each class to create a

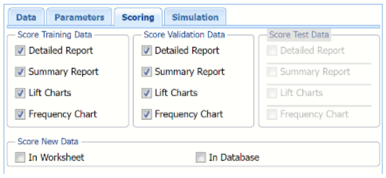
	“strong” classifier. The option *Number of randomly selected features* controls the fixed number of randomly selected features in the algorithm. The default setting is **3**.
Feature Selection Random Seed	If an integer value appears for *Feature Selection Random seed*, Analytic Solver Data Science will use this value to set the feature selection random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Find Best Model Classification Scoring Statistic	**Description**		**Find Best Model Prediction Scoring Statistic**	**Description**
R2	Coefficient of Determination **-** Examines how differences in one variable can be explained by the difference in a second variable, when predicting the outcome of a given event. RMSE = $\frac{SSR}{SST} = \frac{\sum_i(\hat{y}_i - \bar{y})^2}{\sum_i(y_i - \bar{y})^2}$ where $\hat{y}_i$ is the predicted value for obs i y_i is the actual value for obs i $\bar{y}$ is mean of the y values		Accuracy	Accuracy refers to the ability of the classifier to predict a class label correctly.
SSE	Sum of Squared Error – The sum of the squares of the differences between the actual and predicted values. SSE = $\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$ $\hat{y}_i$ is the predicted value for obs i y_i is the actual value for obs i		Specificity	Specificity is defined as the proportion of negative classifications that were actually negative.
MSE	Mean Squared Error – The average of the squared differences between the actual and predicted values. MSE = $\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$ $\hat{y}_i$ is the predicted value for obs i y_i is the actual value for obs i		Sensitivity	Sensitivity is defined as the proportion of positive cases there were classified correctly as positive.
RMSE	Root Mean Squared Error – The standard deviation of the residuals. RSE = $\sqrt{\sum_{i=1}^{n}\frac{(\hat{y}_i - y_i)^2}{n}}$ $\hat{y}_i$ is the predicted value for obs i		Precision	Precision is defined as the proportion of positive results that are truly positive.

	y_i is the actual value for obs i			
MAD	Mean Absolute Deviation - Average distance between each data value and the sample mean; describes variation in a data set. MAD = $\frac{1}{n}\sum_{i=1}^{n}\lvert x_i - \bar{x}\rvert$ where x_i is the i^{th} obs in the sample where $\bar{x}$ is the sample mean		F1	Calculated as 0.743 –2 x (Precision * Sensitivity)/(Precision + Sensitivity) The F-1 Score provides a statistic to balance between Precision and Sensitivity, especially if an uneven class distribution exists. The closer the F-1 score is to 1 (the upper bound) the better the precision and recall.

Scoring Tab

Find Best Model: Classification & Prediction Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

Select the type of output for the Find Best Model method and/or a scoring method (all optional).

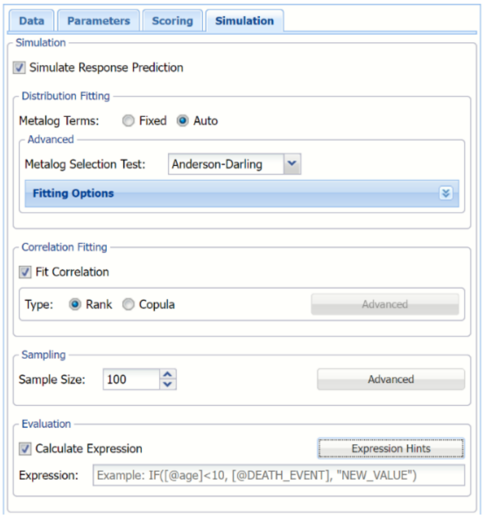
When Frequency Chart is selected, a frequency chart will be displayed when the FBM_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Scoring Data for Training, Validation and Test	
Detailed Report	Select this option to add a Detailed Report to the output. This report shows the classification/prediction of records by row.
Summary Report	Select this option to add a Summary Report to the output. This report summarizes the Detailed Report. For classification models, the Summary Report contains a confusion matrix, error report and various metrics: accuracy, specificity, sensitivity, precision and F1. For regression models, the Summary Report contains five metrics: SSE, MSE, RMSE, MAD and R2.
Lift Charts	Select this option to add Lift charts, gain charts, and ROC curves to the output. For a description of each, see the Find Best Model chapter within the Data Science User Guide.
Frequency Chart	When Frequency Chart is selected, a frequency chart will be displayed when the CFBM_TrainingScore, CFBM_ValidationScore, PFBM_TrainingScore or PFBM_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score New Data	
In Worksheet	Select to score new data in the worksheet immediately after the Find Best Model method is complete.
In Database	Select to score new data within a database immediately after the Find Best Model method is complete.
See the Scoring chapter in the Data Science User Guide for more information on scoring new data.	

Find Best Model dialog, Simulation tab

Find Best Model: Classification/Prediction Simulation tab



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

☑ Calculate Expression

Expression: Example: IF([@age]<10, [@DEATH_EVENT], "NEW_VALUE")

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, *CFBM_Simulation* (for classification) or *PFBM_Simulation* (for prediction) , will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the CFBM_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Discriminant Analysis Classification Method

Introduction

Linear Discriminant analysis (LD) is a *generative* classifier; it models the joint probability distribution of the input and target variables. As a result, this classifier can “generate” new input variables given the target variable.

The discriminant analysis model is built using a set of observations for which the classes are known. This set of observations is sometimes referred to as the training set. Based on the training set, the technique constructs a set of linear functions of the predictors, known as discriminant functions, such that L = b1x1 + b2x2 + … + bnxn + c, where the b's are discriminant coefficients, the x's are the input variables or predictors and c is a constant.

These discriminant functions are used to predict the class of a new observation with an unknown class. For a k class problem, k discriminant functions are constructed. Given a new observation, all k discriminant functions are evaluated and the observation is assigned to the class with the largest discriminant function value.

Discriminant analysis assumes that:

1. The data is normally distributed.
2. Means of each class are specific to that class.
3. All classes have a common covariance matrix.

If these assumptions are realized, DA generates a linear decision boundary.

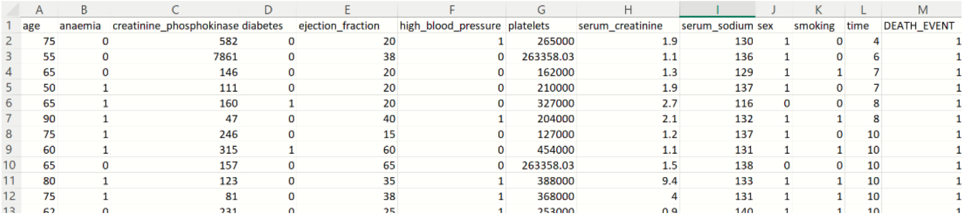
The latest version of Analytic Solver Data Science now contains Quadratic Discriminant Analysis (QDA). QDA produces a quadratic decision boundary, rather than a linear decision boundary. While QDA also assumes that the data is normally distributed, QDA does *not* assume that all classes share the same covariance matrix.

QDA is a more flexible technique when compared to LDA. QDA's performance improves over LDA when the class covariance matrices are disparate. Since each class has a different covariance matrix, the number of parameters that must be estimated increases significantly as the number of dimensions (predictors) increase. As a result, LDA might be a better choice over QDA on datasets with small numbers of observations and large numbers of classes. It’s advisable to try both techniques to determine which one performs best on your model. You can easily switch between LDA and QDA simply by setting this option to true or false.

Linear Discriminant Analysis Example

This example illustrates how to use the Discriminant Analysis classification algorithm using the Heart_failure_clinical_records.xlsx example dataset. Click **Help – Example Models -- Forecasting/Data Science Examples**. See the next example for a description of how to use the Quadratic Discriminant analysis algorithm.

A portion of the dataset is shown in the screenshot below. This dataset contains 13 characteristics pertaining to patients in a heart clinic. Each variable is listed below followed by a short description.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking	time	DEATH_EVENT
2	75	0	582	0	20	1	265000	1.9	130	1	0	4	1
3	55	0	7861	0	38	0	263358.03	1.1	136	1	0	6	1
4	65	0	146	0	20	0	162000	1.3	129	1	1	7	1
5	50	1	111	0	20	0	210000	1.9	137	1	0	7	1
6	65	1	160	1	20	0	327000	2.7	116	0	0	8	1
7	90	1	47	0	40	1	204000	2.1	132	1	1	8	1
8	75	1	246	0	15	0	127000	1.2	137	1	0	10	1
9	60	1	315	1	60	0	454000	1.1	131	1	1	10	1
10	65	0	157	0	65	0	263358.03	1.5	138	0	0	10	1
11	80	1	123	0	35	1	388000	9.4	133	1	1	10	1
12	75	1	81	0	38	1	368000	4	131	1	1	10	1
13	62	0	231	0	25	1	253000	0.9	140	1	1	10	1

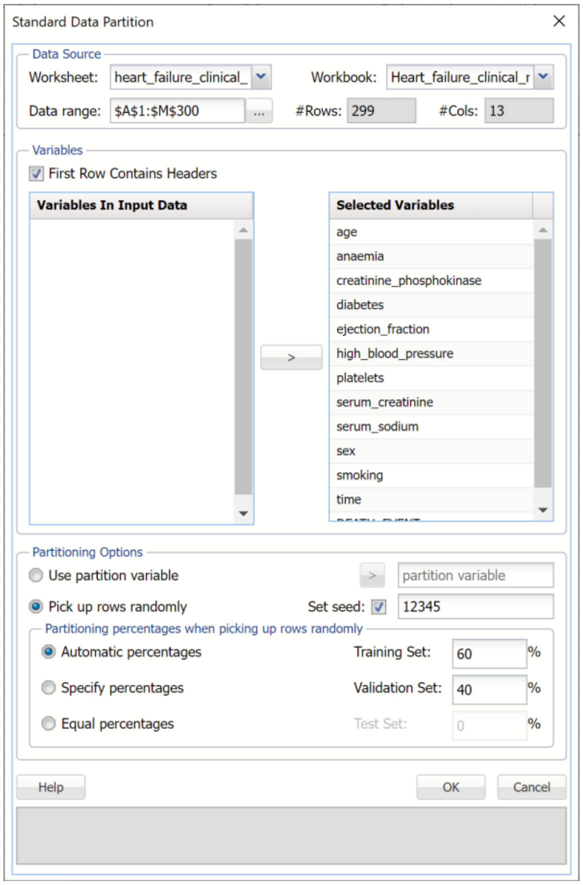
Variable	**Description**
age	Age of patient
anaemia	Decrease of red blood cells or hemoglobin (boolean)
creatinine_phosphokinase	Level of the CPK enzyme in the blood (mcg/L)
diabetes	If the patient has diabetes (boolean)
ejection_fraction	Percentage of blood leaving the heart at each contraction (percentage)
high_blood_pressure	If the patient has hypertension (boolean)
platelets	Platelets in the blood (kiloplatelets/mL)
serum_creatinine	Level of serum creatinine in the blood (mg/dL)
serum_sodium	Level of serum sodium in the blood (mEq/L)
sex	Female (0) or Male (1)
smoking	If the patient smokes or not (boolean)
time	Follow-up period (days)
DEATH_EVENT	If the patient was deceased during the follow-up period (boolean)

All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described in the What's New section of the Analytic Solver Data Science User Guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, DA_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type. Since this new functionality does not support categorical variables, these types of variables will not be included in the model, only continuous variables.

Inputs

1. First, we'll need to perform a standard partition, as explained in the previous chapter, using percentages of **60%** training and **40%** validation. *STDPartition* will be inserted to the right of the Data worksheet. (For more information on how to partition a dataset, please see the previous Data Science Partitioning chapter.)

Standard Partition Dialog containing the Heart Failure dataset



Standard Data Partition

Data Source

Worksheet: heart_failure_clinical_ Workbook: Heart_failure_clinical_r

Data range: $A$1:$M$300 #Rows: 299 #Cols: 13

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	age
	anaemia
	creatinine_phosphokinase
	diabetes
	ejection_fraction
	high_blood_pressure
	platelets
	serum_creatinine
	serum_sodium
	sex
	smoking
	time
	DEATH_EVENT

Partitioning Options

○ Use partition variable — partition variable

● Pick up rows randomly — Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages — Training Set: 60 %

○ Specify percentages — Validation Set: 40 %

○ Equal percentages — Test Set: 0 %

2. With the STDPartition worksheet displayed, click **Classify – Discriminant Analysis** to open the *Discriminant Analysis* dialog. Select the **DEATH_EVENT** variable in the *Variables in Input Data* list box then click > to select as the *Output Variable*. Immediately, the options for *Classes in the Output Variable* are enabled. *#Classes* is prefilled as "2" since the output variable contains two classes, 0 and 1.

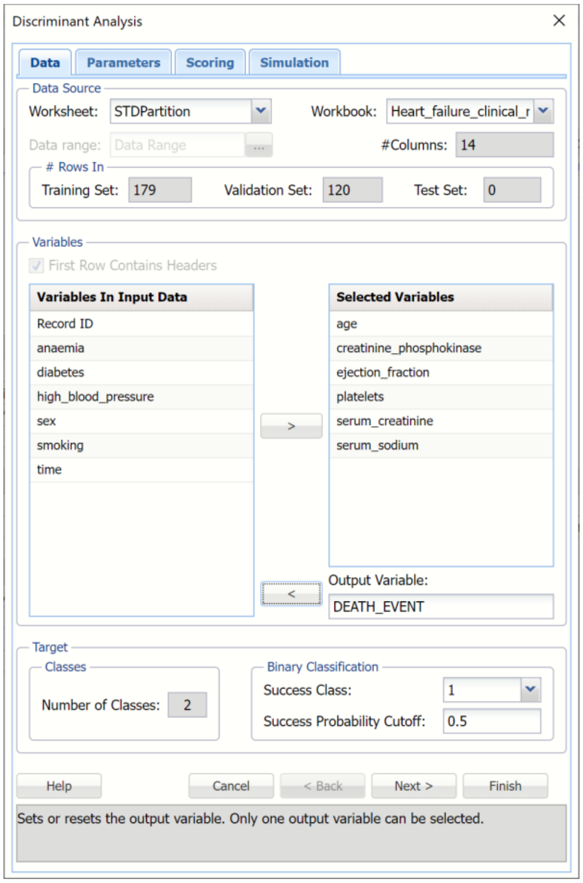
   Success Class is selected by default and Class 1 is to be considered a "success" or the significant class in the Lift Chart. (Note: This option is enabled when the number of classes in the output variable is equal to 2.)

3. Enter a value between 0 and 1 for *Success Probability Cutoff.* If the calculated probability for success for an observation is greater than or equal to this value, than a "success" (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a "non-success" (or a 0) will be predicted for that observation. The default value is 0.5. (Note: This option is only enabled when the # of classes is equal to 2.)

4. Select the following continuous variables (age, creatinine_phosphokinase, ejection_fraction, platelets, serum_creatinine and serum_sodium) in the Variables in Input Data list box, then click > to move to the Selected Variables list box.

Recall that categorical variables are not supported. Anaemia, diabetes, high_blood_pressue, sex, smoking and death_event are all categorical variables that will not be included in the example.

Discriminant Analysis dialog containing the Heart Failure dataset



Discriminant Analysis

Data Source

Worksheet: STDPartition

Workbook: Heart_failure_clinical_r

Data range: Data Range

#Columns: 14

Rows In

Training Set: 179 Validation Set: 120 Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
anaemia
diabetes
high_blood_pressure
sex
smoking
time

Selected Variables
age
creatinine_phosphokinase
ejection_fraction
platelets
serum_creatinine
serum_sodium

Output Variable: DEATH_EVENT

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

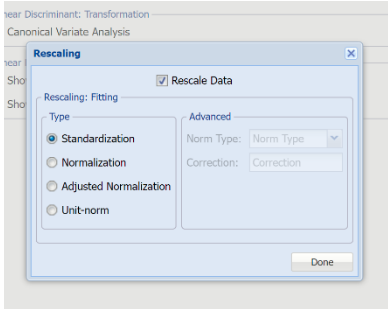
Sets or resets the output variable. Only one output variable can be selected.

5. Click **Next** to advance to the Parameters tab.

6. If you haven't already partitioned your dataset, you can do so from within the Discriminant Analysis method by selecting Partition Data on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the prediction method. Note that a worksheet containing the partitioning results will not be inserted into the workbook. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

7. Click **Rescale Data**, to open the Rescaling Dialog. Rescaling is used to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. *See the important note related to Rescale Data and the new Simulation tab in the Options section (below) in this chapter.*

For more information on rescaling, see the Rescale Continuous Data section within the Transform Continuous Data chapter, that occurs earlier in this guide.



near Discriminant: Transformation

Canonical Variate Analysis

Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

Advanced

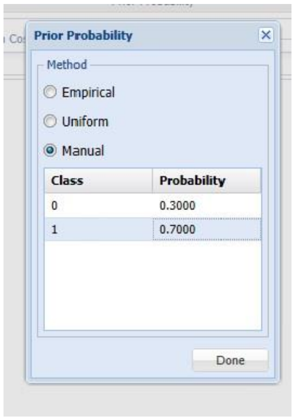
Norm Type: Norm Type

Correction: Correction

Click **Done** to close this dialog without rescaling the data.

8. Click **Prior Probability**. Three options appear in the *Prior Probability* Dialog: *Empirical*, *Uniform* and *Manual.*

Prior Probability dialog



Prior Probability

Method

○ Empirical

○ Uniform

● Manual

Class	Probability
0	0.3000
1	0.7000

If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.

If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.

Select the third option, *Manual*, to manually enter the desired class and probability values of .3 for Class 0 and .7 for Class 1, as shown in the screenshot above.

Click Done to close the dialog.

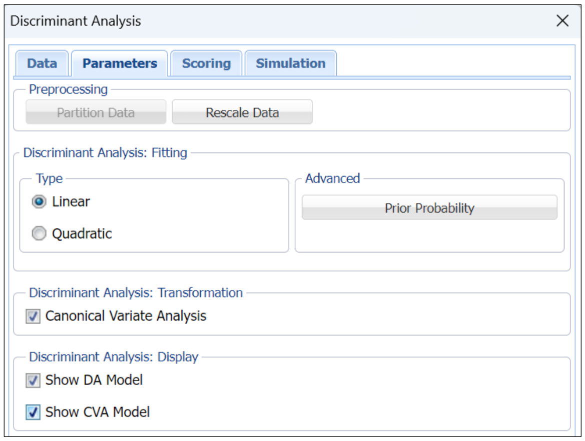
9. Keep the default setting for Type under Discriminant Analysis: Fitting, to use linear discriminant analysis. See the options descriptions below for more information on linear vs quadratic Discriminant Analysis.

10. Select **Canonical Variate Analysis.** When this option is selected, Analytic Solver Data Science produces the canonical variates for the data based on an orthogonal representation of the original variates. This has the effect of choosing a representation which maximizes the distance between the different groups. For a k class problem there are k-1 Canonical variates. Typically, only a subset of the canonical variates is sufficient to discriminate between the classes. For this example, we have two canonical variates which means that if we replace the four original predictors by just two predictors, X_1 and X_2, (which are actually linear combinations of the four original predictors) the discrimination based on these two predictors will perform just as well as the discrimination based on the original predictors.

11. When *Canonical Variate Analysis* is selected, Show **CVA Model** is enabled. Select this option to produce the Canonical Variates in the output.

12. Select **Show DA Model** to print the Linear Discriminant Functions in the output.

Discriminant Analysis dialog, Parameters tab



Discriminant Analysis

Data | Parameters | Scoring | Simulation

Preprocessing

Partition Data | Rescale Data

Discriminant Analysis: Fitting

Type

● Linear

○ Quadratic

Advanced

Prior Probability

Discriminant Analysis: Transformation

☑ Canonical Variate Analysis

Discriminant Analysis: Display

☑ Show DA Model

☑ Show CVA Model

13. Click **Next** to advance to the *Scoring* tab.

14. Select all four options for **Score Training/Validation data**.

When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the Discriminant Analysis output.

When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the Discriminant Analysis output.

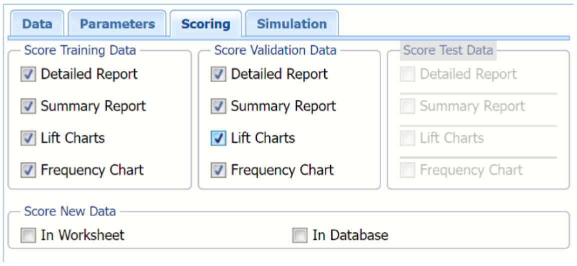
When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and ROC Curve plots in the output.

When Frequency Chart is selected, a frequency chart will be displayed when the DA_TrainingScore and DA_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Since we did not create a test partition, the options for Score test data are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.

Discriminant Analysis dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data

☑ Detailed Report

☑ Summary Report

☑ Lift Charts

☑ Frequency Chart

Score Validation Data

☑ Detailed Report

☑ Summary Report

☑ Lift Charts

☑ Frequency Chart

Score Test Data

□ Detailed Report

□ Summary Report

□ Lift Charts

□ Frequency Chart

Score New Data

□ In Worksheet

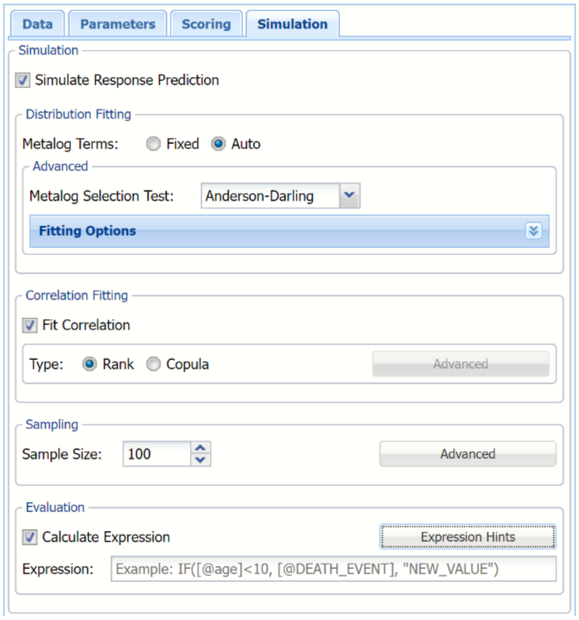
□ In Database

15. Click **Next** to advance to the Simulation tab.

16. Select Simulation Response Prediction to enable all options on the Simulation tab of the Discriminant Analysis dialog. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Anlaytic Solver Upgrade.

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, DA_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type (scale or categorical).

    Discriminant Analysis dialog, Simulation tab



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

☑ Calculate Expression

Expression: Example: IF([@age]<10, [@DEATH_EVENT], "NEW_VALUE")

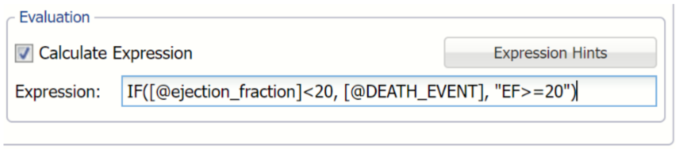
Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the DA_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog. For Expression, enter the following formula to display if the patient suffered catastrophic heart failure (@DEATH_EVENT) when his/her Ejection_Fraction was less than or equal to 20.

IF([@ejection_fraction]<20, [@DEATH_EVENT], “EF>=20”)

Note that variable names are case sensitive.

Evaluation section on the Discriminant Analysis dialog, Simulation tab



Evaluation

☑ Calculate Expression

Expression: IF([@ejection_fraction]<20, [@DEATH_EVENT], "EF>=20")

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

17. Click Finish to run Discriminant Analysis on the example dataset.

Output Worksheets

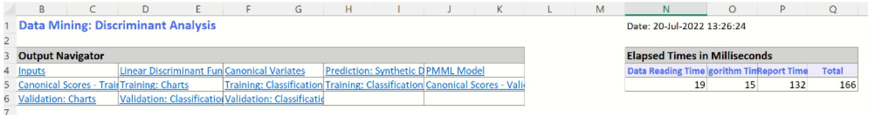
Output sheets containing the Discriminant Analysis results will be inserted into your active workbook to the right of the STDPartition worksheet.

DA_Output

This result worksheet includes 4 segments: Output Navigator, Inputs, Linear Discriminant Functions and Canonical Variates.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

  DA_Output: Output Navigator



Data Mining: Discriminant Analysis

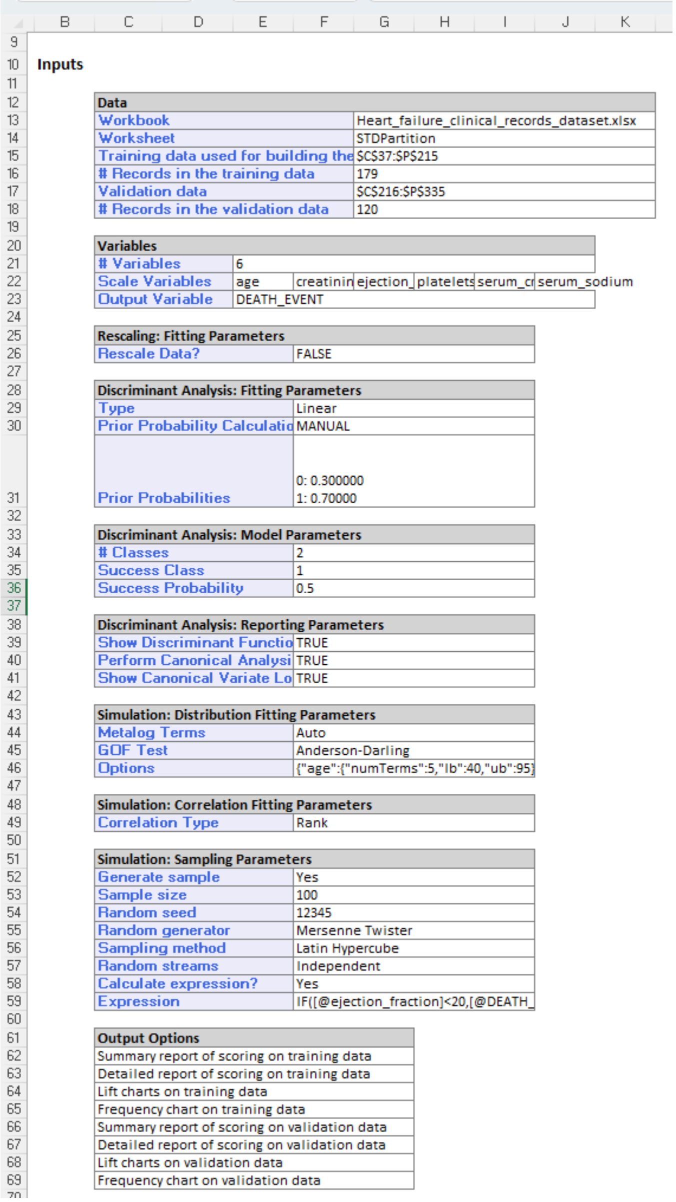
Date: 20-Jul-2022 13:26:24

Output Navigator				
Inputs	Linear Discriminant Fun	Canonical Variates	Prediction: Synthetic D	PMML Model
Canonical Scores - Trai	Training: Charts	Training: Classification	Training: Classification	Canonical Scores - Vali
Validation: Charts	Validation: Classificatio	Validation: Classificatic		

Elapsed Times in Milliseconds			
Data Reading Time	gorithm Tim	Report Time	Total
19	15	132	166

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Discriminant Analysis dialog.

DA_Output: Inputs



Inputs

Data	
Workbook	Heart_failure_clinical_records_dataset.xlsx
Worksheet	STDPartition
Training data used for building the	$C$37:$P$215
# Records in the training data	179
Validation data	$C$216:$P$335
# Records in the validation data	120

Variables						
# Variables	6					
Scale Variables	age	creatinin	ejection_	platelets	serum_cr	serum_sodium
Output Variable	DEATH_EVENT					

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Discriminant Analysis: Fitting Parameters	
Type	Linear
Prior Probability Calculatio	MANUAL
Prior Probabilities	0: 0.300000 1: 0.70000

Discriminant Analysis: Model Parameters	
# Classes	2
Success Class	1
Success Probability	0.5

Discriminant Analysis: Reporting Parameters	
Show Discriminant Functio	TRUE
Perform Canonical Analysi	TRUE
Show Canonical Variate Lo	TRUE

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"age":{"numTerms":5,"lb":40,"ub":95}

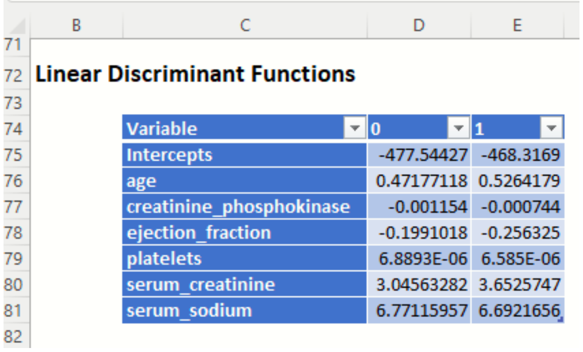
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	Yes
Expression	IF([@ejection_fraction]<20,[@DEATH_

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Linear Discriminant Functions:** In this example, there are 2 functions -- one for each class. Each variable is assigned to the class that contains the higher value.

DA_Output: Linear Discriminant Functions

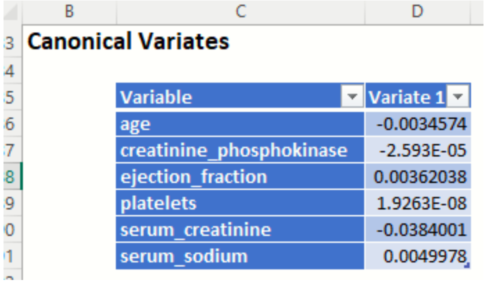


Linear Discriminant Functions

Variable	0	1
Intercepts	-477.54427	-468.3169
age	0.47177118	0.5264179
creatinine_phosphokinase	-0.001154	-0.000744
ejection_fraction	-0.1991018	-0.256325
platelets	6.8893E-06	6.585E-06
serum_creatinine	3.04563282	3.6525747
serum_sodium	6.77115957	6.6921656

- **Canonical Variates:** These functions give a representation of the data that maximizes the separation between the classes. The number of functions is one less than the number of classes (so in this case there is just one function). If we were to plot the cases in this example on a line where xi is the ith case's value for variate1, you would see a clear separation of the data. This output is useful in illustrating the inner workings of the discriminant analysis procedure, but is not typically needed by the end-user analyst.

DA_Output: Canonical Variates



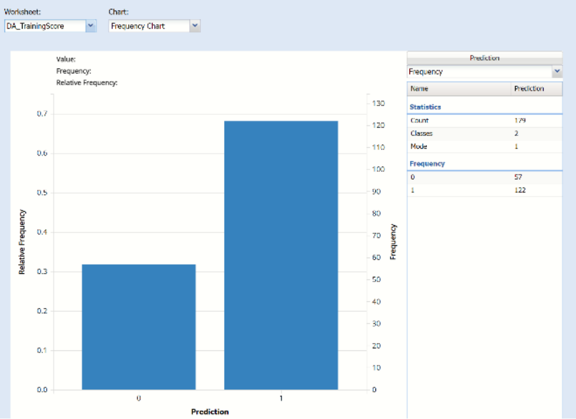
Canonical Variates

Variable	Variate 1
age	-0.0034574
creatinine_phosphokinase	-2.593E-05
ejection_fraction	0.00362038
platelets	1.9263E-08
serum_creatinine	-0.0384001
serum_sodium	0.0049978

DA_TrainingScore

Click the *DA_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Classification Summary and the Training: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: DA_TrainingScore

Chart: Frequency Chart

Prediction

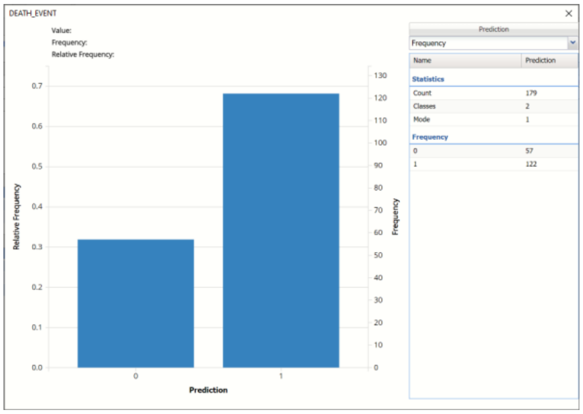
Frequency

Name	Prediction
Statistics	
Count	179
Classes	2
Mode	1
Frequency	
0	57
1	122

- **Frequency Charts:** The output variable frequency chart opens automatically once the *DA_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *DA_TrainingScore* tab.

  Frequency: This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

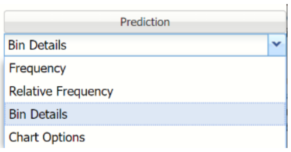
  Frequency Chart on DA_TrainingScore output sheet



Name	Prediction
Statistics	
Count	179
Classes	2
Mode	1
Frequency	
0	57
1	122

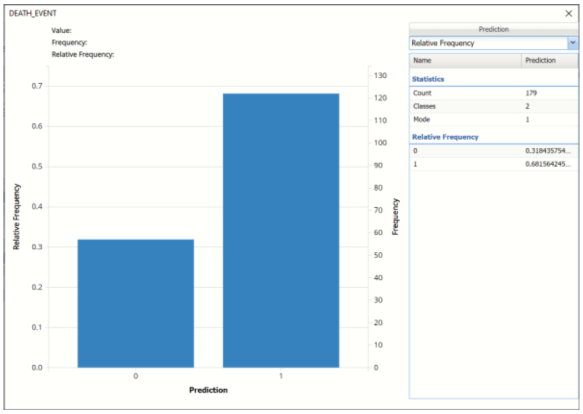
  Click the down arrow next to Frequency to switch to Relative Frequency, Bin Details or Chart Options view.

  Frequency Chart, Frequency View



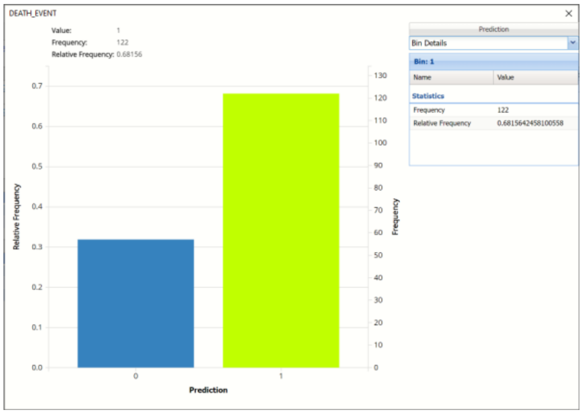
  Relative Frequency: Displays the relative frequency chart.

  Relative Frequency Chart



Name	Prediction
Statistics	
Count	179
Classes	2
Mode	1
Relative Frequency	
0	0.318435754...
1	0.681564245...

  Bin Details: Displays pertinent information pertaining to each bin in the chart.



Value: 1
Frequency: 122
Relative Frequency: 0.68156

Prediction

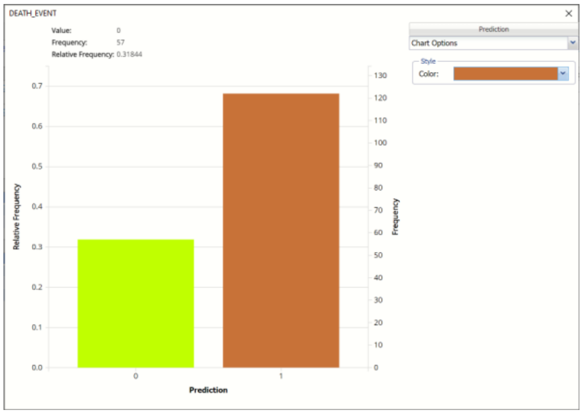
Bin Details

Bin: 1

Name	Value
Statistics	
Frequency	122
Relative Frequency	0.6815642458100558

Chart Options: Use this view to change the color of the bars in the chart.

Chart Options View



Value: 0
Frequency: 57
Relative Frequency: 0.31844

Prediction

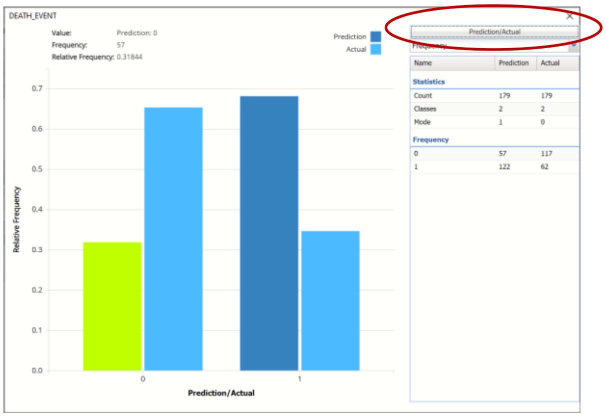
Chart Options

Style

Color:

To see both the actual and predicted frequency, click Prediction and select Actual. This change will be reflected on all charts.

Click Predicted/Actual to change view



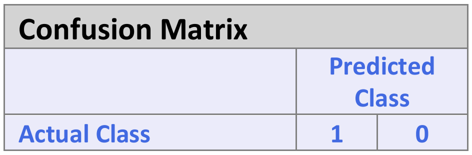
Value: Prediction: 0
Frequency: 57
Relative Frequency: 0.31844

Prediction/Actual

Frequency

Name	Prediction	Actual
Statistics		
Count	179	179
Classes	2	2
Mode	1	0
Frequency		
0	57	117
1	122	62

- **Classification Summary:** In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

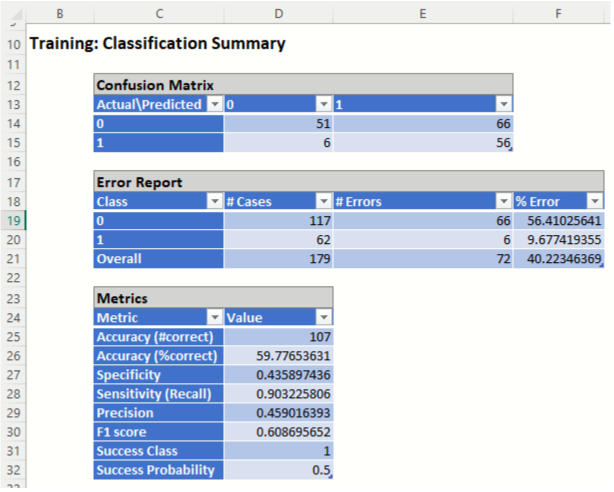


Confusion Matrix		
	Predicted Class	
Actual Class	1	0

1	TP	FN
0	FP	TN

- ○ TP stands for True Positive. These are the number of cases classified as belonging to the Success class that actually were members of the Success class.
- ○ FN stands for False Negative. These are the number of cases that were classified as belonging to the Failure class when they were actually members of the Success class
- ○ FP stands for False Positive. These cases were assigned to the Success class but were actually members of the Failure group
- ○ TN stands for True Negative. These cases were correctly assigned to the Failure group.

DA_TrainingScore: Training: Classification Summary



Training: Classification Summary

Confusion Matrix		
Actual\Predicted	0	1
0	51	66
1	6	56

Error Report			
Class	# Cases	# Errors	% Error
0	117	66	56.41025641
1	62	6	9.677419355
Overall	179	72	40.22346369

Metrics	
Metric	Value
Accuracy (#correct)	107
Accuracy (%correct)	59.77653631
Specificity	0.435897436
Sensitivity (Recall)	0.903225806
Precision	0.459016393
F1 score	0.608695652
Success Class	1
Success Probability	0.5

- True Positive: 56 records belonging to the Success class were correctly assigned to that class
- False Negative: 6 records belonging to the Success class were incorrectly assigned to the Failure class.
- True Negative: 51 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 66 records belonging to the Failure class were incorrectly assigned to the Success class.

The total number of misclassified records was 72 (66 + 6) which results in an error equal to 40.22%.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 59.78% - Refers to the ability of the classifier to predict a class label correctly.
- Specificity: 0.44 - Also called the true negative rate, measures the percentage of failures correctly identified as failures

  Specificity (SPC) or True Negative Rate =TN / (FP + TN)

- Recall (or Sensitivity): 0.90 - Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)

- Precision: 0.46 - The probability of correctly identifying a randomly selected record as one belonging to the Success class

  Precision = TP/(TP+FP)

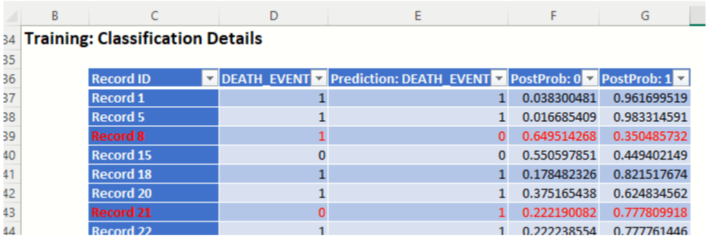
- F-1 Score: 0.61 - Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP / (2 * TP + FP + FN)

- Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

  DA_TrainingScore: Training: Classification Details



Training: Classification Details

Record ID	DEATH_EVENT	Prediction: DEATH_EVENT	PostProb: 0	PostProb: 1
Record 1	1	1	0.038300481	0.961699519
Record 5	1	1	0.016685409	0.983314591
Record 8	1	0	0.649514268	0.350485732
Record 15	0	0	0.550597851	0.449402149
Record 18	1	1	0.178482326	0.821517674
Record 20	1	1	0.375165438	0.624834562
Record 21	0	1	0.222190082	0.777809918
Record 22	1	1	0.222238554	0.777761446

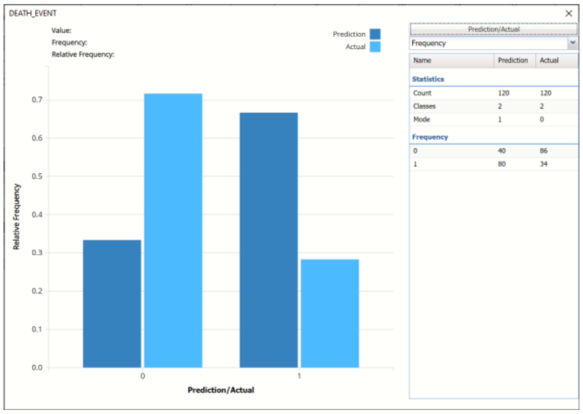
DA_ValidationScore

Click the *DA_ValidationScore* tab to view the newly added Output Variable frequency chart, the Validation: Classification Summary and the Validation: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Validation data.

- **Frequency Charts:** The output variable frequency chart opens automatically once the DA_ValidationScore worksheet is selected. To close this chart, click the “x” in the upper right hand corner. To reopen, click onto another tab and then click back to the DA_ValidationScore tab.

  Click the Frequency chart to display the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode. Selective Relative Frequency from the drop down menu, on the right, to see the relative frequencies of the output variable for both actual and predicted. See above for more information on this chart.

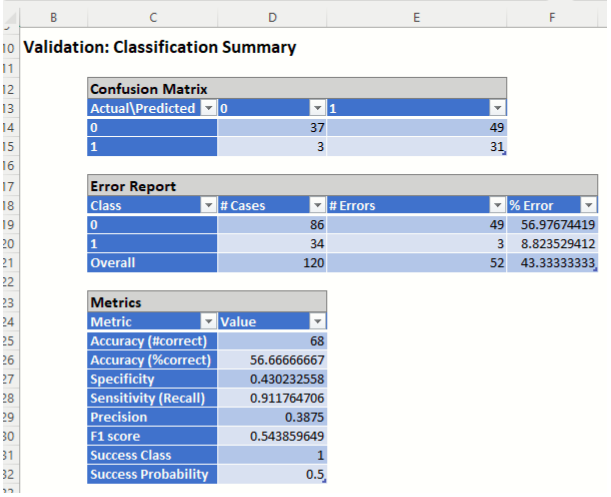
DA_ValidationScore Frequency Chart



Name	Prediction	Actual
Statistics		
Count	120	120
Classes	2	2
Mode	1	0
Frequency		
0	40	86
1	80	34

- **Classification Summary:** This report contains the confusion matrix for the validation data set.

  DA_ValidationScore: Classification Summary



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	37	49
1	3	31

Error Report

Class	# Cases	# Errors	% Error
0	86	49	56.97674419
1	34	3	8.823529412
Overall	120	52	43.33333333

Metrics

Metric	Value
Accuracy (#correct)	68
Accuracy (%correct)	56.66666667
Specificity	0.430232558
Sensitivity (Recall)	0.911764706
Precision	0.3875
F1 score	0.543859649
Success Class	1
Success Probability	0.5

- True Positive: 31 records belonging to the Success class were correctly assigned to that class
- False Negative: 3 records belonging to the Success class were incorrectly assigned to the Failure class.
- True Negative: 37 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 49 records belonging to the Failure class were incorrectly assigned to the Success class.

The total number of misclassified records was 52 (49 + 3) which results in an error equal to 43.33%.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 56.67% - Refers to the ability of the classifier to predict a class label correctly.

    - Specificity: 0.430 - Also called the true negative rate, measures the percentage of failures correctly identified as failures

      Specificity (SPC) or True Negative Rate =TN / (FP + TN)

    - Recall (or Sensitivity): 0.912 - Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

      Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)

    - Precision: 0.388 - The probability of correctly identifying a randomly selected record as one belonging to the Success class

      Precision = TP/(TP+FP)

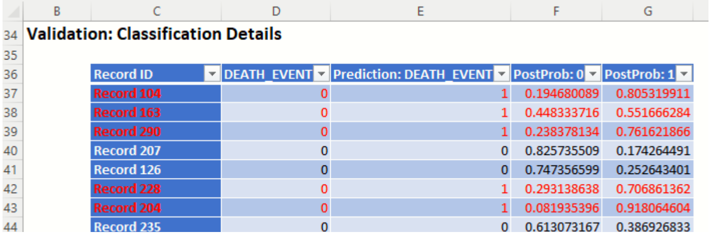
    - F-1 Score: 0.544 - Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

      F1 = 2 * TP / (2 * TP + FP + FN)

    - Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

- **Classification Details**: This table displays how each observation in the validation data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

  DA_ValidationScore: Validation: Classification Details



Validation: Classification Details

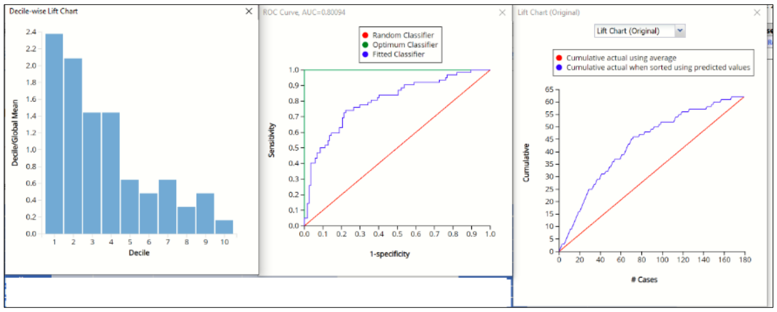
Record ID	DEATH_EVENT	Prediction: DEATH_EVENT	PostProb: 0	PostProb: 1
Record 104	0	1	0.194680089	0.805319911
Record 163	0	1	0.448333716	0.551666284
Record 290	0	1	0.238378134	0.761621866
Record 207	0	0	0.825735509	0.174264491
Record 126	0	0	0.747356599	0.252643401
Record 228	0	1	0.293138638	0.706861362
Record 204	0	1	0.081935396	0.918064604
Record 235	0	0	0.613073167	0.386926833

DA_TrainingLiftChart and DA_ValidationLiftChart

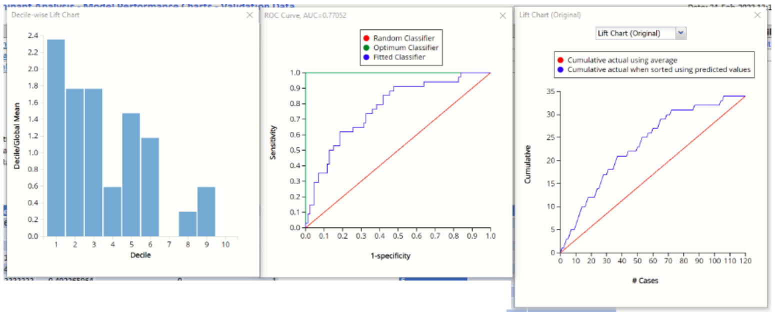
Click **DA_TrainingLiftChart** and **DA_ValidationLiftChart** tabs to navigate to the Training and Validation Data Lift, Decile, ROC Curve and Cumulative Gain Charts.

Lift Charts and ROC Curves are visual aids that help users evaluate the performance of their fitted models. Charts found on the DA_TrainingLiftChart tab were calculated using the Training Data Partition. Charts found on the DA_ValidationLiftChart tab were calculated using the Validation Data Partition. It is good practice to look at both sets of charts to assess model performance on both the Training and Validation partitions.

Decile-wise Lift Chart, ROC Curve, and Lift Charts for Training Partition



Decile-wise Lift Chart, ROC Curve, and Lift Charts for Valid. Partition



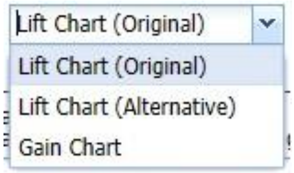
After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted in decreasing order using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the cumulative number of cases in decreasing probability (on the x-axis) vs the cumulative number of true positives on the y-axis. The baseline (red line connecting the origin to the end point of the blue line) is a reference line. For a given number of cases on the x-axis, this line represents the expected number of successes if no model existed, and instead cases were selected at random. This line can be used as a benchmark to measure the performance of the fitted model. The greater the area between the lift curve and the baseline, the better the model. In the Training Lift chart, if we selected 100 cases as belonging to the success class and used the fitted model to pick the members most likely to be successes, the lift curve tells us that we would be right on about 52 of them. Conversely, if we selected 100 random cases, we could expect to be right on about 35 (34.63) of them. In the Validation Lift chart, if we selected 50 cases as belonging to the success class and used the fitted model to pick the members most likely to be successes, the lift curve tells us that we would be right on about 23 of them. Conversely, if we selected 50 random cases, we could expect to be right on about 14 (14.167) of them.

The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. This bars in this chart indicate the factor by which the model outperforms a random assignment, one decile at a time. Records are sorted by their predicted values (scores) and divided into ten equal-sized bins or deciles. The first decile contains 10% of patients that are most likely to experience catastrophic heart failure. The 10th or last decile contains 10% of the patients that are least likely to experience catastrophic heart failure. Ideally, the decile wise lift chart should resemble a stair case with the 1st decile as the tallest bar, the 2nd decile as the 2nd tallest, the 3rd decile as the 3rd tallest, all the way down to the last or 10th decile as the smallest bar. This "staircase" conveys that the model "binned" the records, or in this case patients, correctly from most likely to experience catastrophic heart failure to least likely to experience catastrophic heart failure.

In this particular example, neither chart exhibits the desired “stairstep” effect. Rather, in the training partition, bars 3 and 4 are “even” and bars 7 and 9 are larger than bars 8 and 10. The decile wise validation chart is not much better as the heights for bars 2 and 3 are “even”, bars 5 and 6 are both larger than 4 and bar 9 is larger than 8. In other words, the model appears to do a decent job of identifying the patients at most risk of experiencing catastrophic heart failure but the predictive power of the model begins to fade for patients not exhibiting strong symptoms of heart failure.

The Regression ROC curve was updated in V2017. This new chart compares the performance of the regressor (Fitted Classifier) with an Optimum Classifier Curve and a Random Classifier curve. The Optimum Classifier Curve plots a hypothetical model that would provide perfect classification results. The best possible classification performance is denoted by a point at the top left of the graph at the intersection of the x and y axis. This point is sometimes referred to as the “perfect classification”. The closer the AUC is to 1, the better the performance of the model. In the Validation Partition, AUC = .77052 which suggests that this fitted model is not a good fit to the data.

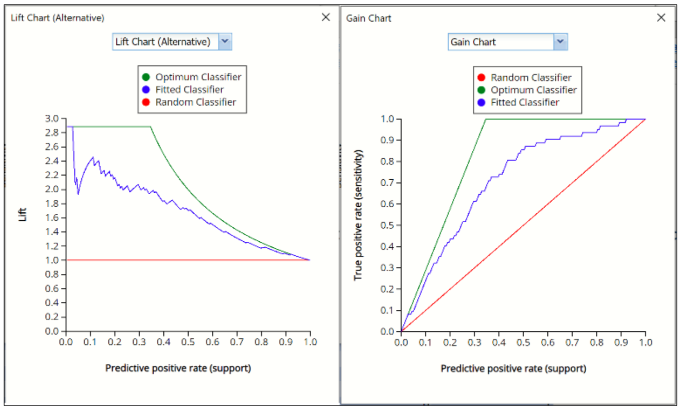
In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.



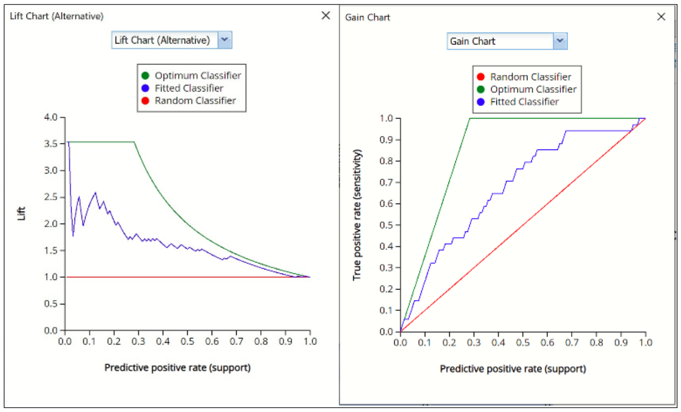
Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Classifier curve, a Fitted Classifier curve, and a Random Classifier curve. The Optimum Classifier curve plots a hypothetical model that would provide perfect classification for our data. The Fitted Classifier curve plots the fitted model and the Random Classifier curve plots the results from using no model or by using a random guess (i.e. for x% of selected observations, x% of the total number of positive observations are expected to be correctly classified).

The Alternative Lift Chart plots Lift against the Predictive Positive Rate or Support.

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition

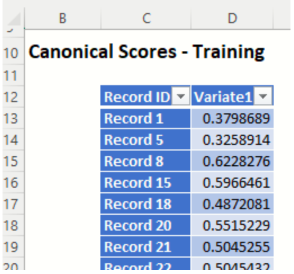


Click the down arrow and select Gain Chart from the menu. In this chart, the True Positive Rate or Sensitivity is plotted against the Predictive Positive Rate or Support.

DA_TrainingCanScores and DA_ValidationCanScores

Click the **Canonical Scores – Training** link in the Output Navigator to navigate to the *DA_TrainingCanScores* worksheet. Canonical Scores are the values of each case for the function. These are intermediate values useful for illustration but are not usually required by the end-user analyst. Canonical Scores are also available for the Validation dataset on the *DA_ValidationCanScores* sheet.

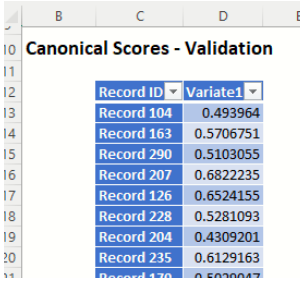
Canonical Scores for Training Partition



Canonical Scores - Training

Record ID	Variate1
Record 1	0.3798689
Record 5	0.3258914
Record 8	0.6228276
Record 15	0.5966461
Record 18	0.4872081
Record 20	0.5515229
Record 21	0.5045255
Record 22	0.5045432

Canonical Scores for Validation Partition



Canonical Scores - Validation

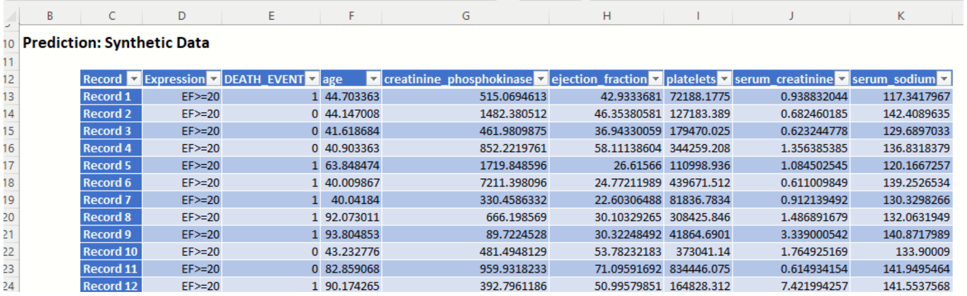
Record ID	Variate1
Record 104	0.493964
Record 163	0.5706751
Record 290	0.5103055
Record 207	0.6822235
Record 126	0.6524155
Record 228	0.5281093
Record 204	0.4309201
Record 235	0.6129163
Record 170	[illegible]

DA_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, DA_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Discriminant Analysis dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data (with or without correlation fitting), the prediction (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted (Simulation)/Predicted (Training) or Expression (Simulation)/Expression (Training) or a combination of two, as long as they are of the same type.

Synthetic Data



Prediction: Synthetic Data

Record	Expression	DEATH_EVENT	age	creatinine_phosphokinase	ejection_fraction	platelets	serum_creatinine	serum_sodium
Record 1	EF>=20	1	44.703363	515.0694613	42.9333681	72188.1775	0.938832044	117.3417967
Record 2	EF>=20	0	44.147008	1482.380512	46.35380581	127183.389	0.682460185	142.4089635
Record 3	EF>=20	0	41.618684	461.9809875	36.94330059	179470.025	0.623244778	129.6897033
Record 4	EF>=20	0	40.903363	852.2219761	58.11138604	344259.208	1.356385385	136.8318379
Record 5	EF>=20	1	63.848474	1719.848596	26.61566	110998.936	1.084502545	120.1667257
Record 6	EF>=20	1	40.009867	7211.398096	24.77211989	439671.512	0.611009849	139.2526534
Record 7	EF>=20	1	40.04184	330.4586332	22.60306488	81836.7834	0.912139492	130.3298266
Record 8	EF>=20	1	92.073011	666.198569	30.10329265	308425.846	1.486891679	132.0631949
Record 9	EF>=20	1	93.804853	89.7224528	30.32248492	41864.6901	3.339000542	140.8717989
Record 10	EF>=20	0	43.232776	481.4948129	53.78232183	373041.14	1.764925169	133.90009
Record 11	EF>=20	0	82.859068	959.9318233	71.09591692	834446.075	0.614934154	141.9495464
Record 12	EF>=20	1	90.174265	392.7961186	50.99579851	164828.312	7.421994257	141.5537568

Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the Discriminant Analysis dialog

IF([@ejection_fraction]<20, [@DEATH_EVENT], “EF>=20”)

The results in this column are either 0, 1, or EF > 20.

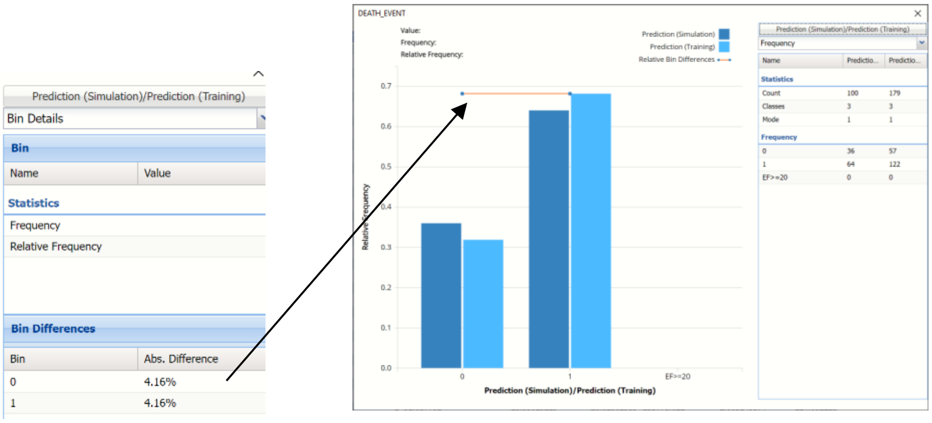
- DEATH_EVENT = 0 indicates that the patient had an ejection_fraction <= 20 but did not suffer catastrophic heart failure.
- DEATH_EVENT = 1 in this column indicates that the patient had an ejection_fraction <= 20 and did suffer catastrophic heart failure.
- EF>20 indicates that the patient had an ejection fraction of greater than 20.

The remainder of the data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide. Note: If the data had been rescaled, i.e. Rescale Data was selected on the Parameters tab, the data shown in this table would have been fit using the rescaled data.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the actual data, the synthetic data and the expression, if it exists. In the screenshot below, the bars in the darker shade of blue are based on the *synthetic* data. The bars in the lighter shade of blue are based on the *training* data.

In the synthetic data (the columns in the darker shade of blue), about 36% of patients survived while about 64% of patients succumbed to the complications of heart failure and in the training partition, about 32% patients survived while about 68% of the patients did not.

Frequency Chart for DA_Simulation output



Prediction (Simulation)/Prediction (Training)

Frequency

Name	Predictio...	Predictio...
Statistics		
Count	100	179
Classes	3	3
Mode	1	1
Frequency		
0	36	57
1	64	122
EF>=20	0	0

Prediction (Simulation)/Prediction (Training)

Bin Details

Bin

Name	Value
Statistics	
Frequency	
Relative Frequency	

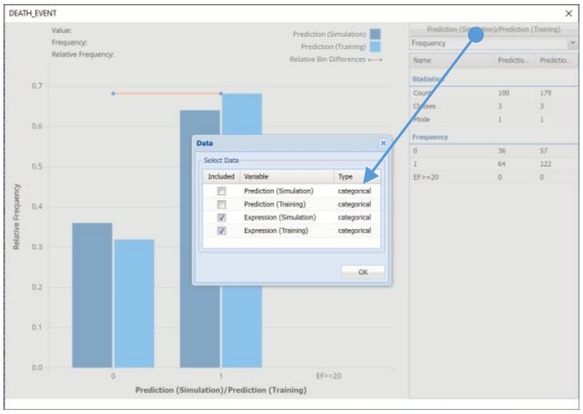
Bin Differences

Bin	Abs. Difference
0	4.16%
1	4.16%

Notice that the Relative Bin Difference curve is flat. Click the down arrow next to Frequency and select Bin Details. This view explains why the curve is flat. Notice that the absolute difference for both bins is 4.16%.

Click *Prediction (Simulation)/Prediction (Training)* and uncheck Prediction (Simulation)/Prediction (Training) and select Expression (Simulation)/Expression (Training) to change the chart view.

Click Expression (Simulation)/Expression (Training) to change the Data view



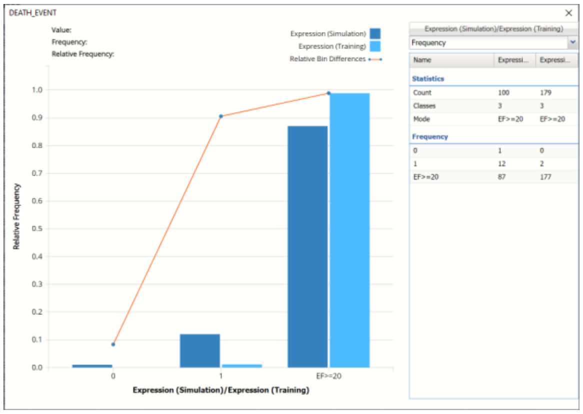
Data

Select Data

Included	Variable	Type
□	Prediction (Simulation)	categorical
□	Prediction (Training)	categorical
☑	Expression (Simulation)	categorical
☑	Expression (Training)	categorical

The chart displays the results of the expression in both datasets. This chart shows that 2 patients with an ejection fraction less than 20 are predicted to survive and 1 patient is not.

Frequency Chart with Expression column

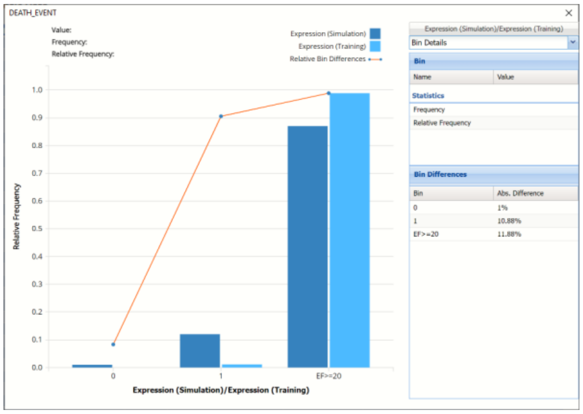


Expression (Simulation)/Expression (Training)

Frequency

Name	Expressi...	Expressi...
Statistics		
Count	100	179
Classes	3	3
Mode	EF>=20	EF>=20
Frequency		
0	1	0
1	12	2
EF>=20	87	177

Click the down arrow next to Frequency to change the chart view to Bin Details.



Bin	Abs. Difference
0	1%
1	10.88%
EF>=20	11.88%

Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

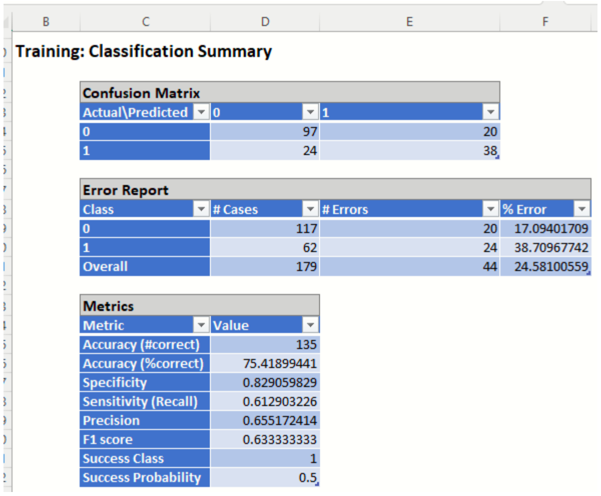
Using Quadratic Discriminant Analysis

It is advisable to try both techniques (linear and quadratic) to determine which one performs best on your model. You can easily switch between LDA and QDA simply by selecting the appropriate option on the Parameters tab of the Discriminant Analysis dialog.

Click back to the **STDPartition** worksheet and open the Discriminant Analysis dialog by clicking **Classify – Discriminant Analysis**.

Click the **Parameters tab** and select **Quadratic** for *Type* to run Quadratic Discriminant analysis.

Click **Finish** to leave all option settings as selected in the earlier example. Click the DA_TrainingScore1 worksheet to view the Training: Classification Summary. Notice immediately that the %Error has been reduced to 24.58%. If you click the DA_ValidationScore1 worksheet, notice that the %Error in the validation set has decreased to 30.83%.



Training: Classification Summary

Confusion Matrix

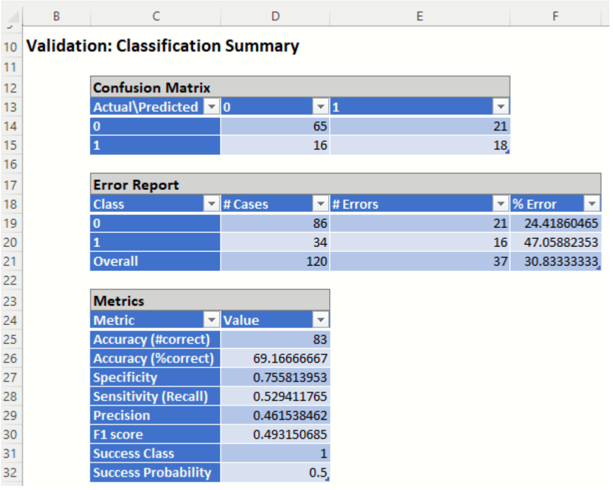
Actual\Predicted	0	1
0	97	20
1	24	38

Error Report

Class	# Cases	# Errors	% Error
0	117	20	17.09401709
1	62	24	38.70967742
Overall	179	44	24.58100559

Metrics

Metric	Value
Accuracy (#correct)	135
Accuracy (%correct)	75.41899441
Specificity	0.829059829
Sensitivity (Recall)	0.612903226
Precision	0.655172414
F1 score	0.633333333
Success Class	1
Success Probability	0.5



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	65	21
1	16	18

Error Report

Class	# Cases	# Errors	% Error
0	86	21	24.41860465
1	34	16	47.05882353
Overall	120	37	30.83333333

Metrics

Metric	Value
Accuracy (#correct)	83
Accuracy (%correct)	69.16666667
Specificity	0.755813953
Sensitivity (Recall)	0.529411765
Precision	0.461538462
F1 score	0.493150685
Success Class	1
Success Probability	0.5

In this instance, quadratic discriminant analysis has resulted in smaller misclassification errors in both the training and validation partitions.

For information on Stored Model Sheets, in this example *DA_Stored*, please refer to the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide.

Discriminant Analysis Options

See below for an explanation of options on all four tabs of the Discriminant *Analysis (*DA) dialog*: Data, Parameters, Scoring and Simulation.*

The following options appear on all four tabs of the Discriminant Analysis dialog.

Help: Click the Help button to access documentation on all *Discriminant Analysis* options.

Cancel: Click the Cancel button to close the dialog without running *Discriminant Analysis.*

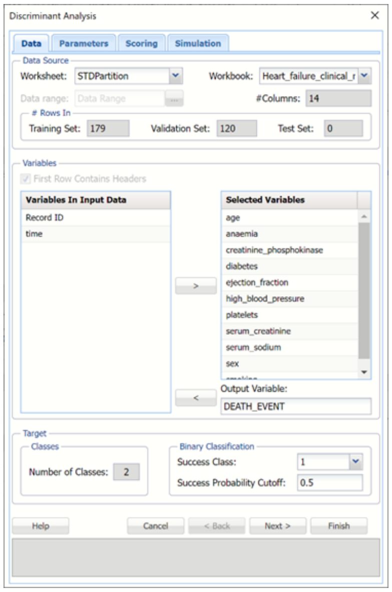
Next: Click the Next button to advance to the next tab.

Finish: Click Finish to accept all option settings on all dialogs, and run *Discriminant Analysis.*

Discriminant Analysis Data Tab

See below for documentation for all options appearing on the Data tab.

Discriminant Analysis Data Tab


Discriminant Analysis

Data Source

Worksheet: STDPartition

Workbook: Heart_failure_clinical_r

Data range: Data Range

#Columns: 14

# Rows In

Training Set: 179

Validation Set: 120

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
time

Selected Variables
age
anaemia
creatinine_phosphokinase
diabetes
ejection_fraction
high_blood_pressure
platelets
serum_creatinine
serum_sodium
sex

Output Variable: DEATH_EVENT

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Data Source

Worksheet: Click the down arrow to select the desired worksheet where the dataset is contained.

Workbook: Click the down arrow to select the desired workbook where the dataset is contained.

Data range: Select or enter the desired data range within the dataset. This data range may either be a portion of the dataset or the complete dataset.

#Columns: Displays the number of columns in the data range. This option is read only.

#Rows In: Training Set, Validation Set, Test Set: Displays the number of rows in each partition, if it exists. This option is read only.

Variables

First Row Contains Headers: Select this checkbox if the first row in the dataset contains column headings.

Variables In Input Data: This field contains the list of the variables, or features, included in the data range.

Selected Variables: This field contains the list of variables, or features, to be included in DA.

- To include a variable in DA, select the variable in the Variables In Input Data list, then click the upper > to move the variable to the Selected Variables list.
- To remove a variable as a selected variable, click the variable in the Selected Variables list, then click the upper < to move the variable back to the Variables In Input Data list.

Output Variable: The selected output variable is displayed here.

- To select the output variable (required), select the variable in the Variables In Input Data list, then click the lower > to move the variable to the Output Variable field.
- To remove the output variable, click < to move the variable back to the Variables In Input Data list.

Number of Classes

(Read Only) This value is the number of classes in the output variable.

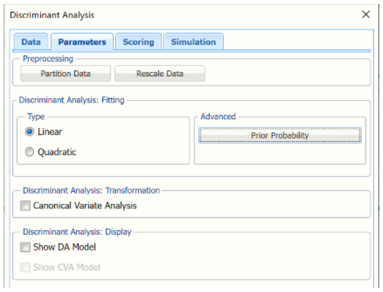
Binary Classification

Set the Success Class and the Success Probability Cutoff here.

Success Class: This option is selected by default. Select the class to be considered a "success" or the significant class in the Lift Chart. This option is enabled when the number of classes in the output variable is equal to 2.

Success Probability Cutoff: Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a "success" (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a "non-success" (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

Discriminant Analysis dialog, Parameters tab



Discriminant Analysis

Data | Parameters | Scoring | Simulation

Preprocessing

Partition Data | Rescale Data

Discriminant Analysis: Fitting

Type

● Linear

○ Quadratic

Advanced

Prior Probability

Discriminant Analysis: Transformation

□ Canonical Variate Analysis

Discriminant Analysis: Display

□ Show DA Model

□ Show CVA Model

Discriminant Analysis, Parameters Tab

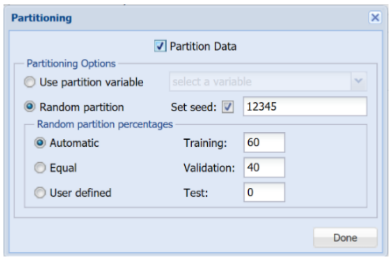
See below for documentation for all options appearing on the Parameters tab.

Preprocessing

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Partitioning dialog



Partitioning

☑ Partition Data

Partitioning Options

○ Use partition variable   select a variable

● Random partition   Set seed: ☑ 12345

Random partition percentages

● Automatic   Training: 60

○ Equal   Validation: 40

○ User defined   Test: 0

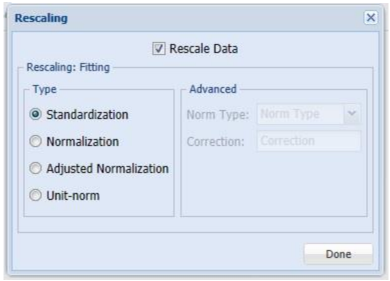
Done

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted

Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Rescaling dialog



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

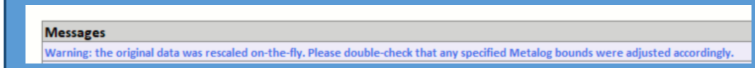
Advanced

Norm Type: Norm Type

Correction: Correction

Notes on Rescaling and the Simulation functionality

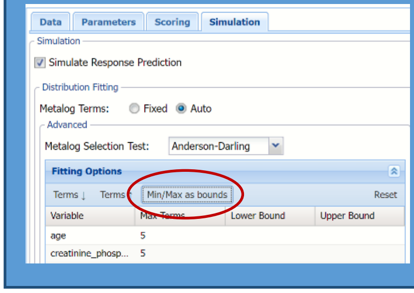
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot to the left, then "Min/Max as bounds" on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the DA_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the "Min/Max as bounds" button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the "Min/Max as bounds" feature is available for the user's convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

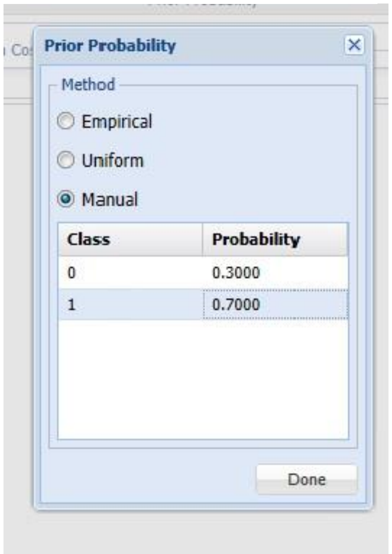
Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Prior Probability dialog



Prior Probability

Method

○ Empirical

○ Uniform

● Manual

Class	Probability
0	0.3000
1	0.7000

Prior Probability

Click **Prior Probability** to open the dialog to the left. Three options appear in the *Prior Probability* Dialog: *Empirical*, *Uniform* and *Manual*.

- If the first option is selected, Empirical, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
- If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.
- Select the third option, *Manual*, to manually enter the desired probability for each class. Probabilities must sum up to 1.

Type: Quadratic or Linear

Discriminant analysis assumes that:

1. The data is normally distributed.
2. Means of each class are specific to that class.
3. All classes have a common covariance matrix.

If these assumptions are realized, DA generates a linear decision boundary.

The latest version of Analytic Solver Data Science now contains Quadratic Discriminant Analysis (QDA). QDA produces a quadratic decision boundary, rather than a linear decision boundary. While QDA also assumes that the data is normally distributed, QDA does *not* assume that all classes share the same covariance matrix.

QDA is a more flexible technique when compared to LDA. QDA's performance improves over LDA when the class covariance matrices are disparate. Since each class has a different covariance matrix, the number of parameters that must be estimated increases significantly as the number of dimensions (predictors) increase. As a result, LDA might be a better choice over QDA on datasets with small numbers of observations and large numbers of classes. It's advisable to try both techniques to determine which one performs best on your model. You can easily switch between LDA and QDA simply by setting this option to linear or quadratic.

Canonical Variate Analysis

When this option is selected, Analytic Solver Data Science produces the canonical variates for the data based on an orthogonal representation of the original variates and sends them to the output sheets, DA_TrainingCanScores (for the training partition), DA_ValidationCanScores (for the validation partition) and DA_TestCanScores (for the test partition). This has the effect of choosing a representation which maximizes the distance between the different groups. For a k class problem there are k-1 Canonical variates. Typically, only a subset of the canonical variates is sufficient to discriminate between the classes. For this example, we have two canonical variates which means that if we replace the four original predictors by just two predictors, X_1 and X_2, (which are actually linear combinations of the four original predictors) the discrimination based on these two predictors will perform just as well as the discrimination based on the original predictors.

Discriminant Model Display

Show CVA Model

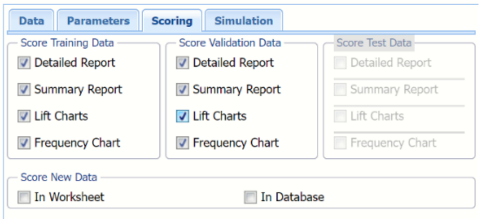
When *Canonical Variate Analysis* is selected, **Show CVA Model** is enabled. Select this option to produce the Canonical Variates in the output.

Show DA Model

Select this option to display the functions that define each class in the output.

Discriminant Analysis, Scoring Tab

Discriminant Analysis Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

Score Training Data

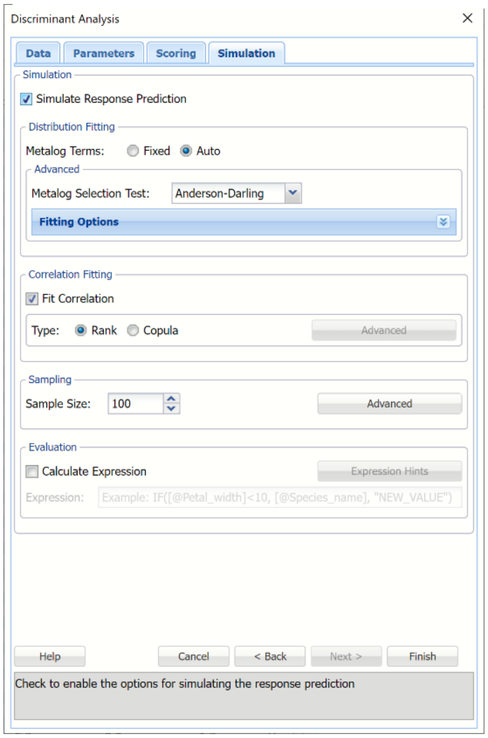
Select these options to show an assessment of the performance of the Discriminant Analysis algorithm in classifying the training data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories.

When Frequency Chart is selected, a frequency chart will be displayed when the DA_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Discriminant Analysis algorithm in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the DA_ValidationScore worksheet is selected.

Discriminant Analysis Simulation tab



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

☑ Fit Correlation

Type: ● Rank ○ Copula

Sample Size: 100

□ Calculate Expression

Expression: Example: IF([@Petal_width]<10, [@Species_name], "NEW_VALUE")

Check to enable the options for simulating the response prediction

Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Discriminant Analysis algorithm in classifying the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the DA_TestScore worksheet is selected.

Score New Data

See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Simulation Tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, DA_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Prediction (Simulation)/Prediction (Training) or Expression (Simulation)/Expression (Training) sources or a combination of any pair may be viewed, if the charts are of the same type. *Check Simulate Response Prediction to enable the options on the Simulation tab.*

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the DA_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Logistic Regression

Introduction

Logistic Regression is a regression model where the dependent (target) variable is categorical. Analytic Solver Data Science provides the functionality to fit a Logistic Model for binary classification problems, i.e. where the dependent variable contains exactly two classes. The fitted model can be used to estimate the posterior probability of the binary outcome based on one or more predictors (features or independent variables). Examples of such binary outcomes could be a college acceptance or rejection, loan application approval or rejection, or classification of a tumor being benign or cancerous.

Logistic Regression is a popular and powerful classification method widely used in various fields due to the model's simplicity and high interpretability. Analytic Solver Data Science implements highly efficient algorithms for Logistic Regression fitting and scoring procedures, which makes this method applicable for large datasets. It's important to note that Logistic Regression is a linear model and cannot capture the non-linear relationships in the data.

Technically, the Logistic Regression fitting procedure aims to fit the coefficients (b_i) of a linear combination of predictor variables (X_i) to estimate the log odds of the binary outcome, i.e. a logit transformation of probability of a particular outcome (p).

Note the similarity between the formulations of Linear and Logistic Regression. Both define the response as a linear combination of predictor variables. However, the linear model predicts a continuous response, which can take any real value, while Logistic Regression requires a response (probability) to be bounded in [0,1] range. This is achieved through the logit transformation as shown below.

$$logit(p) = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \ldots + b_kX_k$$

$$logit(p) = \ln\left(\frac{p}{1-p}\right)$$

Logistic Regression Example

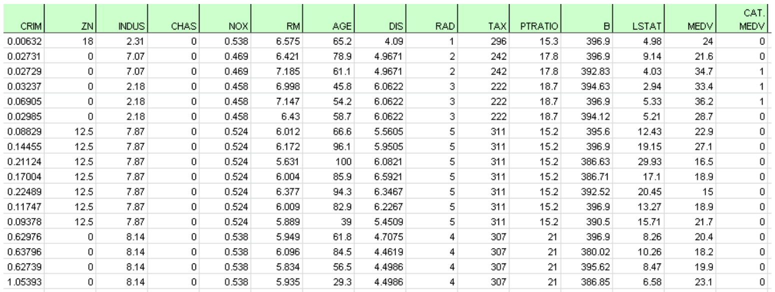
This example illustrates how to fit a model using Analytic Solver Data Science's Logistic Regression algorithm using the Boston_Housing dataset by developing a model for predicting the median price of a house in a census track in the Boston area.

Click **Help – Example Models** on the Data Science ribbon, then **Forecasting/Data Science Examples** and open the example file, **Boston_Housing.xlsx**.

This dataset includes fourteen variables pertaining to housing prices from census tracts in the Boston area collected by the US Census Bureau.

The figure below displays a portion of the data; observe the last column (CAT. MEDV). This variable has been derived from the MEDV variable by assigning a 1 for MEDV levels above 30 (>= 30) and a 0 for levels below 30 (<30) and will not be used in this example.

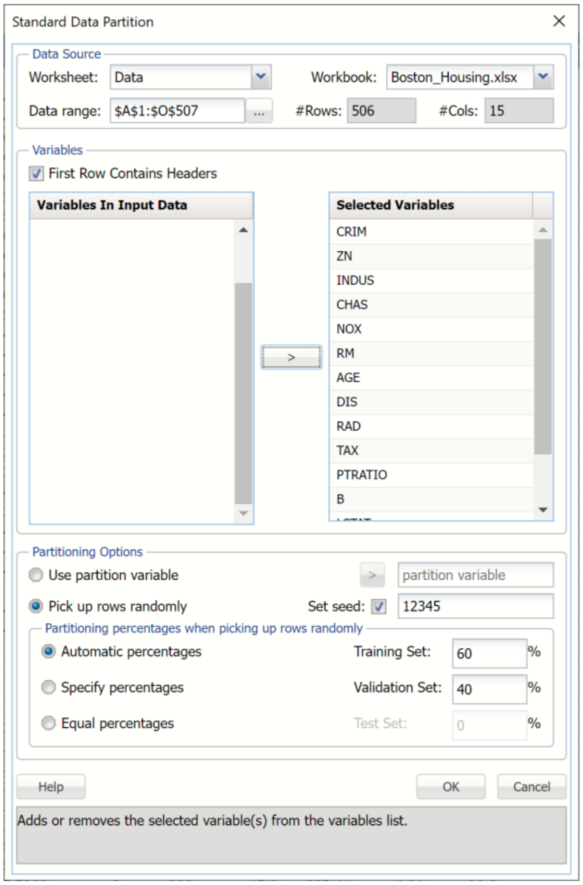
All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described in the What's New section of this guide and then more in depth in the Analytic Solver Data Science Reference Guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CFBM_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type. Since this new functionality does not support categorical variables, these types of variables will not be present in the model, only continuous variables.



CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT. MEDV
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24	0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	0
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	1
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	1
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	1
0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7	0
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	0
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	0
0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	0
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	0
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	0
0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	0
0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	0
0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	0
0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	0
0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	0
1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	0

Inputs

1. First, we partition the data into training and validation sets using the Standard Data Partition defaults of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.



Data Source

Worksheet: Data

Workbook: Boston_Housing.xlsx

Data range: $A$1:$O$507

#Rows: 506

#Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	CRIM
	ZN
	INDUS
	CHAS
	NOX
	RM
	AGE
	DIS
	RAD
	TAX
	PTRATIO
	B

Partitioning Options

○ Use partition variable   partition variable

● Pick up rows randomly   Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages   Training Set: 60 %

○ Specify percentages   Validation Set: 40 %

○ Equal percentages   Test Set: 0 %

Adds or removes the selected variable(s) from the variables list.

2. Click **Classify – Logistic Regression** on the Data Science ribbon. The *Logistic Regression* dialog appears.

3. The categorical variable CAT.MEDV has been derived from the MEDV variable (Median value of owner-occupied homes in $1000's) a 1 for MEDV levels above 30 (>= 30) and a 0 for levels below 30 (<30). This will be our Output Variable.

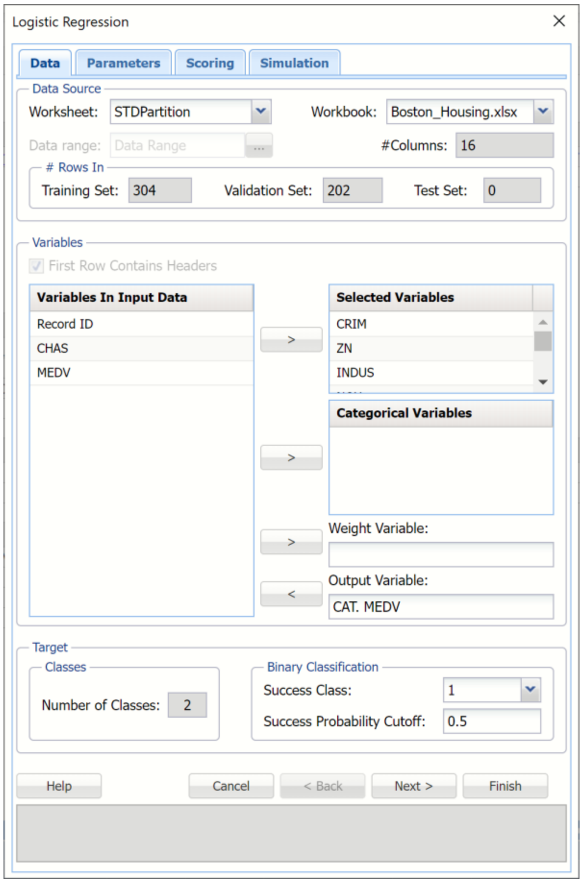
4. Select the nominal categorical variable, CHAS, as a Categorical Variable. This variable is a 1 if the housing tract is located adjacent to the Charles River.

   Select the remaining variables, except Record ID, CHAS and MEDV, as Selected Variables. Since this example showcases the newly added Simulation tab example, no categorical variables will be included in the model. (Recall that Simulation tab functionality does not support categorical variables.)

One major assumption of Logistic Regression is that each observation provides equal information. Analytic Solver Data Science offers an opportunity to provide a *Weight Variable*. Using a *Weight Variable* allows the user to allocate a weight to each record. A record with a large weight will influence the model more than a record with a smaller weight. For the purposes of this example, a *Weight Variable* will not be used.

5. Choose the value that will be the indicator of “Success” by clicking the down arrow next to *Success Class*. In this example, we will use the default of 1.

6. Enter a value between 0 and 1 for *Success Probability Cutoff*. If this value is less than this value, then a 0 will be entered for the class value, otherwise a 1 will be entered for the class value. In this example, we will keep the default of 0.5.



Logistic Regression

Data Source

Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Rows In

Training Set: 304

Validation Set: 202

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
MEDV

Selected Variables
CRIM
ZN
INDUS

Categorical Variables

Weight Variable:

Output Variable: CAT. MEDV

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

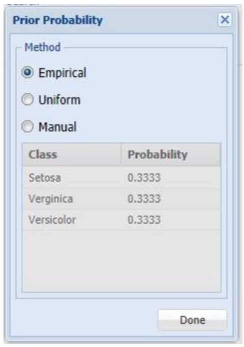
7. Click **Next** to advance to the *Logistic Regression - Parameters* tab.

Analytic Solver Data Science includes the ability to partition or rescale a dataset from within a classification or prediction method by selecting Partition Data or Rescale Data on the Parameters tab. Analytic Solver Data Science will rescale and/or partition your dataset (according to the rescaling and partition options you set) immediately before running the classification method. If partitioning or rescaling has already occurred on the dataset, the option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter. For more information on rescaling your data, see the Transform Continuous Data chapter. Both chapters occur earlier in this guide.

8. Keep *Fit Intercept* selected, the default setting, to fit the Logistic Regression intercept. If this option is not selected, Analytic Solver Data Science will force the intercept term to 0.

9. Keep the default of **50** for the *Iterations*. Estimating the coefficients in the Logistic Regression algorithm requires an iterative non-linear maximization procedure. You can specify a maximum number of iterations to prevent the program from getting lost in very lengthy iterative loops. This value must be an integer greater than 0 or less than or equal to 100 (1< value <= 100).

10. Click **Prior Probability** to open the Prior Probability dialog.



Prior Probability

Method

● Empirical

○ Uniform

○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

11. Analytic Solver Data Science will incorporate prior assumptions about how frequently the different classes occur in each of the partitions.

    - If Empirical is selected, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
    - If Uniform is selected, Analytic Solver Data Science will assume that all classes occur with equal probability.
    - If Manual is selected, the user can enter the desired class and probability value.

    For this example, click **Done** to select the default of Empirical and close the dialog.

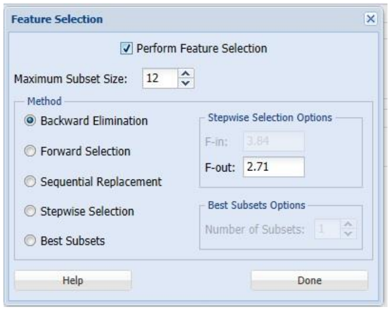
12. Select **Variance – Covariance Matrix.** When this option is selected, Analytic Solver Data Science will display the coefficient covariance matrix in the output. Entries in the matrix are the covariances between the indicated coefficients. The "on-diagonal" values are the estimated variances of the corresponding coefficients.

13. Select **Multicollinearity Diagnostics**. At times, variables can be highly correlated with one another which can result in large standard errors for the affected coefficients. Analytic Solver Data Science will display information useful in dealing with this problem if *Multicollinearity Diagnostics* is selected.

14. Select **Analysis of Coefficients**. When this option is selected, Analytic Solver Data Science will produce a table with all coefficient information such as the Estimate, Odds, Standard Error, etc. When this option is not selected, Analytic Solver Data Science will only print the Estimates.

15. When you have a large number of predictors and you would like to limit the model to only the significant variables, click **Feature Selection** to open the *Feature Selection* dialog and select **Perform Feature Selection** at the top of the dialog. Keep the default selection of 12 for *Maximum Subset Size*. This option can take on values of 1 up to N where N is the number of Selected Variables. The default setting is N.

    Note: If any categorical variables exist in the model, the default setting for Maximum Subset Size will be 15. Categorical variables are expanded into a number of new columns using "one-hot-encoding" (Create Dummies) before Logistic Regression is started. As a result, the default value of 15 is set in this dialog and no upper bound for Maximum Subset Size is enforced as would be if only continuous variables were to appear in the model.



Feature Selection

☑ Perform Feature Selection

Maximum Subset Size: 12

Method

● Backward Elimination

○ Forward Selection

○ Sequential Replacement

○ Stepwise Selection

○ Best Subsets

Stepwise Selection Options

F-in: 3.84

F-out: 2.71

Best Subsets Options

Number of Subsets: 1

Analytic Solver Data Science offers five different selection procedures for selecting the best subset of variables.

- *Backward Elimination* in which variables are eliminated one at a time, starting with the least significant. If this procedure is selected, FOUT is enabled. A statistic is calculated when variables are eliminated. For a variable to leave the regression, the statistic's value must be less than the value of FOUT (default = 2.71).

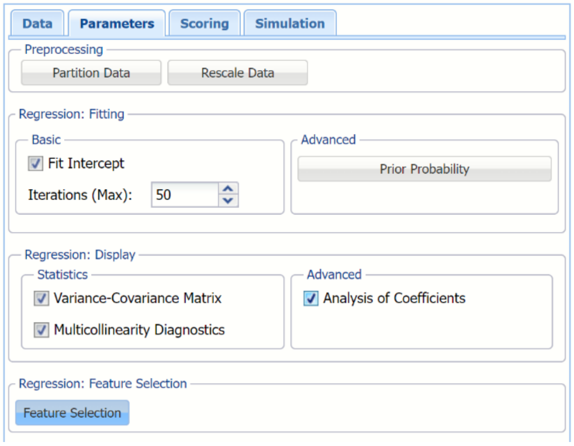
- *Forward Selection* in which variables are added one at a time, starting with the most significant. If this procedure is selected, FIN is enabled. On each iteration of the Forward Selection procedure, each variable is examined for the eligibility to enter the model. The significance of variables is measured as a partial F-statistic. Given a model at a current iteration, we perform an F Test, testing the null hypothesis stating that the regression coefficient would be zero if added to the existing set if variables and an alternative hypothesis stating otherwise. Each variable is examined to find the one with the largest partial F-Statistic. The decision rule for adding this variable into a model is: Reject the null hypothesis if the F-Statistic for this variable exceeds the critical value chosen as a threshold for the F Test (FIN value), or Accept the null hypothesis if the F-Statistic for this variable is less than a threshold. If the null hypothesis is rejected, the variable is added to the model and selection continues in the same fashion, otherwise the procedure is terminated.

- *Sequential Replacement* in which variables are sequentially replaced and replacements that improve performance are retained.

- *Stepwise selection* is similar to Forward selection except that at each stage, Analytic Solver Data Science considers dropping variables that are not statistically significant. When this procedure is selected, the Stepwise selection options FIN and FOUT are enabled. In the stepwise selection procedure a statistic is calculated when variables are added or eliminated. For a variable to come into the regression, the statistic's value must be greater than the value for FIN (default = 3.84). For a variable to leave the regression, the statistic's value must be less than the value of FOUT (default = 2.71). The value for FIN must be greater than the value for FOUT.

- *Best Subsets* where searches of all combinations of variables are performed to observe which combination has the best fit. (This option can become quite time consuming depending on the number of input variables.) If this procedure is selected, Number of best subsets is enabled.

Click **Done** to accept the default choice, Backward Elimimination with an F-out setting of 2.71, and return to the Parameters tab, then click **Next** to advance to the *Scoring* tab.



Preprocessing
Partition Data
Rescale Data

Regression: Fitting
Basic
☑ Fit Intercept
Iterations (Max): 50
Advanced
Prior Probability

Regression: Display
Statistics
☑ Variance-Covariance Matrix
☑ Multicollinearity Diagnostics
Advanced
☑ Analysis of Coefficients

Regression: Feature Selection
Feature Selection

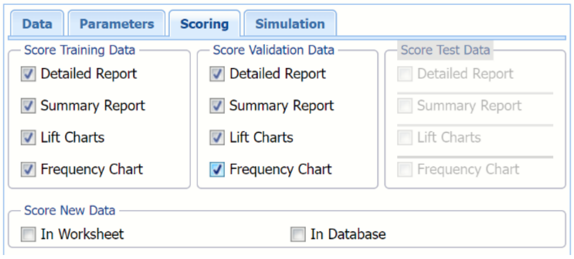
16. Click Next to advance to the Scoring tab.

17. Select **Detailed report**, **Summary report**, **Lift charts** and **Frequency Chart** under both *Score Training Data* and *Score Validation Data.* Analytic Solver Data Science will create a detailed report, complete with the Output Navigator for ease in routing to specific areas in the output, a report that summarizes the regression output for both datasets, and lift charts, ROC curves, and Decile charts for both partitions.

    When **Frequency Chart** is selected under both *Score Training Data* and *Score Validation Data*, a frequency chart will be displayed when the *LogReg_TrainingScore* and *LogReg_ValidationScore* worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

    Since we did not create a test partition when we partitioned our dataset, Score Test Data options are disabled. See the chapter "Data Science Partitioning" for details on how to create a test set.

    For information on scoring in a worksheet or database, please see the "Scoring New Data" chapter in the Analytic Solver Data Science User Guide.



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

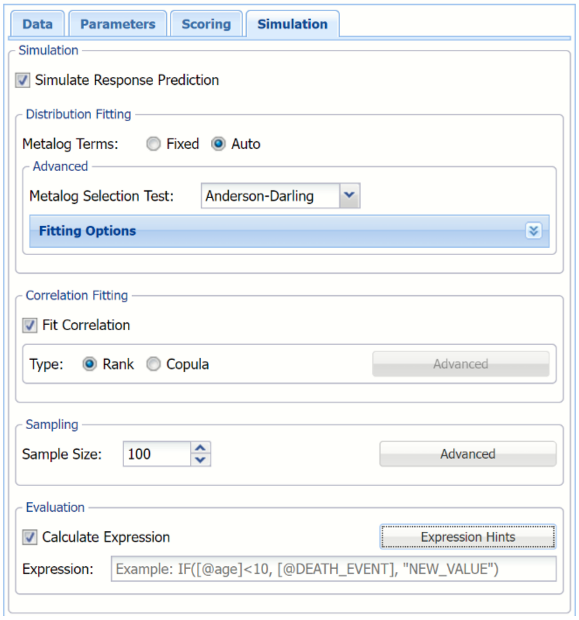
Score New Data
□ In Worksheet
□ In Database

18. Click Next to advance to the Simulation tab.

19. Select **Simulate Response Prediction** to enable all options on the the Simulation tab. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Anlaytic Solver Upgrade.

    Simulation tab: All supervised algorithms include a new Simulation. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, LogReg_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

    Logistic Regression dialog, Simulation tab



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

☑ Calculate Expression

Expression: Example: IF([@age]<10, [@DEATH_EVENT], "NEW_VALUE")

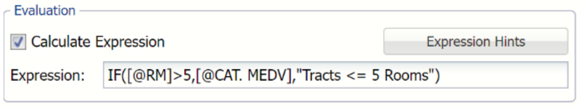
Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the LogReg_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog. For Expression, enter the following formula to display houses that are less than 20 years old.

IF[@RM]>5,[@CAT.MEDV],"Tracks <= 5 Rooms")

Note that variable names are case sensitive.

Evaluation section on the Logistic Regression dialog, Simulation tab


Evaluation

☑ Calculate Expression

Expression: IF([@RM]>5,[@CAT. MEDV],"Tracts <= 5 Rooms")

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

20. Click Finish to run Logistic Regression on the example dataset. The logistic regression output is inserted to the right of the STDPartition worksheet.

Output Worksheets

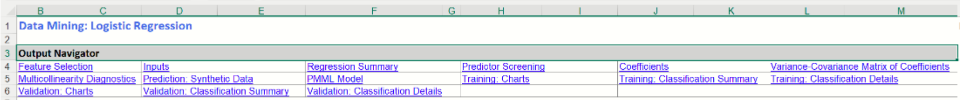
Output sheets containing the Logistic Regression results will be inserted into your active workbook to the right of the STDPartition worksheet.

LogReg_Output

This result worksheet includes 7 segments: Output Navigator, Inputs, Regression Summary, Predictor Screening, Coefficients, Variance-Covariance Matrix of Coefficients and Multicollinearity Diagnostics.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

LogReg_Output: Output Navigator

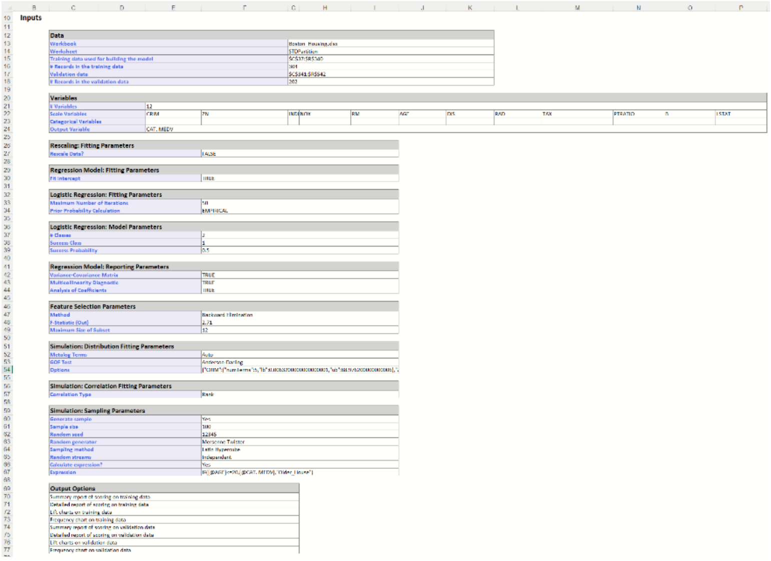


Data Mining: Logistic Regression

Output Navigator					
Feature Selection	Inputs	Regression Summary	Predictor Screening	Coefficients	Variance-Covariance Matrix of Coefficients
Multicollinearity Diagnostics	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Classification Summary	Training: Classification Details
Validation: Charts	Validation: Classification Summary	Validation: Classification Details			

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Logistic Regression dialog.

  LogReg_Output: Inputs



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$17:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	CAT. MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Regression Model: Fitting Parameters	
Fit Intercept	TRUE

Logistic Regression: Fitting Parameters	
Maximum Number of Iterations	50
Prior Probability Calculation	EMPIRICAL

Logistic Regression: Model Parameters	
# Classes	2
Success Class	1
Success Probability	0.5

Regression Model: Reporting Parameters	
Variance-Covariance Matrix	TRUE
Multicollinearity Diagnostic	TRUE
Analysis of Coefficients	TRUE

Feature Selection Parameters	
Method	Backward Elimination
F-Statistic (Out)	2.71
Maximum Size of Subset	12

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.00632000000000000001,"ub":88.97620000000000006},".

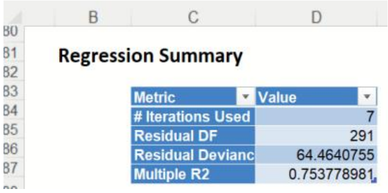
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	Yes
Expression	IF([@AGE]<=20,[@CAT. MEDV],"Older_House")

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Regression Summary:** Summary statistics, found directly below Inputs in the Regression Summary report, show the residual degrees of freedom (#observations - #predictors), a standard deviation type measure for the model (which typically has a chi-square distribution), the number of iterations required to fit the model, and the Multiple R-squared value.

LogReg_Output: Regression Summary



Regression Summary

Metric	Value
# Iterations Used	7
Residual DF	291
Residual Devianc	64.4640755
Multiple R2	0.753778981

The multiple R-squared value shown here is the r-squared value for a logistic regression model , defined as

$R^2 = (D_0\text{-}D)/D_0$,

where D is the Deviance based on the fitted model and D_0 is the deviance based on the null model. The null model is defined as the model containing no predictor variables apart from the constant.

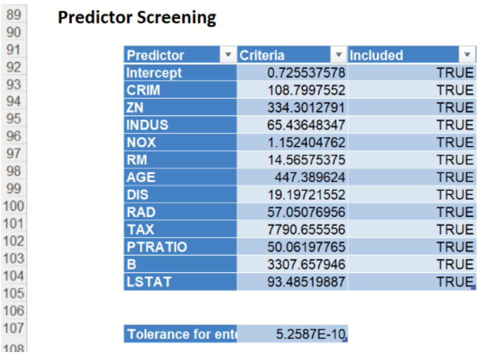
- **Predictor Screening:** Scroll down to the Predictor Screening report. In Analytic Solver Data Science, a preprocessing feature selection step is included to take advantage of automatic variable screening and elimination using Rank-Revealing QR Decomposition. This allows Analytic Solver Data Science to identify the variables causing multicollinearity, rank deficiencies and other problems that would otherwise cause the algorithm to fail. Information about “bad” variables is used in Variable Selection and Multicollinearity Diagnostics and in computing other reported statistics.

  Included and excluded predictors are shown in the table below. In this model there were no excluded predictors. All predictors were eligible to enter the model passing the tolerance threshold of 5.26E-10. This denotes a tolerance beyond which a variance – covariance matrix is not exactly singular to within machine precision. The test is based on the diagonal elements of the triangular factor R resulting from Rank-Revealing QR Decomposition. Predictors that do not pass the test are excluded.

  Note: If a predictor is excluded, the corresponding coefficient estimates will be 0 in the regression model and the variable – covariance matrix would contain all zeros in the rows and columns that correspond to the excluded predictor. Multicollinearity diagnostics, variable selection and other remaining output will be calculated for the reduced model.

  The design matrix may be rank-deficient for several reasons. The most common cause of an ill-conditioned regression problem is the presence of feature(s) that can be exactly or approximately represented by a linear combination of other feature(s). For example, assume that among predictors you have 3 input variables X, Y, and Z where Z = a * X + b * Y where a and b are constants. This will cause the design matrix to not have a full rank. Therefore, one of these 3 variables will not pass the threshold for entrance and will be excluded from the final regression model.

LogReg: Output: Predictor Screening



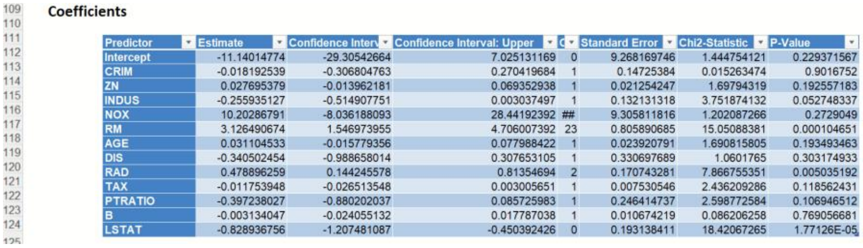
Predictor Screening

Predictor	Criteria	Included
Intercept	0.725537578	TRUE
CRIM	108.7997552	TRUE
ZN	334.3012791	TRUE
INDUS	65.43648347	TRUE
NOX	1.152404762	TRUE
RM	14.56575375	TRUE
AGE	447.389624	TRUE
DIS	19.19721552	TRUE
RAD	57.05076956	TRUE
TAX	7790.655556	TRUE
PTRATIO	50.06197765	TRUE
B	3307.657946	TRUE
LSTAT	93.48519887	TRUE

Tolerance for ent	5.2587E-10

- **Coefficients:** Model terms are shown in the Coefficients output.

  LogReg_Output: Coefficients



Coefficients

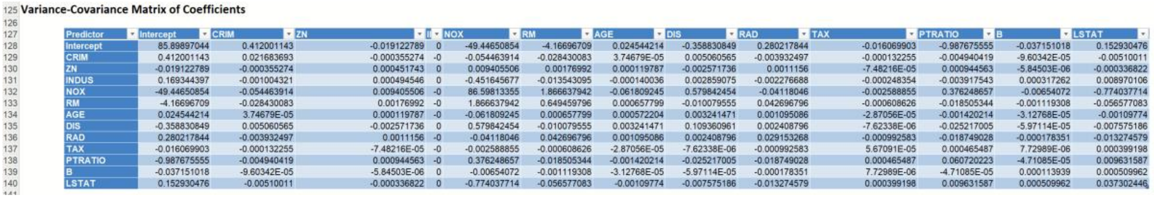
Predictor	Estimate	Confidence Interv	Confidence Interval: Upper	C	Standard Error	Chi2-Statistic	P-Value
Intercept	-11.14014774	-29.30542664	7.025131169	0	9.268169746	1.444754121	0.229371567
CRIM	-0.018192539	-0.306804763	0.270419684	1	0.14725384	0.015263474	0.9016752
ZN	0.027695379	-0.013962181	0.069352938	1	0.021254247	1.69794319	0.192557183
INDUS	-0.255935127	-0.514907751	0.003037497	1	0.132131318	3.751874132	0.052748337
NOX	10.20286791	-8.036188093	28.44192392	##	9.305811816	1.202087266	0.2729049
RM	3.126490674	1.546973955	4.706007392	23	0.805890685	15.05088381	0.000104651
AGE	0.031104533	-0.015779356	0.077988422	1	0.023920791	1.690815805	0.193493463
DIS	-0.340502454	-0.988658014	0.307653105	1	0.330697689	1.0601765	0.303174933
RAD	0.478896259	0.144245578	0.81354694	2	0.170743281	7.866755351	0.005035192
TAX	-0.011753948	-0.026513548	0.003005651	1	0.007530546	2.436209286	0.118562431
PTRATIO	-0.397238027	-0.880202037	0.085725983	1	0.246414737	2.598772584	0.106946512
B	-0.003134047	-0.024055132	0.017787038	1	0.010674219	0.086206258	0.769056681
LSTAT	-0.828936756	-1.207481087	-0.450392426	0	0.193138411	18.42067265	1.77126E-05

This table contains the coefficient estimate, the standard error of the coefficient, the p-value, the odds ratio for each variable (which is simply e^x where x is the value of the coefficient) and confidence interval for the odds. (Note for the Intercept term, the Odds Ratio is calculated as exp^0.)

Note: If a variable has been eliminated by Rank-Revealing QR Decomposition, the variable will appear in red in the Coefficients table with a 0 Coefficient, Std. Error, CI Lower, CI Upper, and RSS Reduction and N/A for the t-Statistic and P-Values.

- **Variance-Covariance Matrix of Coefficients:** This square matrix contains the variances of the fitted model's coefficient estimates in the center diagonal elements and the pair-wise covariances between coefficient estimates in the non-diagonal elements.

LogReg_Output: Variance - Covariance Matrix of Coefficients

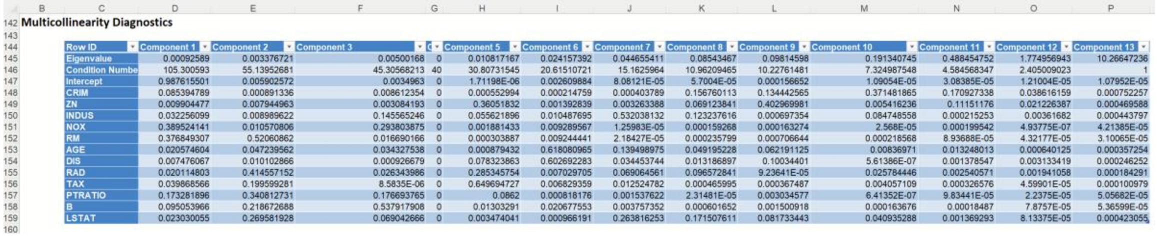


Variance-Covariance Matrix of Coefficients

Predictor	Intercept	CRIM	ZN	I	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Intercept	85.89897044	0.412001143	-0.019122789	0	-49.44650854	-4.16696709	0.024544214	-0.358830849	0.280217844	-0.016069903	-0.987675555	-0.037151018	0.152930476
CRIM	0.412001143	0.021683693	-0.000355274	-0	-0.054463914	-0.028430083	3.74679E-05	0.005060565	-0.003932497	-0.000132255	-0.004940419	-9.60342E-05	-0.00510011
ZN	-0.019122789	-0.000355274	0.000451743	0	0.009405506	0.00176992	0.000119787	-0.002571736	0.0011156	-7.48216E-05	0.000944563	-5.84503E-06	-0.000336822
INDUS	0.169344397	-0.001004321	0.000494546	0	-0.451645677	-0.013543095	-0.000140036	0.002859075	-0.002276688	-0.000248354	-0.003917543	0.000317262	0.008970106
NOX	-49.44650854	-0.054463914	0.009405506	-0	86.59813355	1.866637942	-0.061809245	0.579842454	-0.04118046	-0.002588855	0.376248657	-0.00654072	-0.774037714
RM	-4.16696709	-0.028430083	0.00176992	-0	1.866637942	0.649459796	0.000657799	-0.010079555	0.042696796	-0.000608626	-0.018505344	-0.001119308	-0.056577083
AGE	0.024544214	3.74679E-05	0.000119787	-0	-0.061809245	0.000657799	0.000572204	0.003241471	0.001095086	-2.87056E-05	-0.001420214	-3.12768E-05	-0.00109774
DIS	-0.358830849	0.005060565	-0.002571736	0	0.579842454	-0.010079555	0.003241471	0.109360961	0.002408796	-7.62338E-06	-0.025217005	-5.97114E-05	-0.007575186
RAD	0.280217844	-0.003932497	0.0011156	-0	-0.04118046	0.042696796	0.001095086	0.002408796	0.029153268	-0.000992583	-0.018749028	-0.000178351	-0.013274579
TAX	-0.016069903	-0.000132255	-7.48216E-05	-0	-0.002588855	-0.000608626	-2.87056E-05	-7.62338E-06	-0.000992583	5.67091E-05	0.000465487	7.72989E-06	0.000399198
PTRATIO	-0.987675555	-0.004940419	0.000944563	-0	0.376248657	-0.018505344	-0.001420214	-0.025217005	-0.018749028	0.000465487	0.060720223	-4.71085E-05	0.009631587
B	-0.037151018	-9.60342E-05	-5.84503E-06	0	-0.00654072	-0.001119308	-3.12768E-05	-5.97114E-05	-0.000178351	7.72989E-06	-4.71085E-05	0.000113939	0.000509962
LSTAT	0.152930476	-0.00510011	-0.000336822	0	-0.774037714	-0.056577083	-0.00109774	-0.007575186	-0.013274579	0.000399198	0.009631587	0.000509962	0.037302446

- **Multicollinearity Diagnostics:** Collinearity Diagnostics help assess whether two or more variables so closely track one another as to provide essentially the same information.

LogReg_Output: Multicollinearity Diagnostics



Multicollinearity Diagnostics

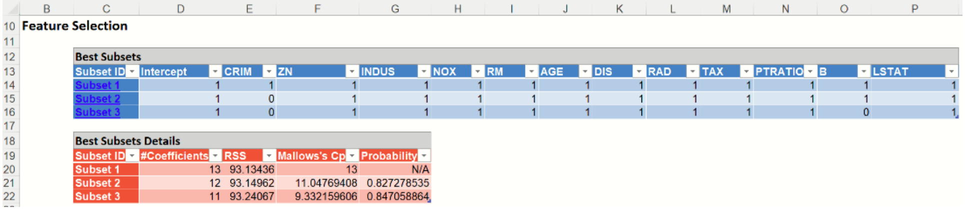
Row ID	Component 1	Component 2	Component 3	C	Component 5	Component 6	Component 7	Component 8	Component 9	Component 10	Component 11	Component 12	Component 13
Eigenvalue	0.00092589	0.003376721	0.00500168	0	0.010817167	0.024157392	0.044655411	0.08543467	0.09814598	0.191340745	0.488454752	1.774956943	10.26647236
Condition Numbe	105.300593	55.13952681	45.30568213	40	30.80731545	20.61510721	15.1625964	10.96209465	10.22761481	7.324987548	4.584568347	2.405009023	1
Intercept	0.987615501	0.005902572	0.0034963	0	1.71198E-06	0.002609884	8.08121E-05	5.7004E-05	0.000156652	1.09054E-05	3.08385E-05	1.21004E-05	1.07952E-05
CRIM	0.085394789	0.000891336	0.008612354	0	0.000552994	0.000214759	0.000403789	0.156760113	0.134442565	0.371481865	0.170927338	0.038616159	0.000752257
ZN	0.009904477	0.007944963	0.003084193	0	0.36051832	0.001392839	0.003263388	0.069123841	0.402969981	0.005416236	0.11151176	0.021226387	0.000469588
INDUS	0.032256099	0.008989622	0.145565246	0	0.055621896	0.010487695	0.532038132	0.123237616	0.000697354	0.084748558	0.000215253	0.00361682	0.000443797
NOX	0.389524141	0.010570806	0.293803875	0	0.001881433	0.009289567	1.25983E-05	0.000159268	0.000163274	2.568E-05	0.000199542	4.93775E-07	4.21385E-05
RM	0.376849307	0.52060862	0.016690166	0	0.000303887	0.009244441	2.18427E-05	0.000235799	0.000706644	0.000218568	8.93688E-05	4.32177E-05	3.10065E-05
AGE	0.020574604	0.047239562	0.034327538	0	0.000879432	0.618080965	0.139498975	0.049195228	0.062191125	0.00836971	0.013248013	0.000640125	0.000357254
DIS	0.007476067	0.010102866	0.000926679	0	0.078323863	0.602692283	0.034453744	0.013186897	0.10034401	5.61386E-07	0.001378547	0.003133419	0.000246252
RAD	0.020114803	0.414557152	0.026343986	0	0.285345754	0.007029705	0.069064561	0.096572841	9.23641E-05	0.025784446	0.002540571	0.001941058	0.000184291
TAX	0.039868566	0.199599281	8.5835E-06	0	0.649694727	0.006829359	0.012524782	0.000465995	0.000367487	0.004057109	0.000326576	4.59901E-05	0.000100979
PTRATIO	0.173281896	0.340812731	0.176693765	0	0.0862	0.000818176	0.001537622	2.31481E-05	0.003034577	6.41352E-07	9.83441E-05	2.2375E-05	5.05682E-05
B	0.095053966	0.218672688	0.537917908	0	0.01303291	0.020677553	0.003757352	0.000601652	0.001500918	0.000163676	0.00018487	7.8757E-05	5.36599E-05
LSTAT	0.023030055	0.269581928	0.069042666	0	0.003474041	0.000966191	0.263816253	0.171507611	0.081733443	0.040935288	0.001369293	8.13375E-05	0.000423055

The columns represent the variance components (related to principal components in multivariate analysis), while the rows represent the variance proportion decomposition explained by each variable in the model. The eigenvalues are those associated with the singular value decomposition of the variance-covariance matrix of the coefficients, while the condition numbers are the ratios of the square root of the largest eigenvalue to all the rest. In general, multicollinearity is likely to be a problem with a high condition number (more than 20 or 30), and high variance decomposition proportions (say more than 0.5) for two or more variables.

LogReg_FS

Since we selected *Perform Feature Selection* on the *Feature Selection* dialog, Analytic Solver Data Science has produced the following output on the LogReg_FS tab which displays the variables that are included in the subsets. This table contains the two subsets with the highest Residual Sum of Squares values.

LogReg_FS: Feature Selection



Feature Selection

Best Subsets

Subset ID	Intercept	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Subset 1	1	1	1	1	1	1	1	1	1	1	1	1	1
Subset 2	1	0	1	1	1	1	1	1	1	1	1	1	1
Subset 3	1	0	1	1	1	1	1	1	1	1	1	0	1

Best Subsets Details

Subset ID	#Coefficients	RSS	Mallows's Cp	Probability
Subset 1	13	93.13436	13	N/A
Subset 2	12	93.14962	11.04769408	0.827278535
Subset 3	11	93.24067	9.332159606	0.847058864

In this table, every model includes a constant term (since Fit Intercept was selected) and one or more variables as the additional coefficients. We can use any of these models for further analysis simply by clicking the hyperlink under Subset ID in the far left column. The Logistic Regression dialog will open. Click Finish to run Logistic Regression using the variable subset as listed in the table.

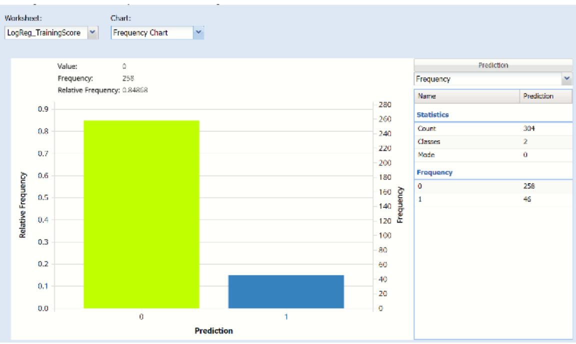
The choice of model depends on the calculated values of various error values and the probability. RSS is the residual sum of squares, or the sum of squared deviations between the predicted probability of success and the actual value (1 or 0). "Mallows Cp" is a measure of the error in the best subset model, relative to the error incorporating all variables. Adequate models are those for which Cp is roughly equal to the number of parameters in the model (including the constant), and/or Cp is at a minimum. "Probability" is a quasi hypothesis test of the proposition that a given subset is acceptable; if Probability < .05 we can rule out that subset.

The considerations about RSS, Cp and Probability in this example would lead us to believe that the subset with 12 coefficients is the best model in this example.

LogReg_TrainingScore

Click the Training: Classification Details link in the Output Navigator to open the Training: Classification Summary and view the newly added Output Variable frequency chart, the Training: Classification Summary and the Training: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: LogReg_TrainingScore

Chart: Frequency Chart

Value: 0

Frequency: 258

Relative Frequency: 0.84868

Prediction

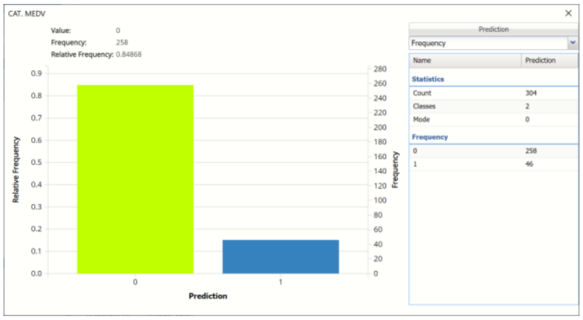
Frequency

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Frequency	
0	258
1	46

- **Frequency Charts:** The output variable frequency chart opens automatically once the *LogReg_TrainingScore* worksheet is selected. To close this chart, click the “x” in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *LogReg_TrainingScore* tab. To change the location of the chart, grab the title bar of the dialog and drag the chart to the desired location on the screen.

  Frequency: This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

  Frequency Chart on LogReg_TrainingScore output sheet



CAT. MEDV

Value: 0

Frequency: 258

Relative Frequency: 0.84868

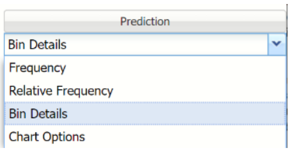
Prediction

Frequency

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Frequency	
0	258
1	46

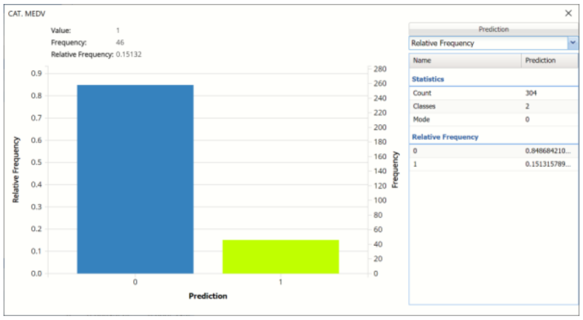
Click the down arrow next to Frequency to switch to Relative Frequency, Bin Details or Chart Options view.

Frequency Chart, Frequency View



Relative Frequency: Displays the relative frequency chart.

Relative Frequency Chart



CAT. MEDV

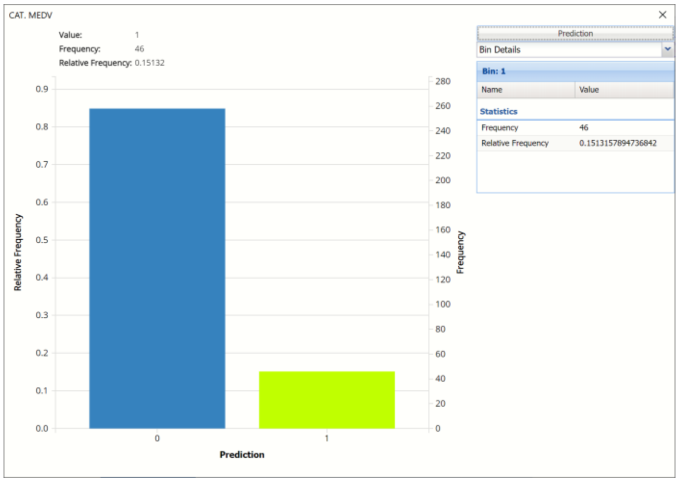
Value: 1
Frequency: 46
Relative Frequency: 0.15132

Prediction

Relative Frequency

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Relative Frequency	
0	0.848684210...
1	0.151315789...

Bin Details: See pertinent information about each bin in the chart.



CAT. MEDV

Value: 1
Frequency: 46
Relative Frequency: 0.15132

Prediction

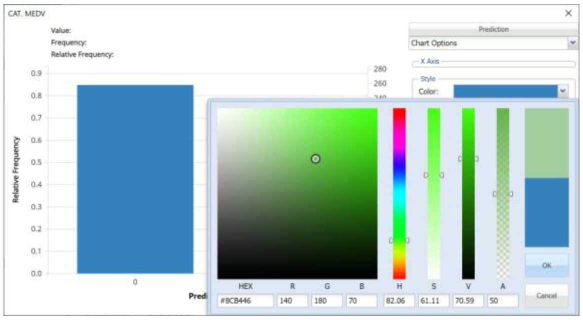
Bin Details

Bin: 1

Name	Value
Statistics	
Frequency	46
Relative Frequency	0.1513157894736842

Chart Options: Use this view to change the color of the bars in the chart.

Chart Options View



CAT. MEDV

Value:
Frequency:
Relative Frequency:

Prediction

Chart Options

X Axis

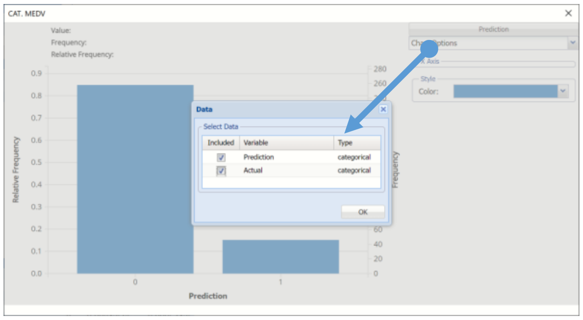
Style

Color:

HEX	R	G	B	H	S	V	A
#8CB446	140	180	70	82.06	61.11	70.59	50

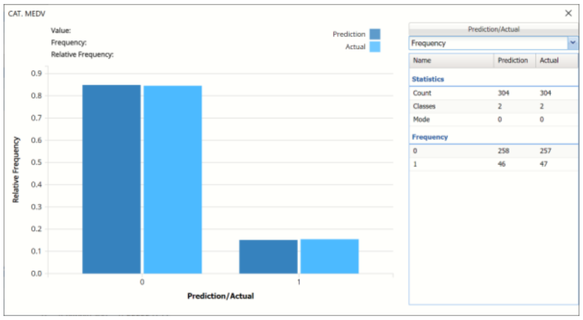
- To see both the actual and predicted frequency, click Prediction and select Actual. This change will be reflected on all charts.

Click Prediction, then select Actual



Included	Variable	Type
☑	Prediction	categorical
☑	Actual	categorical

Predicted/Actual view



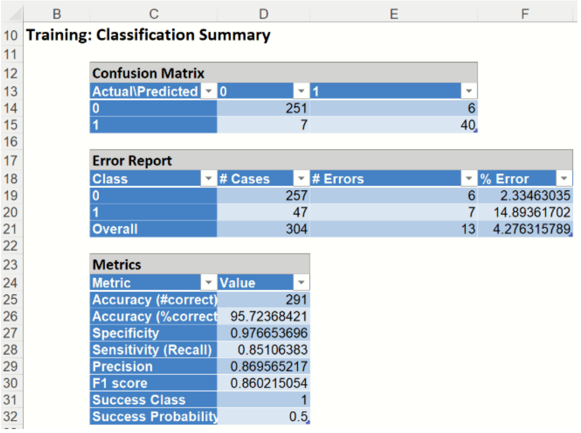
Name	Prediction	Actual
Statistics		
Count	304	304
Classes	2	2
Mode	0	0
Frequency		
0	258	257
1	46	47

- **Classification Summary:** A Confusion Matrix is used to evaluate the performance of a classification method. This matrix summarizes the records that were classified correctly and those that were not.

		prediction outcome		
		p	*n*	total
actual value	*p'*	True Positive	False Negative	P'
	n'	False Positive	True Negative	N'
	total	P	N	

- True Positive cases (TP) are the number of cases classified as belonging to the Success class that actually were members of the Success class.
- False Negative cases (FN) are the number of cases that were classified as belonging to the Failure class when they were actually members of the Success class (i.e. if a cancerous tumor is considered a "success", then imagine patients with cancerous tumors who were told their tumors were benign).
- False Positive (FP) cases were assigned to the Success class but were actually members of the Failure group (i.e. patients who were told they tested positive for cancer when, in fact, their tumors were benign).
- True Negative (TN) cases were correctly assigned to the Failure group.

LogReg_TrainingScore: Classification Summary



Training: Classification Summary

Confusion Matrix		
Actual\Predicted	0	1
0	251	6
1	7	40

Error Report			
Class	# Cases	# Errors	% Error
0	257	6	2.33463035
1	47	7	14.89361702
Overall	304	13	4.276315789

Metrics	
Metric	Value
Accuracy (#correct)	291
Accuracy (%correct	95.72368421
Specificity	0.976653696
Sensitivity (Recall)	0.85106383
Precision	0.869565217
F1 score	0.860215054
Success Class	1
Success Probability	0.5

In the Training Dataset, we see 40 records belonging to the Success class were correctly assigned to that class while 7 records belonging to the Success class were incorrectly assigned to the Failure class. In addition, 251 records belonging to the Failure class were correctly assigned to this same class while 6 records belonging to the Failure class were incorrectly assigned to the Success class. The total number of misclassified records was 13 (7+6) which results in an error equal to 4.28%.

- Precision is the probability of correctly identifying a randomly selected record as one belonging to the Success class (i.e. the probability of correctly identifying a random patient with cancer as having cancer).

  Precision = TP/ (TP+FP)

- Recall (or Sensitivity) measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people with cancer who are correctly identified as having cancer).

  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)

- Specificity (also called the true negative rate) measures the percentage of failures correctly identified as failures (i.e. the proportion of people with no cancer being categorized as not having cancer.)

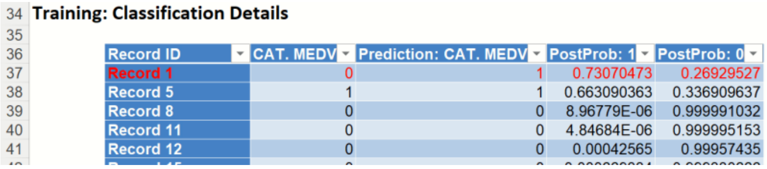
  Specificity (SPC) or True Negative Rate =TN / (FP + TN)

- The F-1 score, which fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP /(2TP+ FP + FN)

- **Training: Classification Details:** This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

  LogReg_TrainingScore: Classification Details



Training: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 1	PostProb: 0
Record 1	0	1	0.73070473	0.26929527
Record 5	1	1	0.663090363	0.336909637
Record 8	0	0	8.96779E-06	0.999991032
Record 11	0	0	4.84684E-06	0.999995153
Record 12	0	0	0.00042565	0.99957435

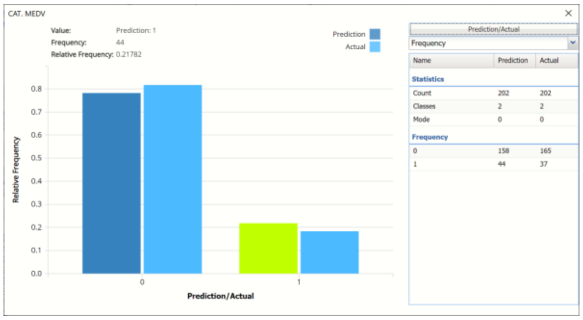
LogReg_ValidationScore

Click the *LogReg_ValidationScore* tab to view the newly added Output Variable frequency chart, the Validation: Classification Summary and the Validation: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Validation data.

- **Frequency Charts:** The output variable frequency chart opens automatically once the LogReg_ValidationScore worksheet is selected. To close this chart, click the "x" in the upper right hand corner. To reopen, click onto another tab and then click back to the LogReg_ValidationScore tab.

  Click the Frequency chart to display the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode. Selective Relative Frequency from the drop down menu, on the right, to see the relative frequencies of the output variable for both actual and predicted. See above for more information on this chart.

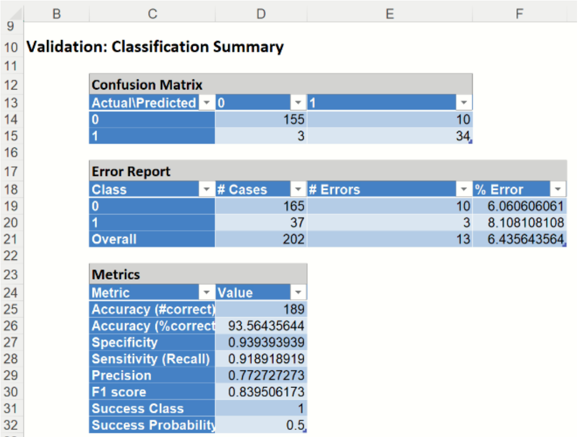
  LogReg_ValidationScore Frequency Chart



Name	Prediction	Actual
Statistics		
Count	202	202
Classes	2	2
Mode	0	0
Frequency		
0	158	165
1	44	37

- **Classification Summary:** This report contains the confusion matrix for the validation data set.

  LogReg_ValidationScore: Classification Summary



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	155	10
1	3	34

Error Report

Class	# Cases	# Errors	% Error
0	165	10	6.060606061
1	37	3	8.108108108
Overall	202	13	6.435643564

Metrics

Metric	Value
Accuracy (#correct)	189
Accuracy (%correct	93.56435644
Specificity	0.939393939
Sensitivity (Recall)	0.918918919
Precision	0.772727273
F1 score	0.839506173
Success Class	1
Success Probability	0.5

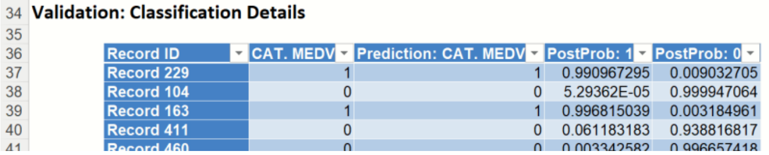
In the Validation Dataset…

- 34 records were correctly classified as belonging to the Success class
- 155 cases were correctly classified as belonging to the Failure class.

- False Positives: 10 records were incorrectly classified as belonging to the Success class when they were, in fact, members of the Failure class.
- False Negatives: 3 records were incorrectly classified as belonging to the Failure class, when they were members of the Success class.
- This resulted in a total classification error of 6.44%

Scroll down to view the Validation: Classification Details table. Again, misclassified records appear in red.

LogReg_ValidationScore: Classification Details



34 **Validation: Classification Details**

35

	Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 1	PostProb: 0
37	Record 229	1	1	0.990967295	0.009032705
38	Record 104	0	0	5.29362E-05	0.999947064
39	Record 163	1	1	0.996815039	0.003184961
40	Record 411	0	0	0.061183183	0.938816817
41	Record 460	0	0	0.003342582	0.996657418

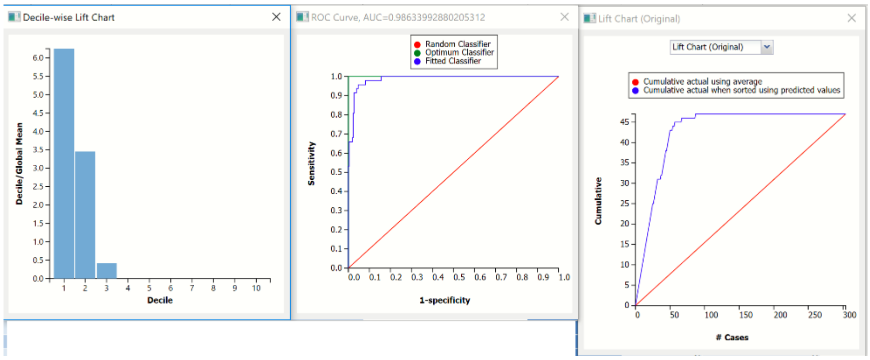
LogReg_TrainingLiftChart and LogReg_ValidationLiftChart

Click **LogReg_TrainingLiftChart** and **LogReg_ValidationLiftChart** tab to navigate to the Training and Validation Data Lift, Decile, and ROC Curve charts.

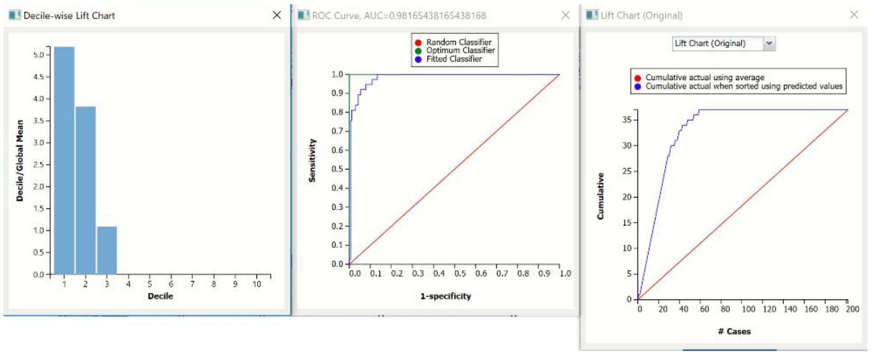
Lift Charts and ROC Curves are visual aids that help users evaluate the performance of their fitted models. Charts found on the LogReg_TrainingLiftChart tab were calculated using the Training Data Partition. Charts found on the LogReg_ValidationLiftChart tab were calculated using the Validation Data Partition. It is good practice to look at both sets of charts to assess model performance on the Training and Validation partitions.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select LogReg_TrainingLiftChart or LogReg_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, ROC Curve, and Lift Charts for Training Partition



Decile-wise Lift Chart, ROC Curve, and Lift Charts for Valid. Partition

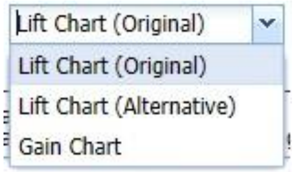


After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted in decreasing order using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the cumulative number of cases in decreasing probability (on the x-axis) vs the cumulative number of true positives on the y-axis. The baseline (red line connecting the origin to the end point of the blue line) is a reference line. For a given number of cases on the x-axis, this line represents the expected number of successes if no model existed, and instead cases were selected at random. This line can be used as a benchmark to measure the performance of the fitted model. The greater the area between the lift curve and the baseline, the better the model. In the Training Lift chart, if we selected 100 cases as belonging to the success class and used the fitted model to pick the members most likely to be successes, the lift curve tells us that we would be right on about 45 of them. Conversely, if we selected 100 random cases, we could expect to be right on about 15 of them. The Validation Lift chart tells us that if we selected 100 cases as belonging to the success class and used the fitted model to pick the members most likely to be successes, the lift curve tells us that we would be right on about 37 of them. If we selected 100 random cases, we could expect to be right on about 15 of them.

The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. The bars in this chart indicate the factor by which the model outperforms a random assignment, one decile at a time. Refer to the validation graph above. In the first decile, taking the most expensive predicted housing prices in the dataset, the predictive performance of the model is about 4.5 times better as simply assigning a random predicted value.

The Regression ROC curve was updated in V2017. This new chart compares the performance of the regressor (Fitted Classifier) with an Optimum Classifier Curve and a Random Classifier curve. The Optimum Classifier Curve plots a hypothetical model that would provide perfect classification results. The best possible classification performance is denoted by a point at the top left of the graph at the intersection of the x and y axis. This point is sometimes referred to as the “perfect classification”. The closer the AUC is to 1, the better the performance of the model. In the Validation Partition, AUC = .98 which suggests that this fitted model is a good fit to the data.

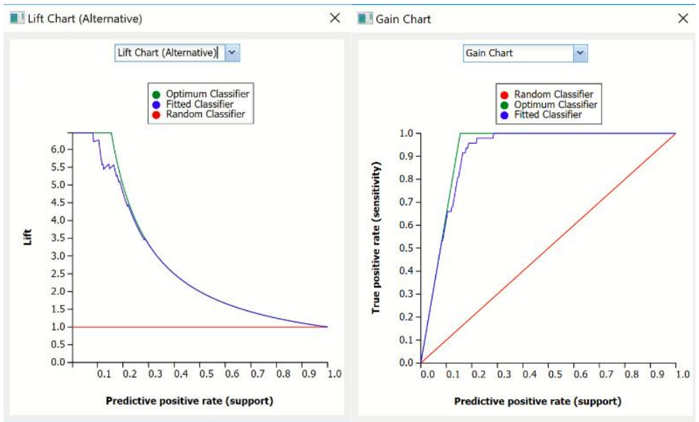
In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.



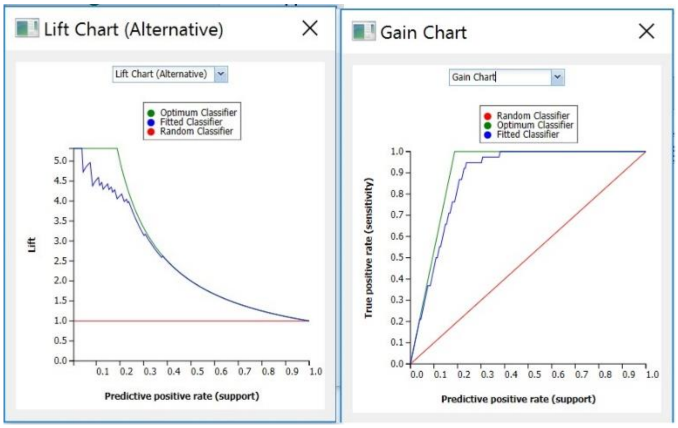
Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Classifier curve, a Fitted Classifier curve, and a Random Classifier curve. The Optimum Classifier curve plots a hypothetical model that would provide perfect classification for our data. The Fitted Classifier curve plots the fitted model and the Random Classifier curve plots the results from using no model or by using a random guess (i.e. for x% of selected observations, x% of the total number of positive observations are expected to be correctly classified).

The Alternative Lift Chart plots Lift against the Predictive Positive Rate or Support.

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition



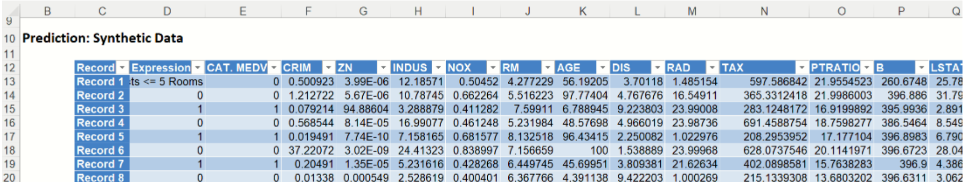
Click the down arrow and select Gain Chart from the menu. In this chart, the True Positive Rate or Sensitivity is plotted against the Predictive Positive Rate or Support.

LogReg_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, LogReg_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Logistic Regression dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data (with or without correlation fitting), the prediction (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. A chart is also displayed with the option to switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

LogReg_Simulation: Synthetic Data



Prediction: Synthetic Data

Record	Expression	CAT. MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	ts <= 5 Rooms	0	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.70118	1.485154	597.586842	21.9554523	260.6748	25.78
Record 2	0	0	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767676	16.54911	365.3312418	21.9986003	396.886	31.79
Record 3	1	1	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223803	23.99008	283.1248172	16.9199892	395.9936	2.891
Record 4	0	0	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966019	23.98736	691.4588754	18.7598277	386.5464	8.549
Record 5	1	1	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082	1.022976	208.2953952	17.177104	396.8983	6.790
Record 6	0	0	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538889	23.99968	628.0737546	20.1141971	396.6723	28.04
Record 7	1	1	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809381	21.62634	402.0898581	15.7638283	396.9	4.386
Record 8	0	0	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422203	1.000269	215.1339308	13.6803202	396.6311	3.062

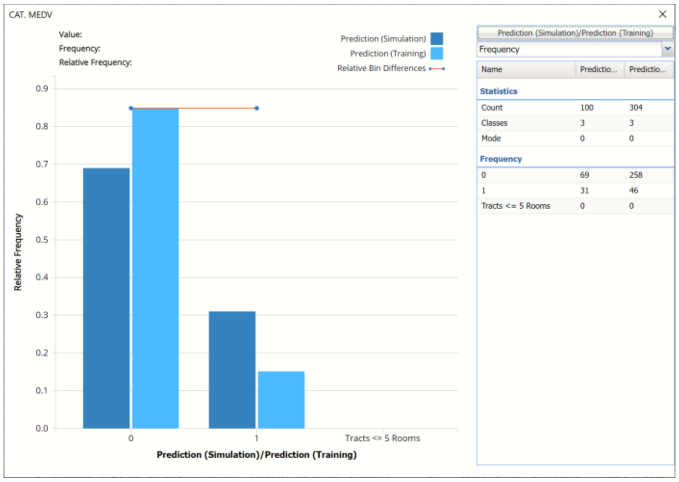
Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the Logistic Regression dialog

Expression: IF([@RM]>5,[@CAT. MEDV],"racks <= 5 Rooms")

The rest of the data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

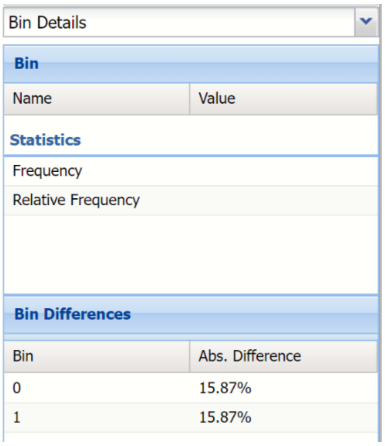
The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the actual data and the synthetic data. In the screenshot below, the bars in the darker shade of blue are based on the synthetic data. The bars in the lighter shade of blue are based on the predicted values for the training partition. In the synthetic data, about 70% of the housing tracts where CAT. MEDV = 0, have more than 5 rooms and about 85% of the housing tracts in the training partition where CAT. MEDV = 0 have more than 5 rooms.

Frequency Chart for LogReg_Simulation output



Name	Predictio...	Predictio...
Statistics		
Count	100	304
Classes	3	3
Mode	0	0
Frequency		
0	69	258
1	31	46
Tracts <= 5 Rooms	0	0

Click the array next to Frequency and select Bin Details. Notice that the abosulte difference in each bin is the same. Hence the flat Relative Bin Difference curve in the chart.



Bin Details

Bin

Name	Value

Statistics

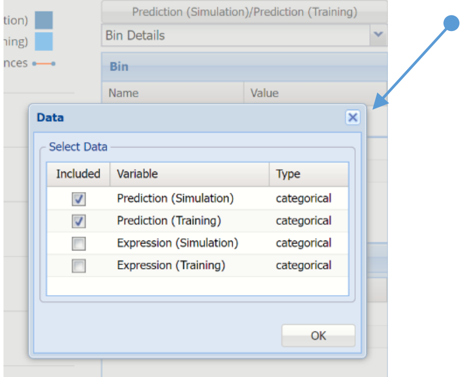
Frequency	
Relative Frequency	

Bin Differences

Bin	Abs. Difference
0	15.87%
1	15.87%

Click *Prediction (Simulation) / Prediction (Training)* to change the chart view to *Expression (Simulation)* and *Expression (Training)*

Click Prediction (Simulation) / Prediction (Training) to change the Data view



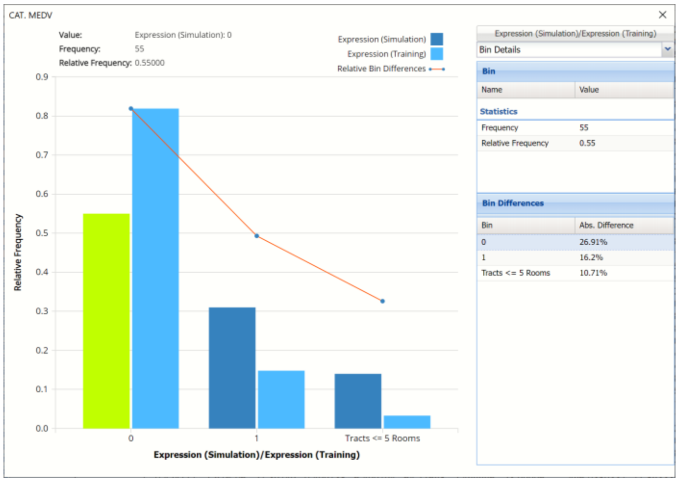
Data

Select Data

Included	Variable	Type
☑	Prediction (Simulation)	categorical
☑	Prediction (Training)	categorical
☐	Expression (Simulation)	categorical
☐	Expression (Training)	categorical

OK

Expression view



Value: Expression (Simulation): 0
Frequency: 55
Relative Frequency: 0.55000

Expression (Simulation)/Expression (Training)

Bin Details

Bin

Name	Value

Statistics

Frequency	55
Relative Frequency	0.55

Bin Differences

Bin	Abs. Difference
0	26.91%
1	16.2%
Tracts <= 5 Rooms	10.71%

This chart shows the relative bin differences are decreasing. Only about 15% of the housing tracts in the synthetic data were predicted has having less than 5 rooms. Less than 5% of the housing tracts in the training data were predicted as having 5 rooms or less.

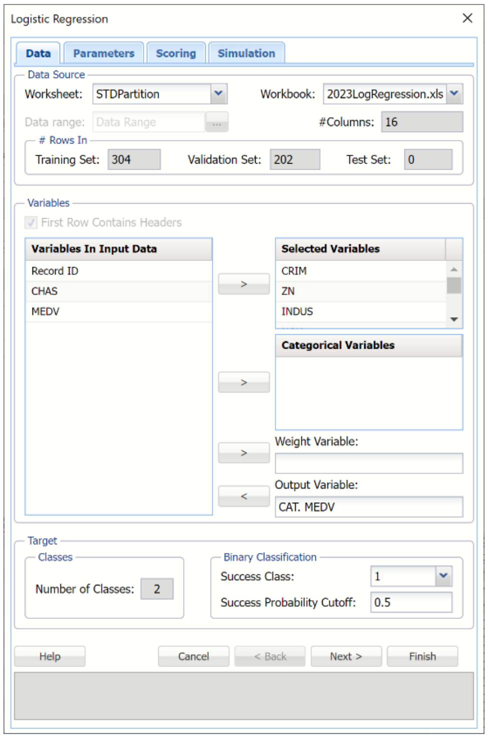
Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

For information on Stored Model Sheets, in this example *LogReg_Stored*, please refer to the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide.

Logistic Regression Options

The following options appear on one of the four *Logistic Regression* dialog tabs.

Logistic Regression Data Tab



Logistic Regression

Data Source

Worksheet: STDPartition

Workbook: 2023LogRegression.xlsx

Data range: Data Range

#Columns: 16

# Rows In

Training Set: 304

Validation Set: 202

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
MEDV

Selected Variables
CRIM
ZN
INDUS

Categorical Variables

Weight Variable:

Output Variable: CAT. MEDV

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Logistic Regression Dialog, Data Tab

Variables In Input Data

All variables in the dataset are listed here.

Selected Variables

Variables listed here will be utilized in the Logistic Regression algorithm.

Weight Variable

One major assumption of Logistic Regression is that each observation provides equal information. Analytic Solver Data Science offers an opportunity to provide a Weight variable. Using a Weight variable allows the user to allocate a weight to each record. A record with a large weight will influence the model more than a record with a smaller weight.

Output Variable

Select the variable whose outcome is to be predicted. The classes in the output variable must be equal to 2.

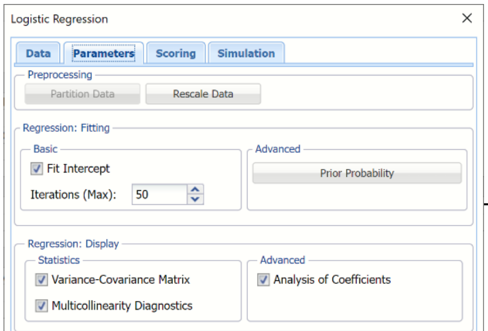
Number of Classes

Displays the number of classes in the Output variable.

Success Class

This option is selected by default. Select the class to be considered a “success” or the significant class in the Lift Chart. This option is enabled when the number of classes in the output variable is equal to 2.

Logistic Regression Parameters Tab



Logistic Regression

Preprocessing

Regression: Fitting

Basic

☑ Fit Intercept

Iterations (Max): 50

Advanced

Regression: Display

Statistics

☑ Variance-Covariance Matrix

☑ Multicollinearity Diagnostics

Advanced

☑ Analysis of Coefficients

Success Probability Cutoff

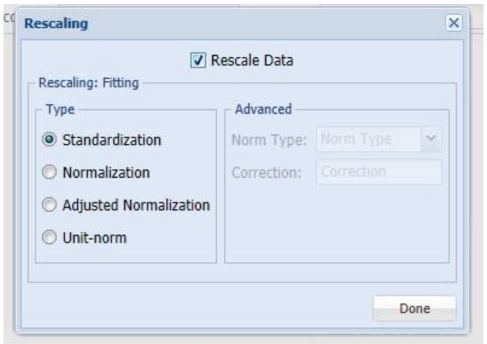
Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a “success” (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a “non-success” (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

Logistic Regression Dialog, Parameters Tab

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Rescale Data



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

Advanced

Norm Type: Norm Type

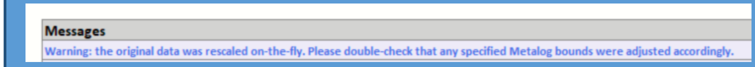
Correction: Correction

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Note: Rescaling has no substantial effect in Logistic Regression other than proportional scaling.

Notes on Rescaling and the Simulation functionality

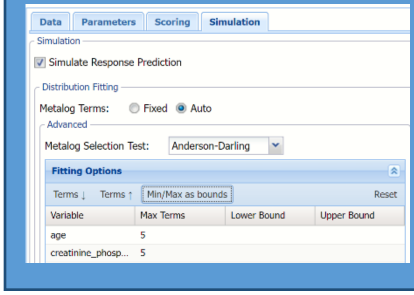
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the LogReg_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction

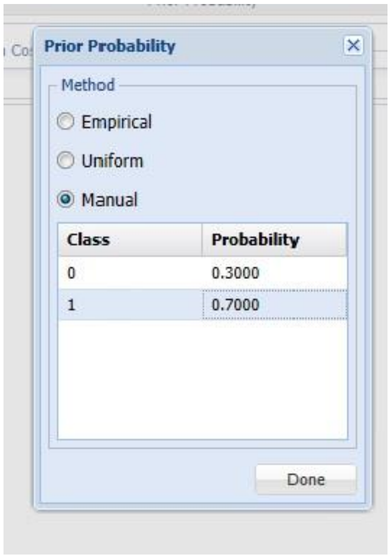
Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Prior Probability

Click **Prior Probability** to open the dialog below. Three options appear in the *Prior Probability* Dialog: *Empirical*, *Uniform* and *Manual*.



○ Empirical

○ Uniform

● Manual

Class	Probability
0	0.3000
1	0.7000

- If the first option is selected, Empirical, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
- If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.

- Select the third option, *Manual*, to manually enter the desired class and probability.

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by selecting Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Fit Intercept

When this option is selected, the default setting, Analytic Solver Data Science will fit the Logistic Regression intercept. If this option is not selected, Analytic Solver Data Science will force the intercept term to 0.

Iterations (Max)

Estimating the coefficients in the Logistic Regression algorithm requires an iterative non-linear maximization procedure. You can specify a maximum number of iterations to prevent the program from getting lost in very lengthy iterative loops. This value must be an integer greater than 0 or less than or equal to 100 (1< value <= 100).

Variance – Covariance Matrix

When this option is selected, Analytic Solver Data Science will display the coefficient covariance matrix in the output. Entries in the matrix are the covariances between the indicated coefficients. The "on-diagonal" values are the estimated variances of the corresponding coefficients.

Multicollinearity Diagnostics

At times, variables can be highly correlated with one another which can result in large standard errors for the affected coefficients. Analytic Solver Data Science will display information useful in dealing with this problem if *Multicollinearity Diagnostics* is selected.

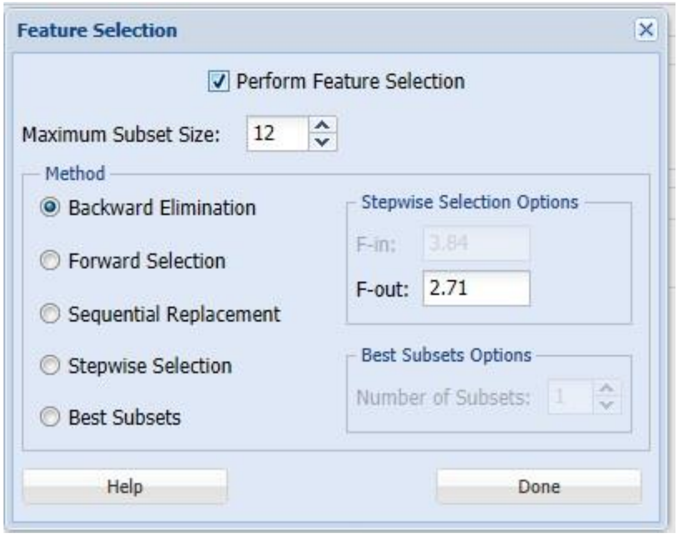
Analysis Of Coefficients

When this option is selected, Analytic Solver Data Science will produce a table with all coefficient information such as the Estimate, Odds, Standard Error, etc. When this option is not selected, Analytic Solver Data Science will only print the Estimates.

Feature Selection

When you have a large number of predictors and you would like to limit the model to only significant variables, click **Feature Selection** to open the *Feature Selection* dialog and select **Perform Feature Selection** at the top of the dialog.

Maximum Subset Size can take on values of 1 up to N where N is the number of Selected Variables. If no Categorical Variables exist, the default for this option is N. If one or more Categorical Variables exist, the default is "15".



Feature Selection

☑ Perform Feature Selection

Maximum Subset Size: 12

Method

● Backward Elimination

○ Forward Selection

○ Sequential Replacement

○ Stepwise Selection

○ Best Subsets

Stepwise Selection Options

F-in: 3.84

F-out: 2.71

Best Subsets Options

Number of Subsets: 1

Analytic Solver Data Science offers five different selection procedures for selecting the best subset of variables.

- *Backward Elimination* in which variables are eliminated one at a time, starting with the least significant. If this procedure is selected, FOUT is enabled. A statistic is calculated when variables are eliminated. For a variable to leave the regression, the statistic's value must be less than the value of FOUT (default = 2.71).

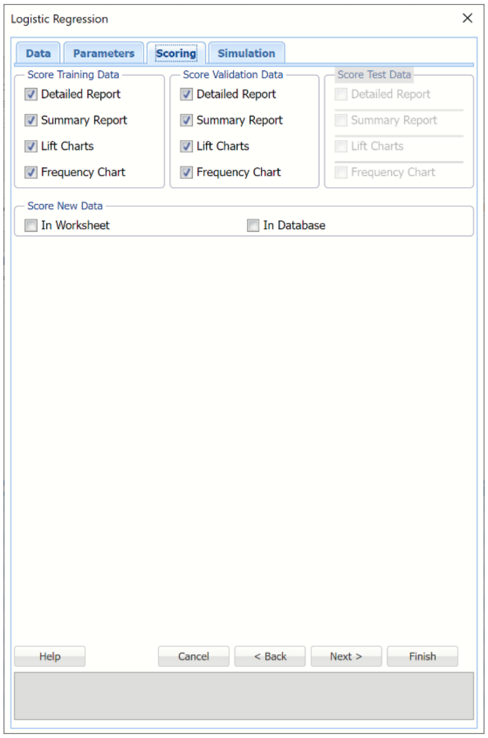
- *Forward Selection* in which variables are added one at a time, starting with the most significant. If this procedure is selected, FIN is enabled. On each iteration of the Forward Selection procedure, each variable is examined for the eligibility to enter the model. The significance of variables is measured as a partial F-statistic. Given a model at a current iteration, we perform an F Test, testing the null hypothesis stating that the regression coefficient would be zero if added to the existing set if variables and an alternative hypothesis stating otherwise. Each variable is examined to find the one with the largest partial F-Statistic. The decision rule for adding this variable into a model is: Reject the null hypothesis if the F-Statistic for this variable exceeds the critical value chosen as a threshold for the F Test (FIN value), or Accept the null hypothesis if the F-Statistic for this variable is less than a threshold. If the null hypothesis is rejected, the variable is added to the model and selection continues in the same fashion, otherwise the procedure is terminated.

- *Sequential Replacement* in which variables are sequentially replaced and replacements that improve performance are retained.

- *Stepwise selection* is similar to Forward selection except that at each stage, Analytic Solver Data Science considers dropping variables that are not statistically significant. When this procedure is selected, the Stepwise selection options FIN and FOUT are enabled. In the stepwise selection procedure a statistic is calculated when variables are added or eliminated. For a variable to come into the regression, the statistic's value must be greater than the value for FIN (default = 3.84). For a variable to leave the regression, the statistic’s value must be less than the value of FOUT (default = 2.71). The value for FIN must be greater than the value for FOUT.

- *Best Subsets* where searches of all combinations of variables are performed to observe which combination has the best fit. (This option can become quite time consuming depending on the number of input variables.) If this procedure is selected, Number of best subsets is enabled.

Logistic Regression Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

Logistic Regression Dialog, Scoring Tab

When Frequency Chart is selected, a frequency chart will be displayed when the LogReg_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user’s preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Training Data

Select these options to show an assessment of the performance of the algorithm in classifying the training data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Score Validation Data

These options are enabled when a validation dataset is present. Select these options to show an assessment of the performance of the algorithm in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Score Test Data

These options are enabled when a test dataset is present. Select these options to show an assessment of the performance of the algorithm in classifying the test data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

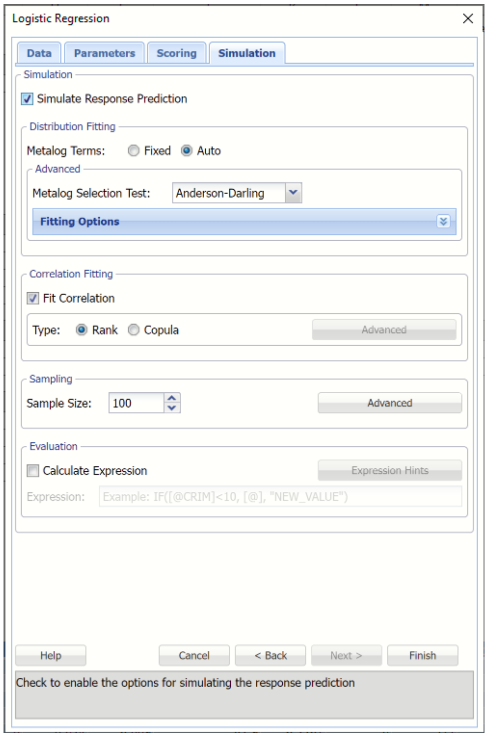
Score New Data

For information on scoring in a worksheet or database, please see the chapters “Scoring New Data” and “Scoring Test Data” in the Analytic Solver Data Science User Guide.

Simulation Tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature

Logistic Regression Simulation tab



Logistic Regression

Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

□ Calculate Expression

Expression: Example: IF([@CRIM]<10, [@], "NEW_VALUE")

Check to enable the options for simulating the response prediction

(described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, LogReg_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type. Check Simulate Response Prediction to enable the options on the Simulation tab.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the LogReg_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

k – Nearest Neighbors Classification Method

Introduction

K-nearest neighbors is a simple but powerful classifier. This method classifies a given record based on the predominant classification of it's "k" nearest neighbor records.

The k-Nearest Neighbors Classifier performs the following steps for each record in the dataset.

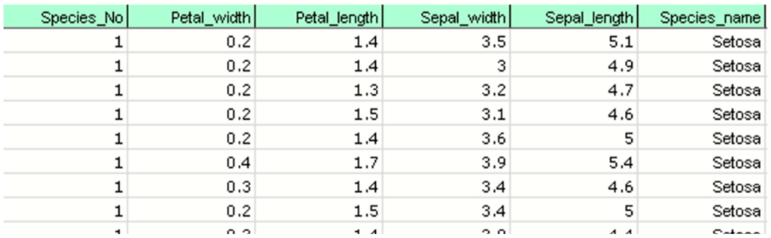
1. The Euclidean Distance between the given record and all remaining records is calculated. In order for this distance measure to be accurate, all variables must be scaled appropriately.
2. The classification of the "k" nearest neighbors is examined. The predominant classification is assigned to the given row.
3. This procedure is repeated for all remaining rows.

Analytic Solver Data Science allows the user to select a maximum value for k and builds models in parallel on all values of k up to the maximum specified value. Additional scoring can be performed on the best of these models.

As k increases, the computing time will also increase. If a high value of k is selected, such as 18 or 20, the risk of underfitting the data is high. Conversely, a low value of k, such as 1 or 2, runs the risk of overfitting the data. In most applications, k is in units of tens rather than in hundreds or thousands.

k-Nearest Neighbors Classification Example

The example below illustrates' the use of Analytic Solver Data Science's k-Nearest Neighbors classification method using the well-known Iris dataset. This dataset was introduced by R. A. Fisher and reports four characteristics of three species of the Iris flower. A portion of the dataset is shown below.

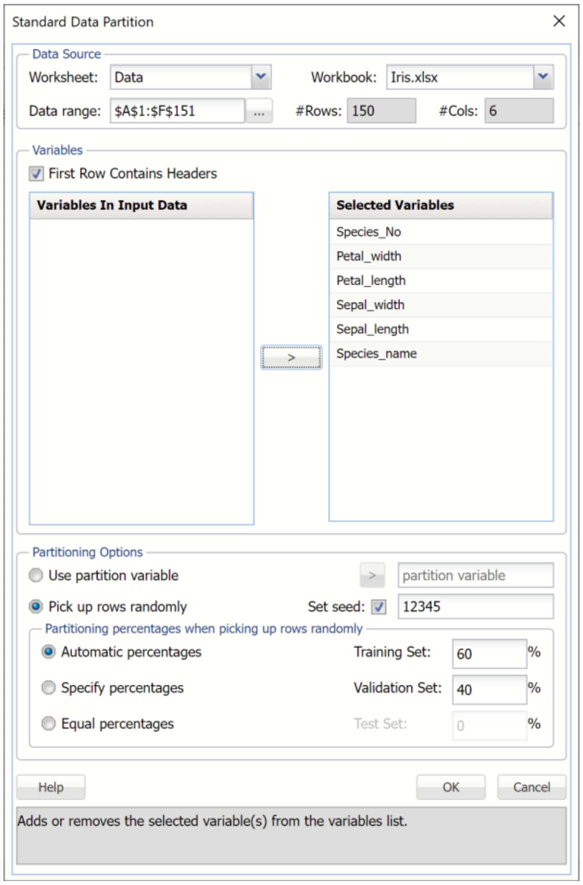


Species_No	Petal_width	Petal_length	Sepal_width	Sepal_length	Species_name
1	0.2	1.4	3.5	5.1	Setosa
1	0.2	1.4	3	4.9	Setosa
1	0.2	1.3	3.2	4.7	Setosa
1	0.2	1.5	3.1	4.6	Setosa
1	0.2	1.4	3.6	5	Setosa
1	0.4	1.7	3.9	5.4	Setosa
1	0.3	1.4	3.4	4.6	Setosa
1	0.2	1.5	3.4	5	Setosa

Inputs

1. Partition the data using a standard partition with percentages of 60% training and 40% validation (the default settings for the Automatic choice). For more information on how to partition a dataset, please see the previous *Data Science Partitioning* chapter.

Standard Data Partitioning dialog



Standard Data Partition

Data Source

Worksheet: Data

Workbook: Iris.xlsx

Data range: $A$1:$F$151

#Rows: 150

#Cols: 6

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	Species_No
	Petal_width
	Petal_length
	Sepal_width
	Sepal_length
	Species_name

Partitioning Options

○ Use partition variable

● Pick up rows randomly

Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages

○ Specify percentages

○ Equal percentages

Training Set: 60 %

Validation Set: 40 %

Test Set: 0 %

Adds or removes the selected variable(s) from the variables list.

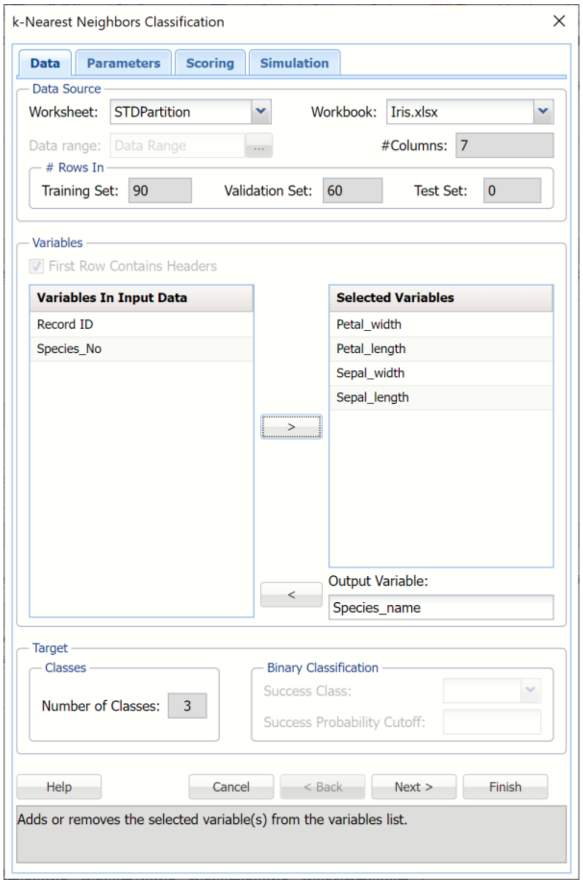
2. Click **Classify – k-Nearest Neighbors** to open the *k-Nearest Neighbors Classification* dialog.

3. Select **Petal_width**, **Petal_length**, **Sepal_width**, and **Sepal_length** under *Variables in Input Data* then click > to select as *Selected Variables*. Select **Species_name** as the *Output Variable*.

   Note: Since the variable *Species_No* is perfectly predictive of the output variable, *Species_name*, it will not be included in the model.

   Once the *Output Variable* is selected, *Number of Classes* (3) will be filled automatically. Since our output variable contains more than 2 classes, *Success Class* and *Success Probability Cutoff* are disabled.

k-Nearest Neighbors Classification dialog, Data tab



Worksheet: STDPartition
Workbook: Iris.xlsx
#Columns: 7
Training Set: 90
Validation Set: 60
Test Set: 0

Variables In Input Data	Selected Variables
Record ID	Petal_width
Species_No	Petal_length
	Sepal_width
	Sepal_length

Output Variable: Species_name
Number of Classes: 3

Adds or removes the selected variable(s) from the variables list.

4. Click **Next** to advance to the *Parameters* tab.

   Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by selecting Partition Data on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

5. Click **Rescale Data**, to open the Rescaling Dialog. Recall that the Euclidean distance measurement performs best when each variable is rescaled. Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

   Click "Done" to close the dialog without rescaling the data.

6. Enter **10** for *# Neighbor*. (This number is based on standard practice from the literature.) This is the parameter k in the k-Nearest Neighbor algorithm. If the number of observations (rows) is less than 50 then the value of k should be between 1 and the total number of observations (rows). If the number of rows is greater than 50, then the value of k should be between 1

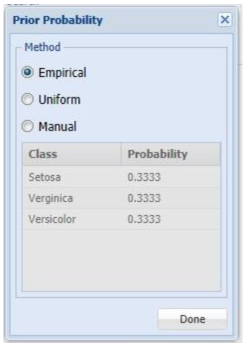
and 50. Note that if k is chosen as the total number of observations in the training set, then for any new observation, all the observations in the training set become nearest neighbors. The default value for this option is 1.

7. Select **Search 1..K** under *Nearest Neighbors Search*. When this option is selected, Analytic Solver Data Science will display the output for the best k between 1 and the value entered for *# Neighbors*. If *Fixed K* is selected, the output will be displayed for the specified value of k.

8. Click **Prior Probability** to open the is selected. Analytic Solver Data Science will incorporate prior assumptions about how frequently the different classes occur and will assume that the probability of encountering a particular class in the data set is the same as the frequency with which it occurs in the training dataset.

   - If Empirical is selected, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
   - If Uniform is selected, Analytic Solver Data Science will assume that all classes occur with equal probability.
   - If Manual is selected, the user can enter the desired class and probability value.

   Prior Probability dialog



Prior Probability

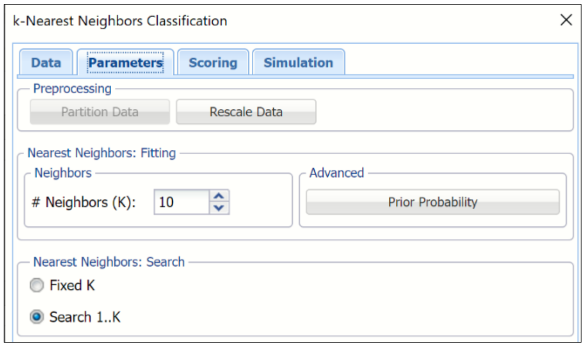
Method

● Empirical
○ Uniform
○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

For this example, click **Done** to select the default of Empirical and close the dialog.

k-Nearest Neighbors dialog, Parameters tab



k-Nearest Neighbors Classification

Neighbors (K): 10

○ Fixed K
● Search 1..K

9. Click **Next** to advance to the *Scoring* tab.

10. *Summary Report* under both *Score Training Data* and *Score Validation Data* is selected by default.

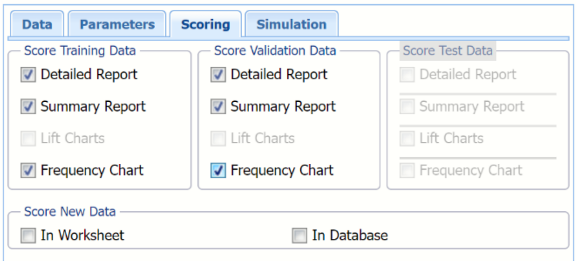
Select **Detailed Report** under both *Score Training Data* and *Score Validation Data*. Analytic Solver Data Science will create detailed and summary reports for both the training and validation sets.

When **Frequency Chart** is selected under both *Score Training Data* and *Score Validation Data*, a frequency chart will be displayed when the *KNNC_TrainingScore* and *KNNC_ValidationScore* worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Lift charts are disabled since there are more than 2 categories in our *Output Variable*, *Species_name*. Since we did not create a test partition, the options for *Score test data* are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

For more information on the *Score new data* options, please see the "Scoring New Data" chapter in the Analytic Solver Data Science User Guide.

k-Nearest Neighbors dialog, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☐ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☐ Lift Charts
☑ Frequency Chart

Score Test Data
☐ Detailed Report
☐ Summary Report
☐ Lift Charts
☐ Frequency Chart

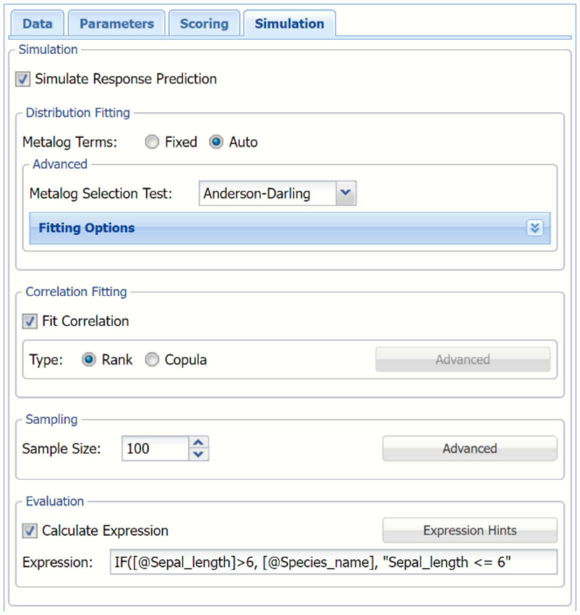
Score New Data
☐ In Worksheet
☐ In Database

11. Click **Next** to advance to the Simulation tab. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

12. Select Simulation Response Prediction to enable all options on the Simulation tab of the Discriminant Analysis dialog.

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, _Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

k-Nearest Neighbors Classification dialog, Simulation tab



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

☑ Fit Correlation

Type: ● Rank ○ Copula

Sample Size: 100

☑ Calculate Expression

Expression: IF([@Sepal_length]>6, [@Species_name], "Sepal_length <= 6"

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the KNNC_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog. For Expression, enter the following formula to display Species_name if Sepal_length is greater than 6.

IF([@Sepal_length] >6, [@Species_name], “Sepal <= 6”)

Note that variable names are case sensitive.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

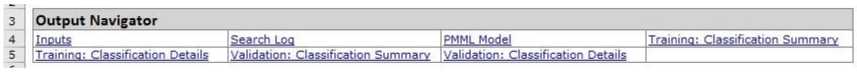
13. Click **Finish** to run k-Nearest Neighbors on the example dataset.

Output Worksheets

Worksheets containing the results are inserted to the right of the STDPartition worksheet.

KNNC_Output

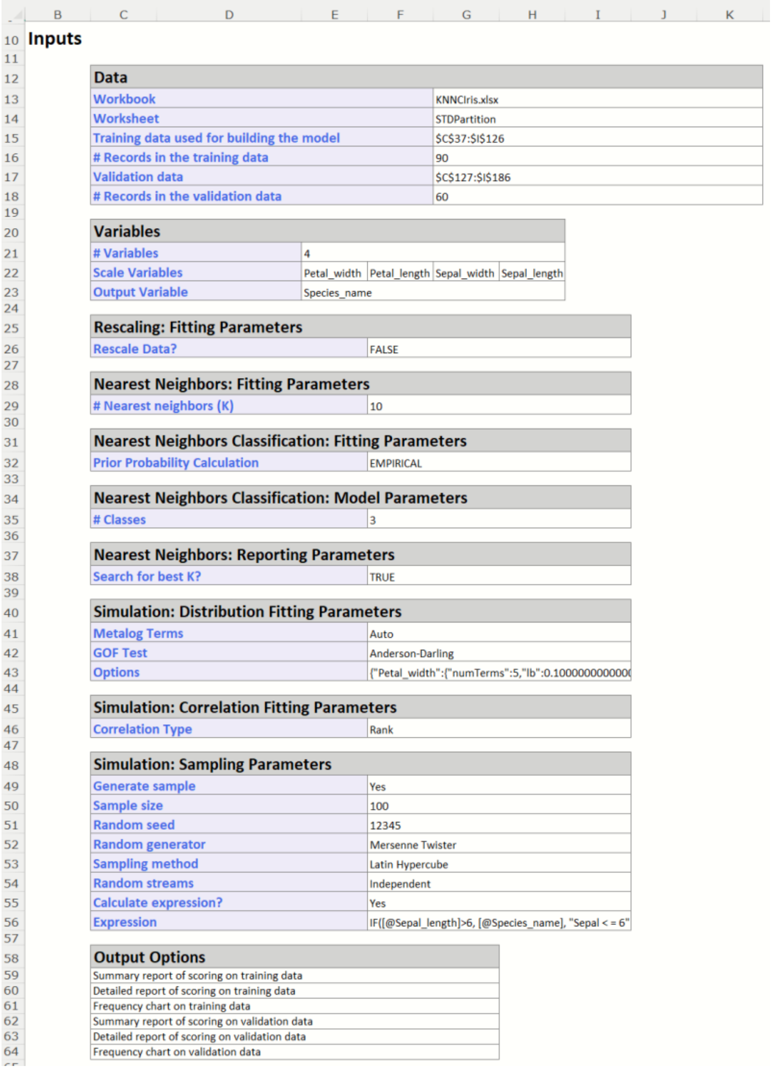
Double click the *KNNC_Output* sheet. The Output Navigator is included at the top of each output worksheet. The top part of this sheet contains all of our inputs. At the top of this sheet is the *Output Navigator*.



3	**Output Navigator**			
4	Inputs	Search Log	PMML Model	Training: Classification Summary
5	Training: Classification Details	Validation: Classification Summary	Validation: Classification Details	

Navigate to any report in the output by clicking a link in the table.

Scroll down to the Inputs section to view all selected inputs to the k-Nearest Neighbor Classification method.



Inputs

Data	
Workbook	KNNCIris.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$I$126
# Records in the training data	90
Validation data	$C$127:$I$186
# Records in the validation data	60

Variables				
# Variables	4			
Scale Variables	Petal_width	Petal_length	Sepal_width	Sepal_length
Output Variable	Species_name			

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Nearest Neighbors: Fitting Parameters	
# Nearest neighbors (K)	10

Nearest Neighbors Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Nearest Neighbors Classification: Model Parameters	
# Classes	3

Nearest Neighbors: Reporting Parameters	
Search for best K?	TRUE

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"Petal_width":{"numTerms":5,"lb":0.1000000000000

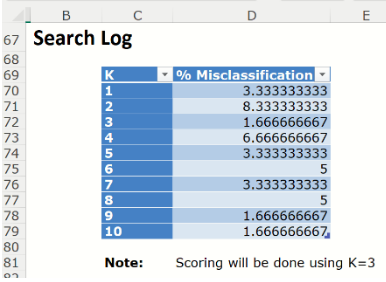
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	Yes
Expression	IF([@Sepal_length]>6, [@Species_name], "Sepal < = 6"

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Frequency chart on validation data

Scroll down a bit further to view the Search log. (This output is produced because we selected Seach 1..k on the Parameters tab. If this option had not been selected, this output would not be produced.)

Search Log output



Search Log

K	% Misclassification
1	3.333333333
2	8.333333333
3	1.666666667
4	6.666666667
5	3.333333333
6	5
7	3.333333333
8	5
9	1.666666667
10	1.666666667

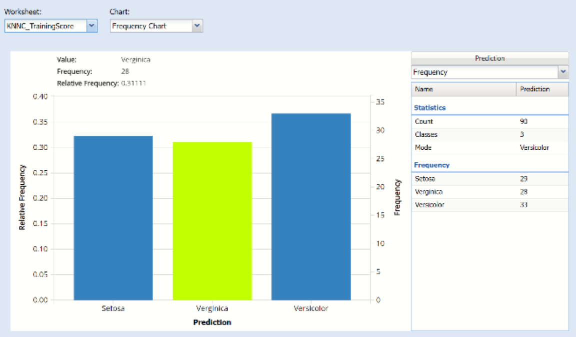
Note: Scoring will be done using K=3

The Search Log for the different k's lists the % Misclassification errors for all values of k for the validation data set, if present. The k with the smallest % Misclassification is selected as the "Best k". Scoring is performed later using this best value of k.

KNNC_TrainingScore

Click the *KNNC_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Classification Summary and the Training: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

> Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select the desired worksheet under Worksheet and the desired chart under Chart.



Worksheet: KNNC_TrainingScore

Chart: Frequency Chart

Value: Verginica
Frequency: 28
Relative Frequency: 0.31111

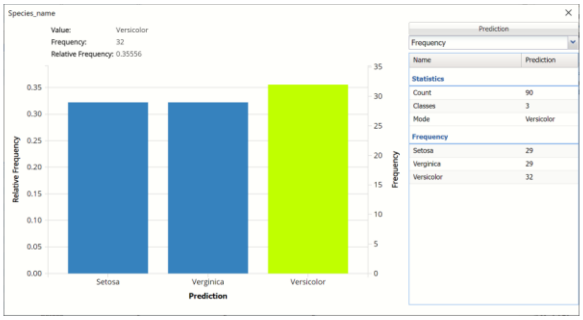
Prediction: Frequency

Name	Prediction
Statistics	
Count	90
Classes	3
Mode	Versicolor
Frequency	
Setosa	29
Verginica	28
Versicolor	33

- **Frequency Charts:** The output variable frequency chart opens automatically once the *KNNC_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *KNnC_TrainingScore* tab. To move the chart, grab the title bar and drag the chart to the desired location on the screen.

  Frequency: This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

  Frequency Chart on KNNC_TrainingScore output sheet



Species_name

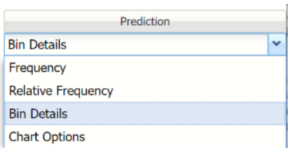
Value: Versicolor
Frequency: 32
Relative Frequency: 0.35556

Prediction: Frequency

Name	Prediction
Statistics	
Count	90
Classes	3
Mode	Versicolor
Frequency	
Setosa	29
Verginica	29
Versicolor	32

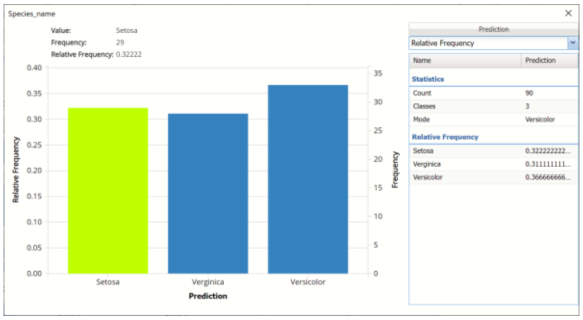
Click the down arrow next to Frequency to switch to Relative Frequency, Bin Details or Chart Options view.

Frequency Chart, Frequency View



Relative Frequency: Displays the relative frequency chart.

Relative Frequency Chart

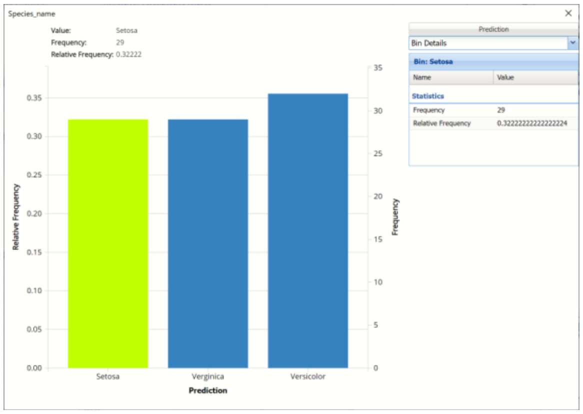


Value: Setosa
Frequency: 29
Relative Frequency: 0.32222

Relative Frequency

Name	Prediction
Statistics	
Count	90
Classes	3
Mode	Versicolor
Relative Frequency	
Setosa	0.322222222...
Verginica	0.311111111...
Versicolor	0.366666666...

Bin Details: This view displays information pertaining to each bin in the chart.



Value: Setosa
Frequency: 29
Relative Frequency: 0.32222

Bin Details

Bin: Setosa

Name	Value
Statistics	
Frequency	29
Relative Frequency	0.32222222222222224

Chart Options: Use this view to change the color of the bars in the chart.

Chart Options View

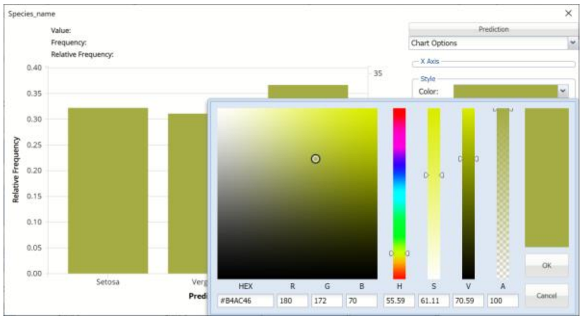


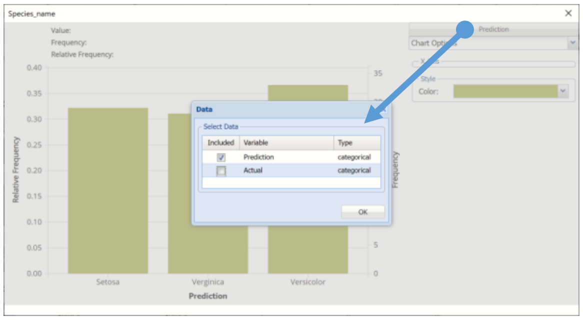
Chart Options

Color:

HEX	R	G	B	H	S	V	A
#B4AC46	180	172	70	55.59	61.11	70.59	100

- To see both the actual and predicted frequency, click Prediction and select Actual. This change will be reflected on all charts.

Click Predicted/Actual to change view

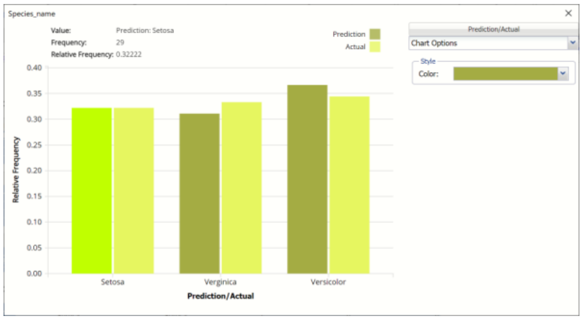


Data

Select Data

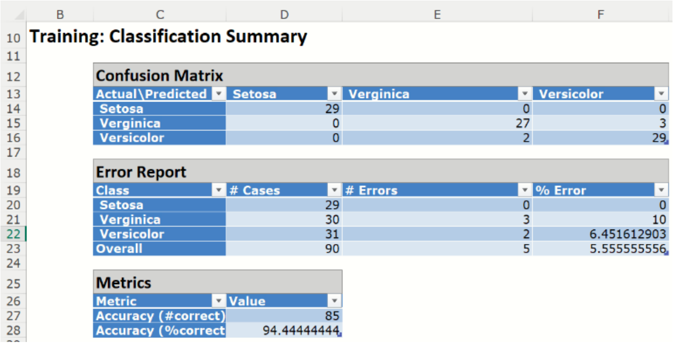
Included	Variable	Type
☑	Prediction	categorical
☐	Actual	categorical

KNNC_TrainingScore Frequency Chart with Actual and Predicted



- **Classification Summary:** In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

  KNNC_TrainingScore: Training: Classification Summary



Training: Classification Summary

Confusion Matrix

Actual\Predicted	Setosa	Verginica	Versicolor
Setosa	29	0	0
Verginica	0	27	3
Versicolor	0	2	29

Error Report

Class	# Cases	# Errors	% Error
Setosa	29	0	0
Verginica	30	3	10
Versicolor	31	2	6.451612903
Overall	90	5	5.555555556

Metrics

Metric	Value
Accuracy (#correct)	85
Accuracy (%correct	94.44444444

This Summary report tallies the actual and predicted classifications. (Predicted classifications were generated by applying the model to the validation data.) Correct classification counts are along the diagonal from the upper left to the lower right.

- There were 5 records mislabeled in the Training partition:
- Three records were assigned to the Versicolor class when they should have been assigned to the Verginica class
- Two records were assigned to the Verginica class that should have been assigned to the Versicolor class.
- The total misclassification error is 5.55% (5 misclassified records / 90 total records).

Any misclassified records will appear under Training: Classification Details in red.

Metrics

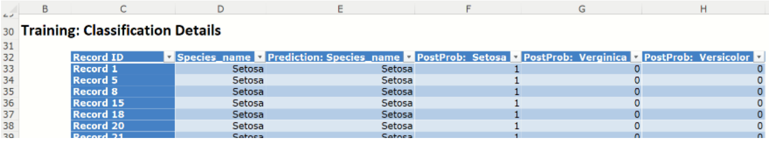
The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct = 85 and %Correct = 94.4%): Refers to the ability of the classifier to predict a class label correctly.

- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns.

Records assigned to a class other than what was predicted are highlighted in red.

KNNC_TrainingScore: Training: Classification Details



Training: Classification Details

Record ID	Species_name	Prediction: Species_name	PostProb: Setosa	PostProb: Verginica	PostProb: Versicolor
Record 1	Setosa	Setosa	1	0	0
Record 5	Setosa	Setosa	1	0	0
Record 8	Setosa	Setosa	1	0	0
Record 15	Setosa	Setosa	1	0	0
Record 18	Setosa	Setosa	1	0	0
Record 20	Setosa	Setosa	1	0	0
Record 21	Setosa	Setosa	1	0	0

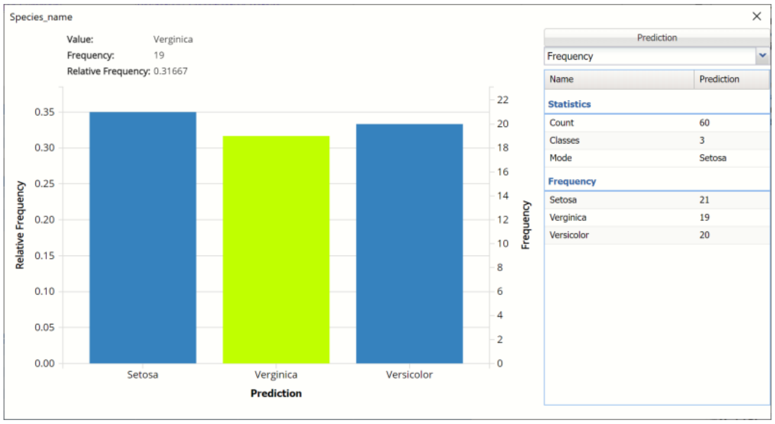
KNNC_ValidationScore

Click the *KNNC_ValidationScore* tab to view the newly added Output Variable frequency chart, the Validation: Classification Summary and the Validation: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Validation data.

- **Frequency Charts:** The output variable frequency chart opens automatically once the KNNC_ValidationScore worksheet is selected. To close this chart, click the "x" in the upper right hand corner. To reopen, click onto another tab and then click back to the KNNC_ValidationScore tab.

  Click the Frequency chart to display the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode. Selective Relative Frequency from the drop down menu, on the right, to see the relative frequencies of the output variable for both actual and predicted. See above for more information on this chart.

  KNNC_ValidationScore Frequency Chart



Species_name

Value: Verginica
Frequency: 19
Relative Frequency: 0.31667

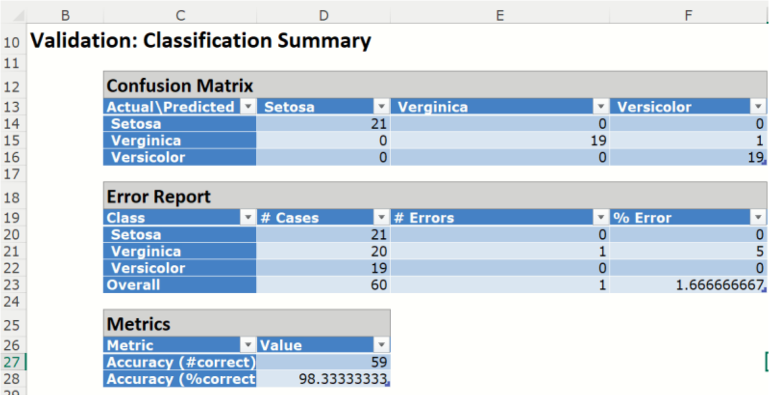
Prediction

Frequency

Name	Prediction
Statistics	
Count	60
Classes	3
Mode	Setosa
Frequency	
Setosa	21
Verginica	19
Versicolor	20

- **Classification Summary:** This report contains the confusion matrix for the validation data set.

KNNC_ValidationScore: Classification Summary



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	Setosa	Verginica	Versicolor
Setosa	21	0	0
Verginica	0	19	1
Versicolor	0	0	19

Error Report

Class	# Cases	# Errors	% Error
Setosa	21	0	0
Verginica	20	1	5
Versicolor	19	0	0
Overall	60	1	1.666666667

Metrics

Metric	Value
Accuracy (#correct)	59
Accuracy (%correct	98.33333333

When the fitted model was applied to the Validation partition, 1 record was misclassified.

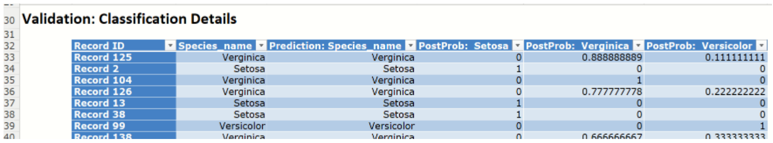
Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct = 59/60 and %Correct = 98.3%): Refers to the ability of the classifier to predict a class label correctly.

- **Classification Details**: This table displays how each observation in the validation data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Note that the largest PostProb value depicts the predicted value.

KNNC_ValidationScore: Validation: Classification Details



Validation: Classification Details

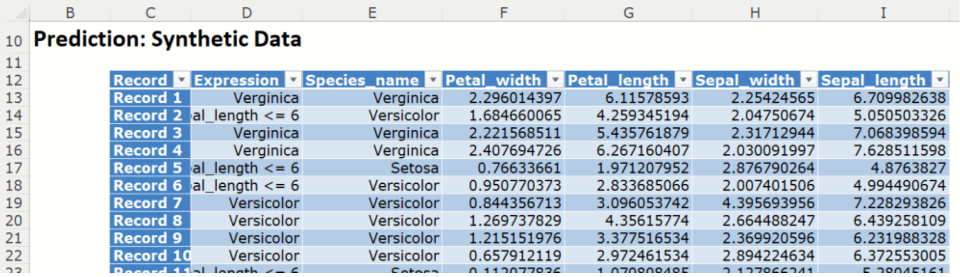
Record ID	Species_name	Prediction: Species_name	PostProb: Setosa	PostProb: Verginica	PostProb: Versicolor
Record 125	Verginica	Verginica	0	0.888888889	0.111111111
Record 2	Setosa	Setosa	1	0	0
Record 104	Verginica	Verginica	0	1	0
Record 126	Verginica	Verginica	0	0.777777778	0.222222222
Record 13	Setosa	Setosa	1	0	0
Record 38	Setosa	Setosa	1	0	0
Record 99	Versicolor	Versicolor	0	0	1
Record 138	Verginica	Verginica	0	0.666666667	0.333333333

KNNC_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, KNNC_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the k-Nearest Neighbors dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the actual output variable values for the training partition and the Excel – calculated Expression column, if populated in the dialog. A chart is also displayed with the option to switch between the Synthetic, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



Prediction: Synthetic Data

Record	Expression	Species_name	Petal_width	Petal_length	Sepal_width	Sepal_length
Record 1	Verginica	Verginica	2.296014397	6.11578593	2.25424565	6.709982638
Record 2	al_length <= 6	Versicolor	1.684660065	4.259345194	2.04750674	5.050503326
Record 3	Verginica	Verginica	2.221568511	5.435761879	2.31712944	7.068398594
Record 4	Verginica	Verginica	2.407694726	6.267160407	2.030091997	7.628511598
Record 5	al_length <= 6	Setosa	0.76633661	1.971207952	2.876790264	4.8763827
Record 6	al_length <= 6	Versicolor	0.950770373	2.833685066	2.007401506	4.994490674
Record 7	Versicolor	Versicolor	0.844356713	3.096053742	4.395693956	7.228293826
Record 8	Versicolor	Versicolor	1.269737829	4.35615774	2.664488247	6.439258109
Record 9	Versicolor	Versicolor	1.215151976	3.377516534	2.369920596	6.231988328
Record 10	Versicolor	Versicolor	0.657912119	2.972461534	2.894224634	6.372553005

Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the k-Nearest Neighbors dialog

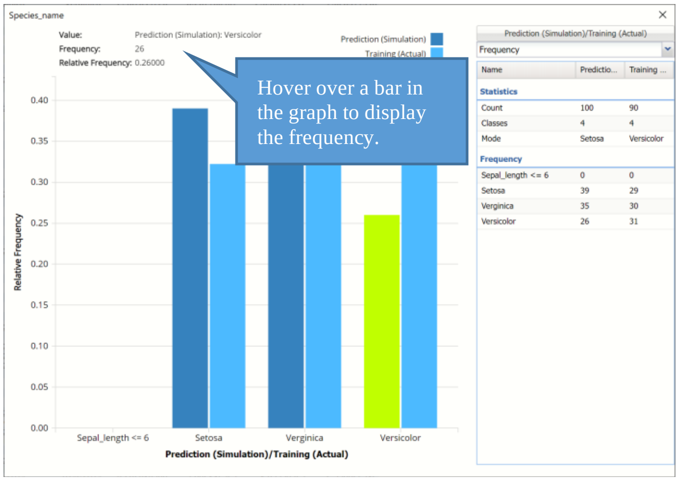
IF([@Sepal_length]>6, [@Species_name], “Sepal_length <= 6”)

The results in this column are either Setosa, Verginica, Versicolor, if a record’s Sepal_width is greater than 6, or Sepal_length <= 6, if the record’s Sepal_width is less than or equal to 6.

The remainder of the data in this report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the actual output variable in the training partition, the synthetic data and the expression, if it exists. In the screenshot below, the bars in the darker shade of blue are based on the synthetic data. The bars in the lighter shade of blue are based on the actual values in the training partition.

Frequency Chart for KNNC_Simulation output

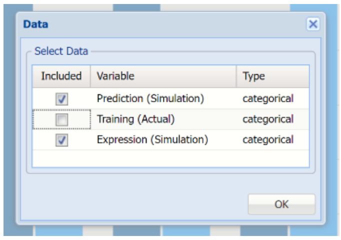


Value: Prediction (Simulation): Versicolor
Frequency: 26
Relative Frequency: 0.26000

Hover over a bar in the graph to display the frequency.

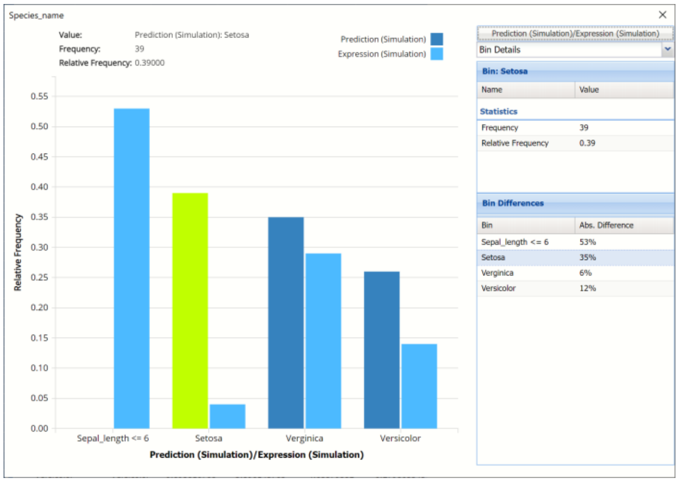
Name	Predictio...	Training ...
Statistics		
Count	100	90
Classes	4	4
Mode	Setosa	Versicolor
Frequency		
Sepal_length <= 6	0	0
Setosa	39	29
Verginica	35	30
Versicolor	26	31

Click *Prediction (Simulation)* / *Training (Actual)* to change the chart view to *Prediction (Simulation / Expression (Simulation).*



Included	Variable	Type
☑	Prediction (Simulation)	categorical
□	Training (Actual)	categorical
☑	Expression (Simulation)	categorical

Frequency Chart with Expression column



Value: Prediction (Simulation): Setosa
Frequency: 39
Relative Frequency: 0.39000

Bin: Setosa

Name	Value
Statistics	
Frequency	39
Relative Frequency	0.39

Bin Differences

Bin	Abs. Difference
Sepal_length <= 6	53%
Setosa	35%
Verginica	6%
Versicolor	12%

This chart compares the number of records in the synthetic data vs the result of the expression on the synthetic data. The dark blue columns represent the predictions in the synthetic data. In the 100 synthetic data records, 39 records are predicted to be classified as Setosa, 35 are predicted to be classified as Verginica and 26 are predicted to be classified as Versicolor. The light blue columns represent the result of the expression as applied to the synthetic data. In other words, out of 39 records in the synthetic data predicted to be classified as Setosa, only 4 are predicted to have sepal_lengths greater than 6.

Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

For information on Stored Model Sheets, in this example *DA_Stored,* please refer to the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide.

k-Nearest Neighbors Classification Options

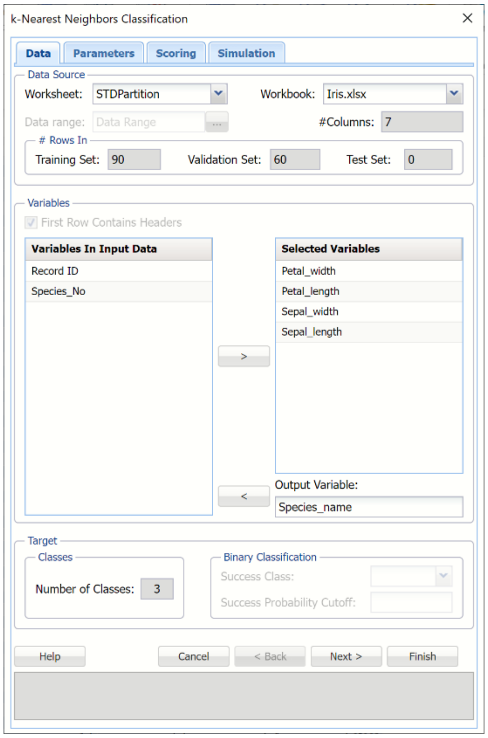
The following options appear on the *k-Nearest Neighbors Classification* dialogs.

k-Nearest Neighbors Classification, Data Tab

Variables in input data

The variables in the dataset appear here.

k-Nearest Neighbors Classification, Data tab



Data Source

Worksheet: STDPartition

Workbook: Iris.xlsx

Data range: Data Range

#Columns: 7

Rows In

Training Set: 90

Validation Set: 60

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
Record ID	Petal_width
Species_No	Petal_length
	Sepal_width
	Sepal_length

Output Variable: Species_name

Target

Classes

Number of Classes: 3

Binary Classification

Success Class:

Success Probability Cutoff:

Selected variables

The variables selected as input variables appear here

Output variable

The variable to be classified is entered here.

Number of Classes

The number of classes in the output variable appear here.

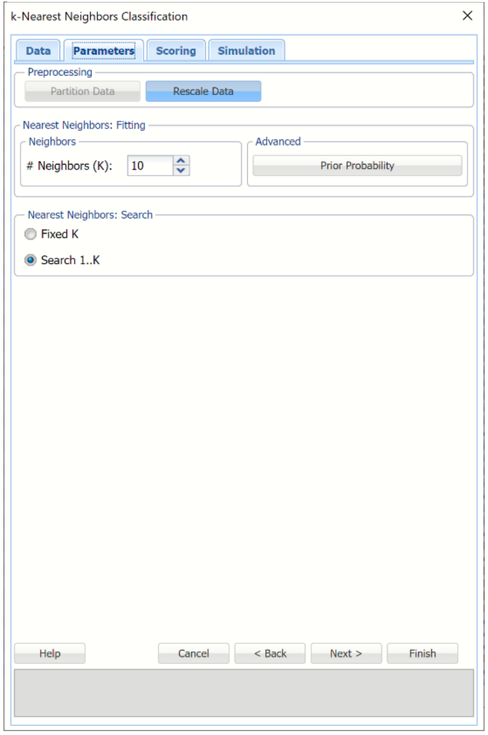
Success Class

This option is selected by default. Select the class to be considered a "success" or the significant class in the Lift Chart. This option is enabled when the number of classes in the output variable is equal to 2.

Success Probability Cutoff

Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a "success" (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a "non-success" (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

k-Nearest Neighbors Classification, Data tab



Preprocessing

Nearest Neighbors: Fitting

Neighbors

Neighbors (K): 10

Advanced

Nearest Neighbors: Search

○ Fixed K

● Search 1..K

k-Nearest Neighbors Classification, Parameters tab

# Neighbors (k)

This is the parameter k in the k-Nearest Neighbor algorithm.

Nearest Neighbors Search

If *Search 1..K* is selected, Analytic Solver Data Science will display the output for the best k between 1 and the value entered for *# Neighbors (k).*

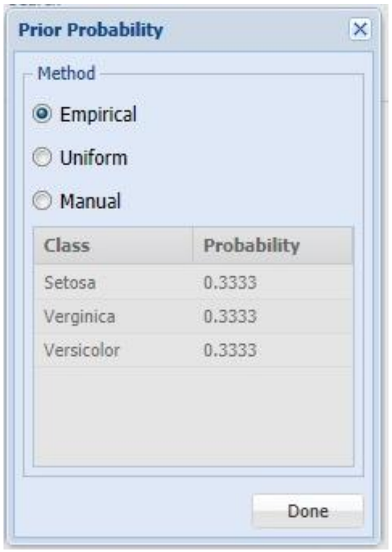
If *Fixed K* selected, the output will be displayed for the specified value of k.

Prior Probabilities

Analytic Solver Data Science will incorporate prior assumptions about how frequently the different classes occur and will assume that the probability of encountering a particular class in the data set is the same as the frequency with which it occurs in the training dataset.

- If Empirical is selected, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
- If Uniform is selected, Analytic Solver Data Science will assume that all classes occur with equal probability.

Prior Probability dialog



Prior Probability

Method

● Empirical
○ Uniform
○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

Done

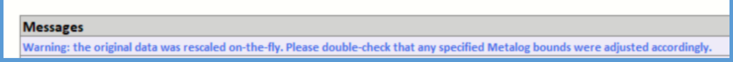
- If Manual is selected, the user can enter the desired class and probability value.

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Notes on Rescaling and the Simulation functionality

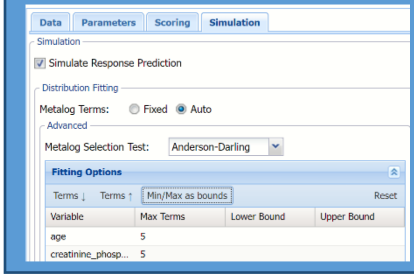
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the KNNC_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

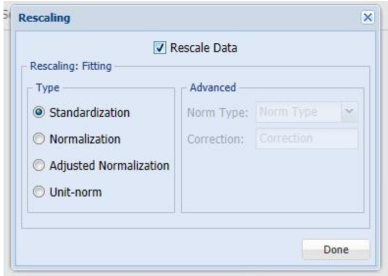
Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by selecting Partition Options on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Rescaling dialog



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization
○ Normalization
○ Adjusted Normalization
○ Unit-norm

Advanced

Norm Type: Norm Type

Correction: Correction

Done

k-Nearest Neighbors Classification, Scoring tab

When Frequency Chart is selected, a frequency chart will be displayed when the KNNC_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user’s preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Training Data

Select these options to show an assessment of the performance of the algorithm in classifying the training data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Score Validation Data

These options are enabled when a validation dataset is present. Select these options to show an assessment of the performance of the algorithm in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Score Test Data

These options are enabled when a test dataset is present. Select these options to show an assessment of the performance of the algorithm in classifying the test data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Score New Data

For information on scoring in a worksheet or database, please see the chapters "Scoring New Data" and "Scoring Test Data" in the Analytic Solver Data Science User Guide.

k-Nearest Neighbors Classification, Simulation tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, KNNC_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the KNNC_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

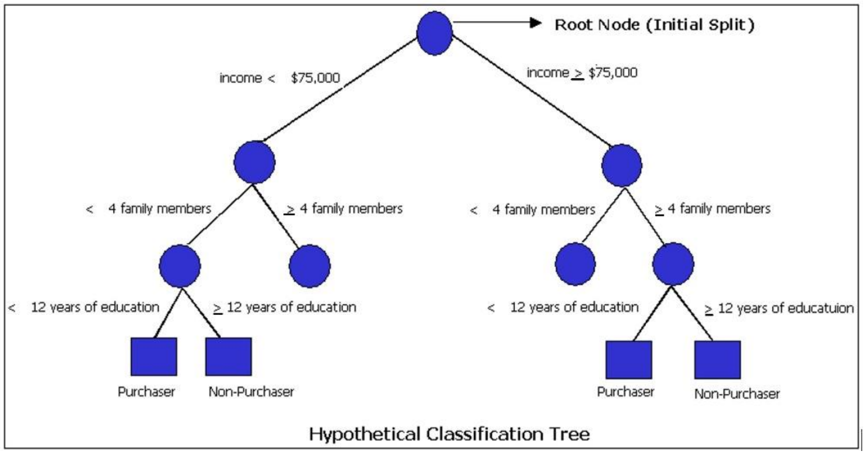
Classification Tree Classification Method

Introduction

Classification tree methods (also known as decision tree methods) are a good choice when the data science task is classification or prediction of outcomes. The goal of this algorithm is to generate rules that can be easily understood, explained, and translated into SQL or a natural query language.

A Classification tree labels, records and assigns variables to discrete classes and can also provide a measure of confidence that the classification is correct. The tree is built through a process known as binary recursive partitioning. This is an iterative process of splitting the data into partitions, and then splitting it up further on each of the branches.

Initially, a training set is created where the classification label (say, "purchaser" or "non-purchaser") is known (pre-classified) for each record. In the next step, the algorithm systematically assigns each record to one of two subsets on the some basis, for example income >= $75,000 or income < $75,000). The object is to attain as homogeneous set of labels (say, "purchaser" or "non-purchaser") as possible in each partition. This splitting (or partitioning) is then applied to each of the new partitions. The process continues until no more useful splits can be found. The heart of the algorithm is the rule that determines the initial split rule (see figure below).



Hypothetical Classification Tree

As explained above, the process starts with a training set consisting of pre-classified records (target field or dependent variable with a known class or label such as "purchaser" or "non-purchaser"). The goal is to build a tree that distinguishes among the classes. For simplicity, assume that there are only two target classes and that each split is a binary partition. The splitting criterion easily generalizes to multiple classes, and any multi-way partitioning can be achieved through repeated binary splits. To choose the best splitter at a node, the algorithm considers each input field in turn. In essence, each field is sorted. Then, every possible split is tried and considered, and the best split is the one which produces the largest decrease in diversity of the classification label within each partition (this is just another way of saying "the increase in homogeneity"). This is repeated for all fields, and the winner is chosen as the best splitter for that node. The process is continued at subsequent nodes until a full tree is generated.

Analytic Solver Data Science uses the Gini index as the splitting criterion, which is a very commonly used measure of inequality. The index fluctuates between a value of 0 and 1. A Gini index of 0 would indicate that all records in the node belong to the same category. A Gini index of 1 would indicate that each record in the node belongs to a different category. For a complete discussion of this index, please see Leo Breiman's and Richard Friedman's book, *Classification and Regression Trees* (3).

Pruning the tree

Pruning is the process of removing leaves and branches to improve the performance of the decision tree when moving from the training data (where the classification is known) to real-world applications (where the classification is unknown). The tree-building algorithm makes the best split at the root node where there are the largest number of records and, hence, considerable information. Each subsequent split has a smaller and less representative population with which to work. Towards the end, idiosyncrasies of training records at a particular node display patterns that are peculiar only to those records. These patterns can become meaningless and sometimes harmful for prediction if you try to extend rules based on them to larger populations.

For example, say the classification tree is trying to predict height and it comes to a node containing one tall person X and several other shorter people. The algorithm can decrease diversity at that node by a new rule imposing "people named X are tall" and thus classify the training data. In the real world this rule is obviously inappropriate. Pruning methods solve this problem **--** they let the tree grow to maximum size, then remove smaller branches that fail to generalize. (Note: In practice, we do not include irrelevant fields such as "name", this is simply used an illustration.)

Since the tree is "grown" from the training data set, when it has reached full structure it usually suffers from over-fitting (i.e. it is "explaining" random elements of the training data that are not likely to be features of the larger population of data). This results in poor performance on real life data. Therefore, trees must be pruned using the validation data set.

Single Tree Classification Tree Example

This example illustrates how to create a classification tree using the single classification tree method using the Boston_Housing.xlsx example dataset.

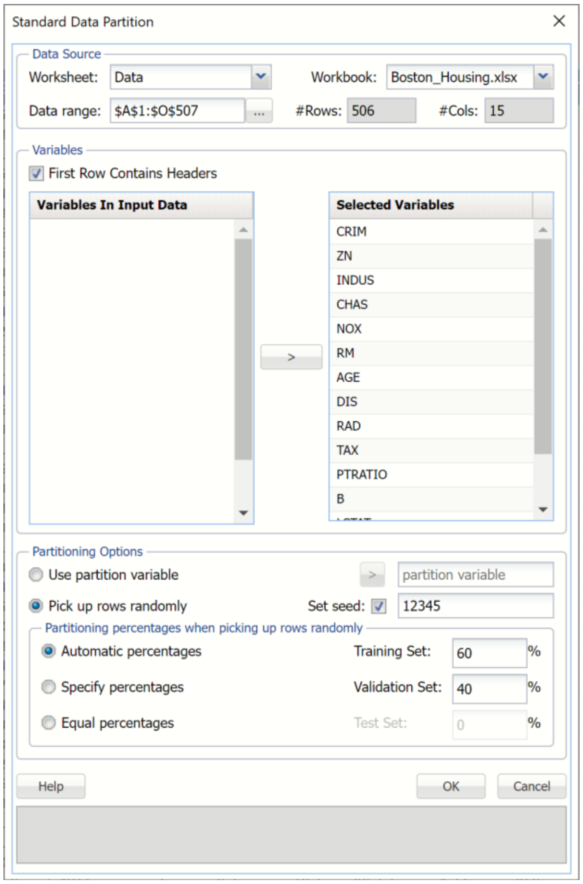
Click **Help – Example Models**, then **Forecasting/Data Science Examples** to open the **Boston_Housing.xlsx** dataset. This dataset includes fourteen variables pertaining to housing prices from census tracts in the Boston area collected by the US Census Bureau.

All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described in the What's New section of this guide and then more in depth in the Analytic Solver Data Science Reference Guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CFBM_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type. Since this new functionality does not support categorical variables, these types of variables will not be present in the model, only continuous variables.

Inputs

1. First, we partition the data into training and validation sets using the Standard Data Partition defaults of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.

   Standard Data Partition dialog



Standard Data Partition

Data Source

Worksheet: Data   Workbook: Boston_Housing.xlsx

Data range: $A$1:$O$507   #Rows: 506   #Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables: CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B

Partitioning Options

○ Use partition variable   partition variable

● Pick up rows randomly   Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages   Training Set: 60 %

○ Specify percentages   Validation Set: 40 %

○ Equal percentages   Test Set: 0 %

2. With the STDPartition worksheet selected, click **Classify – Classification Tree** to open the Classification Tree dialog. Note: A cell must be selected within the Data Range, A1:O57.

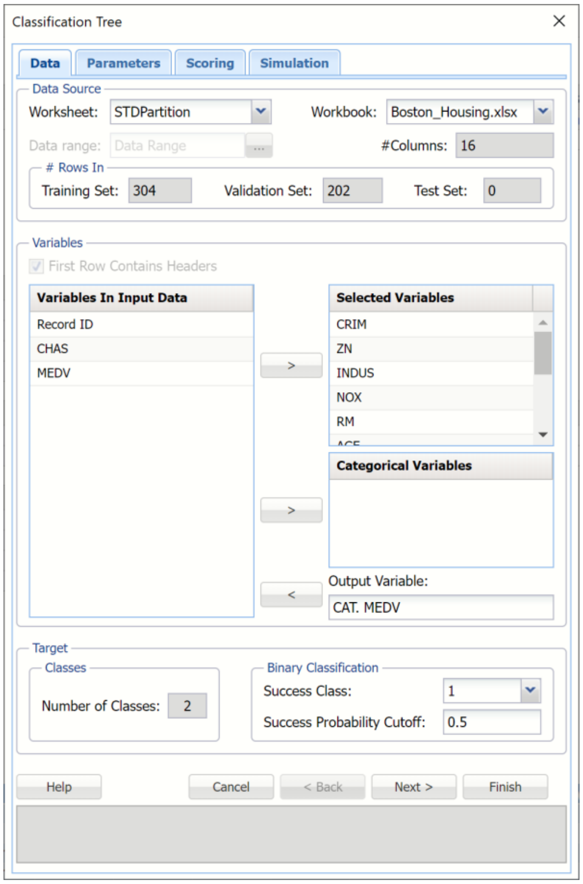
3. Select **CAT. MEDV** as the *Output variable*. Then select **all remaining variables *except CHAS, MEDV and Record ID*** under *Variables in Input Data*, then click > to move them to the *Selected Variables field*.

   Note: MEDV is not included in the Input since CAT. MEDV, derived from MEDV, is included in the model. CHAS is not included in the Input since this is a categorical variable. Recall that the new Simulation functionality included in Analytic Solver Data Science does not support categorical variables.

4. Choose the value that will be the indicator of "Success" by clicking the down arrow next to *Success Class*. In this example, we will use the default of 1.

5. Enter a value between 0 and 1 for *Success Probability Cutoff*. If the Probability of success (probability of the output variable = 1) is less than this value, then a 0 will be entered for the class value, otherwise a 1 will be entered for the class value. In this example, we will keep the default of 0.5.

Classification Tree dialog, Data tab



Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Rows In

Training Set: 304

Validation Set: 202

Test Set: 0

☑ First Row Contains Headers

Variables In Input Data: Record ID, CHAS, MEDV

Selected Variables: CRIM, ZN, INDUS, NOX, RM

Categorical Variables:

Output Variable: CAT. MEDV

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

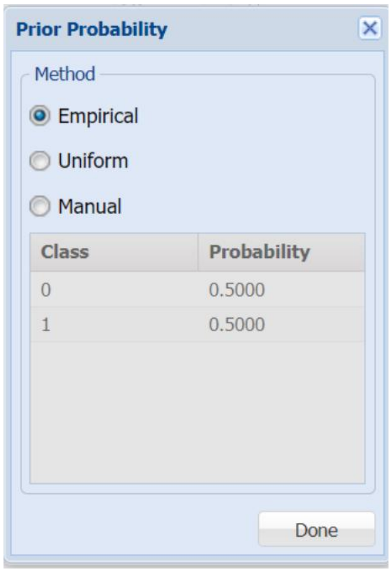
6. Click **Next** to advance to the *Classification Tree – Parameters* tab.

As discussed in previous sections, Analytic Solver Data Science includes the ability to partition and scale a dataset from within a classification or prediction method by clicking Partition Data and/or Rescale Data on the Parameters tab. Analytic Solver Data Science will partition and/or rescale your dataset (according to the partition and rescaling options you set) immediately before running the classification method. If either partitioning and/or rescaling has already occurred on the dataset, the option(s) will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter. For more information on scaling your data, please see the Transform Continuous Data chapter.

7. In the *Tree Growth* section, leave all selections at their default settings. To limit the growth of Tree Levels, Nodes, Splits or the number of Records in a Terminal Node, select the desired component(s) and enter the desired value(s), i.e. if 10 is entered for Levels, the tree will be limited to 10 levels.

8. Click **Prior Probability**. Three options appear in the *Prior Probability* Dialog: *Empirical, Uniform* and *Manual.*

Classification Tree dialog, Prior Probability dialog



Prior Probability

Method

● Empirical

○ Uniform

○ Manual

Class	Probability
0	0.5000
1	0.5000

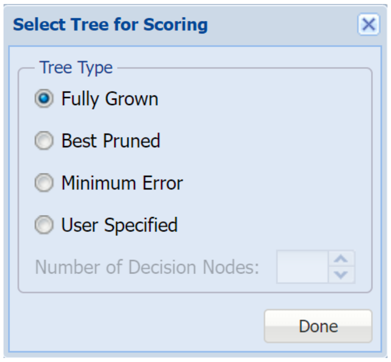
- If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
- If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.
- Select the third option, *Manual*, to manually enter the desired class and probability value.

Click **Done** to accept the default section, Empirical, and close the dialog.

9. Select *Prune (Using Validation Set)***.** **(**This option is enabled when a Validation Dataset exists.) Analytic Solver Data Science will prune the tree using the validation set when this option is selected. (Pruning the tree using the validation set reduces the error from over-fitting the tree to the training data.)

10. Click Tree for Scoring and select Fully Grown.

Classification Tree, Select Tree for Scoring dialog



Select Tree for Scoring

Tree Type

● Fully Grown

○ Best Pruned

○ Minimum Error

○ User Specified

Number of Decision Nodes:

11. Select Show Feature Importance to include the Feature Importance Data Table in the output. This table shows the relative importance of the feature measured as the reduction of the error criterion during the tree growth.

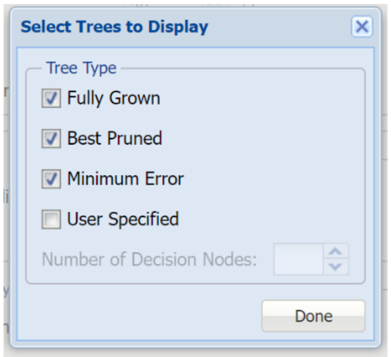
12. Leave *Maximum Number of Levels* at the default setting of 7. This option specifies the maximum number of levels in the tree to be displayed in the output.

13. Select *Trees to Display* to select the types of trees to display: Fully Grown, Best Pruned, Minimum Error or User Specified.

    - Select *Fully Grown* to "grow" a complete tree using the training data.
    - Select *Best Pruned* to create a tree with the fewest number of nodes, subject to the constraint that the error be kept below a specified level (minimum error rate plus the standard error of that error rate).
    - Select *Minimum error* to produce a tree that yields the minimum classification error rate when tested on the validation data.
    - To create a tree with a specified number of decision nodes select *User Specified* and enter the desired number of nodes.

    Select **Fully Grown**, **Best Pruned**, and **Minimum Error**. Then click Done to close the dialog.

Classification Tree, Select Trees to Display dialog



Tree Type

☑ Fully Grown

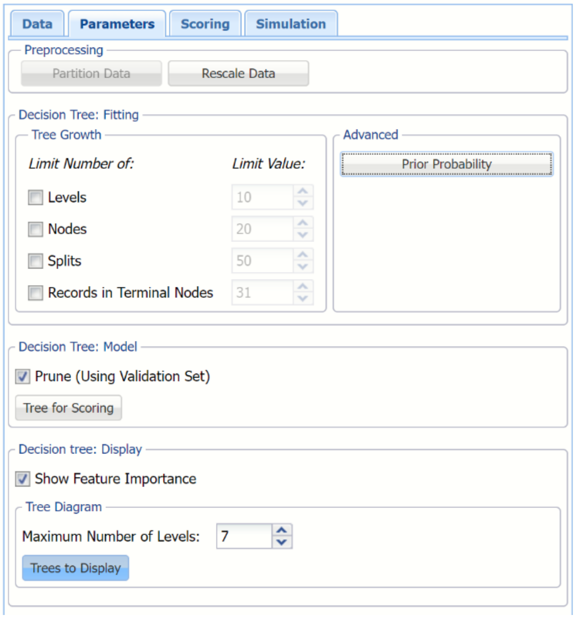
☑ Best Pruned

☑ Minimum Error

□ User Specified

Number of Decision Nodes:

Classification Tree, Parameters tab



Limit Number of: *Limit Value:*

☐ Levels 10

☐ Nodes 20

☐ Splits 50

☐ Records in Terminal Nodes 31

☑ Prune (Using Validation Set)

☑ Show Feature Importance

Maximum Number of Levels: 7

14. Click **Next** to advance to the *Classification Tree* **-** *Scoring* tab.

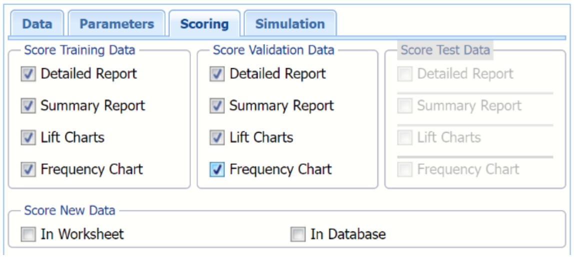
15. Select **Detailed report**, **Summary report**, **Lift charts** and **Frequency Chart** under both *Score Training Data* and *Score Validation Data*. Analytic Solver Data Science will create a detailed report, complete with the Output Navigator for ease in routing to specific areas in the output, a report that summarizes the regression output for both datasets, and lift charts, ROC curves, and Decile charts for both partitions.

    When **Frequency Chart** is selected under both *Score Training Data* and *Score Validation Data*, a frequency chart will be displayed when the *CT_TrainingScore* and *CT_ValidationScore* worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user’s preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

    Since we did not create a test partition when we partitioned our dataset, Score Test Data options are disabled. See the chapter “Data Science Partitioning” for details on how to create a test set.

    For information on scoring in a worksheet or database, please see the “Scoring New Data” chapter in the Analytic Solver Data Science User Guide.

Classification Tree dialog, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

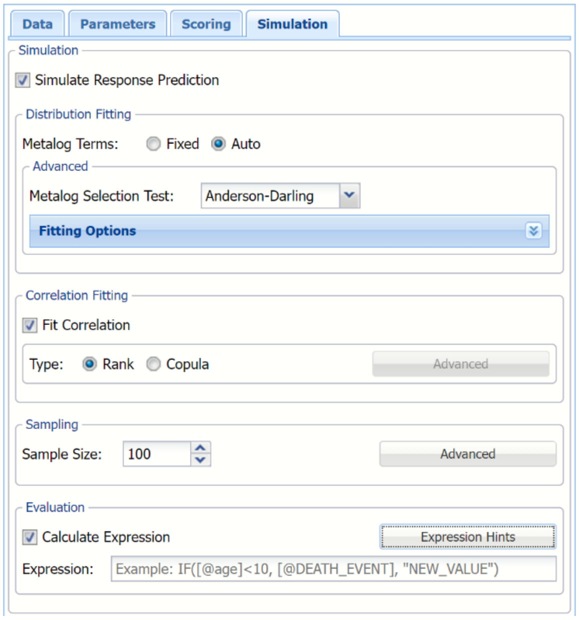
Score New Data
□ In Worksheet
□ In Database

16. Click Next to advance to the Simulation tab.

17. Select **Simulate Response Prediction** to enable all options on the the Simulation tab. (This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Anlaytic Solver Upgrade.)

Simulation tab: All supervised algorithms include a new Simulation. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CT_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Classification Tree dialog, Simulation tab



Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula

Sampling
Sample Size: 100

Evaluation
☑ Calculate Expression
Expression: Example: IF([@age]<10, [@DEATH_EVENT], "NEW_VALUE")

18. **Evaluation:** Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the CT_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

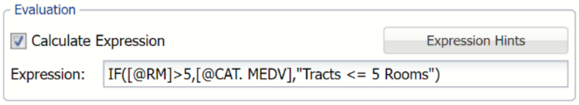
For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog.

For Expression, enter the following formula to display census tracts with an average number of bedrooms greater than or equal to 5.

IF[@RM]>5,[@CAT.MEDV],"Tracts <= 5 Rooms")

Note that variable names are case sensitive.

Evaluation section on the Classification Tree dialog, Simulation tab



☑ Calculate Expression

Expression: IF([@RM]>5,[@CAT. MEDV],"Tracts <= 5 Rooms")

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

19. Click Finish to run Classification Trees on the example dataset. Output worksheets are inserted to the right of the STDPartition worksheet.

Output

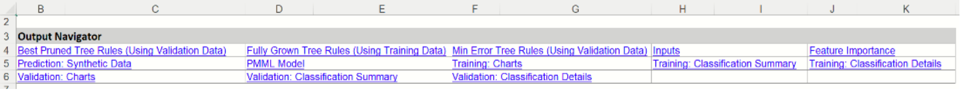
Output containing the results from Classification Trees will be inserted into the active workbook to the right of the STDPartition worksheet and also in the Model tab of the task pane under Reports – Clasification Tree.

CT_Output

This result worksheet includes 4 segments: Output Navigator, Inputs, Training Log, Prune Log and Feature Importance.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

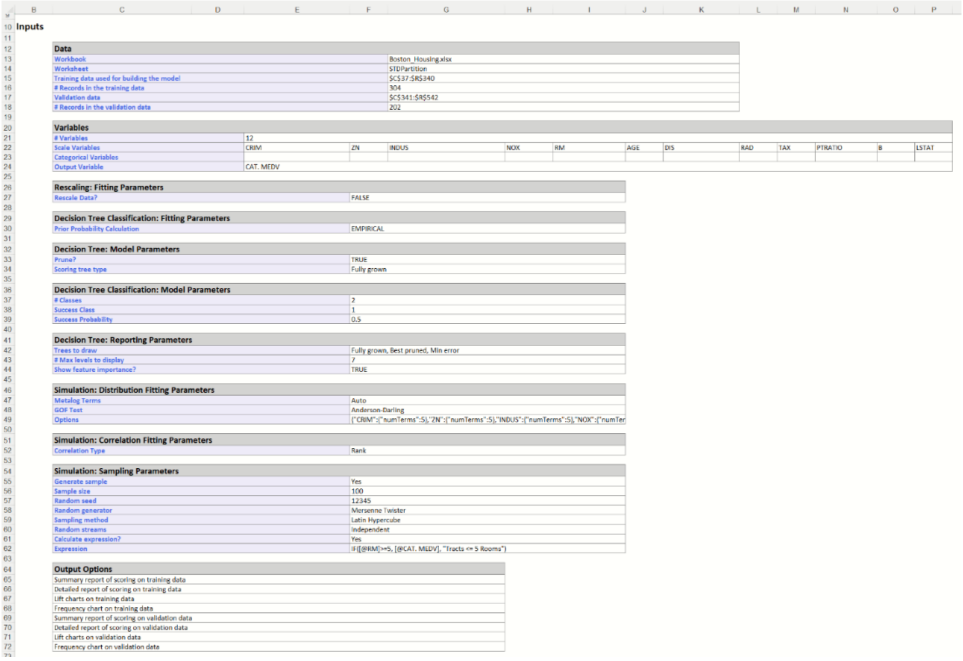
CT_Output: Output Navigator



Output Navigator				
Best Pruned Tree Rules (Using Validation Data)	Fully Grown Tree Rules (Using Training Data)	Min Error Tree Rules (Using Validation Data)	Inputs	Feature Importance
Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Classification Summary	Training: Classification Details
Validation: Charts	Validation: Classification Summary	Validation: Classification Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Classification Tree dialog.

CT_Output: Inputs



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	CAT. MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Decision Tree Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Decision Tree: Model Parameters	
Prune?	TRUE
Scoring tree type	Fully grown

Decision Tree Classification: Model Parameters	
# Classes	2
Success Class	1
Success Probability	0.5

Decision Tree: Reporting Parameters	
Trees to draw	Fully grown, Best pruned, Min error
# Max levels to display	7
Show feature importance?	TRUE

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5},"ZN":{"numTerms":5},"INDUS":{"numTerms":5},"NOX":{"numTer

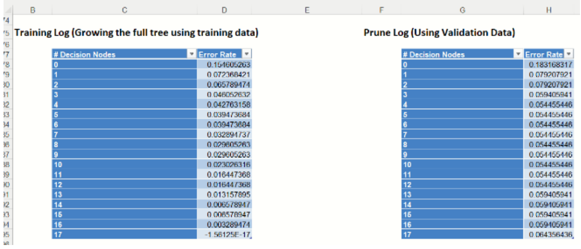
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	Yes
Expression	IF([@RM]>=5, [@CAT. MEDV], "Tracts <= 5 Rooms")

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Training Log and Prune Log:** The training log shows the misclassification (error) rate as each additional node is added to the tree, starting with 0 nodes and ending with 17. The error rate is -1.56E-17. Scoring will be performed using this tree.

  Note that since scoring on the training data will be performed using this tree, the total % Error for the Confusion Matrix on the CT_TrainingScore output sheet will be equal to the error rate of the fully gown tree, reported as a percentage.



Training Log (Growing the full tree using training data)

# Decision Nodes	Error Rate
0	0.154605263
1	0.072368421
2	0.065789474
3	0.046052632
4	0.042763158
5	0.039473684
6	0.039473684
7	0.032894737
8	0.029605263
9	0.029605263
10	0.023026316
11	0.016447368
12	0.016447368
13	0.013157895
14	0.006578947
15	0.006578947
16	0.003289474
17	-1.56125E-17

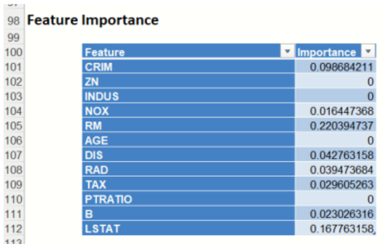
Prune Log (Using Validation Data)

# Decision Nodes	Error Rate
0	0.183168317
1	0.079207921
2	0.079207921
3	0.059405941
4	0.054455446
5	0.054455446
6	0.054455446
7	0.054455446
8	0.054455446
9	0.054455446
10	0.054455446
11	0.054455446
12	0.054455446
13	0.059405941
14	0.059405941
15	0.059405941
16	0.059405941
17	0.064356436

Analytic Solver Data Science chooses the number of decision nodes for the pruned tree and the minimum error tree from the values of Validation MSE. In the Prune log shown above, the smallest Validation MSE error belongs to the trees with 4, 5, 6, 7, 8, 9, 10, 11 and 12 decision nodes. Where there is a tie, meaning when multiple trees have the exact same Error Rate, the tree with the smaller number of nodes is selected. In this case, the tree with four (4) decision nodes is the Minimum Error Tree – the tree with the smallest misclassification error in the validation dataset.

- **Feature Importance:** Select *Feature Importance* to include the *Features Importance* table in the output. This table displays the variables that are included in the model along with their Importance value. The larger the Importance value, the bigger the influence the variable has on the predicted classification. In this instance, the census tracts with homes with many rooms will be predicted as having a larger selling price.

Feature Importance Report

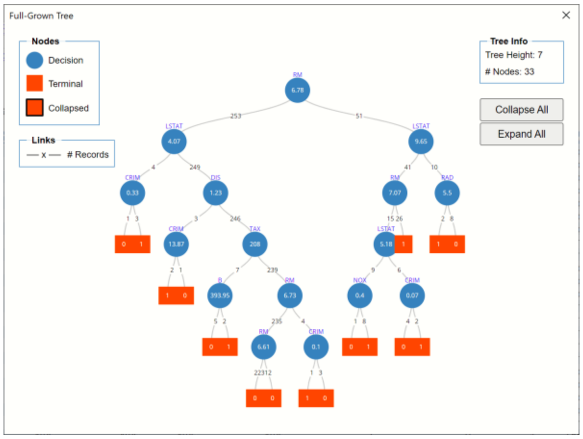


Feature Importance

	Feature	Importance
101	CRIM	0.098684211
102	ZN	0
103	INDUS	0
104	NOX	0.016447368
105	RM	0.220394737
106	AGE	0
107	DIS	0.042763158
108	RAD	0.039473684
109	TAX	0.029605263
110	PTRATIO	0
111	B	0.023026316
112	LSTAT	0.167763158

CT_FullTree

Click *CT_FullTree* to view the full tree.



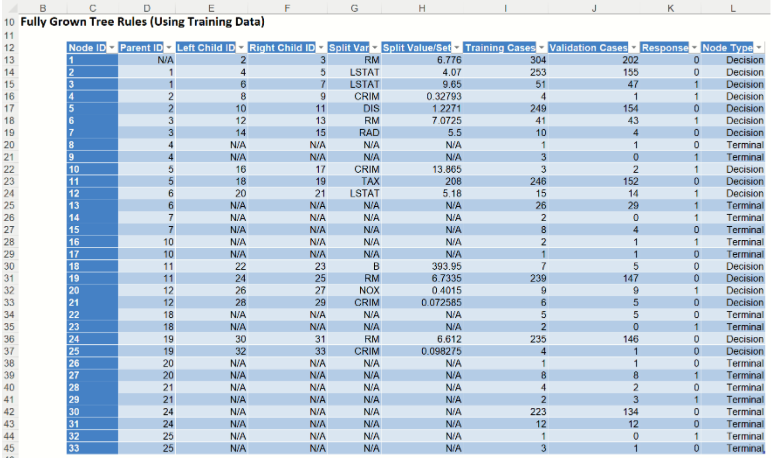
Tree Info
Tree Height: 7
Nodes: 33

Recall that the objective of this example is to classify each case as a 0 (low median value) or a 1 (high median value). Consider the top decision node (denoted by a circle). The label above this node indicates the variable represented at this node (i.e. the variable selected for the first split) in this case, RM (Average # of Rooms). The value inside the node indicates the split threshold. (Hover over the decision node to read the decision rule.) If the RM value for a specific record is greater than or equal to 6.78 (RM >= 6.78), the record will be assigned to the right node. If the RM value for the record is less than 6.78, the value will be assigned to the left node. There are 51 records with values for the RM variable greater than or equal to 6.78 while 253 records contained RM values less than 6.78. We can think of records with an RM value less than 6.78 (RM < 6.78) as tentatively classified as "0" (low median value). Any record where RM >= 6.78 can be tentatively classified as a "1" (high median value).

Let’s follow the tree as it descends to the left for a couple levels. The 253 records with RM values less than 6.78 are further split as we move down the tree. The second split occurs with the LSTAT variable (percent of the population that is of lower socioeconomic status). The LSTAT values for 4 records (out of 253) fell below the split value of 4.07. These records are tentatively classified as a “1” – high median value. The LSTAT values for the remaining 249 records are greater than or equal to 4.07, and are tentatively classified as “0" – low median value.

Following the tree to the left, the 4 records with a LSTAT value < 4.07 are split on the CRIM variable node, CRIM = per capita crime rate by town. Records with CRIM values greater than or equal .33 are classified as a 1 and records with CRIM values less than .33 are classified as a 0, in the terminal nodes. No further splits occur on terminal nodes.

The structure of the full tree will be clear by reading the **Full – Grown Tree Rules** also on the CT_FullTree tab.



Fully Grown Tree Rules (Using Training Data)

Node ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	RM	6.776	304	202	0	Decision
2	1	4	5	LSTAT	4.07	253	155	0	Decision
3	1	6	7	LSTAT	9.65	51	47	1	Decision
4	2	8	9	CRIM	0.32793	4	1	1	Decision
5	2	10	11	DIS	1.2271	249	154	0	Decision
6	3	12	13	RM	7.0725	41	43	1	Decision
7	3	14	15	RAD	5.5	10	4	0	Decision
8	4	N/A	N/A	N/A	N/A	1	1	0	Terminal
9	4	N/A	N/A	N/A	N/A	3	0	1	Terminal
10	5	16	17	CRIM	13.865	3	2	1	Decision
11	5	18	19	TAX	208	246	152	0	Decision
12	6	20	21	LSTAT	5.18	15	14	1	Decision
13	6	N/A	N/A	N/A	N/A	26	29	1	Terminal
14	7	N/A	N/A	N/A	N/A	2	0	1	Terminal
15	7	N/A	N/A	N/A	N/A	8	4	0	Terminal
16	10	N/A	N/A	N/A	N/A	2	1	1	Terminal
17	10	N/A	N/A	N/A	N/A	1	1	0	Terminal
18	11	22	23	B	393.95	7	5	0	Decision
19	11	24	25	RM	6.7335	239	147	0	Decision
20	12	26	27	NOX	0.4015	9	9	1	Decision
21	12	28	29	CRIM	0.072585	6	5	0	Decision
22	18	N/A	N/A	N/A	N/A	5	5	0	Terminal
23	18	N/A	N/A	N/A	N/A	2	0	1	Terminal
24	19	30	31	RM	6.612	235	146	0	Decision
25	19	32	33	CRIM	0.098275	4	1	0	Decision
26	20	N/A	N/A	N/A	N/A	1	1	0	Terminal
27	20	N/A	N/A	N/A	N/A	8	8	1	Terminal
28	21	N/A	N/A	N/A	N/A	4	2	0	Terminal
29	21	N/A	N/A	N/A	N/A	2	3	1	Terminal
30	24	N/A	N/A	N/A	N/A	223	134	0	Terminal
31	24	N/A	N/A	N/A	N/A	12	12	0	Terminal
32	25	N/A	N/A	N/A	N/A	1	0	1	Terminal
33	25	N/A	N/A	N/A	N/A	3	1	0	Terminal

Node ID 1: The first entry in this table shows a split on the RM variable with a split value of 6.776 (rounded to 6.78). The 304 total records in the training partition and 202 records in the validation partition were split between nodes 2 (LeftChild ID) and 3 (Rightchild ID).

Node ID 2:

- In the training partition, 253 records were assigned to this node (from node 1) which has a “0” value (Response). These cases were split on the LSTAT variable using a value of 4.07: 249 records were assigned to node 5 and 4 records were assigned to node 4.
- In the Validation Partition, 155 records were assigned to this node (from node 1). These cases were split on the same variable (LSTAT) and value (4.07): 154 records were assigned to node 5 and 1 record was assigned to node 4.

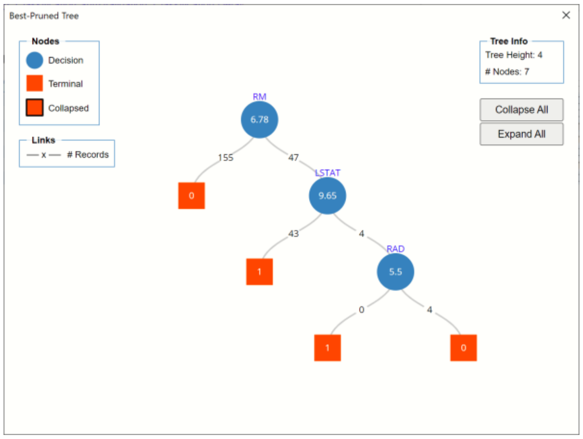
Node ID 4:

- In the training partition, 4 records, assigned from Node 2, were split on the CRIM variable using a value of 0.33. This node has a tentative classification of 1 (Response). Three records were assigned to node 9 and classified as 1. 1 record was assigned to node 8 and classified as a 0. Both nodes 8 and 9 are terminal nodes.
- In the validation partition, 1 node was assigned from Node 2. This record was assigned to terminal node 8 using the CRIM variable and a value of 0.33 and classified as 0.

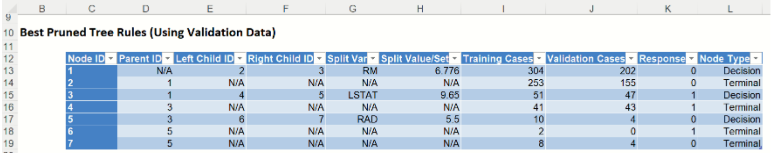
The table can be used to follow the tree all the way down to level 33.

CT_BestTree

Click the *CT_BestTree* tab to view the Best Pruned Tree and the Rules for the Best Pruned Tree.



The Best Pruned Tree is based on the validation data set, and is the smallest tree whose misclassification rate is within one standard error of the misclassification rate of the Minimum Error Tree



Best Pruned Tree Rules (Using Validation Data)

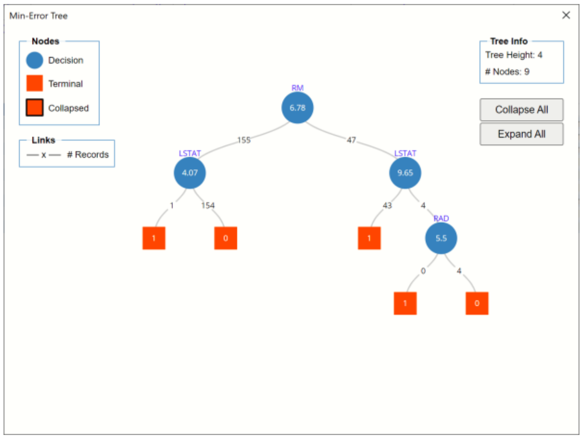
Node ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	RM	6.776	304	202	0	Decision
2	1	N/A	N/A	N/A	N/A	253	155	0	Terminal
3	1	4	5	LSTAT	9.65	51	47	1	Decision
4	3	N/A	N/A	N/A	N/A	41	43	1	Terminal
5	3	6	7	RAD	5.5	10	4	0	Decision
6	5	N/A	N/A	N/A	N/A	2	0	1	Terminal
7	5	N/A	N/A	N/A	N/A	8	4	0	Terminal

The Validation Partition records are split in the Tree according to the following rules:

- Node 1: 202 cases were split using the RM variable with a value of 6.78.
    - 155 records were assigned to Node 2, a terminal node, and classified as 0.
    - 47 records were assigned to Node 3, a decision node, and tentatively classified as 1.
- Node 3: 47 cases were split using the LSTAT variable with a value of 9.65.
    - 43 records were assigned to Node 4, a terminal node, and classified as 1.
    - 4 records were assigned to Node 5, a decision node, and tentatively classified as 0.
- Node 5: 4 records were split using the RAD variable with a value of 5.5.
    - All 4 records were assigned to node 7 and classified as 0.

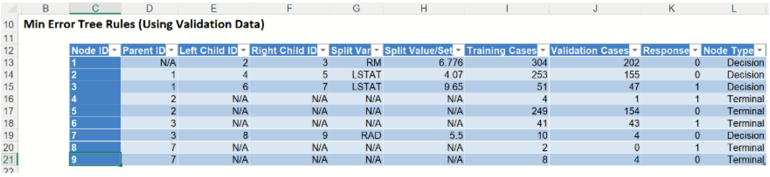
CT_MinErrorTree

Click **CT_MinErrorTree** to view the Minimum Error Tree.



Tree Info
Tree Height: 4
Nodes: 9

The "minimum error tree" is the tree that yields a minimum classification error rate when tested on the validation data. The misclassification (error) rate is measured as the tree is pruned. The tree that produces the lowest error rate is selected. The Min Error Tree Rules can also be found on the CT_MinErrorTree sheet.



Min Error Tree Rules (Using Validation Data)

Node ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	RM	6.776	304	202	0	Decision
2	1	4	5	LSTAT	4.07	253	155	0	Decision
3	1	6	7	LSTAT	9.65	51	47	1	Decision
4	2	N/A	N/A	N/A	N/A	4	1	1	Terminal
5	2	N/A	N/A	N/A	N/A	249	154	0	Terminal
6	3	N/A	N/A	N/A	N/A	41	43	1	Terminal
7	3	8	9	RAD	5.5	10	4	0	Decision
8	7	N/A	N/A	N/A	N/A	2	0	1	Terminal
9	7	N/A	N/A	N/A	N/A	8	4	0	Terminal

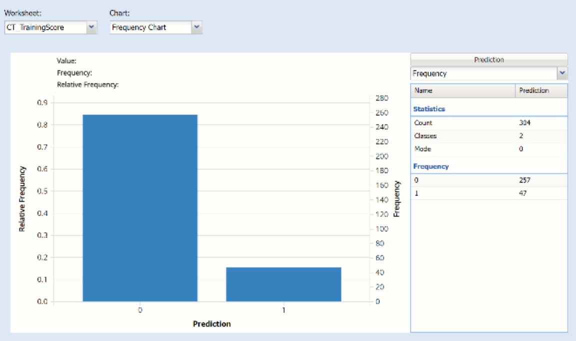
The Validation Partition records are split in the Min Error Tree according to the following rules:

- Node 1: 202 records were split using the RM variable with a value of 6.78.
  - 155 records were assigned to Node 2, a decision node, and tentatively classified as 0.
  - 47 records were assigned to Node 3, a decision node, and tentatively classified as 1.
- Node 2: 155 records were split using the LSTAT variable with a value of 4.07.
  - 1 record was assigned to Node 4, a terminal node, and classified as 1.
  - 154 records were assigned to Node 5, a terminal node, and classified as 0.
- Node 3: 47 records were split using the LSTAT variable (again) with a value of 9.65.
  - 43 records were assigned to node 6, a terminal node, and classified as 1.
  - 4 records were assigned to node 7, a decision node, and classified as 0.
- Node 7: 4 records were split using the RAD variable using a value of 5.5.
  - All 4 records were assigned to node 9, a terminal node, and classified as 0.

CT_TrainingScore

Click the *CT_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Classification Summary and the Training: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: CT_TrainingScore

Chart: Frequency Chart

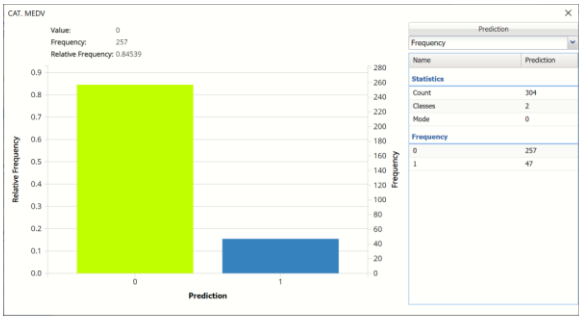
Value:
Frequency:
Relative Frequency:

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Frequency	
0	257
1	47

- **Frequency Charts:** The output variable frequency chart opens automatically once the *CT_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *CT_TrainingScore* tab. To move, click the title bar on the dialog and drag the chart to the desired location.

  Frequency: This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

  Frequency Chart on CT_TrainingScore output sheet



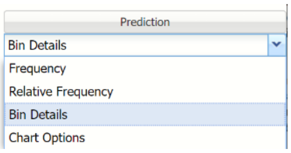
CAT. MEDV

Value: 0
Frequency: 257
Relative Frequency: 0.84539

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Frequency	
0	257
1	47

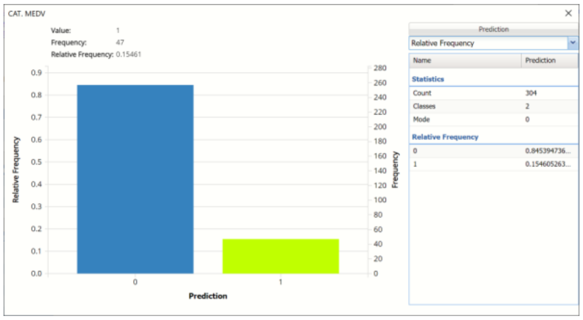
  Click the down arrow next to Frequency to switch to Relative Frequency, Bin Details or Chart Options view.

  Frequency Chart, Frequency View



  Relative Frequency: Displays the relative frequency chart.

Relative Frequency Chart



CAT. MEDV

Value: 1
Frequency: 47
Relative Frequency: 0.15461

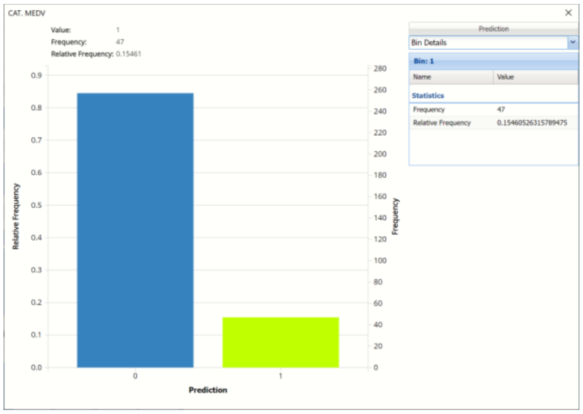
Prediction

Relative Frequency

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Relative Frequency	
0	0.845394736...
1	0.154605263...

Bin Details: Use this view to find metrics related to each bin in the chart.

Bin Details view



CAT. MEDV

Value: 1
Frequency: 47
Relative Frequency: 0.15461

Prediction

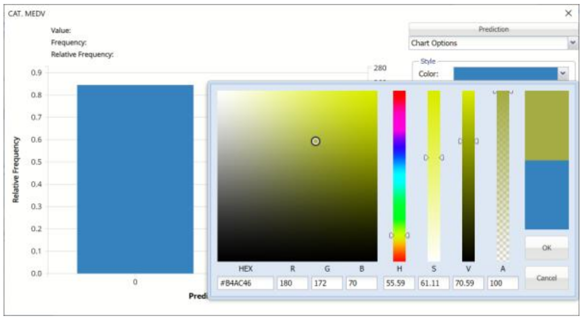
Bin Details

Bin: 1

Name	Value
Statistics	
Frequency	47
Relative Frequency	0.15460526315789475

Chart Options: Use this view to change the color of the bars in the chart.

Chart Options View



CAT. MEDV

Value:
Frequency:
Relative Frequency:

Prediction

Chart Options

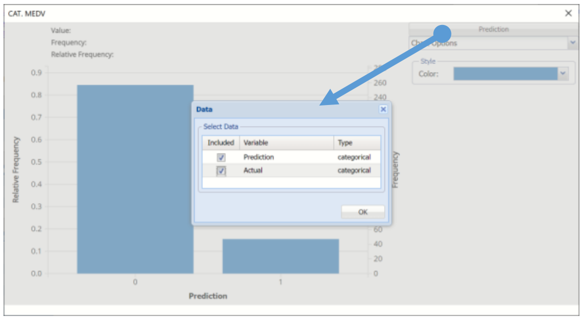
Style

Color:

HEX	R	G	B	H	S	V	A
#B4AC46	180	172	70	55.59	61.11	70.59	100

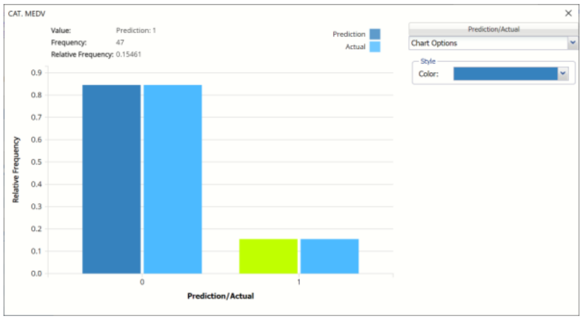
- To see both the actual and predicted frequency, click Prediction and select Actual. This change will be reflected on all charts.

Selecting Prediction/Actual



Included	Variable	Type
☑	Prediction	categorical
☑	Actual	categorical

Click Prediction/Actual to change view



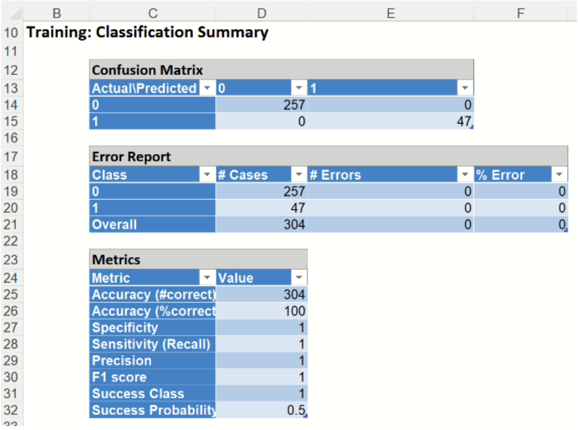
Value: Prediction: 1
Frequency: 47
Relative Frequency: 0.15461

- **Classification Summary:** In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

Confusion Matrix		
	Predicted Class	
Actual Class	1	0
1	TP	FN
0	FP	TN

  - TP stands for True Positive. These are the number of cases classified as belonging to the Success class that actually were members of the Success class.
  - FN stands for False Negative. These are the number of cases that were classified as belonging to the Failure class when they were actually members of the Success class
  - FP stands for False Positive. These cases were assigned to the Success class but were actually members of the Failure group
  - TN stands for True Negative. These cases were correctly assigned to the Failure group.

CT_TrainingScore: Training: Classification Summary



Training: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	257	0
1	0	47

Error Report

Class	# Cases	# Errors	% Error
0	257	0	0
1	47	0	0
Overall	304	0	0

Metrics

Metric	Value
Accuracy (#correct)	304
Accuracy (%correct	100
Specificity	1
Sensitivity (Recall)	1
Precision	1
F1 score	1
Success Class	1
Success Probability	0.5

- True Positive: All 47 records belonging to the Success class were correctly assigned to that class
- True Negative: All 257 records belonging to the Failure class were correctly assigned to this same class

The were no misclassified records. The metrics below reflect the accuracy of the algorithm.

Note: Scoring was performed using the Full Tree. Recall that the error rate for this tree was -1.56E-17 as shown on the CT_Output worksheet. This error rate will be converted to a percentage (1.56E-17*100) in the confusion matrix. In this instance, the % Error has been rounded to 0%.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 100% - Refers to the ability of the classifier to predict a class label correctly.
- Specificity: 1 - Also called the true negative rate, measures the percentage of failures correctly identified as failures

  Specificity (SPC) or True Negative Rate =TN / (FP + TN)
- Recall (or Sensitivity): 1 - Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

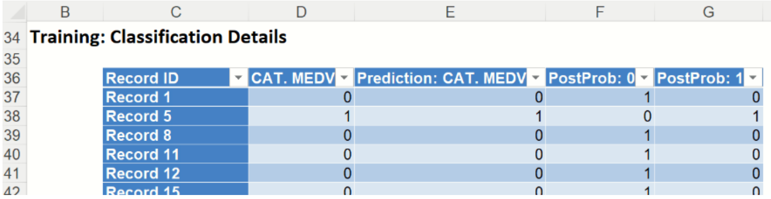
  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)
- Precision: 1 - The probability of correctly identifying a randomly selected record as one belonging to the Success class

  Precision = TP/(TP+FP)
- F-1 Score: 1 - Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP / (2 * TP + FP + FN)
- Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns.

  CT_TrainingScore: Training: Classification Details



Training: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 0	PostProb: 1
Record 1	0	0	1	0
Record 5	1	1	0	1
Record 8	0	0	1	0
Record 11	0	0	1	0
Record 12	0	0	1	0
Record 15	0	0	1	0

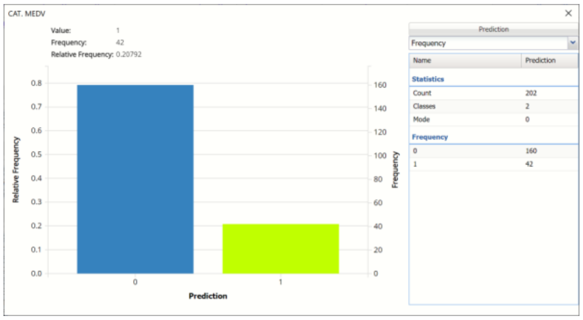
CT_ValidationScore

Click the *CT_ValidationScore* tab to view the newly added Output Variable frequency chart, the Validation: Classification Summary and the Validation: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Validation data.

- **Frequency Charts:** The output variable frequency chart opens automatically once the CT_ValidationScore worksheet is selected. To close this chart, click the "x" in the upper right hand corner. To reopen, click onto another tab and then click back to the CT_ValidationScore tab.

  Click the Frequency chart to display the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode. Selective Relative Frequency from the drop down menu, on the right, to see the relative frequencies of the output variable for both actual and predicted. See above for more information on this chart.

  CT_ValidationScore Frequency Chart



CAT. MEDV

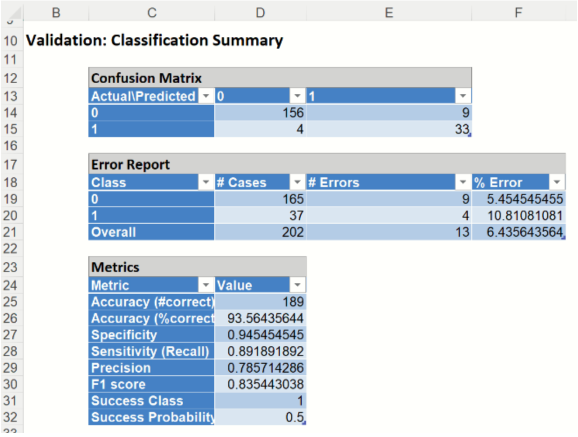
Value: 1
Frequency: 42
Relative Frequency: 0.20792

Prediction
Frequency

Name	Prediction
Statistics	
Count	202
Classes	2
Mode	0
Frequency	
0	160
1	42

- **Classification Summary:** This report contains the confusion matrix for the validation data set.

CT_ValidationScore: Classification Summary



Validation: Classification Summary

Confusion Matrix		
Actual\Predicted	0	1
0	156	9
1	4	33

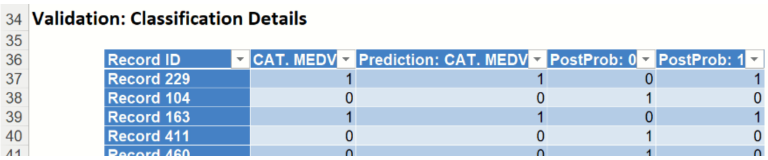
Error Report			
Class	# Cases	# Errors	% Error
0	165	9	5.454545455
1	37	4	10.81081081
Overall	202	13	6.435643564

Metrics	
Metric	Value
Accuracy (#correct)	189
Accuracy (%correct	93.56435644
Specificity	0.945454545
Sensitivity (Recall)	0.891891892
Precision	0.785714286
F1 score	0.835443038
Success Class	1
Success Probability	0.5

- True Positive: 33 records belonging to the Success class were correctly assigned to that class
- False Negative: 3 records belonging to the Success class were incorrectly assigned to the Failure class.
- True Negative: 156 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 9 records belonging to the Failure class were incorrectly assigned to the Success class.
- There were 13 cases misclassified in the validation dataset resulting in a % error of 6.44%.

- **Classification Details**: This table displays how each observation in the validation data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

CT_ValidationScore: Validation: Classification Details



Validation: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 0	PostProb: 1
Record 229	1	1	0	1
Record 104	0	0	1	0
Record 163	1	1	0	1
Record 411	0	0	1	0
Record 460	0	0	1	0

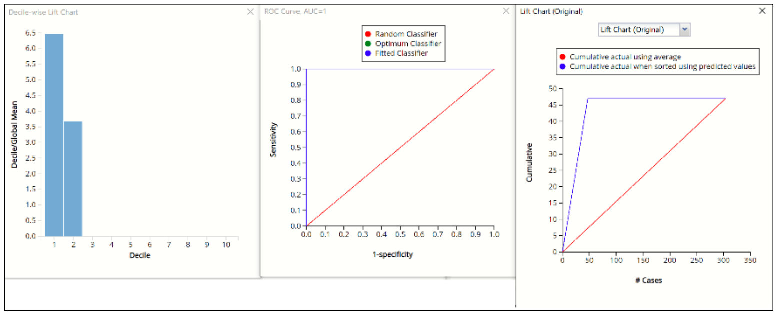
CT_TrainingLiftChart and CT_ValidationLiftChart

Click the CT_ValidationLiftChart tab to find the Lift Chart, ROC Curve, and Decile Chart for the Validation partition. Click the CT_TrainingLiftChart tab to display these same charts created using the training partition.

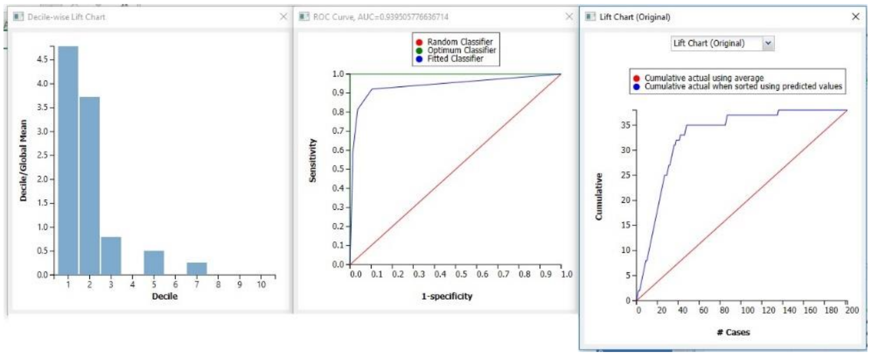
Lift Charts and ROC Curves are visual aids that help users evaluate the performance of their fitted models. Charts found on the CT_TrainingLiftChart tab were calculated using the Training Data Partition. Charts found on the CT_ValidationLiftChart tab were calculated using the Validation Data Partition. It is good practice to look at both sets of charts to assess model performance on both the Training and Validation partitions.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select CT_TrainingLiftChart or CT_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, ROC Curve, and Lift Charts for Training Partition



Decile-wise Lift Chart, ROC Curve, and Lift Charts for Valid Partition

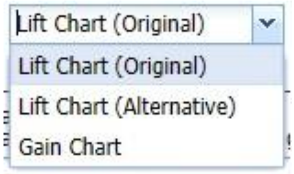


After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted in decreasing order using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the cumulative number of cases in decreasing probability (on the x-axis) vs the cumulative number of true positives on the y-axis. The baseline (red line connecting the origin to the end point of the blue line) is a reference line. For a given number of cases on the x-axis, this line represents the expected number of successes if no model existed, and instead cases were selected at random. This line can be used as a benchmark to measure the performance of the fitted model. The greater the area between the lift curve and the baseline, the better the model. In the Training Lift chart, if we selected 100 cases as belonging to the success class and used the fitted model to pick the members most likely to be successes, the lift curve tells us that we would be right on all of them. Conversely, if we selected 100 random cases, we could expect to be right on about 15 of them.

The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. This bars in this chart indicate the factor by which the model outperforms a random assignment, one decile at a time. Refer to the validation graph above. In the first decile, taking the most expensive predicted housing prices in the dataset, the predictive performance of the model is about 5 times better as simply assigning a random predicted value.

The Regression ROC curve was updated in V2017. This new chart compares the performance of the regressor (Fitted Classifier) with an Optimum Classifier Curve and a Random Classifier curve. The Optimum Classifier Curve plots a hypothetical model that would provide perfect classification results. The best possible classification performance is denoted by a point at the top left of the graph at the intersection of the x and y axis. This point is sometimes referred to as the “perfect classification”. The closer the AUC is to 1, the better the performance of the model. In the Validation Partition, AUC = .94 which suggests that this fitted model is a good fit to the data.

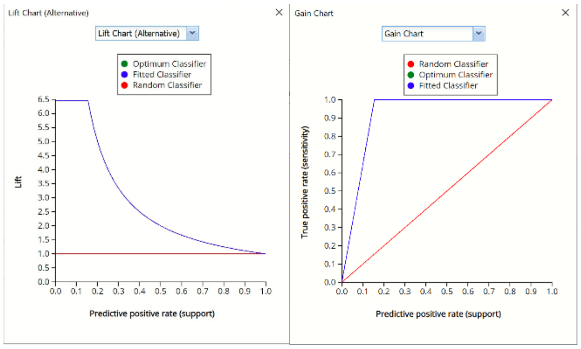
In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.



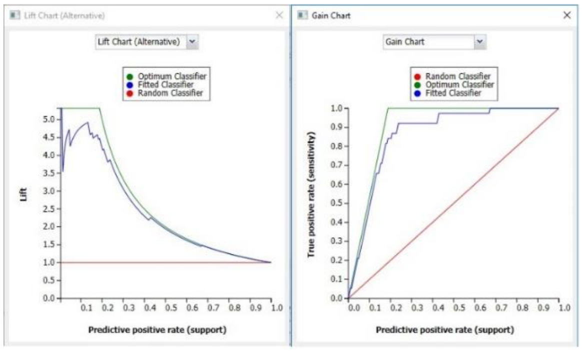
Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Classifier curve, a Fitted Classifier curve, and a Random Classifier curve. The Optimum Classifier curve plots a hypothetical model that would provide perfect classification for our data. The Fitted Classifier curve plots the fitted model and the Random Classifier curve plots the results from using no model or by using a random guess (i.e. for x% of selected observations, x% of the total number of positive observations are expected to be correctly classified).

The Alternative Lift Chart plots Lift against the Predictive Positive Rate or Support.

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition



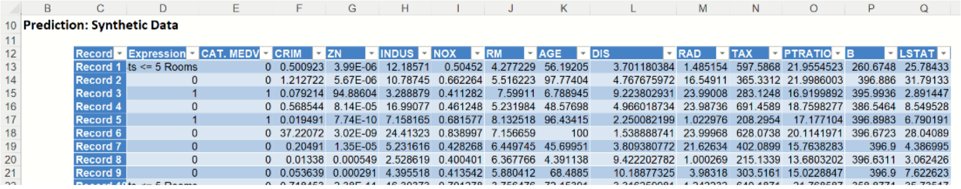
Click the down arrow and select Gain Chart from the menu. In this chart, the True Positive Rate or Sensitivity is plotted against the Predictive Positive Rate or Support.

CT_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, CT_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Classification Tree dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predicted values for the training partition (using the fitted model) and the Excel – calculated Expression results, if populated in the dialog. A chart is displayed with the option to switch between the Predicted Simulation and Training sources and the Expression results for the Simulation and Training data, or a combination of two as long as they are of the same type.

Synthetic Data



Prediction: Synthetic Data

Record	Expression	CAT. MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	ts <= 5 Rooms	0	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.701180384	1.485154	597.5868	21.9554523	260.6748	25.78433
Record 2	0	0	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767675972	16.54911	365.3312	21.9986003	396.886	31.79133
Record 3	1	1	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223802931	23.99008	283.1248	16.9199892	395.9936	2.891447
Record 4	0	0	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966018734	23.98736	691.4589	18.7598277	386.5464	8.549528
Record 5	1	1	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082199	1.022976	208.2954	17.177104	396.8983	6.790191
Record 6	0	0	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538888741	23.99968	628.0738	20.1141971	396.6723	28.04089
Record 7	0	0	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809380772	21.62634	402.0899	15.7638283	396.9	4.386995
Record 8	0	0	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422202782	1.000269	215.1339	13.6803202	396.6311	3.062426
Record 9	0	0	0.053639	0.000291	4.395518	0.413542	5.880412	68.4885	10.18877325	3.98318	303.5161	15.0228847	396.9	7.622623

Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the Discriminant Analysis dialog

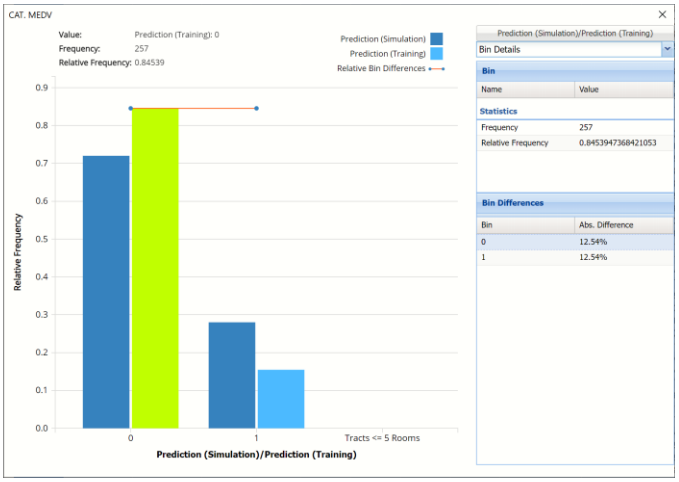
IF[@RM]>=5,[@CAT.MEDV],"Tracts < 5 Rooms")

The results in this column are either 0, 1, or Tracts <= 5 Rooms.

The remainder of the data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training partition and the synthetic data. In the screenshot below, the bars in the darker shade of blue are based on the synthetic data. The bars in the lighter shade of blue are based on the predictions for the training partition.   In the synthetic data, a little over 70% of the census tracts are predicted to have a classification equal to 0, or low median value, while almost 30% of census tracts are predicted to have a classification equal to 1, or high median value.

Frequency Chart for CT_Simulation output



CAT. MEDV

Value: Prediction (Training): 0

Frequency: 257

Relative Frequency: 0.84539

Prediction (Simulation)/Prediction (Training)

Bin Details

Bin

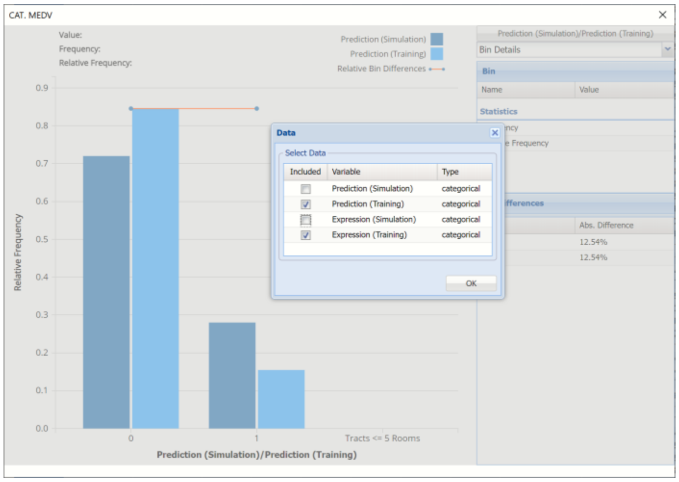
Name	Value
Statistics	
Frequency	257
Relative Frequency	0.8453947368421053

Bin Differences

Bin	Abs. Difference
0	12.54%
1	12.54%

Click *Prediction (Simulation) / Prediction (Training)* to change the chart view to *Prediction (Training)/Expression (Training).*

Data dialog



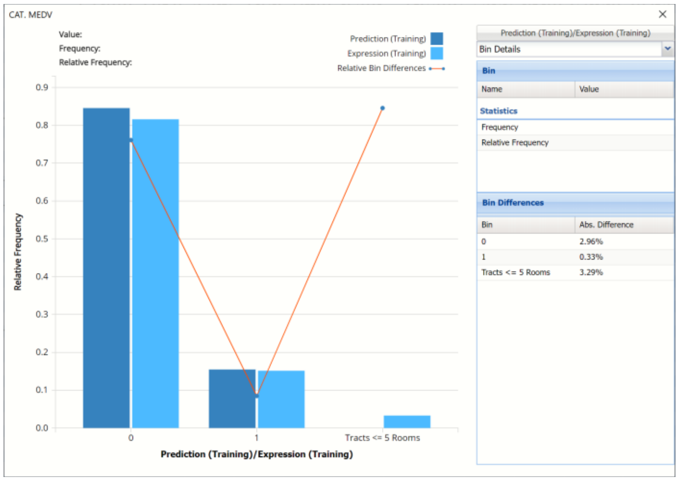
Data

Select Data

Included	Variable	Type
□	Prediction (Simulation)	categorical
☑	Prediction (Training)	categorical
□	Expression (Simulation)	categorical
☑	Expression (Training)	categorical

OK

Frequency Chart Prediction (Training) / Expression (Training)



CAT. MEDV

Value:

Frequency:

Relative Frequency:

Prediction (Training)/Expression (Training)

Bin Details

Bin

Name	Value
Statistics	
Frequency	
Relative Frequency	

Bin Differences

Bin	Abs. Difference
0	2.96%
1	0.33%
Tracts <= 5 Rooms	3.29%

The chart above reports the predictions from the training partition and compares them to the results of the expression evaluated on the training partition. About 83% of the records in the training partition were classified as CAT. MEDV = 0. About 15% of the records in the training partition were classified as CAT. MEDV = 1. The Expression columns (the lighter blue columns) report the number of records that were classified as 0 or 1 where RM > 5. This chart clearly shows that a small number of rooms in the training partition have values for RM less than or equal to 5.

Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

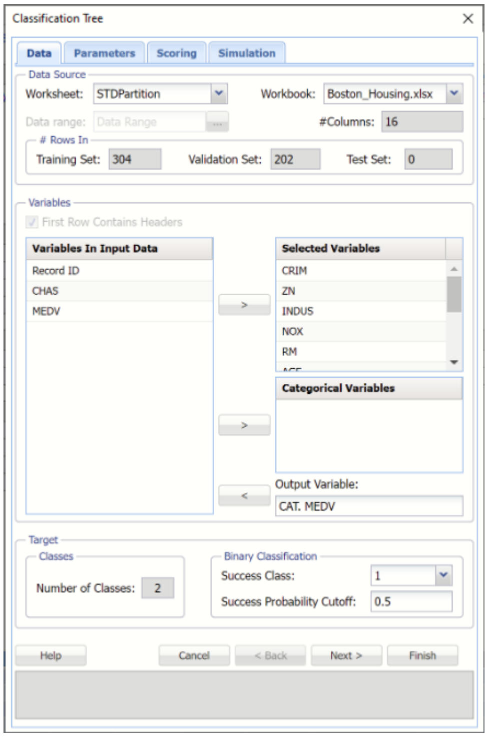
For information on Stored Model Sheets, in this example *DA_Stored*, please refer to the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide.

Classification Tree Options

The following options appear on one of the three *Classification Tree* dialogs.

Classification Tree dialog, Data tab

Classification Tree dialog, Data tab



Classification Tree

Data Source

Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Rows In

Training Set: 304

Validation Set: 202

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM

Categorical Variables

Output Variable: CAT. MEDV

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Variables In Input Data

The variables included in the dataset appear here.

Selected Variables

Variables selected to be included in the output appear here.

Output Variable

The dependent variable or the variable to be classified appears here.

Categorical Variables

Place categorical variables from the Variables listbox to be included in the model by clicking the > command button. This classification algorithm will accept non-numeric categorical variables.

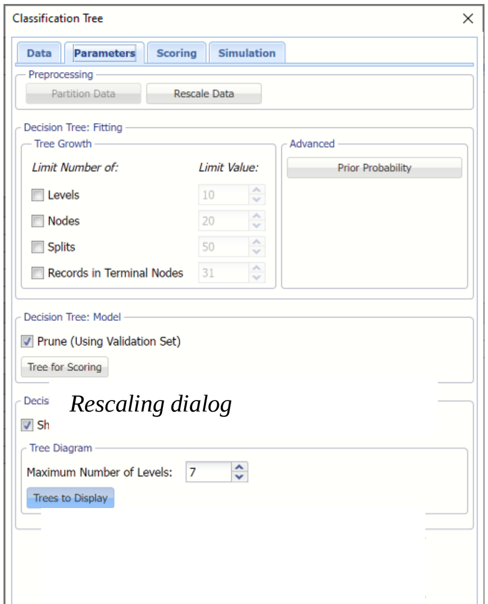
Number of Classes

Displays the number of classes in the Output variable.

Success Class

This option is selected by default. Select the class to be considered a “success” or the significant class in the Lift Chart. This option is enabled when the number of classes in the output variable is equal to 2.

Classification Tree dialog, Parameters tab


Classification Tree

Preprocessing: Partition Data, Rescale Data

Decision Tree: Fitting
Tree Growth

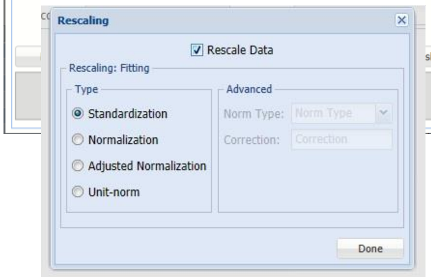
Limit Number of:	Limit Value:
□ Levels	10
□ Nodes	20
□ Splits	50
□ Records in Terminal Nodes	31

Advanced: Prior Probability

Decision Tree: Model
☑ Prune (Using Validation Set)
Tree for Scoring

☑ Sh
Tree Diagram
Maximum Number of Levels: 7
Trees to Display

Rescaling dialog


Rescaling
☑ Rescale Data
Rescaling: Fitting
Type
● Standardization
○ Normalization
○ Adjusted Normalization
○ Unit-norm
Advanced
Norm Type: Norm Type
Correction: Correction

Success Probability Cutoff

Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a “success” (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a “non-success” (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

Classification Tree dialog, Parameters tab

Partition Data

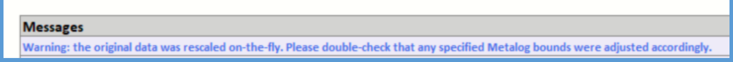
Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Rescale Data

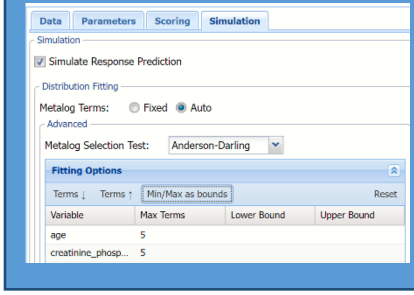
Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Notes on Rescaling and the Simulation functionality

If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot to the left, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the CT_Simulation output sheet, as shown below.


Messages
Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



Simulation
☑ Simulate Response Prediction
Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Tree Growth

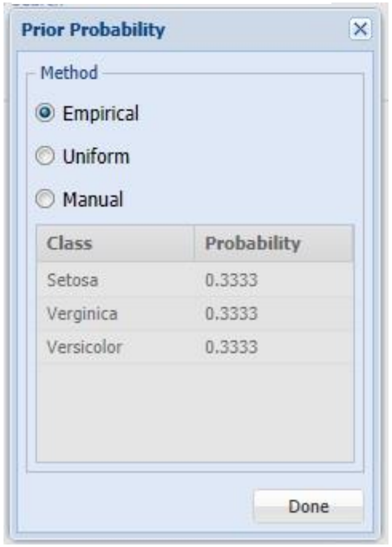
In the *Tree Growth* section, select Levels, Nodes, Splits, and Records in Terminal Nodes. Values entered for these options limit tree growth, i.e. if 10 is entered for Levels, the tree will be limited to 10 levels.

Prior Probability

Three options appear in the *Prior Probability* Dialog: *Empirical*, *Uniform* and *Manual.*

- If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
- If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.
- Select the third option, *Manual*, to manually enter the desired class and probability value.

Prior Probability dialog



Prior Probability

Method

● Empirical

○ Uniform

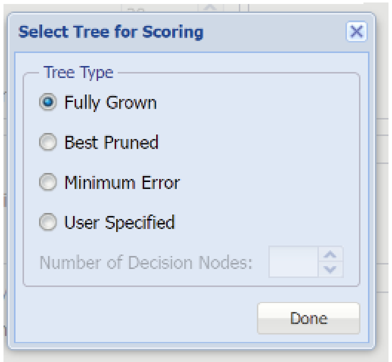
○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

Prune (Using Validation Set)

If a validation partition exists, this option is enabled. When this option is selected, Analytic Solver Data Science will prune the tree using the validation set. Pruning the tree using the validation set reduces the error from over-fitting the tree to the training data.

Select Tree for Scoring



Select Tree for Scoring

Tree Type

● Fully Grown

○ Best Pruned

○ Minimum Error

○ User Specified

Number of Decision Nodes:

Show Feature Importance

Select *Feature Importance* to include the *Features Importance* table in the output. This table displays the variables that are included in the model along with their Importance value.

Maximum Number of Levels

This option specifies the maximum number of levels in the tree to be displayed in the output.

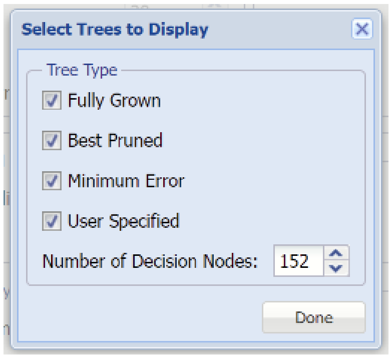
Note: If a tree is limited to X levels in the output (intentionally or due to a limited Analytic Solver license), Analytic Solver will draw the first X levels of the diagram.

Trees to Display

Select *Trees to Display* to select the types of trees to display: Fully Grown, Best Pruned, Minimum Error or User Specified.

- Select *Fully Grown* to “grow” a complete tree using the training data.
- Select *Best Pruned* to create a tree with the fewest number of nodes, subject to the constraint that the error be kept below a specified level (minimum error rate plus the standard error of that error rate).

Select Trees to Display



Select Trees to Display

Tree Type

☑ Fully Grown

☑ Best Pruned

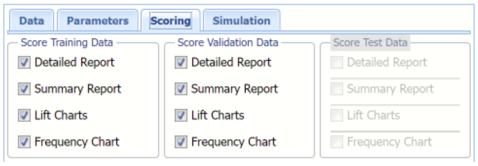
☑ Minimum Error

☑ User Specified

Number of Decision Nodes: 152

- Select *Minimum error* to produce a tree that yields the minimum classification error rate when tested on the validation data.
- To create a tree with a specified number of decision nodes select *User Specified* and enter the desired number of nodes.

Classification Tree dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Classification Tree dialog, Scoring tab

Score Training Data

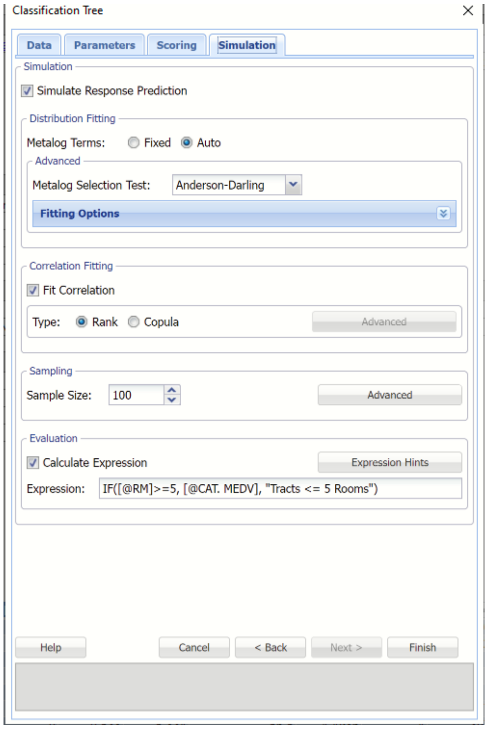
Select these options to show an assessment of the performance of the Classification Tree algorithm in classifying the training data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories.

When Frequency Chart is selected, a frequency chart will be displayed when the CT_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Classification Tree algorithm in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the CT_ValidationScore worksheet is selected.

Classification Tree dialog, Simulation tab



Classification Tree

Data | Parameters | Scoring | Simulation

Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula

Sampling
Sample Size: 100

Evaluation
☑ Calculate Expression
Expression: IF([@RM]>=5, [@CAT. MEDV], "Tracts <= 5 Rooms")

Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Classification Tree algorithm in classifying the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the CT_TestScore worksheet is selected.

Score New Data

See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Classification Tree dialog, Simulation tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CT_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the CT_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Naïve Bayes Classification Method

Introduction

Suppose your data consists of fruits, described by their color and shape. Bayesian classifiers operate by saying "If you see a fruit that is red and round, which type of fruit is it most likely to be? In the future, classify red and round fruit as that type of fruit."

A difficulty arises when you have more than a few variables and classes – an enormous number of observations (records) would be required to estimate these probabilities.

The Naive Bayes classification method avoids this problem by not requiring a large number of observations for each possible combination of the variables. Rather, the variables are assumed to be independent of one another and, therefore the probability that a fruit that is red, round, firm, 3" in diameter, etc. will be an apple can be calculated from the independent probabilities that a fruit is red, that it is round, that it is firm, that it is 3" in diameter, etc.

In other words, Naïve Bayes classifiers assume that the effect of a variable value on a given class is independent of the values of other variables. This assumption is called class conditional independence and is made to simplify the computation. In this sense, it is considered to be “Naïve”.

This assumption is a fairly strong assumption and is often not applicable. However, bias in estimating probabilities often may not make a difference in practice -- it is the order of the probabilities, not their exact values, which determine the classifications.

Studies comparing classification algorithms have found the Naïve Bayesian classifier to be comparable in performance with classification tree and neural network classifiers. It has also been found that these classifiers exhibit high accuracy and speed when applied to large databases.

A more technical description of the Naïve Bayesian classification method follows.

Bayes Theorem

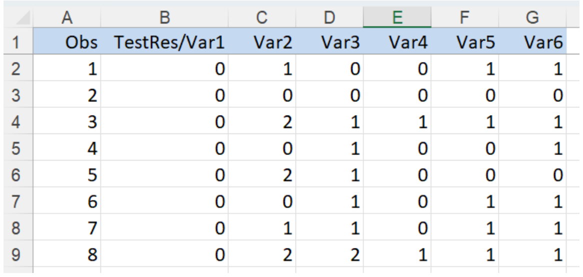
Let X be the data record (case) whose class label is unknown. Let H be some hypothesis, such as "data record X belongs to a specified class C." For classification, we want to determine P (H|X) -- the probability that the hypothesis H holds, given the observed data record X.

P (H|X) is the posterior probability of H conditioned on X. For example, the probability that a fruit is an apple, given the condition that it is red and round. In contrast, P(H) is the prior probability, or apriori probability, of H. In this example P(H) is the probability that any given data record is an apple, regardless of how the data record looks. The posterior probability, P (H|X), is based on more information (such as background knowledge) than the prior probability, P(H), which is independent of X.

Similarly, P (X|H) is posterior probability of X conditioned on H. That is, it is the probability that X is red and round given that we know that it is true that X is an apple. P(X) is the prior probability of X, i.e., it is the probability that a data record from our set of fruits is red and round. Bayes theorem is useful in that it provides a way of calculating the posterior probability, P(H|X), from P(H), P(X), and P(X|H). Bayes theorem can be written as: P (H|X) = P(X|H) P(H) / P(X).

Naïve Bayes Classification Example

The following example illustrates Analytic Solver Data Science's Naïve Bayes classification method. Click **Help – Example Models** on the Data Science ribbon, then **Forecasting/Data Science Examples** to open the **Flying_Fitness.xlsx** example dataset. A portion of the dataset appears below.

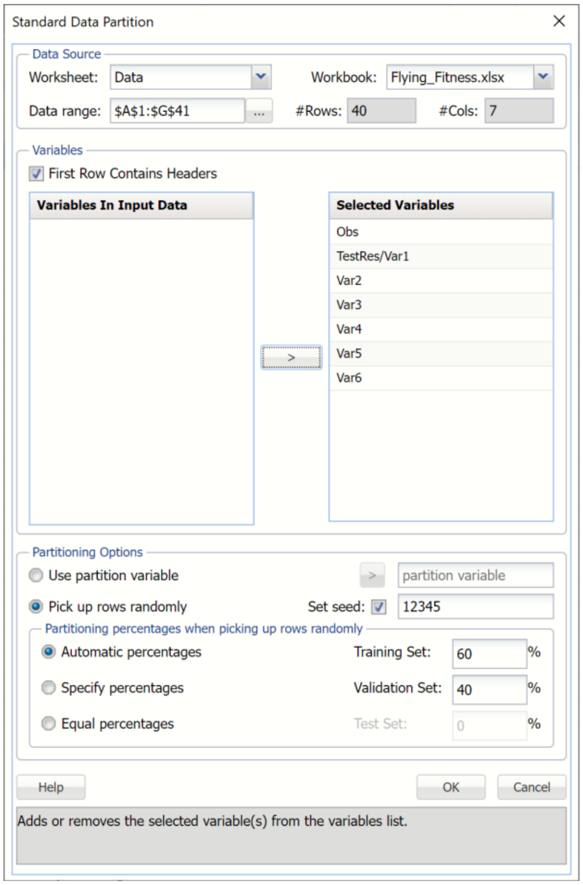


Obs	TestRes/Var1	Var2	Var3	Var4	Var5	Var6
1	0	1	0	0	1	1
2	0	0	0	0	0	0
3	0	2	1	1	1	1
4	0	0	1	0	0	1
5	0	2	1	0	0	0
6	0	0	1	0	1	1
7	0	1	1	0	1	1
8	0	2	2	1	1	1

In this example, we will classify pilots on whether they are fit to fly based on various physical and psychological tests. The output variable, TestRes/Var1 equals 1 if the pilot is fit and 0 if not.

First, we partition the data into training and validation sets using the Standard Data Partition defaults of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.

Standard Data Partition dialog



Standard Data Partition

Data Source

Worksheet: Data

Workbook: Flying_Fitness.xlsx

Data range: $A$1:$G$41

#Rows: 40

#Cols: 7

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	Obs
	TestRes/Var1
	Var2
	Var3
	Var4
	Var5
	Var6

Partitioning Options

○ Use partition variable

● Pick up rows randomly

Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages

○ Specify percentages

○ Equal percentages

Training Set: 60 %

Validation Set: 40 %

Test Set: 0 %

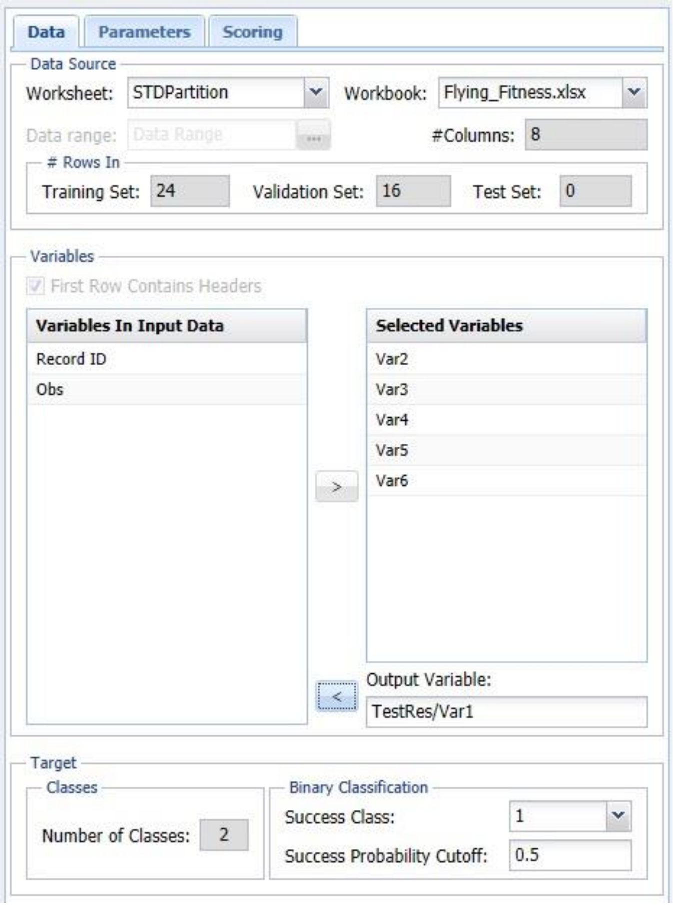
Adds or removes the selected variable(s) from the variables list.

Click **Classify – Naïve Bayes**. The following *Naïve Bayes* dialog appears.

Select **Var2**, **Var3**, **Var4**, **Var5**, and **Var6** as *Selected Variables* and **TestRest/Var1** as the *Output Variable*. The *Number of Classes* statistic will be automatically updated with a value of *2* when the Output Variable is selected. This indicates that the Output variable, *TestRest/Var1*, contains two classes, 0 and 1.

Choose the value that will be the indicator of "Success" by clicking the down arrow next to *Success Class*. In this example, we will use the default of 1 indicating that a value of "1" will be specified as a "success".

Enter a value between 0 and 1 for *Success Probability Cutoff*. If the Probability of success (probability of the output variable = 1) is less than this value, then a 0 will be entered for the class value, otherwise a 1 will be entered for the class value. In this example, we will keep the default of 0.5.



Data Source

Worksheet: STDPartition   Workbook: Flying_Fitness.xlsx

Data range: Data Range   #Columns: 8

# Rows In

Training Set: 24   Validation Set: 16   Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
Record ID	Var2
Obs	Var3
	Var4
	Var5
	Var6

Output Variable: TestRes/Var1

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Click **Next** to advance to the *Naïve Bayes – Parameters* tab.

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

On the *Parameters* tab, click **Prior Probability** to calculate the *Prior class probabilities*. When this option is selected, Analytic Solver Data Science will calculate the class probabilities from the training data . For the first class, Analytic Solver Data Science will calculate the probability using the number of “0” records / total number of points. For the second class, Analytic Solver Data Science will calculate the probability using the number of “1” records / total number of points.

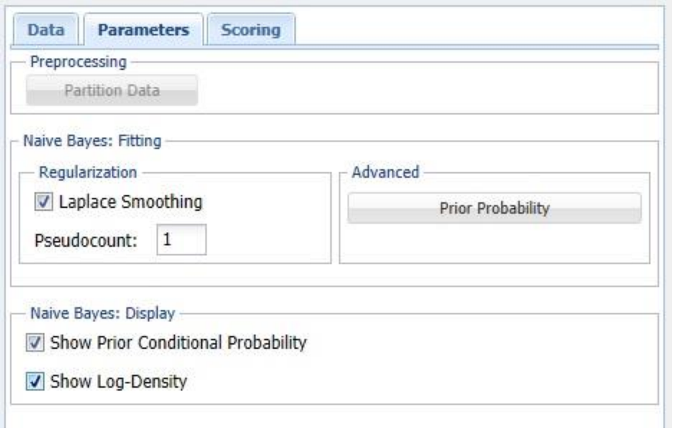
- If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.
- If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.
- Select the third option, *Manual*, to manually enter the desired class and probability value.

Click Done to accept the default setting, Empirical, and close the dialog.

In Analytic Solver, users have the ability to use Laplace smoothing, or not, during model creation. In this example, leave Laplace Smoothing and Pseudocount at their defaults.

If a particular realization of some feature never occurs in a given class in the training partition, then the corresponding frequency-based prior conditional probability estimate will be zero. For example, assume that you have trained a model to classify emails using the Naïve Bayes Classifier with 2 classes: work and personal. Assume that the model rates one email as having a high probability of belonging to the "personal" class. Now assume that there is a 2nd email that is *the same* as the previous email, but this email includes *one word* that is different. Now, if this one word was not present in any of the "personal" emails in the training partition, the estimated probability would be zero. Consequently, the resulting product of all probabilities will be zero, leading to a loss of all the strong evidence of this email to belong to a "personal" class. To mitigate this problem, Analytic Solver Data Science allows you to specify a small correction value, known as a pseudocount, so that no probability estimate is ever set to 0. Normalizing the Naïve Bayes classifier in this way is called Laplace smoothing. Pseudocount set to zero is equivalent to no smoothing. There are arguments in the literature which support a pseudocount value of 1, although in practice fractional values are often used. When Laplace Smoothing is selected, Analytic Solver Data Science will accept any positive value for pseudocount.

Under *Naïve Bayes: Display*, select **Show Prior Conditional Probability** and **Show Log-Density** to add both in the output.



☑ Laplace Smoothing

Pseudocount: 1

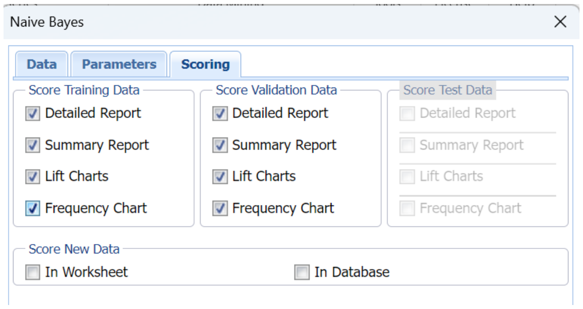
☑ Show Prior Conditional Probability

☑ Show Log-Density

Click **Next** to advance to the Scoring tab.

Select **Detailed report, Lift Charts** and **Frequency Chart**, under both *Score training data* and *Score validation data*. **Summary report** under both *Score Training Data* and *Score Validation Data* are selected by default. These settings will allow us to obtain the complete output results for this classification method. Since we did not create a test partition, the options for *Score test data* are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

For more information on the options for *Score new data*, please see the chapters "Scoring New Data" and "Scoring Test Data" within the Analytic Solver Data Science User Guide.



Naive Bayes

Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

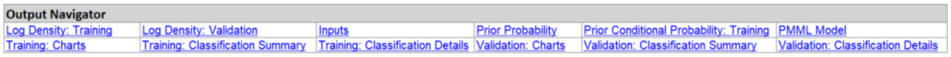
Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

Click **Finish** to generate the output. Results are inserted to the right.

Click *NB_Output* to display the Output Navigator. Click any link to navigate to the selected topic.

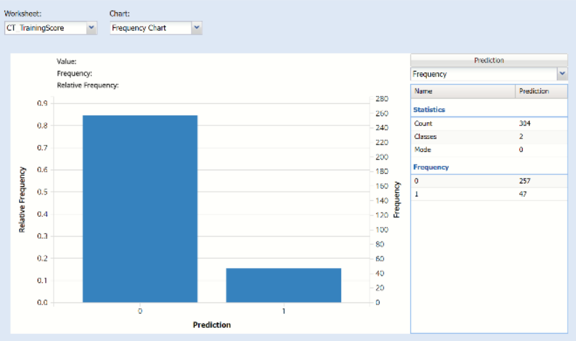


Output Navigator					
Log Density: Training	Log Density: Validation	Inputs	Prior Probability	Prior Conditional Probability: Training	PMML Model
Training: Charts	Training: Classification Summary	Training: Classification Details	Validation: Charts	Validation: Classification Summary	Validation: Classification Details

NB_TrainingScore

Click Training: Classification Details in the Output Navigator top open the NB_TrainingScore output worksheet. Immediately, the Output Variable frequency chart appears. The worksheet contains the Training: Classification Summary and the Training: Classification Details reports. All calculations, charts and predictions on this worksheet apply to the Training data.

> Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: CT_TrainingScore

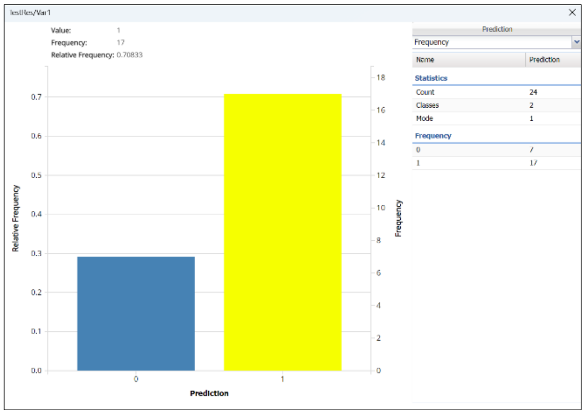
Chart: Frequency Chart

Name	Prediction
Statistics	
Count	304
Classes	2
Mode	0
Frequency	
0	257
1	47

- **Frequency Charts:** The output variable frequency chart opens automatically once the *NB_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *NB_TrainingScore* tab. To move, click the title bar on the dialog and drag the chart to the desired location.

  Frequency: This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

Frequency Chart on NB_TrainingScore output sheet



Value: 1
Frequency: 17
Relative Frequency: 0.70833

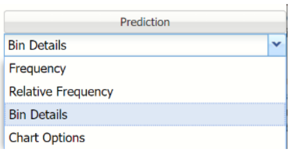
Prediction

Frequency

Name	Prediction
Statistics	
Count	24
Classes	2
Mode	1
Frequency	
0	7
1	17

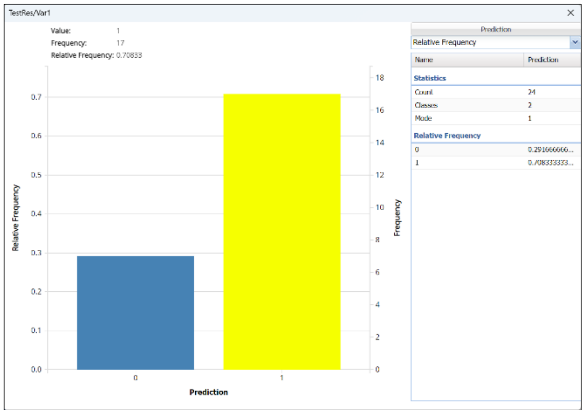
Click the down arrow next to Frequency to switch to Relative Frequency, Bin Details or Chart Options view.

Frequency Chart, Frequency View



Relative Frequency: Displays the relative frequency chart.

Relative Frequency Chart



Value: 1
Frequency: 17
Relative Frequency: 0.70833

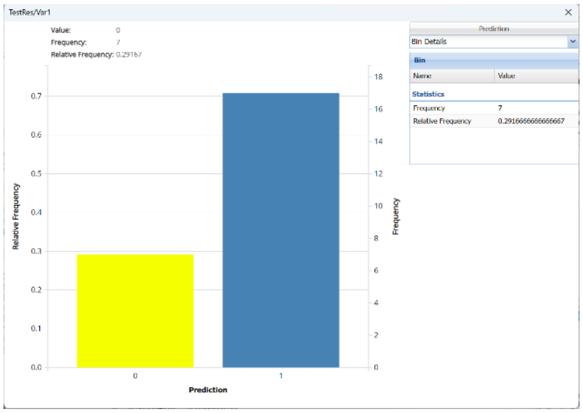
Prediction

Relative Frequency

Name	Prediction
Statistics	
Count	24
Classes	2
Mode	1
Relative Frequency	
0	0.291666666...
1	0.708333333...

Bin Details: Use this view to find metrics related to each bin in the chart.

Bin Details view



Value: 0
Frequency: 7
Relative Frequency: 0.29167

Bin

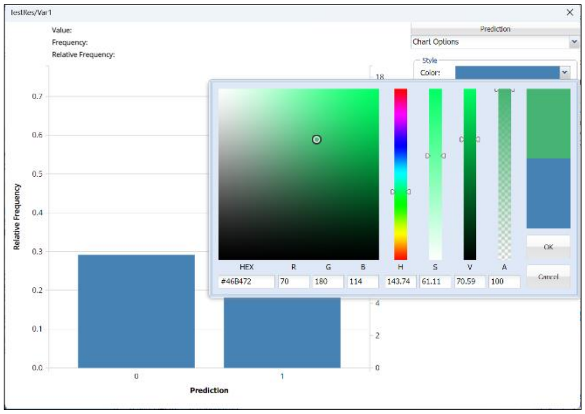
Name	Value

Statistics

Frequency	7
Relative Frequency	0.2916666666666667

Chart Options: Use this view to change the color of the bars in the chart.

Chart Options View



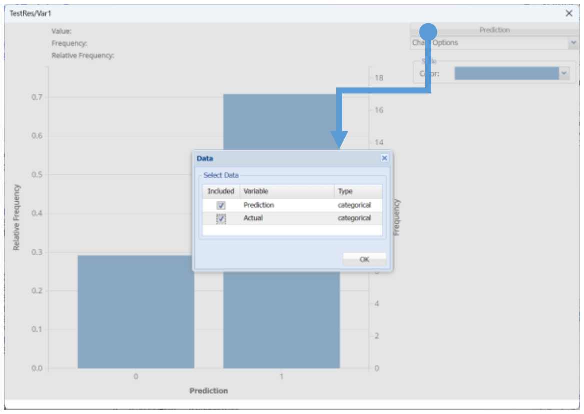
Value:
Frequency:
Relative Frequency:

Style
Color:

HEX	R	G	B	H	S	V	A
#46B472	70	180	114	143.74	61.11	70.59	100

- To see both the actual and predicted frequency, click Prediction and select Actual. This change will be reflected on all charts.

  Selecting Prediction/Actual



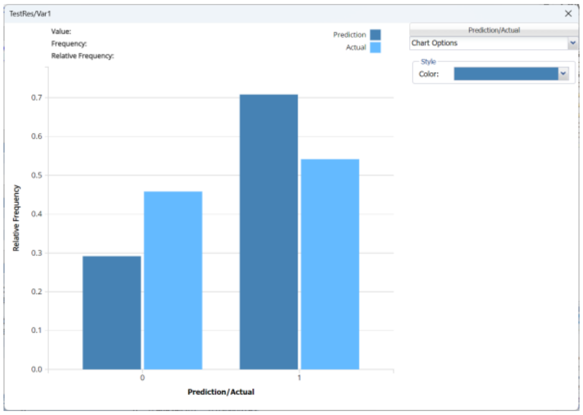
Value:
Frequency:
Relative Frequency:

Data

Select Data

Included	Variable	Type
☑	Prediction	categorical
☑	Actual	categorical

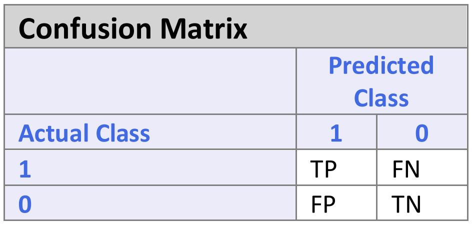
Click Prediction/Actual to change view



- **Classification Summary:** In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

  Recall that in this example, we are classifying pilots on whether they are fit to fly based on various physical and psychological tests. Our output variable, TestRes/Var1 is 1 if the pilot is fit and 0 if not.

  A Confusion Matrix is used to evaluate the performance of a classification method. This matrix summarizes the records that were classified correctly and those that were not.



Confusion Matrix		
	Predicted Class	
Actual Class	1	0
1	TP	FN
0	FP	TN

TP stands for True Positive. These are the number of cases classified as belonging to the Success class that actually were members of the Success class. FN stands for False Negative. These are the number of cases that were classified as belonging to the Failure class when they were actually members of the Success class (i.e. patients with cancerous tumors who were told their tumors were benign). FP stands for False Positive. These cases were assigned to the Success class but were actually members of the Failure group (i.e. patients who were told they tested postive for cancer when, in fact, their tumors were benign). TN stands for True Negative. These cases were correctly assigned to the Failure group.

Precision is the probability of correctly identifying a randomly selected record as one belonging to the Success class (i.e. the probability of correctly identifying a random patient with cancer as having cancer). Recall (or Sensitivity) measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people with cancer who are correctly identified as having cancer). Specificity (also called the true negative rate) measures the percentage of failures correctly identified as failures (i.e. the proportion of people with no cancer being categorized as not having cancer.) The F-1 score, which fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

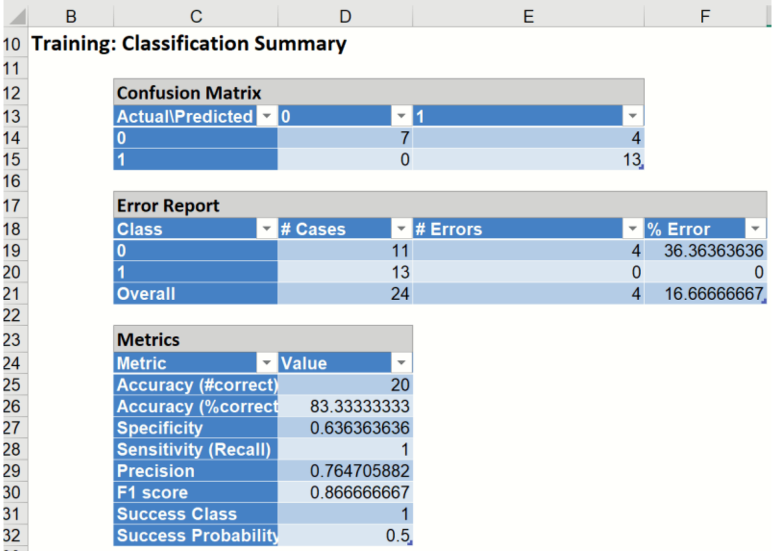
Precision = TP/(TP+FP)

Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)

Specificity (SPC) or True Negative Rate =TN / (FP + TN)

F1 = (2 * TP) /(2TP + FP + FN))

Click the *Training: Classification Summary* and *Validation: Classification Summary* links to view the Classification Summaries for both partitions.



Training: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	7	4
1	0	13

Error Report

Class	# Cases	# Errors	% Error
0	11	4	36.36363636
1	13	0	0
Overall	24	4	16.66666667

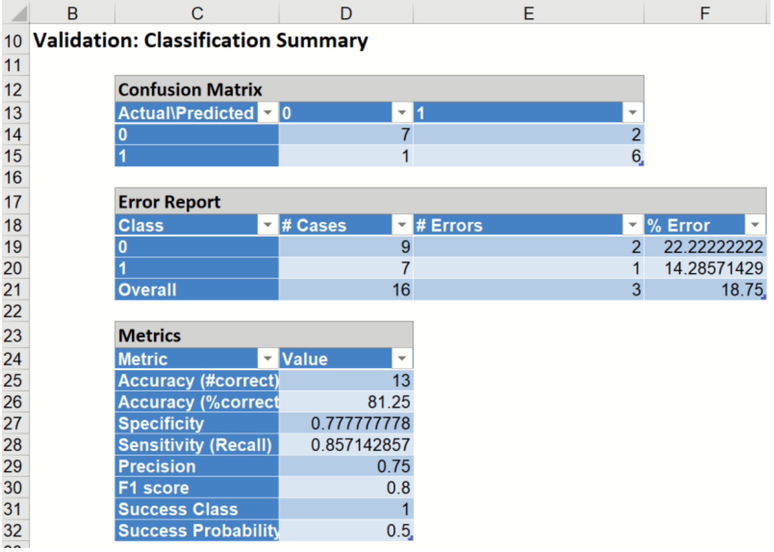
Metrics

Metric	Value
Accuracy (#correct)	20
Accuracy (%correct	83.33333333
Specificity	0.636363636
Sensitivity (Recall)	1
Precision	0.764705882
F1 score	0.866666667
Success Class	1
Success Probability	0.5

In the Training Dataset, we see 4 records were misclassified giving a misclassification error of 16.67%. Four (4) records were misclassified as successes.

NB_ValidationScore

Click the link for Validation: Classification Summary in the Output Navigator to open the Classification Summary for the validation partition.



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	7	2
1	1	6

Error Report

Class	# Cases	# Errors	% Error
0	9	2	22.22222222
1	7	1	14.28571429
Overall	16	3	18.75

Metrics

Metric	Value
Accuracy (#correct)	13
Accuracy (%correct	81.25
Specificity	0.777777778
Sensitivity (Recall)	0.857142857
Precision	0.75
F1 score	0.8
Success Class	1
Success Probability	0.5

However, in the Validation Dataset, 7 records were correctly classified as belonging to the Success class while 1 case was incorrectly assigned to the Failure class. Six (6) cases were correctly classified as belonging to the Failure class while 2 records were incorrectly classified as belonging to the Success class. This resulted in a total classification error of 18.75%.

While predicting the class for the output variable, Analytic Solver Data Science calculates the conditional probability that the variable may be classified to a particular class. In this example, the classes are 0 and 1. For every record in each partition, the conditional probabilities for class - 0 and for class - 1 are calculated. Analytic Solver Data Science assigns the class to the output variable for which the conditional probability is the largest. Misclassified records will be highlighted in red.

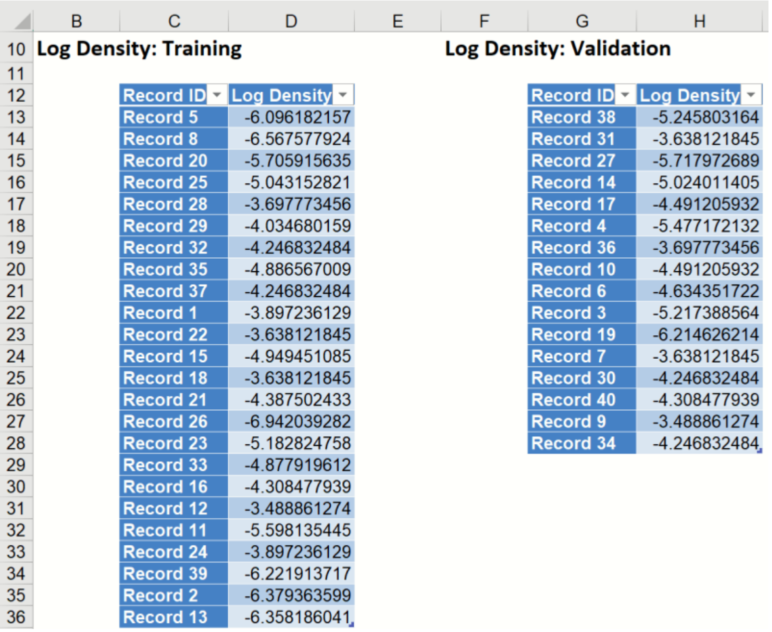
It's possible that a N/A "error" may be displayed in the Classification table. These appear when the Naïve Bayes classifier is unable to classify specific patterns because they have not been seen in the training dataset. Rows of such partitions with unseen values are considered to be outliers. When N/A's are present, Lift charts will not be available for that dataset.

NB_LogDensity

Click the NB_LogDensity tab to view the Log Densities for each partition. Log PDF, or Logarithm of Unconditional Probability Density, is the distribution of the predictors marginalized over the classes and is computed using:

$$\log[P\{X_1,\dots,X_n\}] = \log\left[\sum_{c=1}^{C} P\{X_1,\dots,X_n,Y=c\}\right] = \log\left[\sum_{c=1}^{C} \pi\{Y=c\}\,P\{X_1,\dots,X_n|Y=c\}\right]$$

where$\pi\{Y = c\}$ is a prior class probability



Log Density: Training

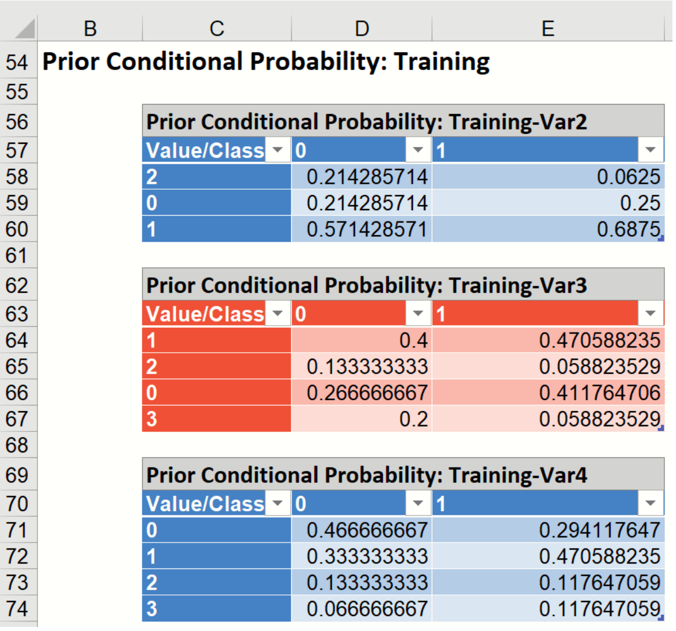
Record ID	Log Density
Record 5	-6.096182157
Record 8	-6.567577924
Record 20	-5.705915635
Record 25	-5.043152821
Record 28	-3.697773456
Record 29	-4.034680159
Record 32	-4.246832484
Record 35	-4.886567009
Record 37	-4.246832484
Record 1	-3.897236129
Record 22	-3.638121845
Record 15	-4.949451085
Record 18	-3.638121845
Record 21	-4.387502433
Record 26	-6.942039282
Record 23	-5.182824758
Record 33	-4.877919612
Record 16	-4.308477939
Record 12	-3.488861274
Record 11	-5.598135445
Record 24	-3.897236129
Record 39	-6.221913717
Record 2	-6.379363599
Record 13	-6.358186041

Log Density: Validation

Record ID	Log Density
Record 38	-5.245803164
Record 31	-3.638121845
Record 27	-5.717972689
Record 14	-5.024011405
Record 17	-4.491205932
Record 4	-5.477172132
Record 36	-3.697773456
Record 10	-4.491205932
Record 6	-4.634351722
Record 3	-5.217388564
Record 19	-6.214626214
Record 7	-3.638121845
Record 30	-4.246832484
Record 40	-4.308477939
Record 9	-3.488861274
Record 34	-4.246832484

NB_Output

Click the **Prior Conditional Probability: Training** link to display the table below. This table shows the probabilities for each case by variable. For example, for Var2, 21% of the records where Var2 = 0 were assigned to Class 0, 57% of the records where Var2 = 1 were assigned to Class 0 and 21% of the records where Var2 = 2 were assigned to Class 0.



Prior Conditional Probability: Training

Prior Conditional Probability: Training-Var2

Value/Class	0	1
2	0.214285714	0.0625
0	0.214285714	0.25
1	0.571428571	0.6875

Prior Conditional Probability: Training-Var3

Value/Class	0	1
1	0.4	0.470588235
2	0.133333333	0.058823529
0	0.266666667	0.411764706
3	0.2	0.058823529

Prior Conditional Probability: Training-Var4

Value/Class	0	1
0	0.466666667	0.294117647
1	0.333333333	0.470588235
2	0.133333333	0.117647059
3	0.066666667	0.117647059

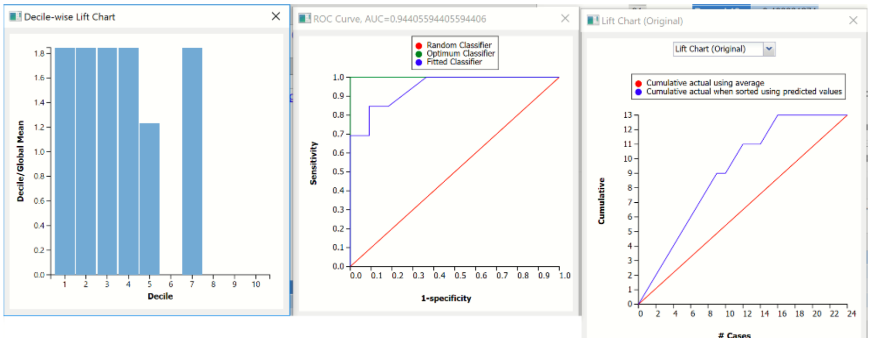
NB_TrainingLiftChart and NB_ValidationLiftChart

Click the NB_TrainingLiftChart and NB_ValidationLiftChart tabs to find the Lift Chart, ROC Curve, and Decile Chart for both the Training and Validation partitions.

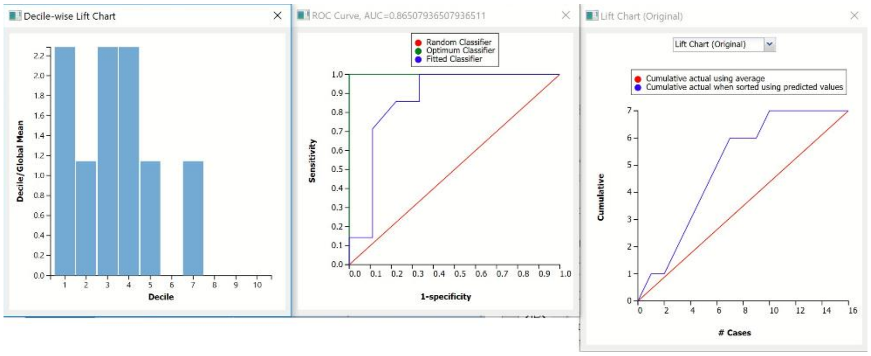
Lift Charts and ROC Curves are visual aids that help users evaluate the performance of their fitted models. Charts found on the NB_TrainingLiftChart tab were calculated using the Training Data Partition. Charts found on the NB_ValidationLiftChart tab were calculated using the Validation Data Partition. It is good practice to look at both sets of charts to assess model performance on both the Training and Validation partitions.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select CT_TrainingLiftChart or CT_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, ROC Curve, and Lift Charts for Training Partition



Decile-wise Lift Chart, ROC Curve, and Lift Charts for Valid. Partition

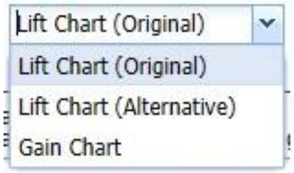


After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted in decreasing order using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the cumulative number of cases in decreasing probability (on the x-axis) vs the cumulative number of true positives on the y-axis. The baseline (red line connecting the origin to the end point of the blue line) is a reference line. For a given number of cases on the x-axis, this line represents the expected number of successes if no model existed, and instead cases were selected at random. This line can be used as a benchmark to measure the performance of the fitted model. The greater the area between the lift curve and the baseline, the better the model. In the Training Lift chart, if we selected 10 cases as belonging to the success class and used the fitted model to pick the members most likely to be successes, the lift curve tells us that we would be right on about 9 of them. Conversely, if we selected 10 random cases, we could expect to be right on about 4 of them. The Validation Lift chart tells us that we could expect to see the Random model perform the same or better on the validation partition than our fitted model.

The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. This bars in this chart indicate the factor by which the model outperforms a random assignment, one decile at a time. Refer to the validation graph above. In the first decile, the predictive performance of the model is about 1.8 times better as simply assigning a random predicted value.

The Regression ROC curve was updated in V2017. This new chart compares the performance of the regressor (Fitted Classifier) with an Optimum Classifier Curve and a Random Classifier curve. The Optimum Classifier Curve plots a hypothetical model that would provide perfect classification results. The best possible classification performance is denoted by a point at the top left of the graph at the intersection of the x and y axis. This point is sometimes referred to as the “perfect classification”. The closer the AUC is to 1, the better the performance of the model. In the Validation Partition, AUC = .43 which suggests that this fitted model is not a good fit to the data.

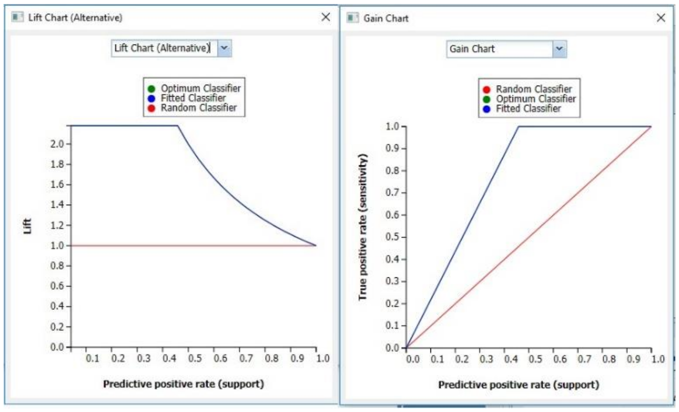
In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.



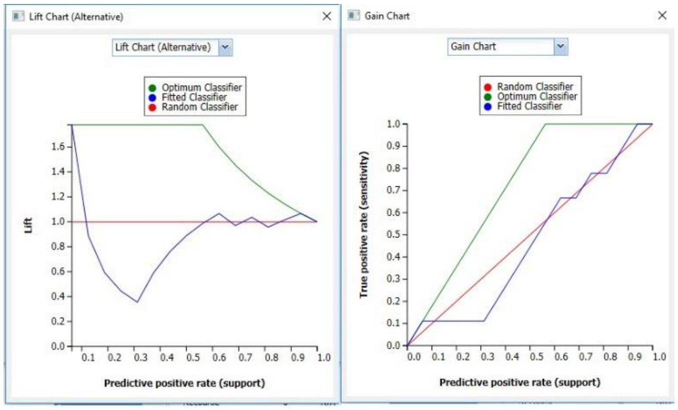
Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Classifier curve, a Fitted Classifier curve, and a Random Classifier curve. The Optimum Classifier curve plots a hypothetical model that would provide perfect classification for our data. The Fitted Classifier curve plots the fitted model and the Random Classifier curve plots the results from using no model or by using a random guess (i.e. for x% of selected observations, x% of the total number of positive observations are expected to be correctly classified).

The Alternative Lift Chart plots Lift against the Predictive Positive Rate or Support.

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition

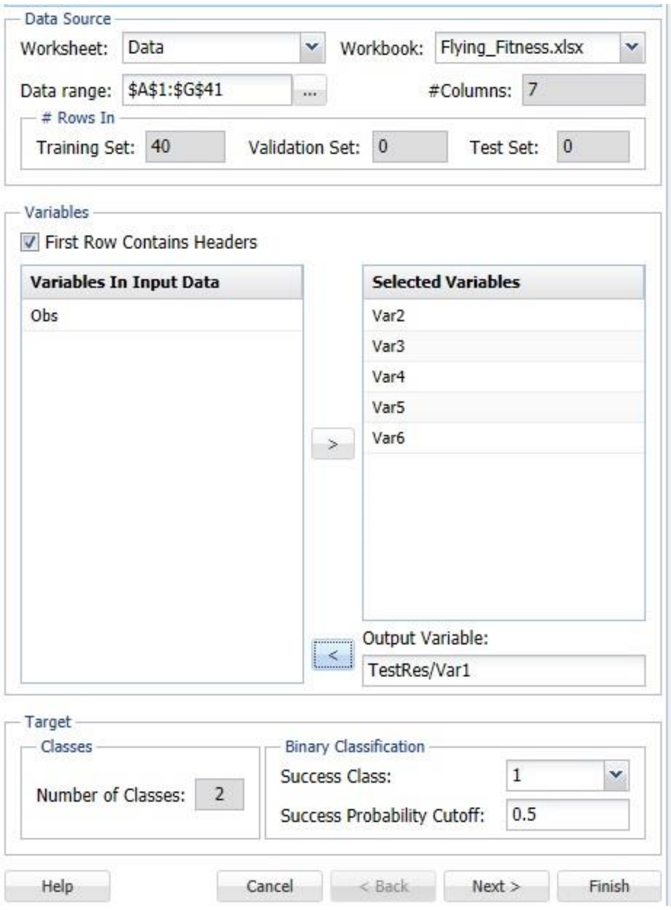


Click the down arrow and select Gain Chart from the menu. In this chart, the True Positive Rate or Sensitivity is plotted against the Predictive Positive Rate or Support.

Please see the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide for information on *NB_Stored*.

Naïve Bayes Classification Method Options

The options below appear on one of the three *Naïve Bayes* classification methods’ dialogs.



Data Source

Worksheet: Data

Workbook: Flying_Fitness.xlsx

Data range: $A$1:$G$41

#Columns: 7

Rows In

Training Set: 40

Validation Set: 0

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Obs

Selected Variables
Var2
Var3
Var4
Var5
Var6

Output Variable: TestRes/Var1

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Variables in input data

The variables included in the dataset appear here.

Selected Variables

Variables selected to be included in the output appear here.

Output Variable

The dependent variable or the variable to be classified appears here.

Number of Classes

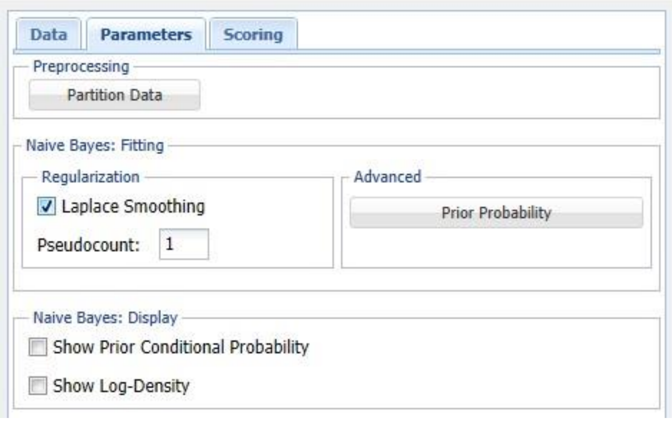
Displays the number of classes in the Output variable.

Success Class

This option is selected by default. Select the class to be considered a “success” or the significant class in the Lift Chart. This option is enabled when the number of classes in the output variable is equal to 2.

Success Probability Cutoff

Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a “success” (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a “non-success” (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.



☑ Laplace Smoothing

Pseudocount: 1

□ Show Prior Conditional Probability

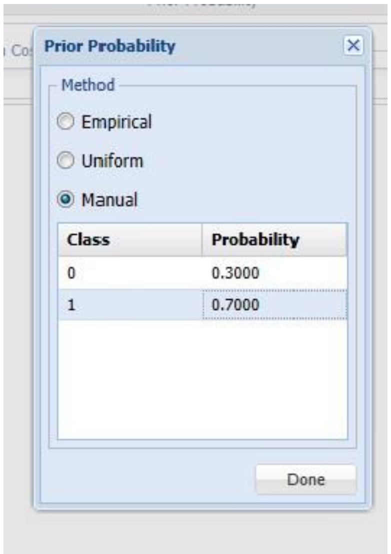
□ Show Log-Density

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Prior Probability

Click **Prior Probability**. Three options appear in the *Prior Probability* Dialog: *Empirical, Uniform* and *Manual.*



Method

○ Empirical

○ Uniform

● Manual

Class	Probability
0	0.3000
1	0.7000

If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.

If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.

Select the third option, *Manual*, to manually enter the desired class and probability value.

Laplace Smoothing

If a particular realization of some feature never occurs in a given class in the training partition, then the corresponding frequency-based prior conditional probability estimate will be zero. For example, assume that you have trained a model to classify emails using the Naïve Bayes Classifier with 2 classes: work and personal. Assume that the model rates one email as having a high probability of belonging to the "personal" class. Now assume that there is a 2nd email that is *the same* as the previous email, but this email includes *one word* that is different. Now, if this one word was not present in any of the “personal” emails in the training partition, the estimated probability would be zero. Consequently, the resulting product of all probabilities will be zero, leading to a loss of all the strong evidence of this email to belong to a “personal” class. To mitigate this problem, Analytic Solver Data Science allows you to specify a small correction value, known as a pseudocount, so that no probability estimate is ever set to 0. Normalizing the Naïve Bayes classifier in this way is called Laplace smoothing. Pseudocount set to zero is equivalent to no smoothing. There are arguments in the literature which support a pseudocount value of 1, although in practice, fractional values are often used. When Laplace Smoothing is selected, Analytic Solver Data Science will accept any positive value for pseudocount.

Show Prior Conditional Probability

Select this option to print Prior Conditional Probability for the training partition in the output.

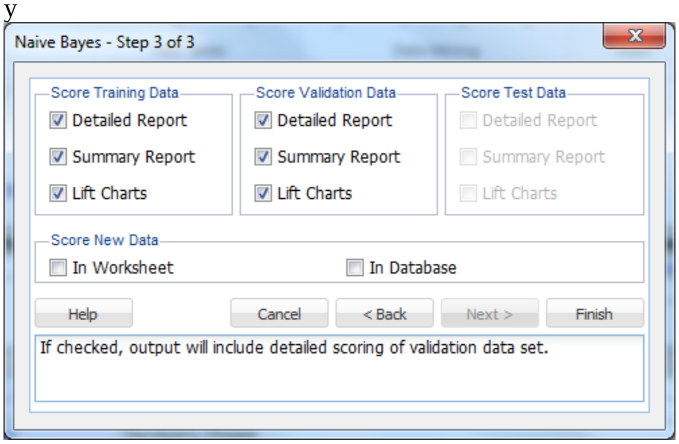
Show Log-Density

Select this option to print the Log-Density values for each partition in the output. Log PDF, or Logarithm of Unconditional Probability Density, is the distribution of the predictors marginalized over the classes and is computed using:

$$\log[P\{X_1, \dots, X_n\}] = \log\left[\sum_{c=1}^{C} P\{X_1, \dots, X_n, Y = c\}\right] = \log\left[\sum_{c=1}^{C} \pi\{Y = c\}\, P\{X_1, \dots, X_n | Y = c\}\right]$$

where$\pi\{Y = c\}$ is a prior class probability

y



Naive Bayes - Step 3 of 3

Score Training Data
- ☑ Detailed Report
- ☑ Summary Report
- ☑ Lift Charts

Score Validation Data
- ☑ Detailed Report
- ☑ Summary Report
- ☑ Lift Charts

Score Test Data
- □ Detailed Report
- □ Summary Report
- □ Lift Charts

Score New Data
- □ In Worksheet
- □ In Database

If checked, output will include detailed scoring of validation data set.

Score Training Data

Select these options to show an assessment of the performance of the tree in classifying the training data. The report is displayed according to your specifications - Detailed, Summary and Lift Charts. Lift charts are only available when the output variable has 2 classes.

Score Validation Data

These options are enabled when a validation dataset is present. Select these options to show an assessment of the performance of the tree in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary and Lift Charts. Lift charts are only available when the output variable has 2 classes.

Score Test Data

These options are enabled when a test dataset is present. Select these options to show an assessment of the performance of the tree in classifying the test data. The report is displayed according to your specifications - Detailed, Summary and Lift Charts. Lift charts are only available when the output variable has 2 classes.

Score New Data

Please see the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide for information on the *Score New Data* options.

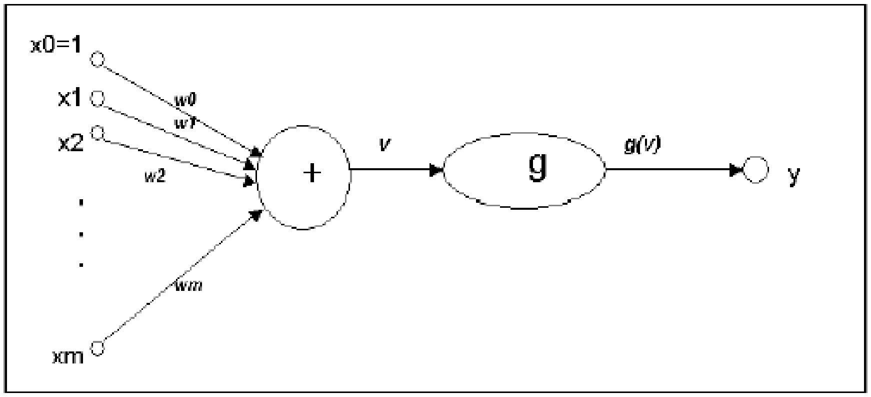
Neural Network Classification Method

Introduction

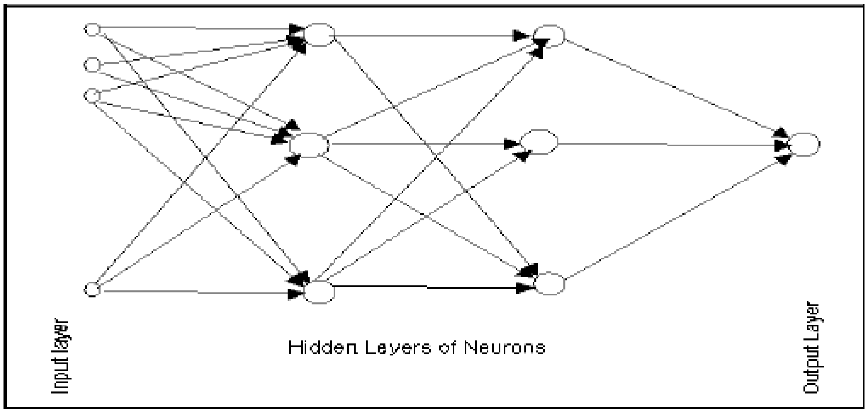
Artificial neural networks are relatively crude electronic networks of "neurons" based on the neural structure of the brain. They process records one at a time, and "learn" by comparing their classification of the record (which, at the outset, is largely arbitrary) with the known actual classification of the record. The errors from the initial classification of the first record is fed back into the network, and used to modify the networks algorithm the second time around, and so on for many iterations.

Roughly speaking, a neuron in an artificial neural network is

1. A set of input values (xi) and associated weights (wi)
2. A function (g) that sums the weights and maps the results to an output (y).



Neurons are organized into layers: input, hidden and output. The input layer is composed not of full neurons, but rather consists simply of the record's values that are inputs to the next layer of neurons. The next layer is the hidden layer. Several hidden layers can exist in one neural network. The final layer is the output layer, where there is one node for each class. A single sweep forward through the network results in the assignment of a value to each output node, and the record is assigned to the class node with the highest value.



Training an Artificial Neural Network

In the training phase, the correct class for each record is known (this is termed supervised training), and the output nodes can therefore be assigned "correct" values -- "1" for the node corresponding to the correct class, and "0" for the others. (In practice, better results have been found using values of “0.9” and “0.1”, respectively.) It is thus possible to compare the network's calculated values for the output nodes to these "correct" values, and calculate an error term for each node (the "Delta" rule). These error terms are then used to adjust the weights in the hidden layers so that, hopefully, during the next iteration the output values will be closer to the "correct" values.

The Iterative Learning Process

A key feature of neural networks is an iterative learning process in which records (rows) are presented to the network one at a time, and the weights associated with the input values are adjusted each time. After all cases are presented, the process is often repeated. During this learning phase, the network “trains” by adjusting the weights to predict the correct class label of input samples. Advantages of neural networks include their high tolerance to noisy data, as well as their ability to classify patterns on which they have not been trained. The most popular neural network algorithm is the back-propagation algorithm proposed in the 1980's.

Once a network has been structured for a particular application, that network is ready to be trained. To start this process, the initial weights (described in the next section) are chosen randomly. Then the training, or learning, begins.

The network processes the records in the training data one at a time, using the weights and functions in the hidden layers, then compares the resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights for application to the next record. This process occurs over and over as the weights are continually tweaked. During the training of a network the same set of data is processed many times as the connection weights are continually refined.

Note that some networks never learn. This could be because the input data does not contain the specific information from which the desired output is derived. Networks also will not converge if there is not enough data to enable complete learning. Ideally, there should be enough data available to create a validation set.

Feedforward, Back-Propagation

The feedforward, back-propagation architecture was developed in the early 1970's by several independent sources (Werbor; Parker; Rumelhart, Hinton and Williams). This independent co-development was the result of a proliferation of articles and talks at various conferences which stimulated the entire industry. Currently, this synergistically developed back-propagation architecture is the most popular, effective, and easy-to-learn model for complex, multi-layered networks. Its greatest strength is in non-linear solutions to ill-defined problems.

The typical back-propagation network has an input layer, an output layer, and at least one hidden layer. There is no theoretical limit on the number of hidden layers but typically there are just one or two. Some studies have shown that the total number of layers needed to solve problems of any complexity is 5 (one input layer, three hidden layers and an output layer). Each layer is fully connected to the succeeding layer.

As noted above, the training process normally uses some variant of the Delta Rule, which starts with the calculated difference between the actual outputs and the desired outputs. Using this error, connection weights are increased in proportion to the error times, which are a scaling factor for global accuracy. This means that the inputs, the output, and the desired output all must be present at the same processing element. The most complex part of this algorithm is determining which input contributed the most to an incorrect output and how must the input be modified to correct the error. (An inactive node would not contribute to the error and would have no need to change its weights.) To solve this problem, training inputs are applied to the input layer of the network, and desired outputs are compared at the output layer. During the learning process, a forward sweep is made through the network, and the output of each element is computed layer by layer. The difference between the output of the final layer and the desired output is back-propagated to the previous layer(s), usually modified by the derivative of the transfer function. The connection weights are normally adjusted using the Delta Rule. This process proceeds for the previous layer(s) until the input layer is reached.

Structuring the Network

The number of layers and the number of processing elements per layer are important decisions. These parameters, to a feedforward, back-propagation topology, are also the most ethereal **-** they are the "art" of the network designer. There is no quantifiable, best answer to the layout of the network for any particular application. There are only general rules picked up over time and followed by most researchers and engineers applying this architecture to their problems.

Rule One: As the complexity in the relationship between the input data and the desired output increases, the number of the processing elements in the hidden layer should also increase.

Rule Two: If the process being modeled is separable into multiple stages, then additional hidden layer(s) may be required. If the process is not separable into stages, then additional layers may simply enable memorization of the training set, and not a true general solution.

Rule Three: The amount of training data available sets an upper bound for the number of processing elements in the hidden layer(s). To calculate this upper bound, use the number of cases in the training data set and divide that number by the sum of the number of nodes in the input and output layers in the network.

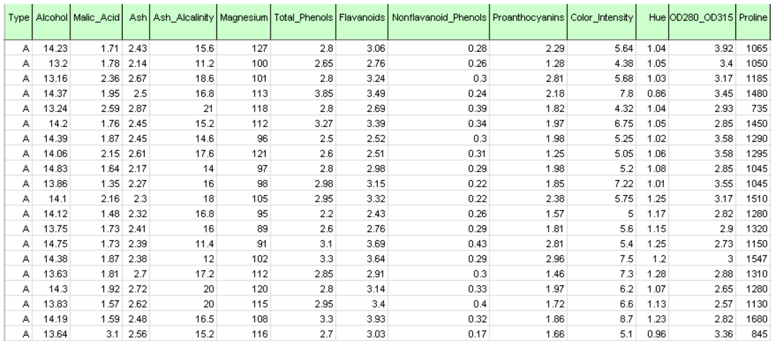
Then divide that result again by a scaling factor between five and ten. Larger scaling factors are used for relatively less noisy data. If too many artificial neurons are used the training set will be memorized, not generalized, and the network will be useless on new data sets.

Automated Neural Network Classification Example

This example focuses on creating a Neural Network using an Automated network architecture. See the section below for an example on creating a Neural Network using a Manual Architecture.

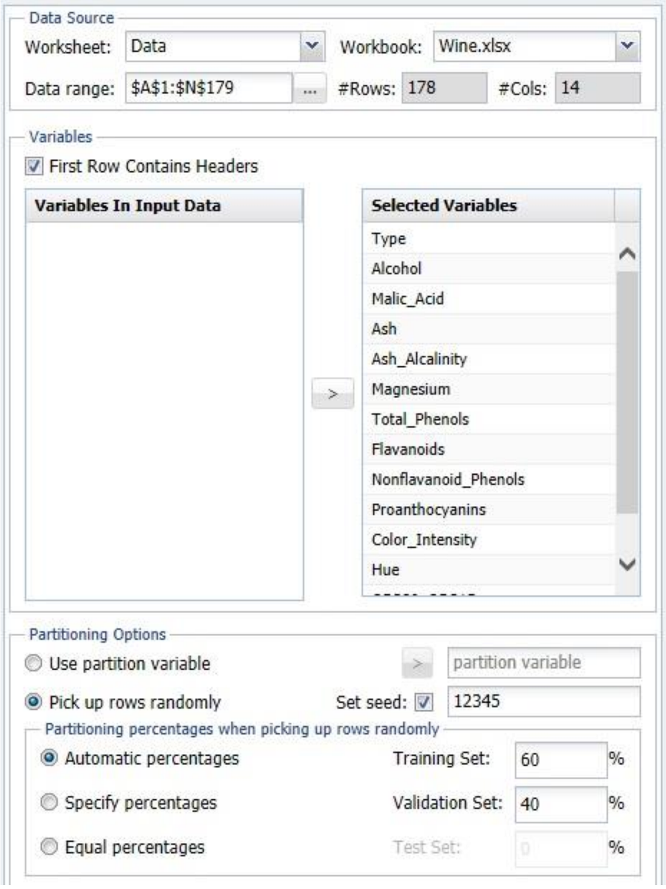
Click **Help – Example Models** on the Data Science ribbon, then click **Forecasting/Data Science Examples** to open the file **Wine.xlsx**.

This file contains 13 quantitative variables measuring the chemical attributes of wine samples from 3 different wineries (*Type* variable). The objective is to assign a wine classification to each record. A portion of this dataset is shown below.



Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
A	14.23	1.71	2.43	15.6	127	2.8	3.06	0.28	2.29	5.64	1.04	3.92	1065
A	13.2	1.78	2.14	11.2	100	2.65	2.76	0.26	1.28	4.38	1.05	3.4	1050
A	13.16	2.36	2.67	18.6	101	2.8	3.24	0.3	2.81	5.68	1.03	3.17	1185
A	14.37	1.95	2.5	16.8	113	3.85	3.49	0.24	2.18	7.8	0.86	3.45	1480
A	13.24	2.59	2.87	21	118	2.8	2.69	0.39	1.82	4.32	1.04	2.93	735
A	14.2	1.76	2.45	15.2	112	3.27	3.39	0.34	1.97	6.75	1.05	2.85	1450
A	14.39	1.87	2.45	14.6	96	2.5	2.52	0.3	1.98	5.25	1.02	3.58	1290
A	14.06	2.15	2.61	17.6	121	2.6	2.51	0.31	1.25	5.05	1.06	3.58	1295
A	14.83	1.64	2.17	14	97	2.8	2.98	0.29	1.98	5.2	1.08	2.85	1045
A	13.86	1.35	2.27	16	98	2.98	3.15	0.22	1.85	7.22	1.01	3.55	1045
A	14.1	2.16	2.3	18	105	2.95	3.32	0.22	2.38	5.75	1.25	3.17	1510
A	14.12	1.48	2.32	16.8	95	2.2	2.43	0.26	1.57	5	1.17	2.82	1280
A	13.75	1.73	2.41	16	89	2.6	2.76	0.29	1.81	5.6	1.15	2.9	1320
A	14.75	1.73	2.39	11.4	91	3.1	3.69	0.43	2.81	5.4	1.25	2.73	1150
A	14.38	1.87	2.38	12	102	3.3	3.64	0.29	2.96	7.5	1.2	3	1547
A	13.63	1.81	2.7	17.2	112	2.85	2.91	0.3	1.46	7.3	1.28	2.88	1310
A	14.3	1.92	2.72	20	120	2.8	3.14	0.33	1.97	6.2	1.07	2.65	1280
A	13.83	1.57	2.62	20	115	2.95	3.4	0.4	1.72	6.6	1.13	2.57	1130
A	14.19	1.59	2.48	16.5	108	3.3	3.93	0.32	1.86	8.7	1.23	2.82	1680
A	13.64	3.1	2.56	15.2	116	2.7	3.03	0.17	1.66	5.1	0.96	3.36	845

First, we partition the data into training and validation sets using a Standard Data Partition with percentages of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.



Data Source

Worksheet: Data

Workbook: Wine.xlsx

Data range: $A$1:$N$179

#Rows: 178

#Cols: 14

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	Type
	Alcohol
	Malic_Acid
	Ash
	Ash_Alcalinity
	Magnesium
	Total_Phenols
	Flavanoids
	Nonflavanoid_Phenols
	Proanthocyanins
	Color_Intensity
	Hue

Partitioning Options

○ Use partition variable — partition variable

● Pick up rows randomly — Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages — Training Set: 60 %

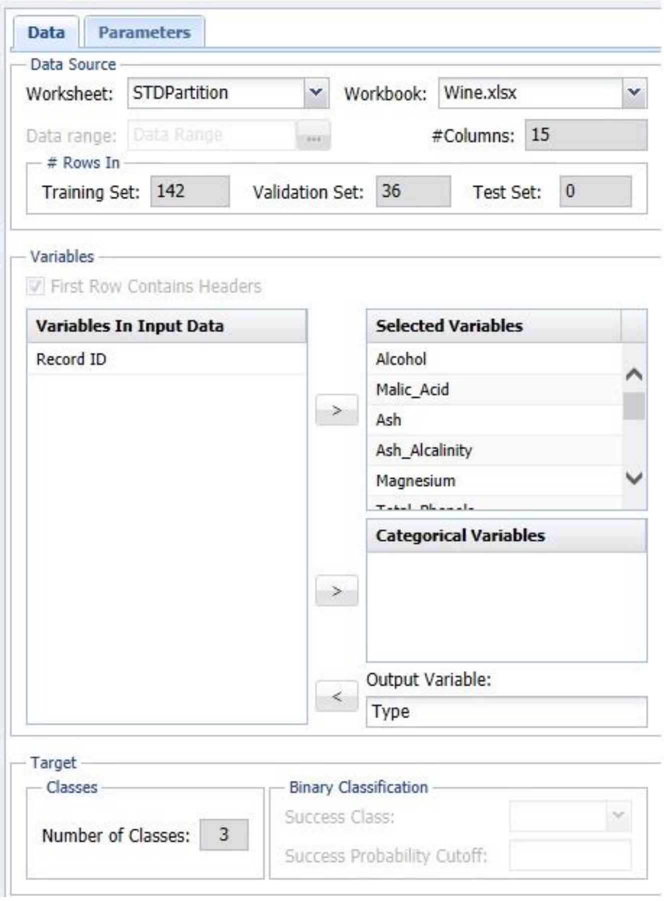
○ Specify percentages — Validation Set: 40 %

○ Equal percentages — Test Set: 0 %

Select a cell on the *StdPartition* worksheet, then click **Classify – Neural Network – Automatic Network** on the Data Science ribbon.

Select **Type** as the *Output variable* and the **remaining variables** as *Selected Variables.*

Since the Output variable contains three classes (A, B, and C) to denote the three different wineries, the options for *Binary Classification* are disabled. (The options under *Binary Classification* are only enabled when the number of classes is equal to 2.)



Worksheet: STDPartition

Workbook: Wine.xlsx

Data range: Data Range

#Columns: 15

Rows In

Training Set: 142

Validation Set: 36

Test Set: 0

☑ First Row Contains Headers

Variables In Input Data
Record ID

Selected Variables
Alcohol
Malic_Acid
Ash
Ash_Alcalinity
Magnesium

Categorical Variables

Output Variable: Type

Number of Classes: 3

Success Class:

Success Probability Cutoff:

Click **Next** to advance to the next tab.

When an automated network is created, several networks are run with increasing complexity in the architecture. The networks are limited to 2 hidden layers and the number of hidden neurons in each layer is bounded by UB1 = (#features + #classes) * 2/3 on the 1st layer and UB2 = (UB1 + #classes) * 2/3 on the 2nd layer.

First, all networks are trained with 1 hidden layer with the number of nodes not exceeding the UB1 and UB2 bounds, then a second layer is added and a 2 – layer architecture is tried until the UB2 limit is satisfied.

The limit on the total number of trained networks is the minimum of 100 and (UB1 * (1+UB2)). In this dataset, there are 13 features in the model and 3 classes in the Type output variable giving the following bounds:

UB1 = FLOOR(13 + 3) * 2/3 = 10.67 ~ 10

UB2 = FLOOR(10 + 3) * 2/3 = 8.67 ~ 8

(where FLOOR rounds a number down to the nearest multiple of significance.)

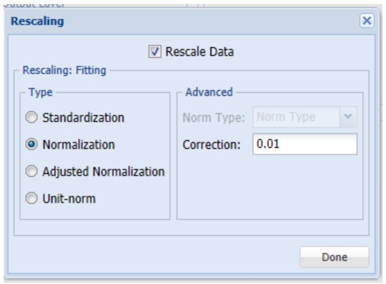
# Networks Trained = MIN {100, (10 * (1 + 8)} = 90

As discussed in previous sections, Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled.

For more information on partitioning, please see the Data Science Partitioning chapter.

Click **Rescale Data** to open the Rescaling dialog. Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

For this example, select **Rescale Data** and then select **Normalization**. Click **Done** to close the dialog.



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

○ Standardization

● Normalization

○ Adjusted Normalization

○ Unit-norm

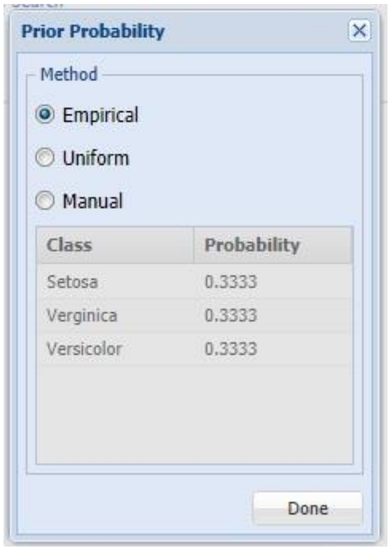
Advanced

Norm Type: Norm Type

Correction: 0.01

Note: When selecting a rescaling technique, it's recommended that you apply Normalization ([0,1)] if Sigmoid is selected for Hidden Layer Activation and Adjusted Normalization ([-1,1]) if Hyperbolic Tangent is selected for Hidden Layer Activation. This applies to both classification and regression. Since we will be using Logistic Sigmoid for   Hidden Layer Activation, Normalization was selected.

Click **Prior Probability**. Three options appear in the *Prior Probability* Dialog: *Empirical, Uniform* and *Manual.*



Prior Probability

Method

● Empirical

○ Uniform

○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

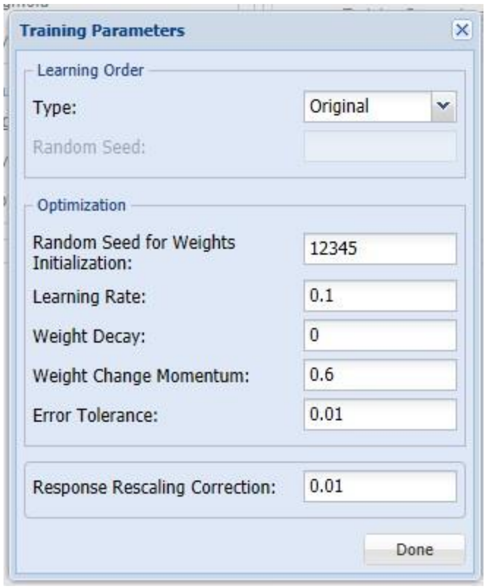
If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.

If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.

Select the third option, *Manual*, to manually enter the desired class and probability value.

Click Done to close the dialog and accept the default setting, Empirical.

Users can change both the Training Parameters and Stopping Rules for the Neural Network. Click Training Parameters to open the Training Parameters dialog. For more information on these options, please see the Neural Network Classification Options section below. For now, simply click Done to accept the option defaults and close the dialog.



Training Parameters

Learning Order

Type: Original

Random Seed:

Optimization

Random Seed for Weights Initialization: 12345

Learning Rate: 0.1

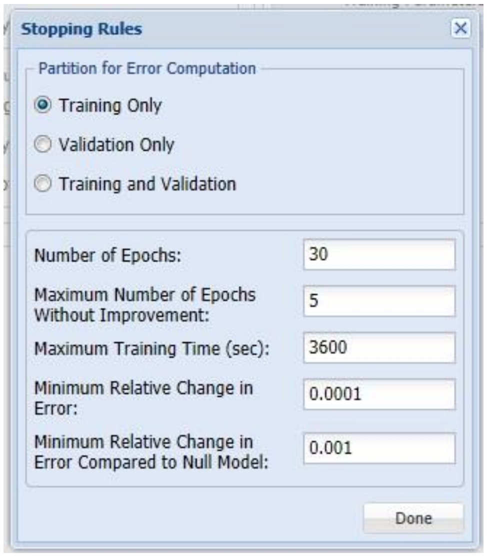
Weight Decay: 0

Weight Change Momentum: 0.6

Error Tolerance: 0.01

Response Rescaling Correction: 0.01

Click Stopping Rules to open the Stopping Rules dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error. For more information on these options, please see the Neural Network Classification Options section below. For now, simply click Done to accept the option defaults and close the dialog.



Stopping Rules

Partition for Error Computation

● Training Only

○ Validation Only

○ Training and Validation

Number of Epochs: 30

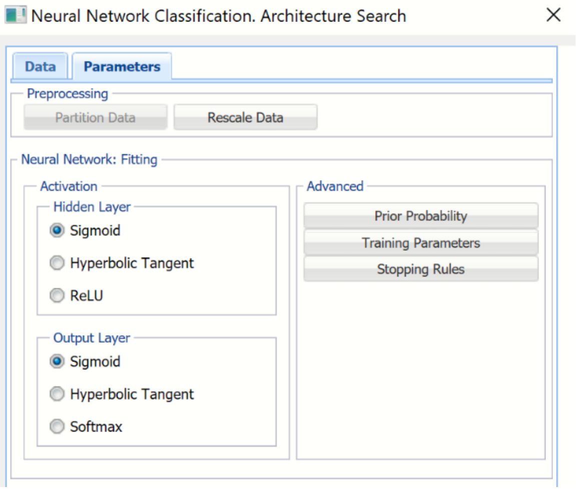
Maximum Number of Epochs Without Improvement: 5

Maximum Training Time (sec): 3600

Minimum Relative Change in Error: 0.0001

Minimum Relative Change in Error Compared to Null Model: 0.001

Keep the default selections for the Hidden Layer and Output Layer options. See the Neural Network Classification Options section below for more information on these options.



Neural Network Classification. Architecture Search

Data | Parameters

Preprocessing: Partition Data | Rescale Data

Neural Network: Fitting

Activation

Hidden Layer
- ● Sigmoid
- ○ Hyperbolic Tangent
- ○ ReLU

Output Layer
- ● Sigmoid
- ○ Hyperbolic Tangent
- ○ Softmax

Advanced: Prior Probability | Training Parameters | Stopping Rules

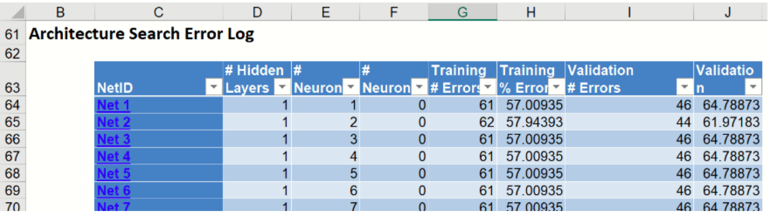
Click **Finish**. Output sheets are inserted to the right of the STDPartition worksheet.

NNC_Output

Click *NNC_Output* to open the first output sheet.

The top section of the output includes the Output Navigator which can be used to quickly navigate to various sections of the output. The Data, Variables, and Parameters/Options sections of the output all reflect inputs chosen by the user.

A little further down is the Architecture Search Error Log, a portion is shown below.

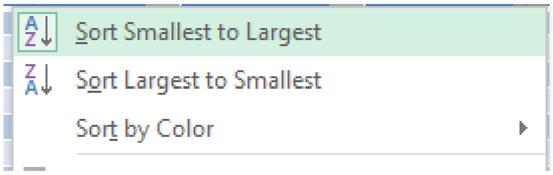


Architecture Search Error Log

NetID	# Hidden Layers	# Neuron	# Neuron	Training # Errors	Training % Error	Validation # Errors	Validation
Net 1	1	1	0	61	57.00935	46	64.78873
Net 2	1	2	0	62	57.94393	44	61.97183
Net 3	1	3	0	61	57.00935	46	64.78873
Net 4	1	4	0	61	57.00935	46	64.78873
Net 5	1	5	0	61	57.00935	46	64.78873
Net 6	1	6	0	61	57.00935	46	64.78873
Net 7	1	7	0	61	57.00935	46	64.78873

Notice the number of networks trained and reported in the Error Report was 90 (# Networks Trained = MIN {100, (10 * (1 + 8)} = 90).

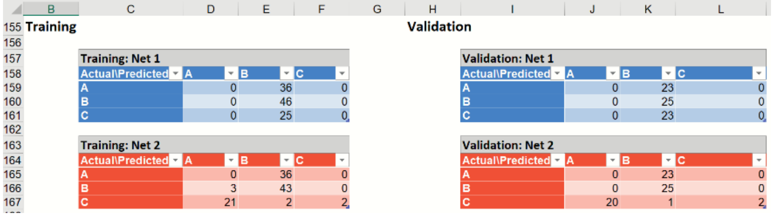
This report may be sorted by each column by clicking the arrow next to each column heading. Click the arrow next to Validation % Error and select **Sort Smallest to Largest** from the menu. Then click the arrow next to Training % Error and do the same to display all networks 0% Error in both the Training and Validation sets.



Click a Net ID, say Net 2, hyperlink to bring up the Neural Network Classification dialog. Click Finish to run the Neural Net Classification method with Manual Architecture using the input and option settings specified for Net 2.

Scroll down on the NNC_Output sheet to see the confusion matrices for each Neural Network listed in the table above. Here's the confusion matrices for Net

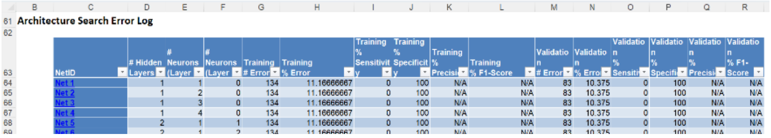
1and 2. These matrices expand upon the information shown in the Error Report for each network ID.



	Training					Validation			
	Training: Net 1					**Validation: Net 1**			
	Actual\Predicted	A	B	C		Actual\Predicted	A	B	C
	A	0	36	0		A	0	23	0
	B	0	46	0		B	0	25	0
	C	0	25	0		C	0	23	0
	Training: Net 2					**Validation: Net 2**			
	Actual\Predicted	A	B	C		Actual\Predicted	A	B	C
	A	0	36	0		A	0	23	0
	B	3	43	0		B	0	25	0
	C	21	2	2		C	20	1	2

Output Variable Containing Two Classes

The layout of this report changes when the number of classes is reduced to two. See the example report below.



Architecture Search Error Log

NetID	# Hidden Layers	# Neurons (Layer	# Neurons (Layer	Training # Error	Training % Error	Training % Sensitivit y	Training % Specificit y	Training % Precisi	Training % F1-Score	Validatio n # Error	Validatio n % Erro	Validatio n % Sensiti	Validatio n % Specifi	Validatio n % Precisi	Validatio n % F1-Score
Net 1	1	1	0	134	11.1666667	0	100	N/A	N/A	83	10.375	0	100	N/A	N/A
Net 2	1	2	0	134	11.1666667	0	100	N/A	N/A	83	10.375	0	100	N/A	N/A
Net 3	1	3	0	134	11.1666667	0	100	N/A	N/A	83	10.375	0	100	N/A	N/A
Net 4	1	4	0	134	11.1666667	0	100	N/A	N/A	83	10.375	0	100	N/A	N/A
Net 5	2	1	1	134	11.1666667	0	100	N/A	N/A	83	10.375	0	100	N/A	N/A
Net 6	2	1	2	134	11.1666667	0	100	N/A	N/A	83	10.375	0	100	N/A	N/A

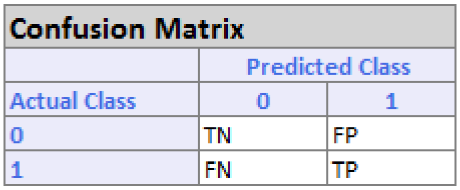
The Error Report provides the total number of errors in the classification -- % Error, % Sensitivity or positive rate, and % Specificity or negative rate -- produced by each network ID for the Training and Validation Sets. This report may be sorted by column by clicking the arrow next to each column heading.

Sensitivity and Specificity measures are unique to the Error Report when the Output Variable contains only two categories. Typically, these two categories can be labeled as success and failure, where one of them is more important than the other (i.e., the success of a tumor being cancerous or benign.) Sensitivity (true positive rate) measures the percentage of actual positives that are correctly identified as positive (i.e., the proportion of people with cancer who are correctly identified as having cancer). Specificity (true negative rate) measures the percentage of failures correctly identified as failures (i.e., the proportion of people with no cancer being categorized as not having cancer). The two are calculated as in the following (displayed in the Confusion Matrix).

Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)

Specificity (SPC) or True Negative Rate =TN / (FP + TN)

If we consider1 as a success, the Confusion Matrix would appear as in the following.



Confusion Matrix		
	Predicted Class	
Actual Class	0	1
0	TN	FP
1	FN	TP

When viewing the Net ID 10, this network has one hidden layer containing 10 nodes. For this neural network, the percentage of errors in the Training Set is 3.95%, and the percentage of errors in the Validation Set is 5.45%.The percent sensitivity is 87.25 % and 89.19% for the training partition and validation partition, respectively. This means that in the Training Set, 87.25% of the records classified as positive were in fact positive, and 89.19% of the records in the Validation Set classified as positive were in fact positive.

Sensitivity and Specificity measures can vary in importance depending upon the application and goals of the application. The values for sensitivity and specificity are pretty low in this network, which could indicate that alternate parameters, a different architecture, or a different model might be in order. Declaring a tumor cancerous when it is benign could result in many unnecessary expensive and invasive tests and treatments. However, in a model where a success does not indicate a potentially fatal disease, this measure might not be viewed as important.

The percentage specificity is 87.23% for the Training Set, and 95.76% in the Validation Set. This means that 87.23% of the records in the training Set and 95.76% of the records in the Validation Set identified as negative, were in fact negative. In the case of a cancer diagnosis, we would prefer that this percentage be higher, or much closer to 100%, as it could potentially be fatal if a person with cancer was diagnosed as not having cancer.

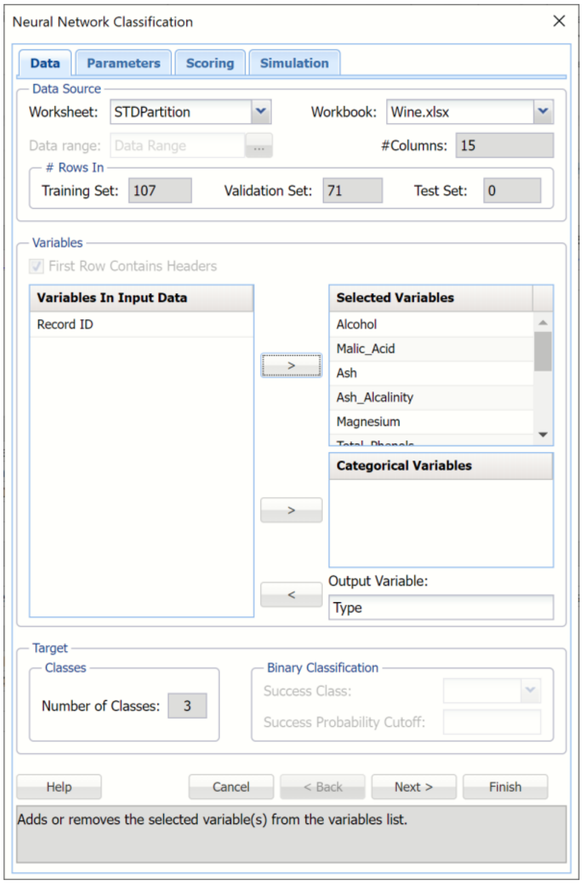
Manual Neural Network Classification Example

This example uses the same partitioned dataset to illustrate the use of the *Manual Network Architecture* selection. This example reuses the partitions created on the STDPartition worksheet in the previous section, Automatic Neural Network Classification Example.

Inputs

1. Select a cell on the *StdPartition* worksheet, then click **Classify – Neural Network – Manual Network** on the Data Science ribbon. The *Neural Network Classification* dialog appears.
2. Select **Type** as the *Output variable* and the **remaining variables** as *Selected Variables.* Since the Output variable contains three classes (A, B, and C) to denote the three different wineries, the options for *Classes in the Output Variable* are disabled. (The options under *Classes in the Output Variable* are only enabled when the number of classes is equal to 2.)

Neural Network Classification dialog, Data tab



Worksheet: STDPartition

Workbook: Wine.xlsx

Data range: Data Range

#Columns: 15

# Rows In

Training Set: 107 Validation Set: 71 Test Set: 0

☑ First Row Contains Headers

Variables In Input Data: Record ID

Selected Variables: Alcohol, Malic_Acid, Ash, Ash_Alcalinity, Magnesium, Total_Phenols

Output Variable: Type

Number of Classes: 3

Success Class:

Success Probability Cutoff:

Adds or removes the selected variable(s) from the variables list.

3. Click **Next** to advance to the next tab.

As discussed in the previous sections, Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

4. Click **Rescale Data** to open the Rescaling dialog.

   Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. *See the important note related to Rescale Data and the new Simulation tab in the Options section (below) within this chapter.*

   For more information on rescaling, see the Rescale Continuous Data section within the Transform Continuous Data chapter, that occurs earlier in this guide.

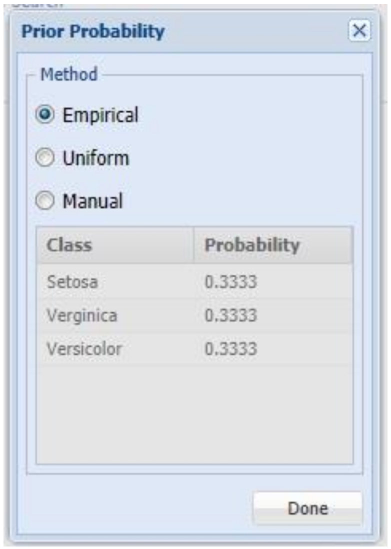
   Note: When selecting a rescaling technique, it's recommended that you apply Normalization ([0,1)] if Sigmoid is selected for Hidden Layer Activation and Adjusted Normalization ([-1,1]) if Hyperbolic Tangent is selected for Hidden Layer Activation. However, in this particular example dataset, the Neural Network algorithm performs best when Standardization was selected, rather than Normalization.

5. Click **Add Layer** to add a hidden layer to the Neural Network. To remove a layer, select the layer to be removed, then click *Remove Layer*. Enter 12 for Neurons.

   Keep the default selections for the Hidden Layer and Output Layer options. See the Neural Network Classification Options section below for more information on these options.

6. Click **Prior Probability**. Three options appear in the *Prior Probability* Dialog: *Empirical*, *Uniform* and *Manual.*

   Prior Probability dialog



Method

● Empirical

○ Uniform

○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

   If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.

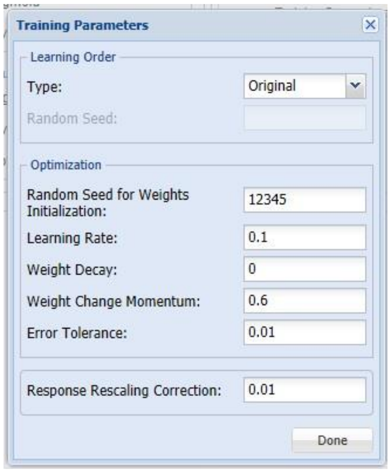
   If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.

   Select the third option, *Manual,* to manually enter the desired class and probability value.

   Click **Done** to close the dialog and accept the default setting, *Empirical*.

7. Click **Training Parameters** to open the *Training Parameters* dialog. See the Neural Network Options section below for more information on these options. For now, click **Done** to accept the default settings and close the dialog.

Training Parameters Dialog



Training Parameters

Learning Order

Type: Original

Random Seed:

Optimization

Random Seed for Weights Initialization: 12345

Learning Rate: 0.1

Weight Decay: 0

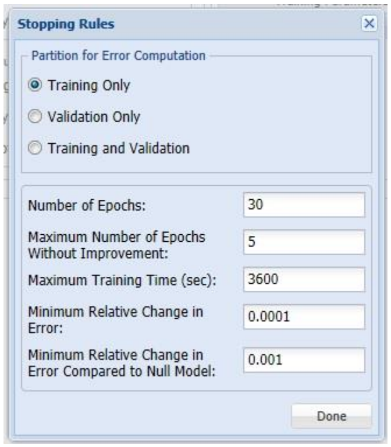
Weight Change Momentum: 0.6

Error Tolerance: 0.01

Response Rescaling Correction: 0.01

8. Click **Stopping Rules** to open the *Stopping Rules* dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error. Again, see the example above or the Neural Network Options section below for more information on these parameters. For now, click **Done** to accept the default settings and close the dialog.

   Stopping Rules dialog



Stopping Rules

Partition for Error Computation

● Training Only

○ Validation Only

○ Training and Validation

Number of Epochs: 30

Maximum Number of Epochs Without Improvement: 5

Maximum Training Time (sec): 3600

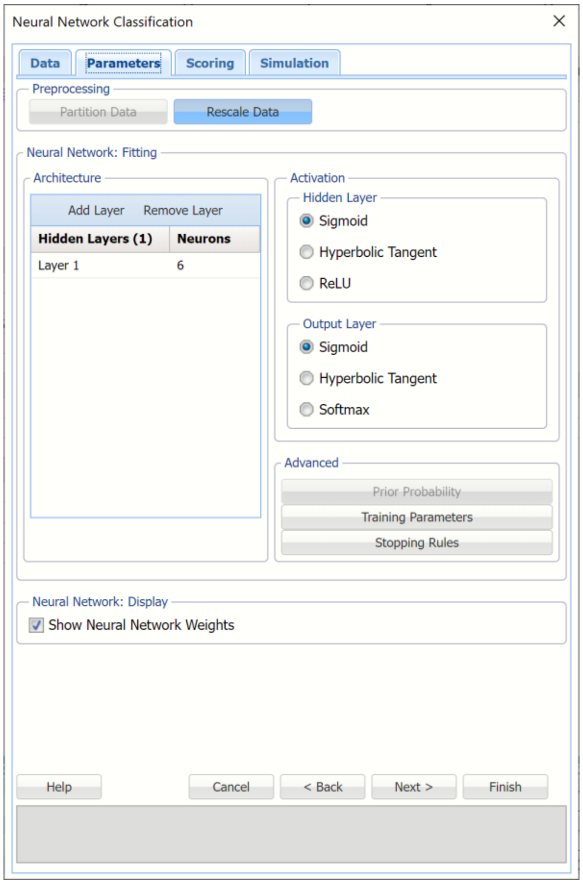
Minimum Relative Change in Error: 0.0001

Minimum Relative Change in Error Compared to Null Model: 0.001

9. Select **Show Neural Network Weights** to include this information in the output.

   Keep the default selections for the Hidden Layer and Output Layer options. See the Neural Network Classification Options section below for more information on these options.

Neural Network Classification dialog, Parameters tab



Preprocessing

Neural Network: Fitting

Architecture

Hidden Layers (1)	Neurons
Layer 1	6

Activation

Hidden Layer

● Sigmoid

○ Hyperbolic Tangent

○ ReLU

Output Layer

● Sigmoid

○ Hyperbolic Tangent

○ Softmax

Advanced

Neural Network: Display

☑ Show Neural Network Weights

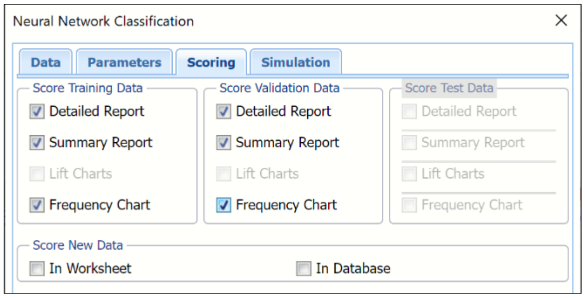
10. Click **Next** to advance to the *Scoring* tab.

11. Select **Detailed Report** and **Summary report** under both *Score Training Data* and *Score Validation Data*. Lift Charts are disabled since the number of classes is greater than 2.

    Since a Test Data partition was not created, the options under Score Test Data are disabled. For information on how to create a test partition, see the "Data Science Partition" chapter.

    For more information on the *Score New Data* options, see the “Scoring New Data” chapter.

Neural Network Classification dialog, Scoring tab



Score Training Data

☑ Detailed Report

☑ Summary Report

□ Lift Charts

☑ Frequency Chart

Score Validation Data

☑ Detailed Report

☑ Summary Report

□ Lift Charts

☑ Frequency Chart

Score Test Data

□ Detailed Report

□ Summary Report

□ Lift Charts

□ Frequency Chart

Score New Data

□ In Worksheet

□ In Database

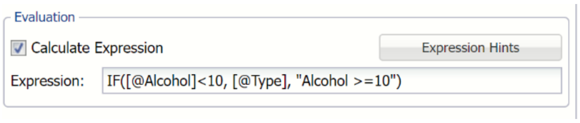
12. Select Simulation Response Prediction to enable all options on the Simulation tab of the Discriminant Analysis dialog. (This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.)

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, _Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

    Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the PFBM_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

    Enter the following Excel formula into the expression field.

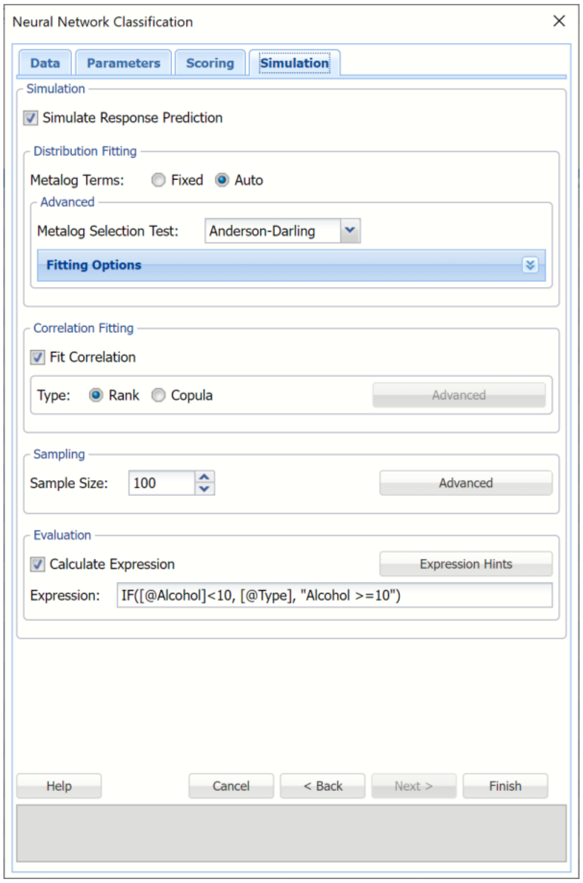
    IF([@Alcohol]<10, [@Type], "Alcohol >=10")



☑ Calculate Expression

Expression: IF([@Alcohol]<10, [@Type], "Alcohol >=10")

Neural Network Classification dialog, Simulation tab


Neural Network Classification

Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

☑ Calculate Expression

Expression: IF([@Alcohol]<10, [@Type], "Alcohol >=10")

13. Click **Finish** to run the Neural Network Clasification algorithm.

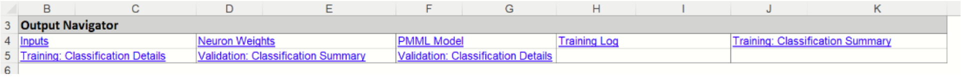
Output Worksheets

Output worksheets are inserted to the right of the STDPartition worksheet.

NNC_Output

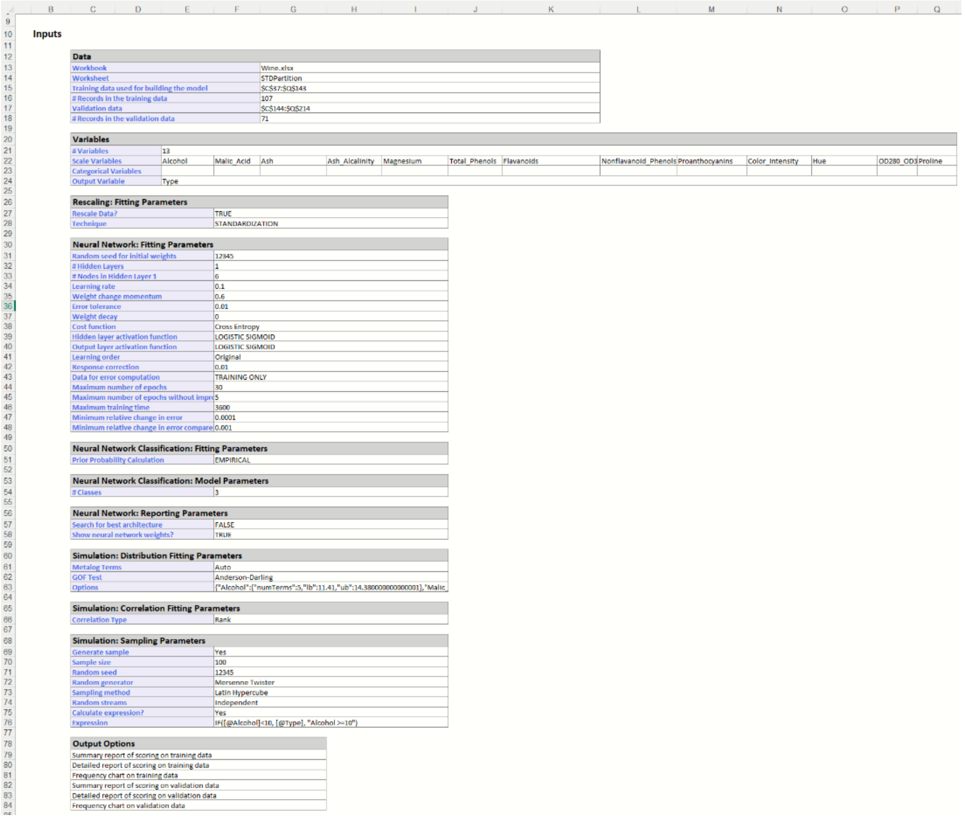
Scroll down to the Inputs section of the output sheet. This section includes all of the input selections from the

Click *NNC_Output1* to view the Output Navigator. Click any link within the table to navigate to the report. Each output worksheet includes the Output Navigator at the top of the sheet.



Output Navigator				
Inputs	Neuron Weights	PMML Model	Training Log	Training: Classification Summary
Training: Classification Details	Validation: Classification Summary	Validation: Classification Details		

Scroll down to the Inputs section. This section includs all the inputs selected on the Neural Network Classification dialog.



Inputs

Data	
Workbook	Wine.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$Q$143
# Records in the training data	107
Validation data	$C$144:$Q$214
# Records in the validation data	71

Variables													
# Variables	13												
Scale Variables	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD3	Proline
Categorical Variables													
Output Variable	Type												

Rescaling: Fitting Parameters	
Rescale Data?	TRUE
Technique	STANDARDIZATION

Neural Network: Fitting Parameters	
Random seed for initial weights	12345
# Hidden Layers	1
# Nodes in Hidden Layer 1	6
Learning rate	0.1
Weight change momentum	0.6
Error tolerance	0.01
Weight decay	0
Cost function	Cross Entropy
Hidden layer activation function	LOGISTIC SIGMOID
Output layer activation function	LOGISTIC SIGMOID
Learning order	Original
Response correction	0.01
Data for error computation	TRAINING ONLY
Maximum number of epochs	30
Maximum number of epochs without impro	5
Maximum training time	3600
Minimum relative change in error	0.0001
Minimum relative change in error compare	0.001

Neural Network Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Neural Network Classification: Model Parameters	
# Classes	3

Neural Network: Reporting Parameters	
Search for best architecture	FALSE
Show neural network weights?	TRUE

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"Alcohol":{"numTerms":5,"lb":11.41,"ub":14.380000000000001},"Malic_

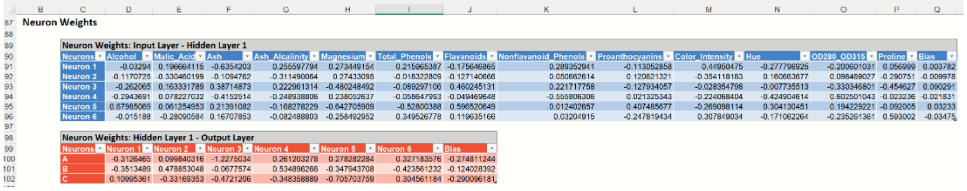
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	Yes
Expression	IF([@Alcohol]<10, [@Type], "Alcohol >=10")

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Frequency chart on validation data

Scroll down to the Neuron Weights report. Analytic Solver Data Science provides intermediate information produced during the last pass through the network. Click the *Neuron Weights* link in the *Output Navigator* to view the Interlayer connections' weights table.

Neuron Weights Report



Neuron Weights

Neuron Weights: Input Layer - Hidden Layer 1

Neurons	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline	Bias
Neuron 1	-0.03294	0.196664115	-0.6354203	0.255597794	0.273449154	0.215965387	-0.175646865	0.289352941	-0.113052858	0.44950475	-0.277796925	-0.200601031	0.056999	0.003782
Neuron 2	-0.1170725	-0.330460199	-0.1094762	-0.311490064	0.27433095	-0.018322809	-0.127140666	0.050662614	0.120621321	-0.354118183	0.160663677	0.098469027	-0.290751	-0.009978
Neuron 3	-0.262065	0.163331789	0.38714873	0.222961314	-0.480248492	-0.089297106	0.460245131	0.221717758	-0.127934057	-0.028354798	-0.007735513	-0.330346801	-0.454627	0.090291
Neuron 4	-0.2943691	0.078227032	-0.4152514	-0.248938806	0.238052637	-0.058647993	-0.049469648	-0.555806306	0.021325343	-0.224068404	-0.424904614	0.602501043	-0.023236	-0.021831
Neuron 5	0.67985069	0.061254953	0.21391082	-0.168278229	-0.642705909	-0.52800388	0.596520649	0.012402657	0.407485677	-0.269098114	0.304130451	0.194229221	-0.092005	0.03233
Neuron 6	-0.015188	-0.28090584	0.16707853	-0.082488803	-0.258492952	0.349526778	0.119635166	0.03204915	-0.247819434	0.307849034	-0.171062264	-0.235261361	0.593002	-0.03475

Neuron Weights: Hidden Layer 1 - Output Layer

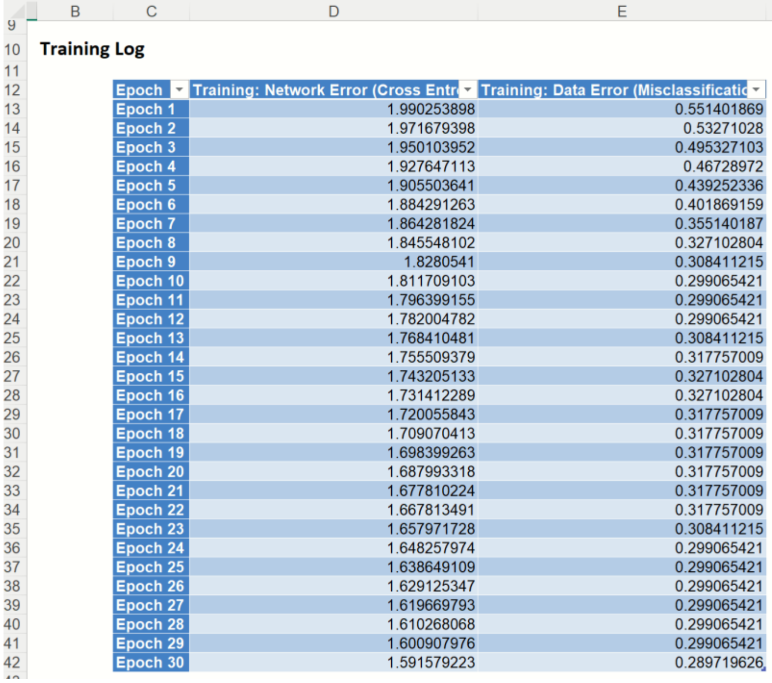
Neurons	Neuron 1	Neuron 2	Neuron 3	Neuron 4	Neuron 5	Neuron 6	Bias
A	-0.3126465	0.099840316	-1.2275034	0.261203278	0.278282284	0.327183576	-0.274811244
B	-0.3513489	0.478853048	-0.0677574	0.534896266	-0.347943708	-0.423561232	-0.124028392
C	0.10995361	-0.33169353	-0.4721206	-0.348358889	-0.705703759	0.304561184	-0.290096181

Recall that a key element in a neural network is the weights for the connections between nodes. In this example, we chose to have one hidden layer containing 6 neurons. Analytic Solver Data Science's output contains a section that contains the final values for the weights between the input layer and the hidden layer, between hidden layers, and between the last hidden layer and the output layer. This information is useful at viewing the "insides" of the neural network; however, it is unlikely to be of use to the data analyst end-user. Displayed above are the final connection weights between the input layer and the hidden layer for our example.

NNC_TrainLog

Click the *Training Log* link in the Output Navigator or click the NNC_TrainLog output tab, to display the Neural Network Training Log. This log displays the Sum of Squared errors and Misclassification errors for each epoch or iteration of the Neural Network. Thirty epochs, or iterations, were performed.

During an epoch, each training record is fed forward in the network and classified. The error is calculated and is back propagated for the weights correction. Weights are continuously adjusted during the epoch. The misclassification error is computed as the records pass through the network. This table does not report the misclassification error after the final weight adjustment. Scoring of the training data is performed using the final weights so the training classification error may not exactly match with the last epoch error in the Epoch log.



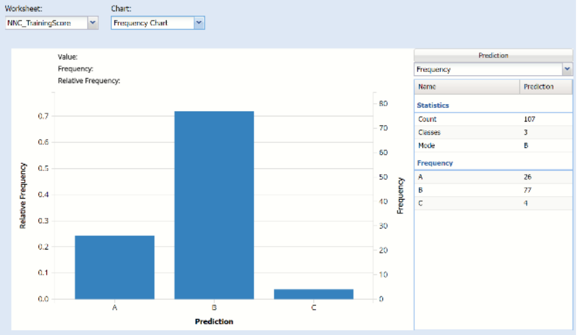
Training Log

Epoch	Training: Network Error (Cross Entr	Training: Data Error (Misclassificatic
Epoch 1	1.990253898	0.551401869
Epoch 2	1.971679398	0.53271028
Epoch 3	1.950103952	0.495327103
Epoch 4	1.927647113	0.46728972
Epoch 5	1.905503641	0.439252336
Epoch 6	1.884291263	0.401869159
Epoch 7	1.864281824	0.355140187
Epoch 8	1.845548102	0.327102804
Epoch 9	1.8280541	0.308411215
Epoch 10	1.811709103	0.299065421
Epoch 11	1.796399155	0.299065421
Epoch 12	1.782004782	0.299065421
Epoch 13	1.768410481	0.308411215
Epoch 14	1.755509379	0.317757009
Epoch 15	1.743205133	0.327102804
Epoch 16	1.731412289	0.327102804
Epoch 17	1.720055843	0.317757009
Epoch 18	1.709070413	0.317757009
Epoch 19	1.698399263	0.317757009
Epoch 20	1.687993318	0.317757009
Epoch 21	1.677810224	0.317757009
Epoch 22	1.667813491	0.317757009
Epoch 23	1.657971728	0.308411215
Epoch 24	1.648257974	0.299065421
Epoch 25	1.638649109	0.299065421
Epoch 26	1.629125347	0.299065421
Epoch 27	1.619669793	0.299065421
Epoch 28	1.610268068	0.299065421
Epoch 29	1.600907976	0.299065421
Epoch 30	1.591579223	0.289719626

NNC_TrainingScore

Click the *NNC_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Classification Summary and the Training: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

> Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select the desired worksheet under Worksheet and the desired chart under Chart.



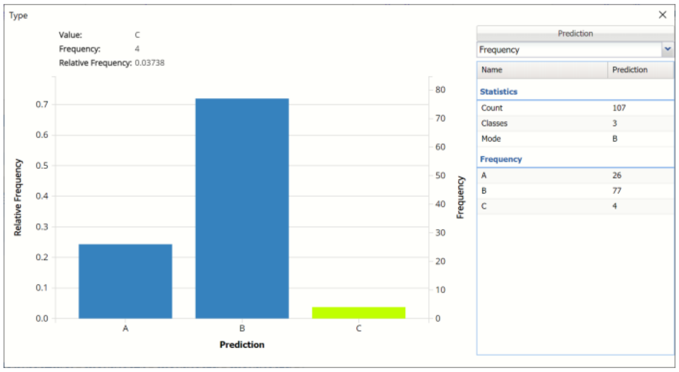
Name	Prediction
Statistics	
Count	107
Classes	3
Mode	B
Frequency	
A	26
B	77
C	4

- **Frequency Charts:** The output variable frequency chart opens automatically once the *NNC_TrainingScore* worksheet is selected. To close this chart, click the “x” in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the

NNC_TrainingScore tab. To change the position of the chart on the screen, simply grab the title bar of the chart and move to the desired location.

Frequency: This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

Frequency Chart on NNC_TrainingScore output sheet



Value: C
Frequency: 4
Relative Frequency: 0.03738

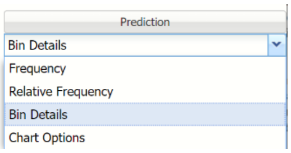
Prediction

Frequency

Name	Prediction
Statistics	
Count	107
Classes	3
Mode	B
Frequency	
A	26
B	77
C	4

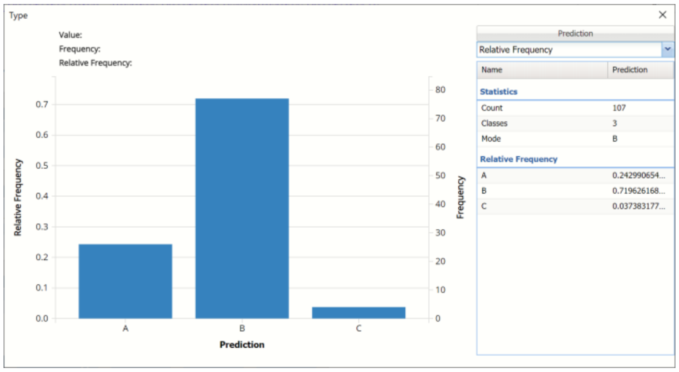
Click the down arrow next to Frequency to switch to Relative Frequency, Bin Details or Chart Options view.

Frequency Chart, Frequency View



Relative Frequency: Displays the relative frequency chart.

Relative Frequency Chart



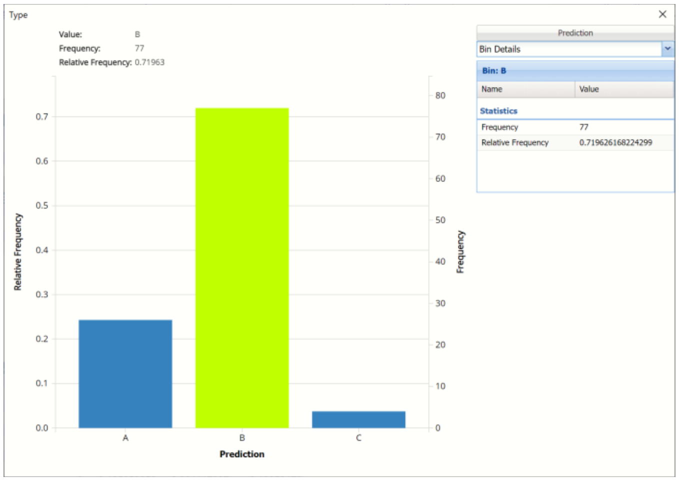
Value:
Frequency:
Relative Frequency:

Prediction

Relative Frequency

Name	Prediction
Statistics	
Count	107
Classes	3
Mode	B
Relative Frequency	
A	0.242990654...
B	0.719626168...
C	0.037383177...

Bin Details: Displays information related to each bin in the chart.



Type

Value: B
Frequency: 77
Relative Frequency: 0.71963

Prediction

Bin Details

Bin: B

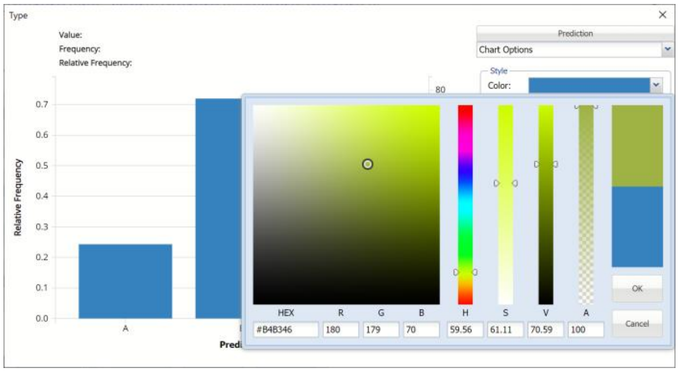
Name	Value

Statistics

Frequency	77
Relative Frequency	0.719626168224299

Chart Options: Use this view to change the color of the bars in the chart.

Chart Options View



Type

Value:
Frequency:
Relative Frequency:

Prediction

Chart Options

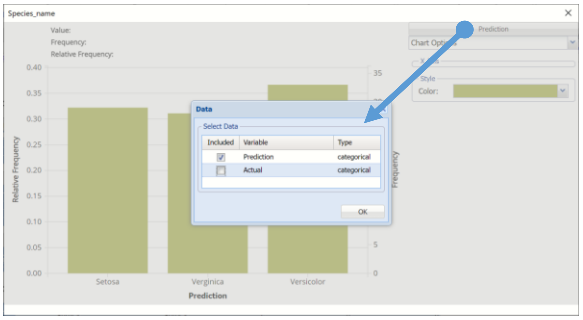
Style

Color:

HEX	R	G	B	H	S	V	A
#B4B346	180	179	70	59.56	61.11	70.59	100

- To see both the actual and predicted frequency, click Prediction and select Actual. This change will be reflected on all charts.

Click Predicted/Actual to change view



Species_name

Value:
Frequency:
Relative Frequency:

Prediction

Chart Opti

Style

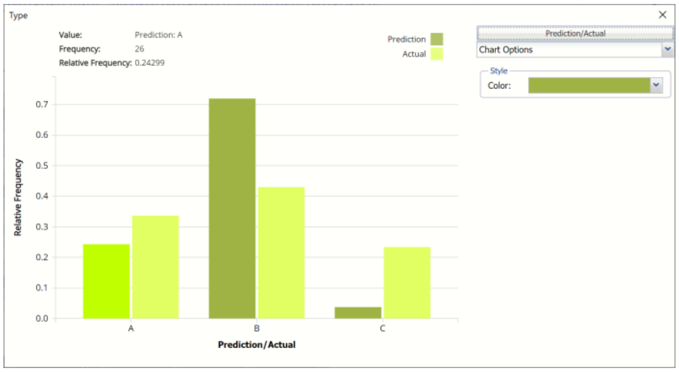
Color:

Data

Select Data

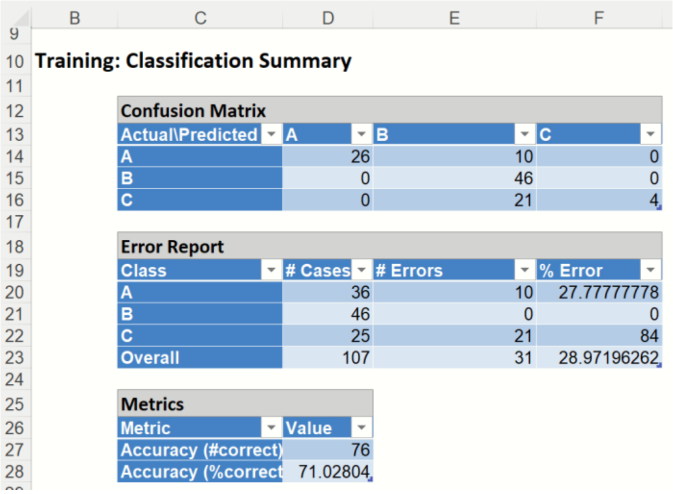
Included	Variable	Type
☑	Prediction	categorical
☐	Actual	categorical

NNC_TrainingScore Frequency Chart with Actual and Predicted



- **Classification Summary:** In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

NNC_TrainingScore: Training: Classification Summary



Training: Classification Summary

Confusion Matrix			
Actual\Predicted	A	B	C
A	26	10	0
B	0	46	0
C	0	21	4

Error Report			
Class	# Cases	# Errors	% Error
A	36	10	27.77777778
B	46	0	0
C	25	21	84
Overall	107	31	28.97196262

Metrics	
Metric	Value
Accuracy (#correct)	76
Accuracy (%correct	71.02804

This Summary report tallies the actual and predicted classifications. (Predicted classifications were generated by applying the model to the validation data.) Correct classification counts are along the diagonal from the upper left to the lower right.

There were 31 misclassified records labeled in the Training partition:

- Ten type A records were incorrectly assigned to type B.
- Twenty-one type C records were incorrectly assigned to type B.

The total misclassification error is 28.97% (31 misclassified records / 107 total records). Any misclassified records will appear under Training: Classification Details in red.

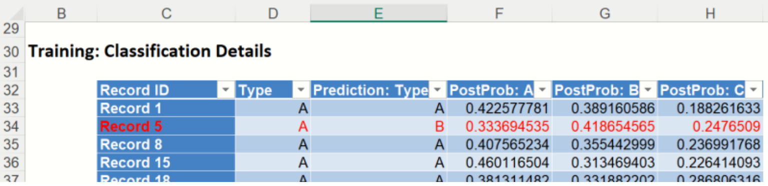
Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct = 76 and %Correct = 71.03%): Refers to the ability of the classifier to predict a class label correctly.

- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

  NNC_TrainingScore: Training: Classification Details



Training: Classification Details

Record ID	Type	Prediction: Type	PostProb: A	PostProb: B	PostProb: C
Record 1	A	A	0.422577781	0.389160586	0.188261633
Record 5	A	B	0.333694535	0.418654565	0.2476509
Record 8	A	A	0.407565234	0.355442999	0.236991768
Record 15	A	A	0.460116504	0.313469403	0.226414093
Record 18	A	A	0.381311482	0.331882202	0.286806316

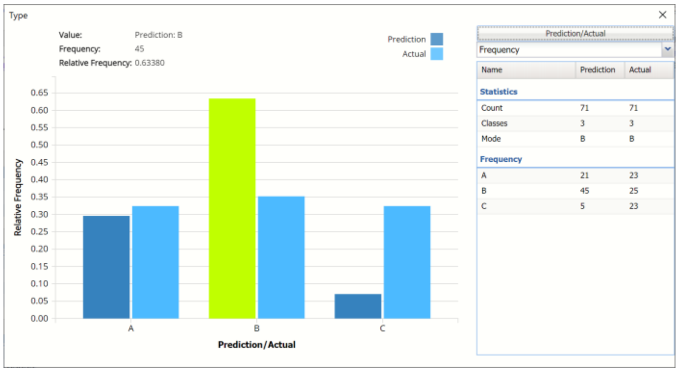
NNC_ValidationScore

Click the *NNC_ValidationScore* tab to view the newly added Output Variable frequency chart, the Validation: Classification Summary and the Validation: Classification Details report. All calculations, charts and predictions on this worksheet apply to the Validation data.

- **Frequency Charts:** The output variable frequency chart opens automatically once the NNC_ValidationScore worksheet is selected. To close this chart, click the "x" in the upper right hand corner. To reopen, click onto another tab and then click back to the NNC_ValidationScore tab. To change the placement of the chart, grab the title bar and move to the desired location on the screen.

  Click the Frequency chart to display the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode. Selective Relative Frequency from the drop down menu, on the right, to see the relative frequencies of the output variable for both actual and predicted. See above for more information on this chart.

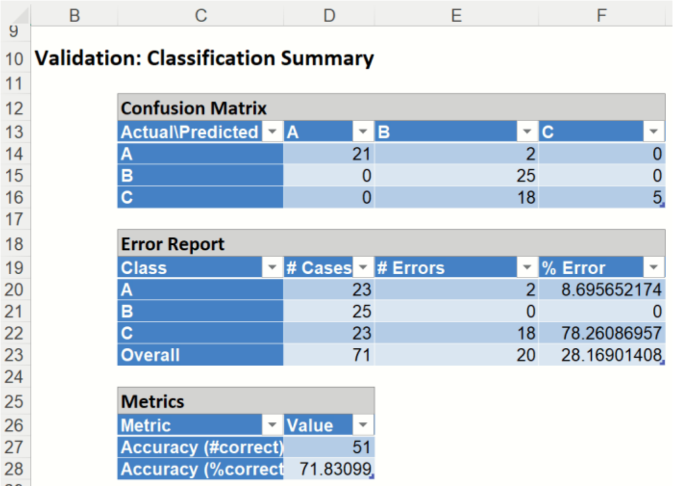
  DA_ValidationScore Frequency Chart



Name	Prediction	Actual
Statistics		
Count	71	71
Classes	3	3
Mode	B	B
Frequency		
A	21	23
B	45	25
C	5	23

- **Classification Summary:** This report contains the confusion matrix for the validation data set.

NNC_ValidationScore: Classification Summary



Validation: Classification Summary

Confusion Matrix			
Actual\Predicted	A	B	C
A	21	2	0
B	0	25	0
C	0	18	5

Error Report			
Class	# Cases	# Errors	% Error
A	23	2	8.695652174
B	25	0	0
C	23	18	78.26086957
Overall	71	20	28.16901408

Metrics	
Metric	Value
Accuracy (#correct)	51
Accuracy (%correct	71.83099

Twenty records were miscalssified by Neural Networks Classification.

- Two (2) type A records were misclassified as type B.
- Eighteen (18) type C records were miscalssified as Type B.

The total number of misclassified records was 20 (18 + 2 which results in an error equal to 28.2%.

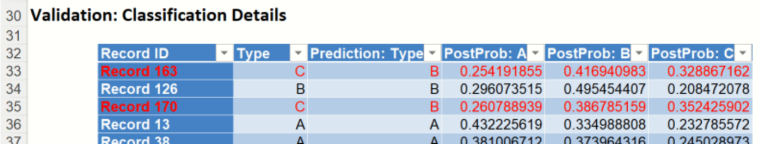
Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct = 51 and %Correct = 71.8%): Refers to the ability of the classifier to predict a class label correctly.

- **Classification Details**: This table displays how each observation in the validation data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

NNC_ValidationScore: Validation: Classification Details



Validation: Classification Details

Record ID	Type	Prediction: Type	PostProb: A	PostProb: B	PostProb: C
Record 163	C	B	0.254191855	0.416940983	0.328867162
Record 126	B	B	0.296073515	0.495454407	0.208472078
Record 170	C	B	0.260788939	0.386785159	0.352425902
Record 13	A	A	0.432225619	0.334988808	0.232785572
Record 38	A	A	0.381006712	0.373964316	0.245028973

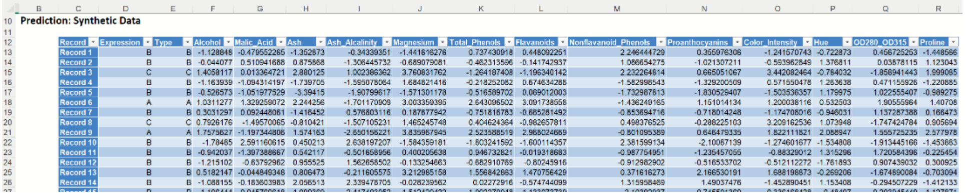
NNC_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, NNC_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Neural Network Classification dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the training partition predictions (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. A dialog is displayed with the option to switch between the

Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type. Note that this data has been rescaled because we selected Rescale Data on the Parameters tab in the Neural Network Classification dialog.

Synthetic Data



Prediction: Synthetic Data

Record	Expression	Type	Alcohol	Malic_Acid	Ash	Ash_Alcalinity	Magnesium	Total_Phenols	Flavanoids	Nonflavanoid_Phenols	Proanthocyanins	Color_Intensity	Hue	OD280_OD315	Proline
Record 1	B	B	-1.128848	-0.479552265	-1.352873	-0.34339351	-1.441616276	0.737430918	0.448092251	2.246444729	0.355976306	-1.241570743	-0.722873	0.456725253	-1.448566
Record 2	B	B	-0.044077	0.510941688	0.875868	-1.306445732	-0.689079081	-0.462313596	-0.141742937	1.086654275	-1.021307211	-0.593962849	1.376811	0.03878115	1.123043
Record 3	C	C	1.4058117	0.013364721	2.880125	1.002386362	3.760831762	-1.264187408	-1.196340142	2.232264814	0.665051067	3.442082464	-0.784032	-1.858941443	1.999085
Record 4	B	B	-1.163939	-1.094314197	-1.739705	-1.595078064	1.684821416	-0.218252082	0.674634288	-1.582998543	-1.329200509	0.571550478	1.263638	0.471155926	-1.220885
Record 5	B	B	-0.526573	-1.051977529	-3.39415	-1.90799617	-1.571301178	-0.516589702	0.069012003	-1.732987813	-1.830529407	-1.503536357	1.179975	1.022555407	-0.989275
Record 6	A	A	1.0311277	1.329259072	2.244256	-1.701170909	3.003359395	2.643096502	3.091738558	-1.436249165	1.151014134	1.200038116	0.532503	1.90555964	1.40708
Record 7	B	B	0.3031297	0.092448061	-1.416452	0.576803116	0.187677942	-0.751816783	-0.665281492	-0.853694716	-0.718014248	-1.174708016	-0.946031	1.137287388	0.166473
Record 8	C	C	0.7926176	-1.49570065	-0.810421	-1.507105231	1.465245748	0.404624364	-0.982657811	0.498376525	-0.288225103	3.209162536	1.073948	-1.747424784	0.905694
Record 9	A	A	1.7575627	-1.197344806	1.574163	-2.650156221	3.835967945	2.523588519	2.968024669	-0.801095389	0.646479335	1.822111821	2.088947	1.555725235	2.577978
Record 10	B	B	-1.78465	2.591160615	0.450213	2.638197207	-1.584359181	-1.803241592	-1.600114357	2.381599134	-2.10067139	-1.274601677	-1.534808	-1.913445166	-1.453863
Record 11	B	B	-0.942037	-1.397388667	0.542117	-0.501658956	0.400205638	0.946732821	-0.019318683	-0.987754951	-1.235457055	-0.88329012	1.315296	1.720584396	-0.225454
Record 12	B	B	-1.215102	-0.63792962	0.955525	1.562658502	-0.133254663	-0.682910769	-0.80245916	-0.912982902	-0.516533702	-0.512112272	-1.761893	0.907439032	0.300925
Record 13	B	B	0.5182147	-0.044849348	0.806473	-0.211605575	3.212985158	1.556842863	1.470756429	0.371616273	2.166530191	1.688198873	-0.269206	-1.674890084	-0.703094
Record 14	B	B	-1.088155	-0.183603983	2.056513	2.339478705	-0.028239562	0.02272916	-0.574744099	1.315958469	1.49037476	-1.452890451	1.153408	-0.294507229	-1.412133

Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the Discriminant Analysis dialog

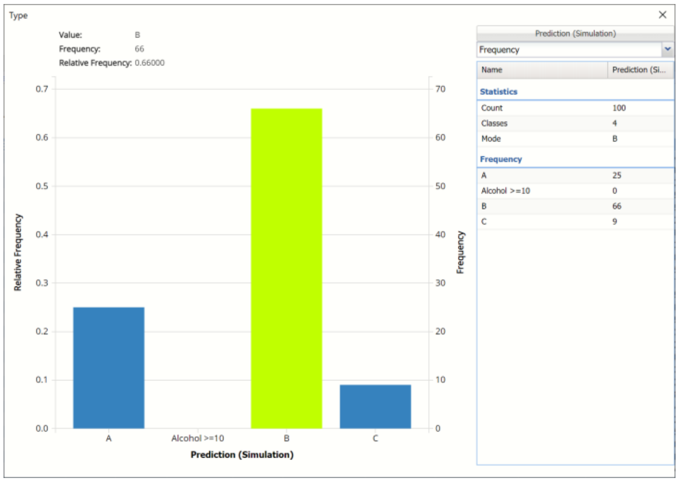
IF([@Alcohol]<10, [@Type], "Alcohol >=10")

The results in this column are either A, B, C or Alcohol >= 10 depending on the alcohol content for each record in the synthetic data.

The remainder of the data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training partition and the synthetic data. The chart below displays frequency information for the predicted values in the synthetic data.

Prediction Frequency Chart for NNC_Simulation output



Type

Value: B
Frequency: 66
Relative Frequency: 0.66000

Prediction (Simulation)

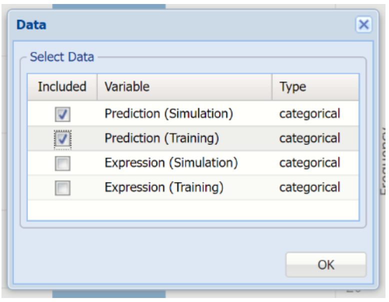
Frequency

Name	Prediction (Si...
Statistics	
Count	100
Classes	4
Mode	B
Frequency	
A	25
Alcohol >=10	0
B	66
C	9

In the synthetic data, 25 records were classified as Type A, 66 for Type B and 9 for Type C.

Click *Prediction (Simulation)* and select Prediction (Training) in the Data dialog to display a frequemcy chart based on the Training partition.

Data Dialog

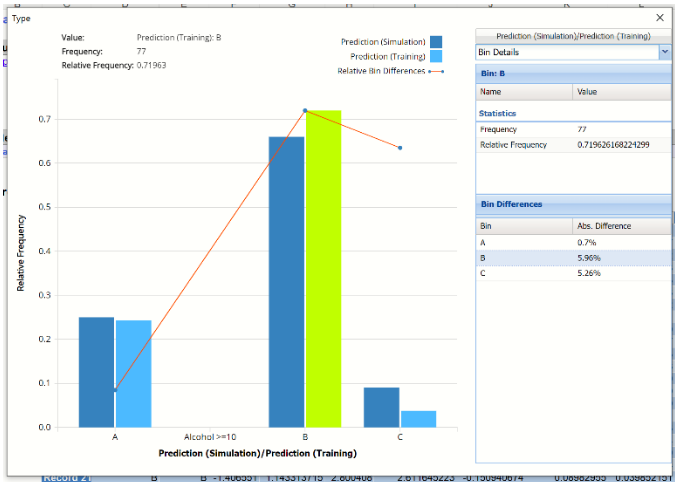


Data

Select Data

Included	Variable	Type
☑	Prediction (Simulation)	categorical
☑	Prediction (Training)	categorical
□	Expression (Simulation)	categorical
□	Expression (Training)	categorical

Prediction (Simulation)/ Prediction (Training) Frequency Chart



Type

Value: Prediction (Training): B
Frequency: 77
Relative Frequency: 0.71963

Prediction (Simulation)/Prediction (Training)

Bin Details

Bin: B

Name	Value
Statistics	
Frequency	77
Relative Frequency	0.719626168224299

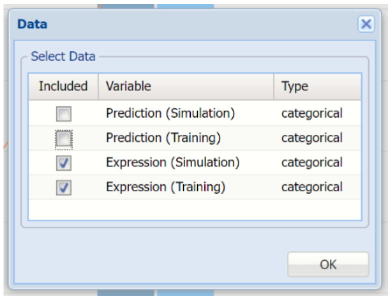
Bin Differences

Bin	Abs. Difference
A	0.7%
B	5.96%
C	5.26%

In this chart, the columns in the darker shade of blue relate to the predicted wine type in the synthetic, or simulated data. The columns in the lighter shade of blue relate to the predicted wine type in the training partition.

Note the red Relative Bin Differences curve. Click the arrow next to Frequency and select Bin Details from the menu. This tab reports the absolute differences between each bin in the chart.

Click *Prediction (Simulation)/Prediction (Training)* and select *Expression (Simulation) and Expression (Training)* in the Data dialog to display both a chart of the results for the expression that was entered in the Simulation tab.

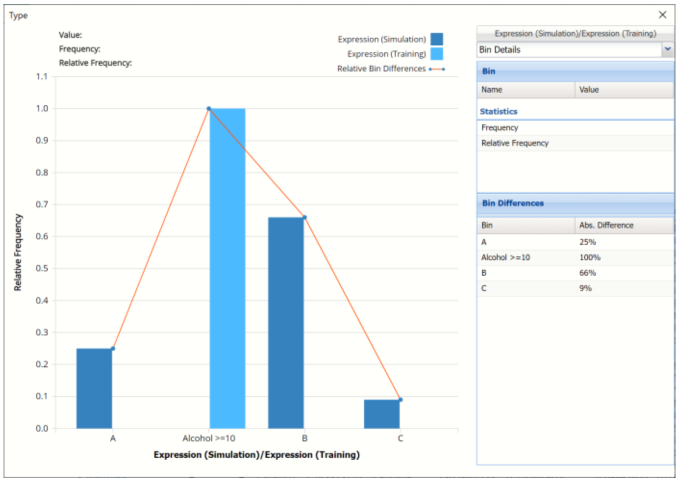


Data

Select Data

Included	Variable	Type
□	Prediction (Simulation)	categorical
□	Prediction (Training)	categorical
☑	Expression (Simulation)	categorical
☑	Expression (Training)	categorical

Chart displaying evaluation of expression



Bin

Name	Value

Statistics

Frequency

Relative Frequency

Bin Differences

Bin	Abs. Difference
A	25%
Alcohol >=10	100%
B	66%
C	9%

The columns in darker blue display the wine type for each record in the simulated, or synthetic, data. In the simulated data, 25% of the records in the data were assigned to type A, 66% were assigned to type B and 9% were assigned to type C. There were no records in the simulated data where the alcohol content was less than 10. As a result, the value for Expression for all records in the synthetic data are labeled as "Alcohol >= 10".

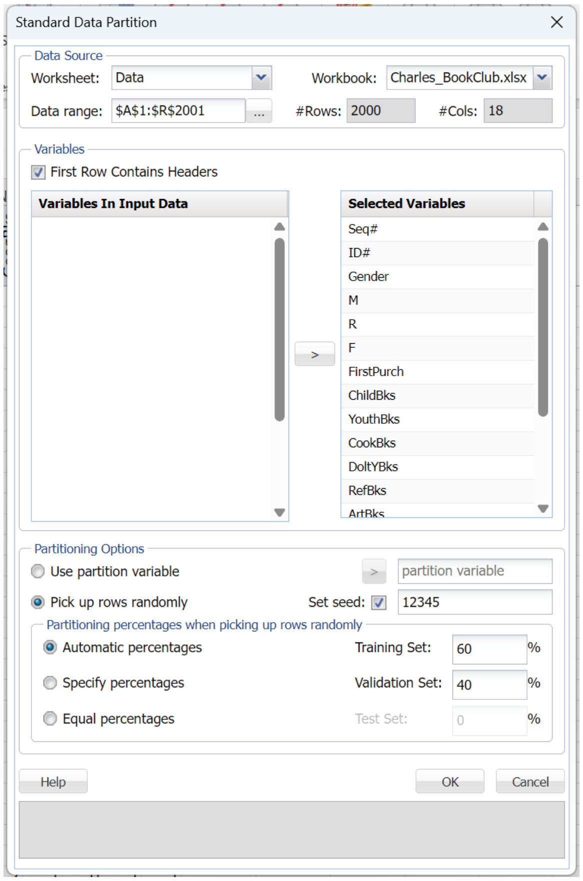
Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

For information on Stored Model Sheets, in this example *DA_Stored*, please refer to the "Scoring New Data" chapter within the Analytic Solver Data Science User Guide.

Automated NNC with 2 Classes in Output Variable

The Error Report for an automated neural network for a dataset with 2 classes in the output variable will look slightly different. Open the file **Charles_BookClub.xlsx** by clicking **Help – Example Models** on the Data Science ribbon, then **Forecasting/Data Science Examples.** This dataset contains data related to book purchases, customer demographics and purchase history as recorded by real-world book seller.

First, we partition the data into training and validation sets using the Standard Data Partition defaults of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.



Standard Data Partition

Data Source

Worksheet: Data

Workbook: Charles_BookClub.xlsx

Data range: $A$1:$R$2001

#Rows: 2000

#Cols: 18

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	Seq#
	ID#
	Gender
	M
	R
	F
	FirstPurch
	ChildBks
	YouthBks
	CookBks
	DoltYBks
	RefBks
	ArtBks

Partitioning Options

○ Use partition variable   partition variable

● Pick up rows randomly   Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages   Training Set: 60 %

○ Specify percentages   Validation Set: 40 %

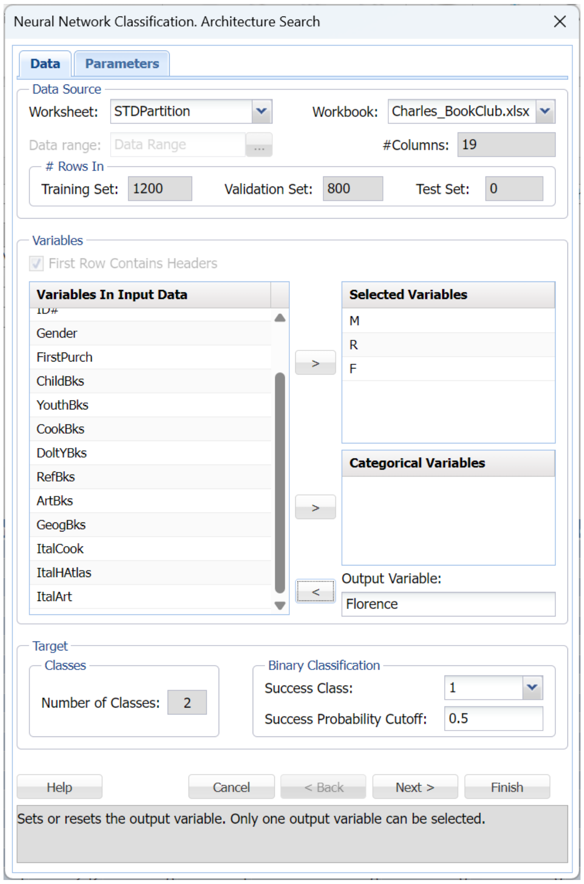
○ Equal percentages   Test Set: 0 %

Select a cell on the *STDPartition* worksheet, then click **Classify – Neural Network – Automatic Network** on the Data Science ribbon. Select **Florence** (has 2 classes) for the *Output variable* and the M, R, and F variables as *Selected variables.*

The *Number of Classes* statistic will be automatically updated with a value of *2* when the Output Variable is selected. This indicates that the Output variable, *CAT.MEDV*, contains two classes, 0 and 1.

Choose the value that will be the indicator of "Success" by clicking the down arrow next to *Success Class*. In this example, we will use the default of 1 indicating that a value of "1" will be specified as a "success".

Enter a value between 0 and 1 for *Success Probability Cutoff*. If the Probability of success (probability of the output variable = 1) is less than this value, then a 0 will be entered for the class value, otherwise a 1 will be entered for the class value. In this example, we will keep the default of 0.5.



Neural Network Classification. Architecture Search

Data Source

Worksheet: STDPartition

Workbook: Charles_BookClub.xlsx

Data range: Data Range

#Columns: 19

Rows In

Training Set: 1200

Validation Set: 800

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data

- ID#
- Gender
- FirstPurch
- ChildBks
- YouthBks
- CookBks
- DoltYBks
- RefBks
- ArtBks
- GeogBks
- ItalCook
- ItalHAtlas
- ItalArt

Selected Variables

- M
- R
- F

Categorical Variables

Output Variable: Florence

Target

Classes

Number of Classes: 2

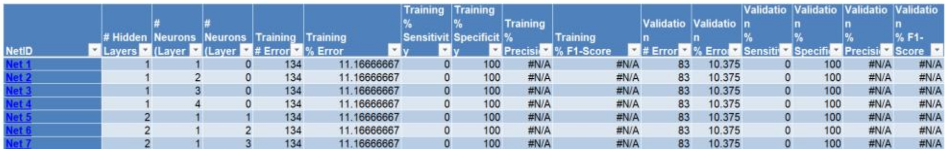
Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

Sets or resets the output variable. Only one output variable can be selected.

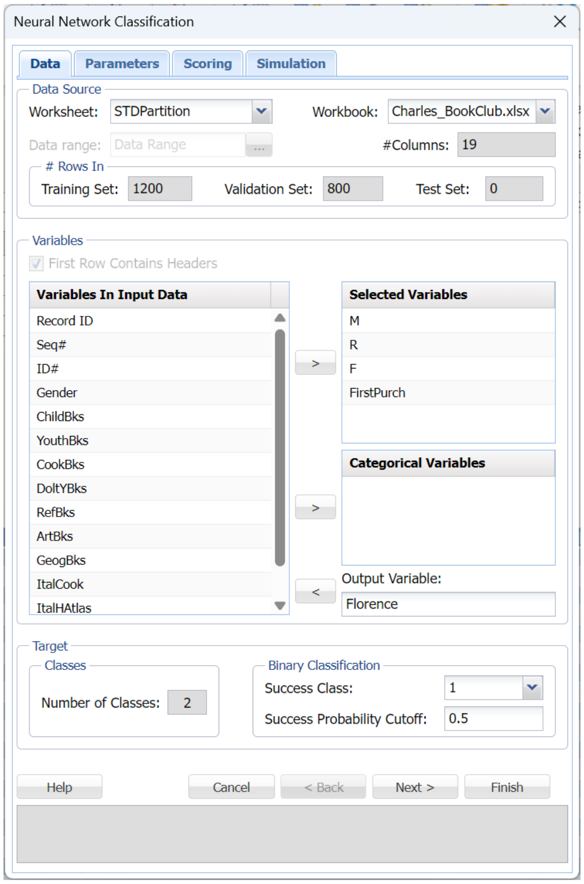
Click **Finish** to accept the default settings for all parameters. Open the sheet, *NNC_Output*, which will be inserted to the right of the STDPartition sheet. Scroll down to the Architecture Search Error Log.



NetID	# Hidden Layers	# Neurons (Layer	# Neurons (Layer	Training # Error	Training % Error	Training % Sensitivity	Training % Specificity	Training % Precisi	Training % F1-Score	Validation # Error	Validation % Erro	Validation % Sensiti	Validation % Specifi	Validation % Precisi	Validation % F1-Score
Net 1	1	1	0	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A
Net 2	1	2	0	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A
Net 3	1	3	0	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A
Net 4	1	4	0	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A
Net 5	2	1	1	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A
Net 6	2	1	2	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A
Net 7	2	1	3	134	11.16666667	0	100	#N/A	#N/A	83	10.375	0	100	#N/A	#N/A

The above error report gives the total number of errors, % Error, % Sensitivity (also known as true positive rate) and % Specificity (also known as true negative rate) in the classification produced by each network ID for the training and validation datasets separately. As shown in the Automatic Neural Network Classification section above, this report may be sorted by column by clicking the arrow next to each column heading. In addition, click the Net ID hyperlinks to re-run the Neural Network Classification method with Manual Architecture with the input and option settings as specified in the specific Net ID.

Let's take a look at Net ID 5. This network has two hidden layers, each containing 1 nodes. For this neural network, the percentage of errors in the training data is 11.16% and the percentage of errors in the validation data is 10.375%. Click the Net5 link in cell C68 to open the Nueral Network Classification dialog. Notice that the Selected Variables and Output Variable have been prefilled in the Data tab. If you click Next, you will find that all Parameters have also been prefilled in the Parameters tab. Click Finish to fit the model.



Neural Network Classification

Data Source

Worksheet: STDPartition   Workbook: Charles_BookClub.xlsx

Data range: Data Range   #Columns: 19

Rows In

Training Set: 1200   Validation Set: 800   Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
Seq#
ID#
Gender
ChildBks
YouthBks
CookBks
DoItYBks
RefBks
ArtBks
GeogBks
ItalCook
ItalHAtlas

Selected Variables
M
R
F
FirstPurch

Categorical Variables

Output Variable: Florence

Target

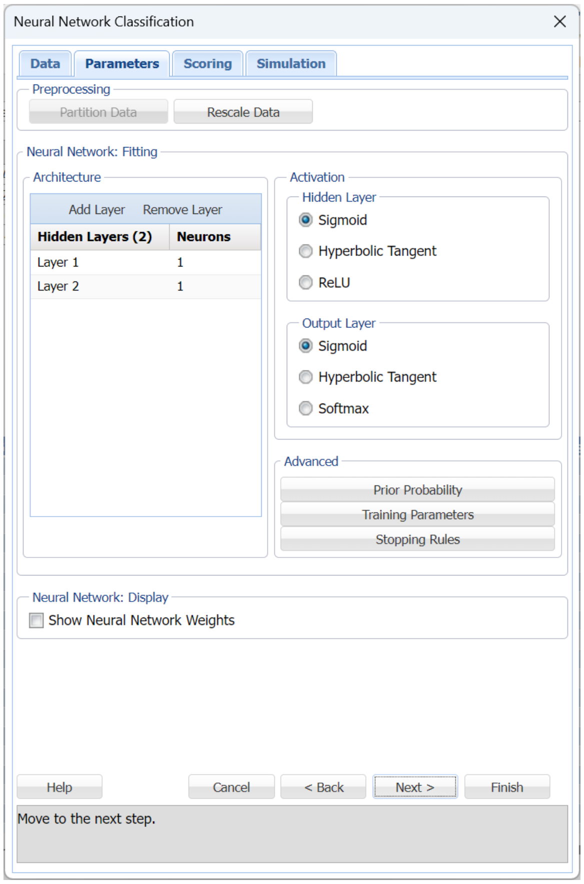
Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5



Neural Network Classification

Neural Network: Fitting

Architecture

Hidden Layers (2)	Neurons
Layer 1	1
Layer 2	1

Activation

Hidden Layer

● Sigmoid

○ Hyperbolic Tangent

○ ReLU

Output Layer

● Sigmoid

○ Hyperbolic Tangent

○ Softmax

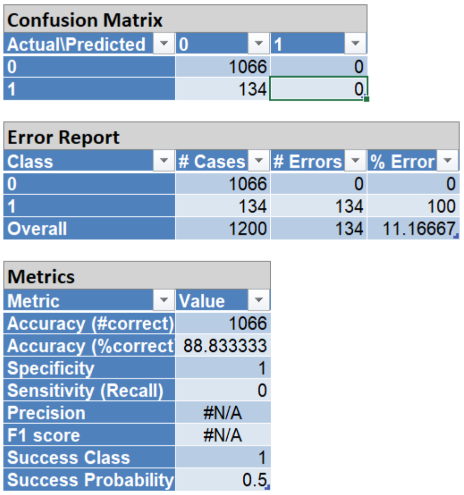
Advanced

Neural Network: Display

□ Show Neural Network Weights

Move to the next step.

Training: Classification Summary

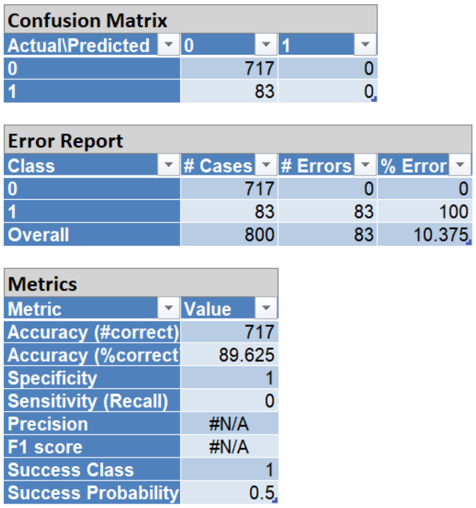


Confusion Matrix		
Actual\Predicted	0	1
0	1066	0
1	134	0

Error Report			
Class	# Cases	# Errors	% Error
0	1066	0	0
1	134	134	100
Overall	1200	134	11.16667

Metrics	
Metric	Value
Accuracy (#correct)	1066
Accuracy (%correct	88.833333
Specificity	1
Sensitivity (Recall)	0
Precision	#N/A
F1 score	#N/A
Success Class	1
Success Probability	0.5

Validation: Classification Summary



Confusion Matrix		
Actual\Predicted	0	1
0	717	0
1	83	0

Error Report			
Class	# Cases	# Errors	% Error
0	717	0	0
1	83	83	100
Overall	800	83	10.375

Metrics	
Metric	Value
Accuracy (#correct)	717
Accuracy (%correct	89.625
Specificity	1
Sensitivity (Recall)	0
Precision	#N/A
F1 score	#N/A
Success Class	1
Success Probability	0.5

The confusion matrices for both partitions are above. Notice the #N/A errors for both Precision and F1 Score in both. Recall that Precision is calculated as True Positives / (True Positives + False Positives) or, in the case of the training partition, 0/(0 + 0). The F1 Score is calculated as 2* (Precision * Recall)/(Precision + Recall), since Precision resulted in an error, F1 also results in an error.

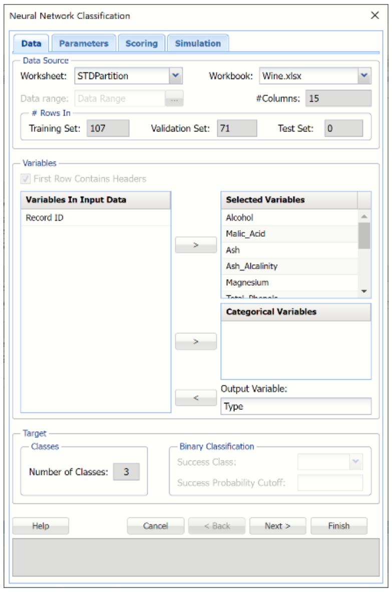
Neural Network Classification Method Options

The options below appear on one of the *Neural Network Classification* dialogs. The Neural Network Classification – Automatic Data tab is identical to the Neural Network Classification Data tab.

Neural Network Classification, Data Tab

See below for option descriptions on the *Neural Network Classification - Data* tab.

Neural Network Classification, Data Tab



Worksheet: STDPartition
Workbook: Wine.xlsx
Data range: Data Range
#Columns: 15
Training Set: 107
Validation Set: 71
Test Set: 0
☑ First Row Contains Headers

Variables In Input Data	Selected Variables
Record ID	Alcohol
	Malic_Acid
	Ash
	Ash_Alcalinity
	Magnesium

Output Variable: Type
Number of Classes: 3
Success Class:
Success Probability Cutoff:

Variables In Input Data

The variables included in the dataset appear here.

Selected Variables

Variables selected to be included in the output appear here.

Categorical Variables

Place categorical variables from the Variables listbox to be included in the model by clicking the > command button. The Neural Network Classification algorithm will accept non-numeric categorical variables.

Output Variable

The dependent variable or the variable to be classified appears here.

Number of Classes

Displays the number of classes in the Output variable.

Success Class

This option is selected by default. Click the drop down arrow to select the value to specify a “success”. This option is only enabled when the # of classes is equal to 2.

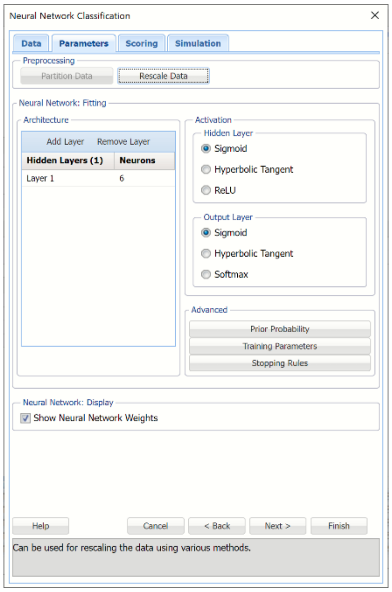
Success Probability Cutoff

Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a “success” (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a “non-success” (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

Neural Network Classification, Parameters Tab

See below for option descriptions on the *Neural Network Classification - Parameters* tab. Note: The *Neural Network Automatic Classification – Parameters* tab does not include Architecture, but is otherwise the same.

Neural Network Classification, Parameters Tab



Can be used for rescaling the data using various methods.

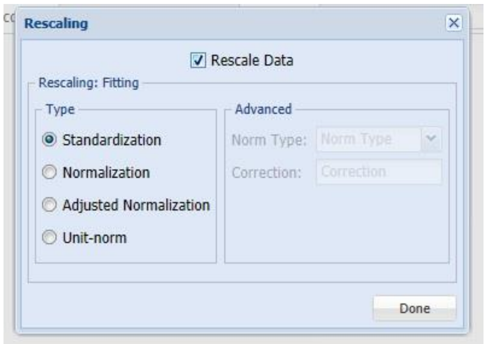
Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Rescale Data

Click **Rescale Data** to open the Rescaling dialog.

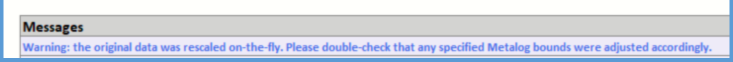
On-the-fly Rescaling Dialog



Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Notes on Rescaling and the Simulation functionality

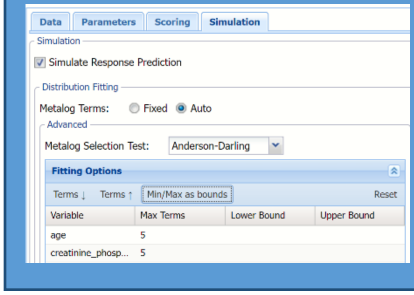
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then "Min/Max as bounds" on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the NNC_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the "Min/Max as bounds" button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the "Min/Max as bounds" feature is available for the user's convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Hidden Layers/Neurons

Click Add Layer to add a hidden layer. To delete a layer, click Remove Layer. Once the layer is added, enter the desired Neurons.

Hidden Layer

Nodes in the hidden layer receive input from the input layer. The output of the hidden nodes is a weighted sum of the input values. This weighted sum is computed with weights that are initially set at random values. As the network "learns", these weights are adjusted. This weighted sum is used to compute the hidden node's output using a *transfer function*. The default selection is *Sigmoid.*

Select *Sigmoid* (the default setting) to use a logistic function for the transfer function with a range of 0 and 1. This function has a "squashing effect" on very small or very large values but is almost linear in the range where the value of the function is between 0.1 and 0.9.

Select *Hyperbolic Tangent* to use the tanh function for the transfer function, the range being -1 to 1. If more than one hidden layer exists, this function is used for all layers.

ReLU (Rectified Linear Unit) is a widely used choice for hidden layers. This function applies a max(0,x) function to the neuron values. When used instead of logistic sigmoid or hyperbolic tangent activations, some adjustments to the Neural Network settings are typically required to achieve a good performance, such as: significantly decreasing the learning rate, increasing the number of learning epochs and network parameters.

Output Layer

As in the hidden layer output calculation (explained in the above paragraph), the output layer is also computed using the same transfer function as described for *Activation: Hidden Layer*. The default selection is *Sigmoid*.

Select *Sigmoid* (the default setting) to use a logistic function for the transfer function with a range of 0 and 1.

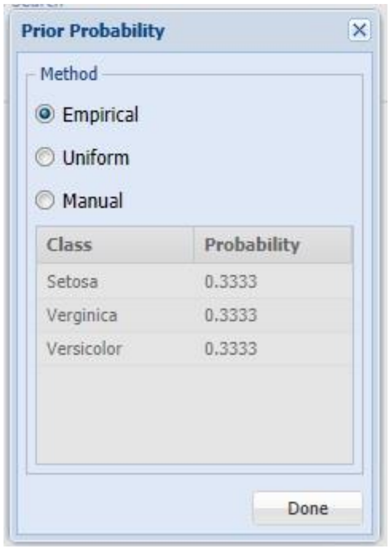
Select *Hyperbolic Tangent* to use the tanh function for the transfer function, the range being **-**1 to 1.

In neural networks, the *Softmax* function is often implemented at the final layer of a classification neural network to impose the constraints that the posterior probabilities for the output variable must be >= 0 and <= 1 and sum to 1. Select *Softmax* to utilize this function.

Prior Probability

Click **Prior Probability**. Three options appear in the *Prior Probability* Dialog: *Empirical, Uniform* and *Manual.*

Prior Probability Dialog



● Empirical

○ Uniform

○ Manual

Class	Probability
Setosa	0.3333
Verginica	0.3333
Versicolor	0.3333

If the first option is selected, *Empirical*, Analytic Solver Data Science will assume that the probability of encountering a particular class in the dataset is the same as the frequency with which it occurs in the training data.

If the second option is selected, *Uniform*, Analytic Solver Data Science will assume that all classes occur with equal probability.

Select the third option, *Manual*, to manually enter the desired class and probability value.

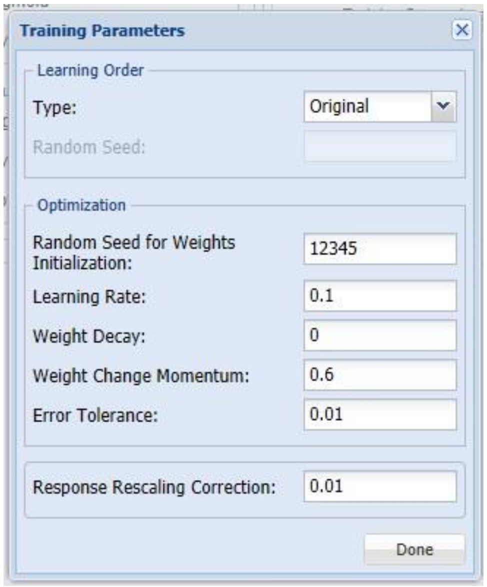
Neuron Weight Initialization Seed

If an integer value appears for *Neuron weight initialization seed,* Analytic Solver Data Science will use this value to set the neuron weight random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the neuron weight are calculated. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Training Parameters

Click Training Parameters to open the Training Parameters dialog to specify parameters related to the training of the Neural Network algorithm.

Training Parameters Dialog



Training Parameters

Learning Order

Type: Original

Random Seed:

Optimization

Random Seed for Weights Initialization: 12345

Learning Rate: 0.1

Weight Decay: 0

Weight Change Momentum: 0.6

Error Tolerance: 0.01

Response Rescaling Correction: 0.01

Learning Order [Original or Random]

This option specifies the order in which the records in the training dataset are being processed. It is recommended to shuffle the training data to avoid the possibility of processing correlated reocrds in order. It also helps the neural network algorithm to converge faster. If Random is selected, Random Seed is enabled. If Original is selected, the algorithm will use the original order of records.

Learning Order [Random Seed]

This option specifies the seed for shuffling the training records. Note that different random shuffling may lead to different results, but as long as the training data is shuffled, different ordering typically does not result in drastic changes in performance.

Random Seed for Weights Initialization

If an integer value appears for *Random Seed for Weights Initialization*, Analytic Solver Data Science will use this value to set the seed for the initial assignment of the neuron values. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the neuron values are calculated. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box.

Learning Rate

This is the multiplying factor for the error correction during backpropagation; it is roughly equivalent to the learning rate for the neural network. A low value produces slow but steady learning, a high value produces rapid but erratic learning. Values for the step size typically range from 0.1 to 0.9.

Weight Decay

To prevent over-fitting of the network on the training data, set a weight decay to penalize the weight in each iteration. Each calculated weight will be multiplied by (1-decay).

Weight Change Momentum

In each new round of error correction, some memory of the prior correction is retained so that an outlier that crops up does not spoil accumulated learning.

Error Tolerance

The error in a particular iteration is backpropagated only if it is greater than the error tolerance. Typically error tolerance is a small value in the range from 0 to 1.

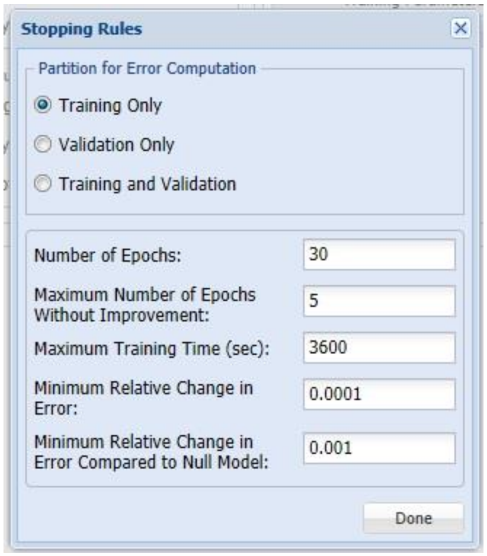
Response Rescaling Correction

This option specifies a small number, which is applied to the Normalization rescaling formula, if the output layer activation is Sigmoid (or Softmax in Classification), and Adjusted Normalization, if the output layer activation is Hyperbolic Tangent. The rescaling correction ensures that all response values stay within the range of activation function.

Stopping Rules

Click Stopping Rules to open the Stopping Rules dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error.

Stopping Rules Dialog



Partition for Error Computation

● Training Only

○ Validation Only

○ Training and Validation

Number of Epochs: 30

Maximum Number of Epochs Without Improvement: 5

Maximum Training Time (sec): 3600

Minimum Relative Change in Error: 0.0001

Minimum Relative Change in Error Compared to Null Model: 0.001

Partition for Error Computation

Specifies which data partition is used to estimate the error after each training epoch.

Number of Epochs

An epoch is one sweep through all records in the training set. Use this option to set the number of epochs to be performed by the algorithm.

Maximum Number of Epochs Without Improvement

The algorithm will stop after this number of epochs has been completed, and no improvement has ben realized.

Maximum Training Time

The algorithm will stop once this time (in seconds) has been exceeded.

Keep Minimum Relative Change in Error

If the relative change in error is less than this value, the algorithm will stop.

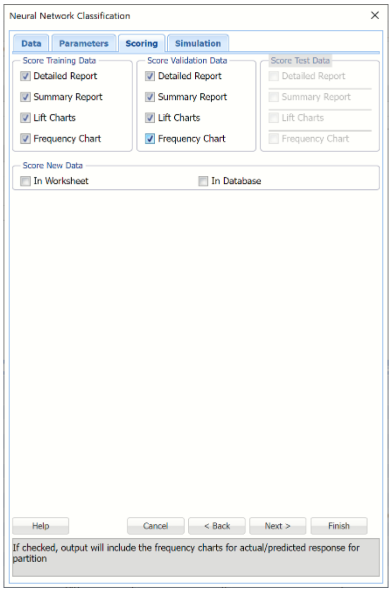
Keep Minimum Relative Change in Error Compared to Null Model

If the relative change in error compared to the Null Model is less than this value, the algorithm will stop. Null Model is the baseline model used for comparing the performance of the neural network model.

Neural Network Classification, Scoring tab

See below for option descriptions on the *Neural Network Classification - Scoring* tab.

Neural Network Classification, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet □ In Database

If checked, output will include the frequency charts for actual/predicted response for partition

When Frequency Chart is selected, a frequency chart will be displayed when the NNC_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Training Data

Select these options to show an assessment of the performance of the algorithm in classifying the training data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

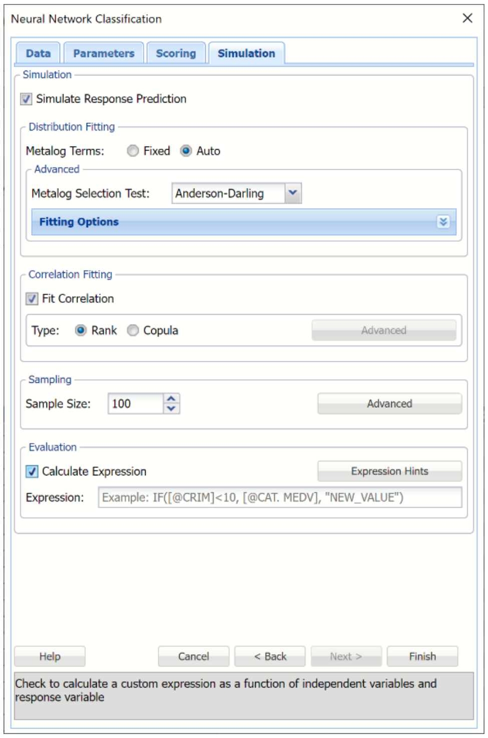
Score Validation Data

These options are enabled when a validation dataset is present. Select these options to show an assessment of the performance of the algorithm in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Score Test Data

These options are enabled when a test dataset is present. Select these options to show an assessment of the performance of the algorithm in classifying the test data. The report is displayed according to your specifications - Detailed, Summary, Lift charts and Frequency. Lift charts are only available when the *Output Variable* contains 2 categories.

Neural Network Classification, Simulation tab



Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula Advanced

Sampling
Sample Size: 100 Advanced

Evaluation
☑ Calculate Expression Expression Hints
Expression: Example: IF([@CRIM]<10, [@CAT. MEDV], "NEW_VALUE")

Check to calculate a custom expression as a function of independent variables and response variable

Score New Data

For information on scoring in a worksheet or database, please see the chapters "Scoring New Data" and "Scoring Test Data" in the Analytic Solver Data Science User Guide.

Neural Neighbors Classification, Simulation

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, NNC_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the NNC_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as

[@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Ensemble Methods for Classification

Analytic Solver Data Science offers two powerful ensemble methods for use with all classification methods: bagging (bootstrap aggregating) and boosting. A third method, random trees, may only be applied to classification tree. Each classification method on their own can be used to find one model that results in good classifications of the new data. We can view the statistics and confusion matrices of the current classifier to see if our model is a good fit to the data, but how would we know if there is a better classifier just waiting to be found? The answer is – we don't. However, ensemble methods allow us to combine multiple "weak" classification models which, when taken together form a new, more accurate "strong" classification model. These methods work by creating multiple diverse classification models, by taking different samples of the original dataset, and then combining their outputs. (Outputs may be combined by several techniques for example, majority vote for classification and averaging for regression. This combination of models effectively reduces the variance in the "strong" model. The three different types of ensemble methods offered in Analytic Solver Data Science (bagging, boosting, and random trees) differ on three items: 1.The selection of training data for each classifier or "weak" model, 2.How the "weak" models are generated and 3. How the outputs are combined. In all three methods, each "weak" model is trained on the entire training dataset to become proficient in some portion of the dataset.

Bagging, or bootstrap aggregating, was one of the first ensemble algorithms ever to be written. It is a simple algorithm, yet very effective. Bagging generates several training data sets by using random sampling with replacement (bootstrap sampling), applies the classification algorithm to each dataset, then takes the majority vote amongst the models to determine the classification of the new data. The biggest advantage of bagging is the relative ease that the algorithm can be parallelized which makes it a better selection for very large datasets.

Boosting, in comparison, builds a "strong" model by successively training models to concentrate on the misclassified records in previous models. Once completed, all classifiers are combined by a weighted majority vote. Analytic Solver Data Science offers three different variations of boosting as implemented by the AdaBoost algorithm (one of the most popular ensemble algorithms in use today): M1 (Freund), M1 (Breiman), and SAMME (Stagewise Additive Modeling using a Multi-class Exponential).

Adaboost.M1 first assigns a weight ($w_b(i)$) to each record or observation. This weight is originally set to 1/n and will be updated on each iteration of the algorithm. An original classification model is created using this first training set (T_b) and an error is calculated as:

$$e_b = \sum_{i-1}^{n} w_b(i) I(C_b(x_i) \neq y_i))$$

where the I() function returns 1 if true and 0 if not.

The error of the classification model in the bth iteration is used to calculate the constant α_b. This constant is used to update the weight $w_b(i)$. In AdaBoost.M1 (Freund), the constant is calculated as:

$$\alpha_b = \ln((1-e_b)/e_b)$$

In AdaBoost.M1 (Breiman), the constant is calculated as:

$$\alpha_b = 1/2\ln((1-e_b)/e_b)$$

In SAMME, the constant is calculated as:

$\alpha_b = 1/2\ln((1-e_b)/e_b + \ln(k-1)$ where k is the number of classes

(When the number of categories is equal to 2, SAMME behaves the same as AdaBoost Breiman.)

In any of the three implementations (Freund, Breiman, or SAMME), the new weight for the (b + 1)th iteration will be

$$w_{b+1}(i) = w_b(i)\exp(\alpha_b I(C_b(x_i) \neq y_i))$$

Afterwards, the weights are all readjusted to sum to 1. As a result, the weights assigned to the observations that were classified incorrectly are increased and the weights assigned to the observations that were classified correctly are decreased. This adjustment forces the next classification model to put more emphasis on the records that were misclassified. (This α constant is also used in the final calculation which will give the classification model with the lowest error more influence.) This process repeats until b = Number of weak learners (controlled by the User). The algorithm then computes the weighted sum of votes for each class and assigns the "winning" classification to the record. Boosting generally yields better models than bagging, however, it does have a disadvantage as it is not parallelizable. As a result, if the number of weak learners is large, boosting would not be suitable.

Random trees, also known as random forests, is a variation of bagging. This method works by training multiple "weak" classification trees using a fixed number of randomly selected features (sqrt[number of features] for classification and number of features/3 for prediction) then takes the mode of each class to create a "strong" classifier. Typically, in this method the number of "weak" trees generated could range from several hundred to several thousand depending on the size and difficulty of the training set. Random Trees are parallelizable since they are a variant of bagging. However, since Random Trees selects a limited amount of features in each iteration, the performance of random trees is faster than bagging.

Classification Ensemble methods are very powerful methods and typically result in better performance than a single tree. This feature addition in Analytic Solver Data Science (introduced in V2015) will provide users with more accurate classification models and should be considered.

Bagging Ensemble Method Example

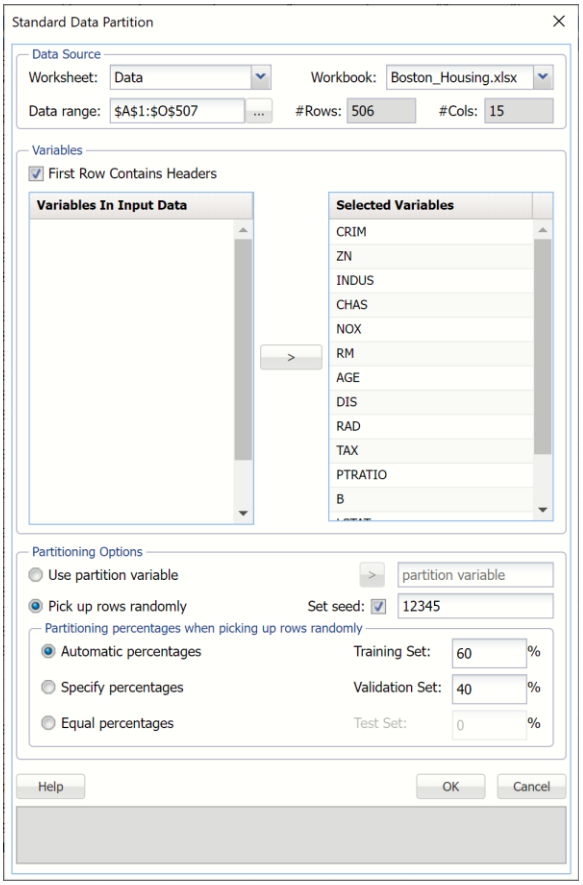
This example illustrates the use of the Bagging Ensemble Classification Method using the Boston Housing dataset. This dataset contains information collected by the US Census Service concerning housing in the Boston, MA area in the 1940's.

1. Click **Help – Example Models**, then **Forecasting/Data Science Examples** to open the **Boston Housing** dataset.

   Descriptions of the "features" or independent variables in this dataset can be found on the Data worksheet tab.

   All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described in the What's New section of this guide and then more in depth in the Analytic Solver Data Science Reference Guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. Since this new functionality does not support categorical input variables, these types of variables will not be present in the model, only continuous, or scale, variables.

2. First, we partition the data into training and validation sets using a Standard Data Partition with percentages of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.



Standard Data Partition

Data Source
Worksheet: Data
Workbook: Boston_Housing.xlsx
Data range: $A$1:$O$507
#Rows: 506
#Cols: 15

Variables
☑ First Row Contains Headers

Variables In Input Data

Selected Variables: CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B

Partitioning Options
○ Use partition variable — partition variable
● Pick up rows randomly — Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly
● Automatic percentages — Training Set: 60 %
○ Specify percentages — Validation Set: 40 %
○ Equal percentages — Test Set: 0 %

3. With the *STDPartition* tab selected, click **Classify – Ensemble – Bagging** to open the Bagging: Classification dialog.

4. Select the following variables under *Variables in Input Data* and then click > next to *Selected Variables* to select these variables as input variables.

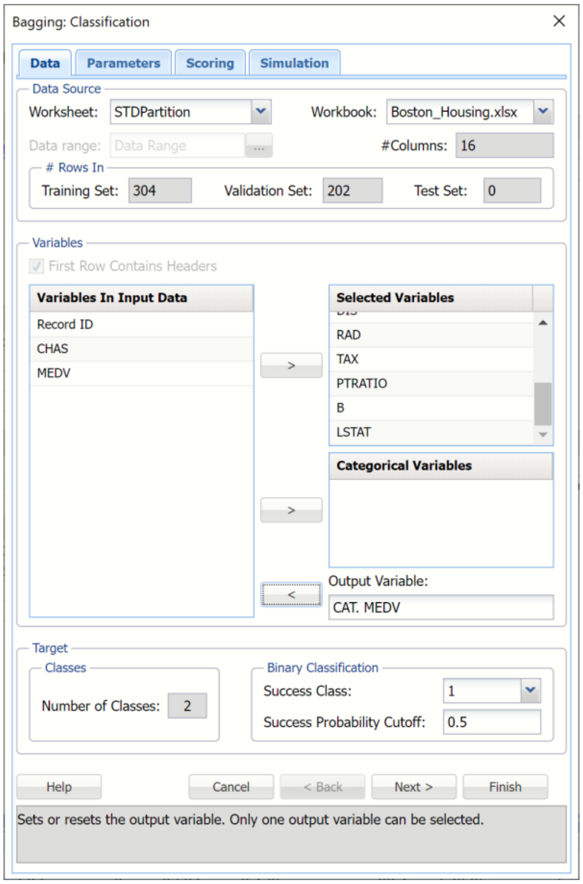
   CRIM, ZN, INDUS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B and **LSTAT**

   Omit Record ID, CHAS and MEDV variables from the input.

5. Select **CAT. MEDV** under *Variables In Input Data*, then click > next to *Output Variable*, to select this variable as the output variable. This variable is derived from the scale MEDV variable.

6. Choose the value that will be the indicator of "Success" by clicking the down arrow next to *Success Class*. In this example, we will use the default of 1.

7. Enter a value between 0 and 1 for *Success Probability Cutoff*. If the Probability of success is less than this value, then a 0 will be entered for the class value, otherwise a 1 will be entered for the class value. In this example, we will keep the default of 0.5.



Bagging: Classification

Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Training Set: 304

Validation Set: 202

Test Set: 0

☑ First Row Contains Headers

Variables In Input Data: Record ID, CHAS, MEDV

Selected Variables: DIS, RAD, TAX, PTRATIO, B, LSTAT

Categorical Variables:

Output Variable: CAT. MEDV

Number of Classes: 2

Success Class: 1

Success Probability Cutoff: 0.5

Sets or resets the output variable. Only one output variable can be selected.

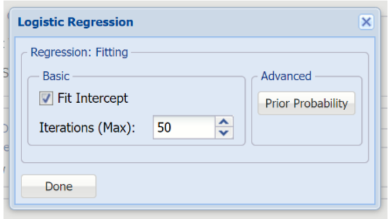
8. Click **Next** to advance to the *Bagging: Classification Parameters* tab.

Analytic Solver Data Science includes the ability to partition or rescale a dataset "on-the-fly" within a classification or regression method by clicking Partition Data or Rescale Data on the Parameters tab. Analytic Solver Data Science will partition or rescale your dataset (according to the partition and rescaling options you set) immediately before running the classification method. If partitioning or rescaling has already occurred on the dataset, these options will be disabled. For more information on partitioning or rescaling your data, please see the Data Science Partitioning and Transform Continuous Data chapters that occur earlier in this guide.

9. Leave the default value of "10" for the *Number of weak learners*. This option controls the number of "weak" classification models that will be created. The ensemble method will stop when the number or classification models created reaches the value set for this option. The algorithm will then compute the weighted sum of votes for each class and assign the "winning" classification to each record.

10. Under Ensemble: Classification click the down arrow beneath Weak Learner to select one of the six featured classifiers. For this example, select Logistic Regression.

    Options pertaining to the Logistic Regression learner may be changed by clicking the Logistic Regression button to the right of Weak Learner.



Logistic Regression

Regression: Fitting

Basic

☑ Fit Intercept

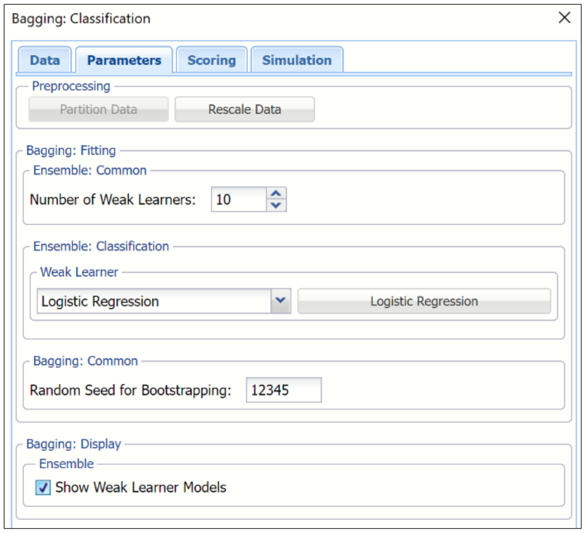
Iterations (Max): 50

Advanced

    All options will be left at their default values. For more information on these options, see the Logistic Regression chapter that occurs earlier in this guide.

11. Leave the default setting for Random Seed for Boostrapping at "12345". Analytic Solver Data Science will use this value to set the bootstrapping random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier.

12. Select Show Weak Learner Models to display the weak learner models in the output.



Bagging: Classification

Preprocessing

Bagging: Fitting

Ensemble: Common

Number of Weak Learners: 10

Ensemble: Classification

Weak Learner

Logistic Regression

Bagging: Common

Random Seed for Bootstrapping: 12345

Bagging: Display

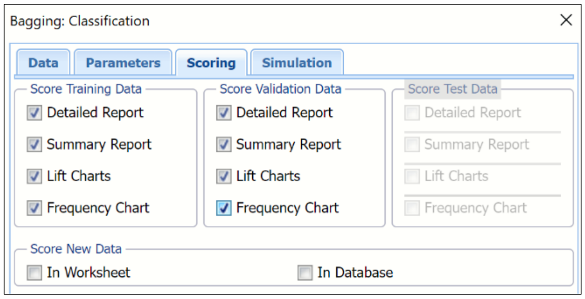
Ensemble

☑ Show Weak Learner Models

13. Click **Next** to advance to the *Bagging: Classification - Scoring* tab.

14. *Summary Report* is selected by default under both Score Training Data and Score Validation Data.

    - Select **Detailed Report** under both *Score Training Data* and *Score Validation Data* to produce a detailed assessment of the performance of the tree in both sets.
    - Select **Lift Charts** to include Lift Charts, ROC Curves and Decile charts for both the Training and Validation datasets.
        - Select **Frequency Chart** under *Score Training/Validation Data* to display a frequency chart on both the CBagging_TrainingScore and CBagging_ValidationScore worksheets. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.
        - Since we did not create a test partition, the options for *Score test data* are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.
        - See the "Scoring New Data" chapter within the Analytic Solver Data Science User Guide for information on the *Score new data* options.



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

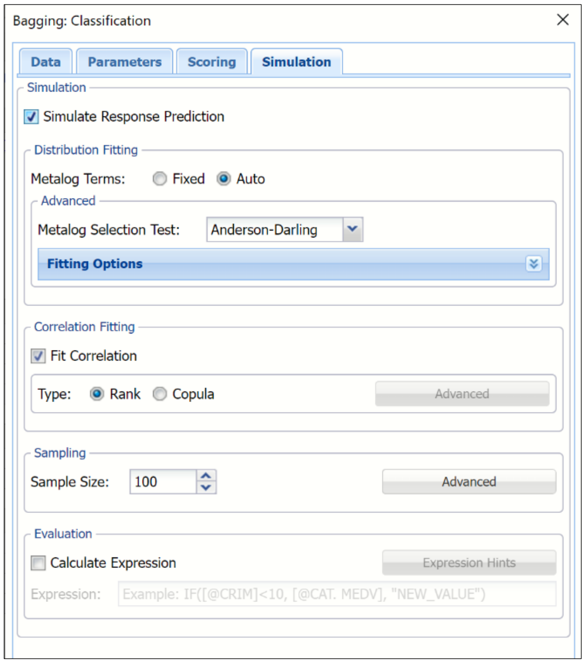
Score New Data
□ In Worksheet
□ In Database

15. Click **Next** to advance to the Simulation tab. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

    Select **Simulation Response Prediction** to enable all options on the Simulation tab.

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CBagging_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Bagging Classification dialog, Simulation tab



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

☑ Fit Correlation

Type: ● Rank ○ Copula

Sample Size: 100

□ Calculate Expression

Expression: Example: IF([@CRIM]<10, [@CAT. MEDV], "NEW_VALUE")

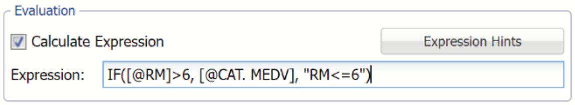
Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the CBagging_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog. For Expression, enter the following formula to display if the patient suffered catastrophic heart failure (@DEATH_EVENT) when his/her Ejection_Fraction was less than or equal to 20.

IF([@RM]>=6, [@CAT. MEDV], “RM<=6”)

Note that variable names are case sensitive.

Evaluation section on the Simulation tab



☑ Calculate Expression

Expression: IF([@RM]>6, [@CAT. MEDV], "RM<=6")

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

16. Click **Finish**.

Output

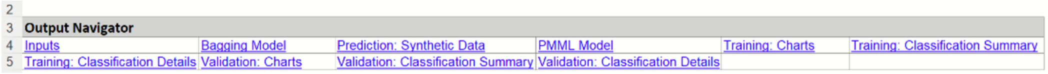
Output from the Bagging algorithm will be inserted to the right of the Data worksheet.

CBagging_Output

This result worksheet includes 4 segments: Output Navigator, Inputs and Bagging Model.

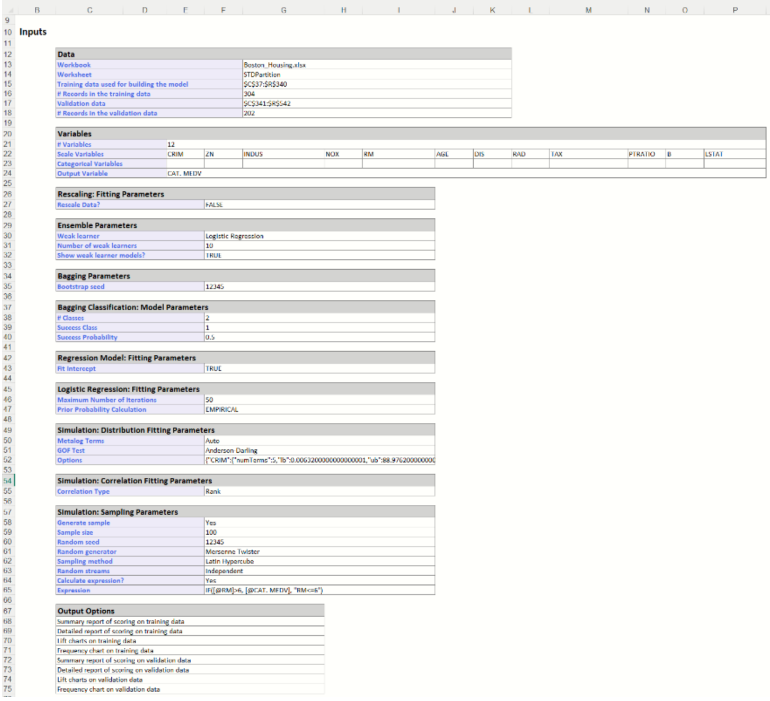
- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

CBagging_Output: Output Navigator



Output Navigator					
Inputs	Bagging Model	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Classification Summary
Training: Classification Details	Validation: Charts	Validation: Classification Summary	Validation: Classification Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Bagging Classification dialog.



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	CAT. MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Ensemble Parameters	
Weak learner	Logistic Regression
Number of weak learners	10
Show weak learner models?	TRUE

Bagging Parameters	
Bootstrap seed	12345

Bagging Classification: Model Parameters	
# Classes	2
Success Class	1
Success Probability	0.5

Regression Model: Fitting Parameters	
Fit Intercept	TRUE

Logistic Regression: Fitting Parameters	
Maximum Number of Iterations	50
Prior Probability Calculation	EMPIRICAL

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.0063200000000000001,"ub":88.97620000000

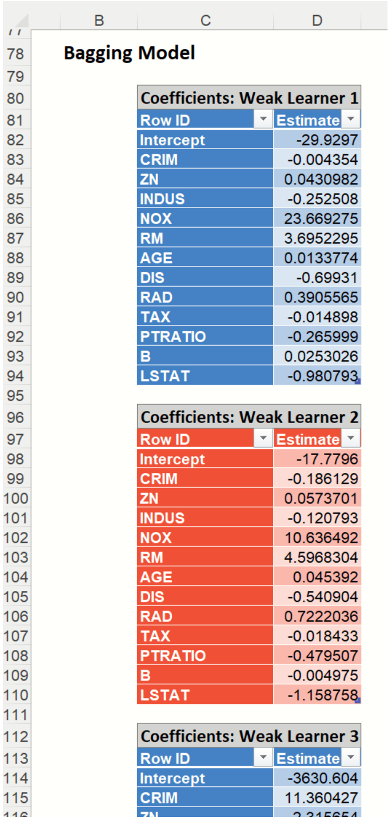
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	Yes
Expression	IF([@RM]>6, [@CAT. MEDV], "RM<=6")

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- The number of Weak Learners in the output is equal to 10 which matches our input on the Parameters tab for the Number of Weak Learners option.

  The Importance percentage for each Variable in each Learner is listed in each table. This percentage measures the variable's contribution in reducing the total misclassification error.



Bagging Model

Coefficients: Weak Learner 1	
Row ID	Estimate
Intercept	-29.9297
CRIM	-0.004354
ZN	0.0430982
INDUS	-0.252508
NOX	23.669275
RM	3.6952295
AGE	0.0133774
DIS	-0.69931
RAD	0.3905565
TAX	-0.014898
PTRATIO	-0.265999
B	0.0253026
LSTAT	-0.980793

Coefficients: Weak Learner 2	
Row ID	Estimate
Intercept	-17.7796
CRIM	-0.186129
ZN	0.0573701
INDUS	-0.120793
NOX	10.636492
RM	4.5968304
AGE	0.045392
DIS	-0.540904
RAD	0.7222036
TAX	-0.018433
PTRATIO	-0.479507
B	-0.004975
LSTAT	-1.158758

Coefficients: Weak Learner 3	
Row ID	Estimate
Intercept	-3630.604
CRIM	11.360427

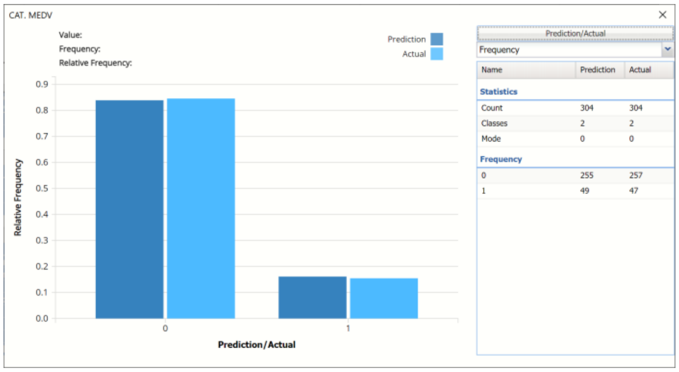
CBagging_TrainingScore

Click the CBagging_TrainingScore Scroll down to view the Classification Summary and Classification Details Reports for the Training partition as well as the Frequency charts. For detailed information on each of these components, see the Logistic Regression chapter that appears earlier in this guide.

- **Frequency Chart:** This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

  Note: To view this frequency charts in the Cloud app, click the Charts icon on the Ribbon, select CBagging_TrainingScore for Worksheet and Frequency for Chart.

Frequency Chart for Training Partition



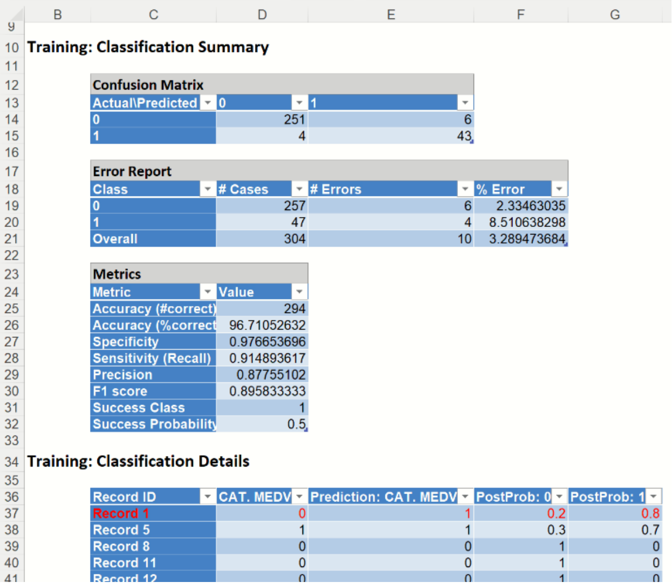
Prediction/Actual

Frequency

Name	Prediction	Actual
Statistics		
Count	304	304
Classes	2	2
Mode	0	0
Frequency		
0	255	257
1	49	47

- **Classification Summary and Classification Details**: In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

Classification Summary and Classification Details Reports



Training: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	251	6
1	4	43

Error Report

Class	# Cases	# Errors	% Error
0	257	6	2.33463035
1	47	4	8.510638298
Overall	304	10	3.289473684

Metrics

Metric	Value
Accuracy (#correct)	294
Accuracy (%correct	96.71052632
Specificity	0.976653696
Sensitivity (Recall)	0.914893617
Precision	0.87755102
F1 score	0.895833333
Success Class	1
Success Probability	0.5

Training: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 0	PostProb: 1
Record 1	0	1	0.2	0.8
Record 5	1	1	0.3	0.7
Record 8	0	0	1	0
Record 11	0	0	1	0
Record 12	0	0	1	0

The Classification Summary displays the confusion matrix for the Training Partition.

- True Positive: 43 records belonging to the Success class were correctly assigned to that class.
- False Negative: 4 records belonging to the Success class were incorrectly assigned to the Failure class.
- True Negative: 251 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 6 records belonging to the Failure class were incorrectly assigned to the Success class.

The total number of misclassified records was 10 (4 + 6) which results in an error equal to 3.29%.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 96.71% **-** Refers to the ability of the classifier to predict a class label correctly.
- Specificity: 0.977 **-** Also called the true negative rate, measures the percentage of failures correctly identified as failures

  Specificity (SPC) or True Negative Rate =TN / (FP + TN)
- Recall (or Sensitivity): 0.914 **-** Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)
- Precision: 0.878 **-** The probability of correctly identifying a randomly selected record as one belonging to the Success class

  Precision = TP/(TP+FP)
- F-1 Score: 0.896 **-** Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP / (2 * TP + FP + FN)
- Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

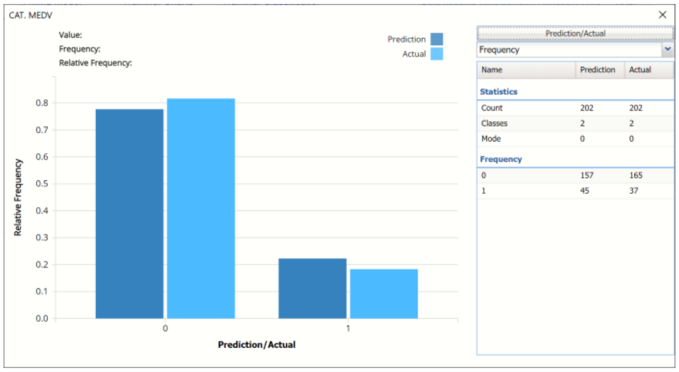
- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

CBagging_ValidationScore

Click the CBagging_ValidationScore Scroll down to view the Classification Summary and Classification Details Reports for the Validation partition as well as the Frequency charts. For detailed information on each of these components, see the Logistic Regression chapter that appears earlier in this guide.

- **Frequency Chart:** This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

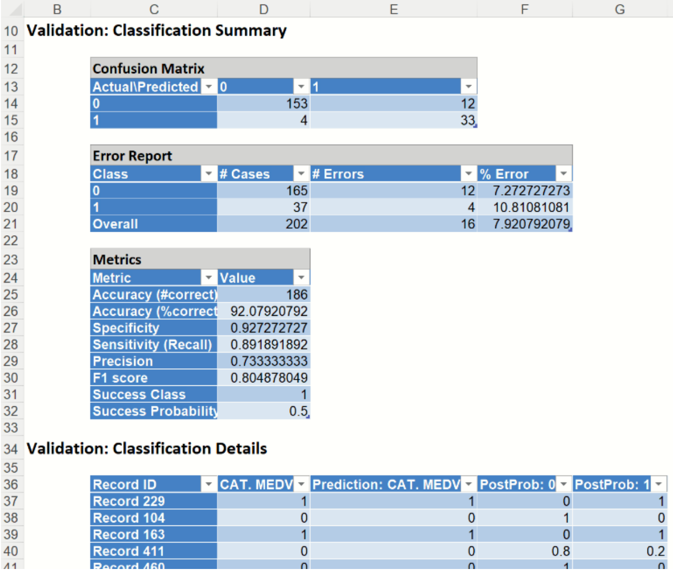
Frequency Chart for Validation Partition



Name	Prediction	Actual
Statistics		
Count	202	202
Classes	2	2
Mode	0	0
Frequency		
0	157	165
1	45	37

- **Classification Summary and Classification Details**: In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the fitted classification model on the validation partition.

Classification Summary and Classification Details Reports



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	153	12
1	4	33

Error Report

Class	# Cases	# Errors	% Error
0	165	12	7.272727273
1	37	4	10.81081081
Overall	202	16	7.920792079

Metrics

Metric	Value
Accuracy (#correct)	186
Accuracy (%correct	92.07920792
Specificity	0.927272727
Sensitivity (Recall)	0.891891892
Precision	0.733333333
F1 score	0.804878049
Success Class	1
Success Probability	0.5

Validation: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 0	PostProb: 1
Record 229	1	1	0	1
Record 104	0	0	1	0
Record 163	1	1	0	1
Record 411	0	0	0.8	0.2
Record 460	0	0	1	0

The Classification Summary displays the confusion matrix for the Training Partition.

- True Positive: 33 records belonging to the Success class were correctly assigned to that class.
- False Negative: 4 records belonging to the Success class were incorrectly assigned to the Failure class.
- True Negative: 153 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 12 records belonging to the Failure class were incorrectly assigned to the Success class.

The total number of misclassified records was 16 (12 + 4) which results in an error equal to 7.92%.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 92.1% **-** Refers to the ability of the classifier to predict a class label correctly.
- Specificity: 0.927 **-** Also called the true negative rate, measures the percentage of failures correctly identified as failures

  Specificity (SPC) or True Negative Rate =TN / (FP + TN)
- Recall (or Sensitivity): 0.892 **-** Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)
- Precision: 0.733 **-** The probability of correctly identifying a randomly selected record as one belonging to the Success class

  Precision = TP/(TP+FP)
- F-1 Score: 0.805 **-** Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP / (2 * TP + FP + FN)
- Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

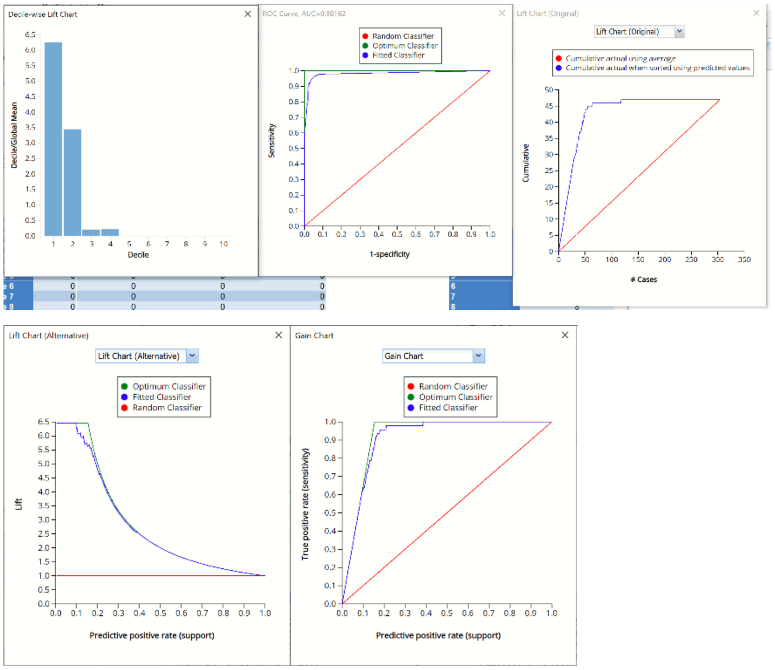
- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

CBagging_TrainingLiftCharts & Bagging_ValidationLiftCharts

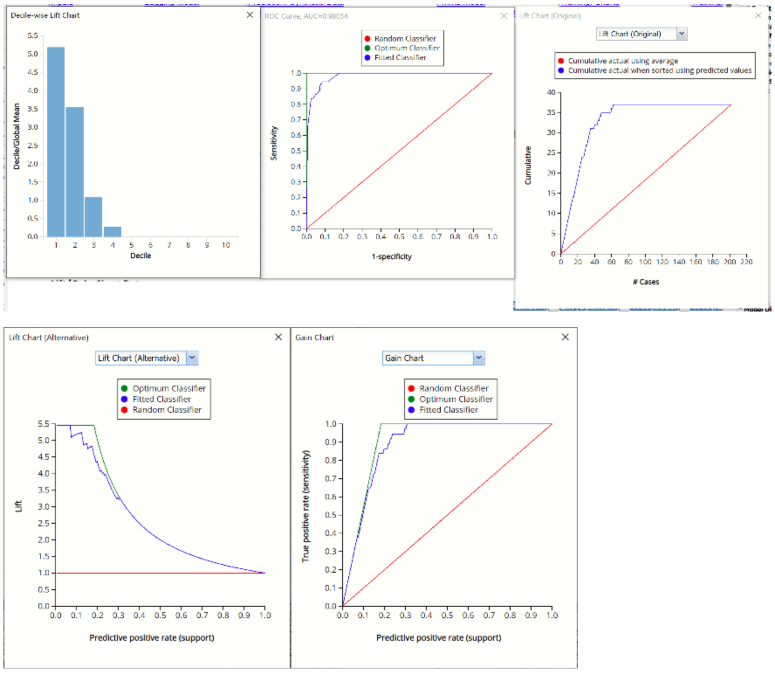
Click the **CBagging_TrainingLiftChart** to navigate to the Lift Charts, shown below. For more information on lift charts, ROC curves, and Decile charts, please see the Logistic Regression chapter that appears previously in this guide.

Lift Charts and ROC Curves are visual aids that help users evaluate the performance of their fitted models. Charts found on the CBagging_TrainingLiftChart tab were calculated using the Training Data Partition. Charts found on the CBagging_ValidationLiftChart tab were calculated using the Validation Data Partition. It is good practice to look at both sets of charts to assess model performance on both the Training and Validation partitions.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select CBagging_TrainingLiftChart or CBagging_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.



Click the **CBagging_ValidationLiftChart** to navigate to the charts, shown below.

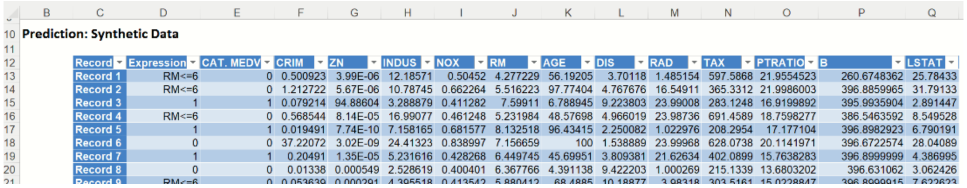


CBagging_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, CBagging_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Bagging Classification dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predictions for the training partition (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



Prediction: Synthetic Data

Record	Expression	CAT. MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	RM<=6	0	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.70118	1.485154	597.5868	21.9554523	260.6748362	25.78433
Record 2	RM<=6	0	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767676	16.54911	365.3312	21.9986003	396.8859965	31.79133
Record 3	1	1	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223803	23.99008	283.1248	16.9199892	395.9935904	2.891447
Record 4	RM<=6	0	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966019	23.98736	691.4589	18.7598277	386.5463592	8.549528
Record 5	1	1	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082	1.022976	208.2954	17.177104	396.8982923	6.790191
Record 6	0	0	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538889	23.99968	628.0738	20.1141971	396.6722574	28.04089
Record 7	1	1	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809381	21.62634	402.0899	15.7638283	396.8999999	4.386995
Record 8	0	0	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422203	1.000269	215.1339	13.6803202	396.631062	3.062426
Record 9	RM<=6	0	0.053639	0.000291	4.395518	0.413542	5.880412	68.4885	10.18877	3.98318	303.5161	15.0228847	396.8999915	7.622623

Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the Bagging Classification dialog

IF([@RM]>6, [@CAT. MEDV], “RM<=6”)

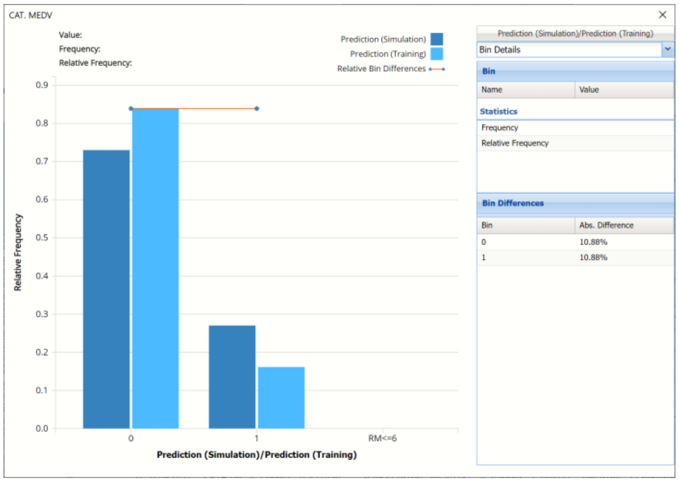
The Expression column will contain each record’s predicted score for the CAT. MEDV variable or the string, “RM<=6”.

The remainder of the data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the predicted values for the output variable in the training partition, the synthetic data and the expression, if it exists.

In the screenshot below, the bars in the darker shade of blue are based on the Prediction, or synthetic, data as generated in the table above for the CAT. MEDV variable. The bars in the lighter shade of blue display the frequency of the predictions for the CAT. MEDV variable in the training partition.

Frequency Chart for CBagging_Simulation output



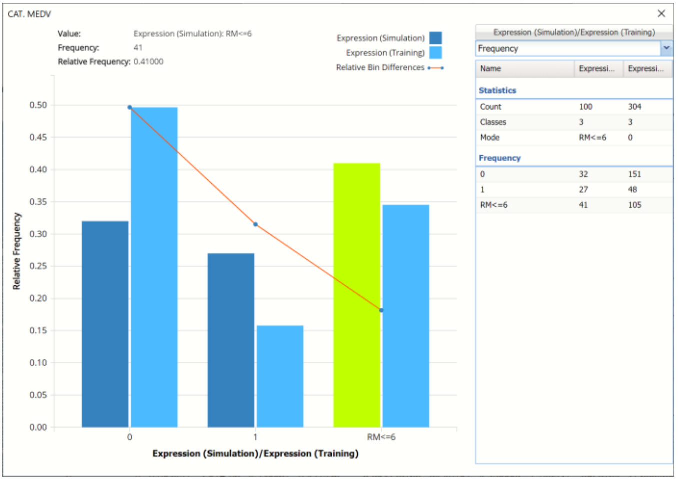
Name	Value

Bin	Abs. Difference
0	10.88%
1	10.88%

The red Relative Bin Differences curve indicate that the absolute difference for each bin are equal. Click the down arrow next to Frequency and select Bin Details to view.

The chart below displays frequency information from the synthetic data and the predictions for the training partition as evaluated by the expression, IF([@RM]>6, [@CAT. MEDV], “RM<=6”)

For example, the bars in the darker shade of blue display the results of the expression as applied to the synthetic data and the bars in the lighter shade of blue display the results of the expression when applied to the predictions for the training partition.



Value: Expression (Simulation): RM<=6
Frequency: 41
Relative Frequency: 0.41000

Expression (Simulation)/Expression (Training)

Frequency

Name	Expressi...	Expressi...
Statistics		
Count	100	304
Classes	3	3
Mode	RM<=6	0
Frequency		
0	32	151
1	27	48
RM<=6	41	105

- In the synthetic data, 32 records with RM > 6 were classified as 0 and 27 records with RM > 6 were classified as 1 (expensive). The remaining records in the synthetic data are shown in the dark blue column on the far left labeled RM <= 6, or 41.
- In the training partition, 151 records with RM > 6 were classified as 0 and 48 records with RM > 6 were classified as 1 (expensive). The remaining records in the training partitiong are shown in the light blue column on the far left labeled RM <= 6, or 105.
- The Relative Bin Differences curve indicates the absolute differences in each bin.

Click the down arrow next to Frequency to change the chart view to Relative Frequency, to change the look by clicking Chart Options or to see details of each bin listed in the chart. Statistics on the right of the chart dialog are discussed earlier in the Logistic Regression chapter. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

Analytic Solver Data Science generates *CBagging_Stored* along with the other output. Please refer to the “Scoring New Data” chapter in the Analytic Solver Data Science User Guide for details.

Boosting Ensemble Method for Classification

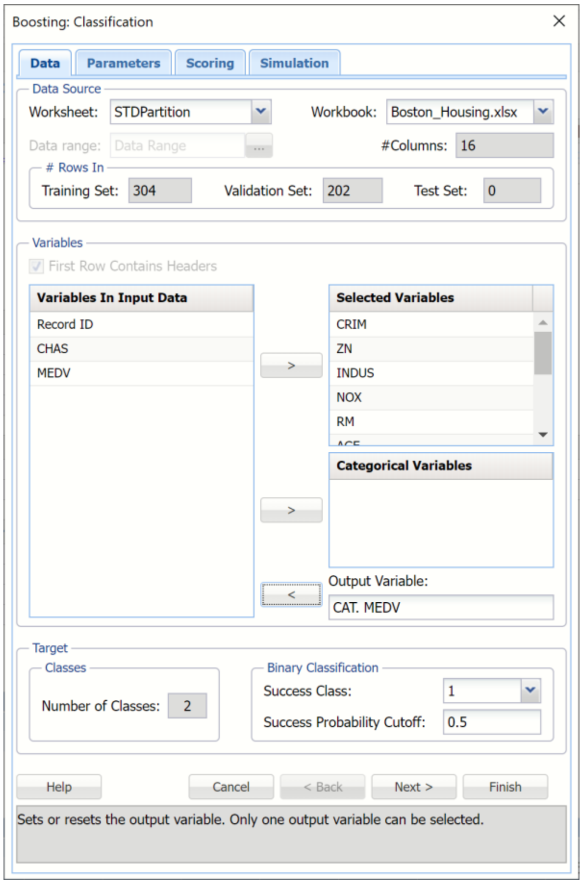
Now let’s use the 2nd ensemble method, boosting. We’ll re-use the same dataset, Boston_Housing.xlsx, with the same partitions.

Inputs

1. With the STDPartition worksheet selected, click **Classify – Ensemble Methods – Boosting** to open the Boosting Classification dialog.

2. Again, select **CAT. MEDV** as the *Output variable*. Then select **all remaining variables *except Record, ID, MEDV and CHAS*** as *Selected Variables.*

3. Keep the default selections for Binary Classification. (For more information on these options, see the Bagging Ensemble Method Example above.)



Worksheet: STDPartition
Workbook: Boston_Housing.xlsx
Data range: Data Range
#Columns: 16
# Rows In
Training Set: 304
Validation Set: 202
Test Set: 0

☑ First Row Contains Headers

Variables In Input Data: Record ID, CHAS, MEDV

Selected Variables: CRIM, ZN, INDUS, NOX, RM, AGE

Output Variable: CAT. MEDV

Number of Classes: 2
Success Class: 1
Success Probability Cutoff: 0.5

Sets or resets the output variable. Only one output variable can be selected.

4. Click **Next** to advance to the *Boosting Classifications Parameters* tab.

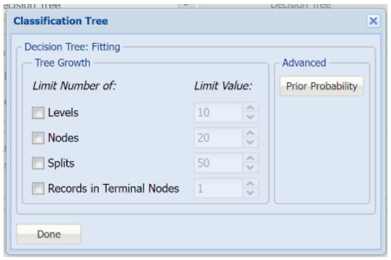
For more information on the partitioning and rescaling data “on-the-fly” using the Partition Data and Rescale Data buttons, see the Bagging example above.

5. Leave the *Number of weak learners* at the default of 10. Recall that this option controls the number of “weak” classification models created.

6. Under Ensemble: Common, click the down arrow beneath Weak Learner to select one of the six featured classifiers. In this example, we will select Decision Tree.

7. Under Ensemble: Classification click the down arrow beneath Weak Leaner to select one of the six featured classifiers. For this example, select Decision Tree.

   Options pertaining to the Decision Tree learner may be changed by clicking the Decision Tree button to the right of Weak Learner.



Classification Tree

Decision Tree: Fitting

Tree Growth

Limit Number of: *Limit Value:*

☐ Levels 10

☐ Nodes 20

☐ Splits 50

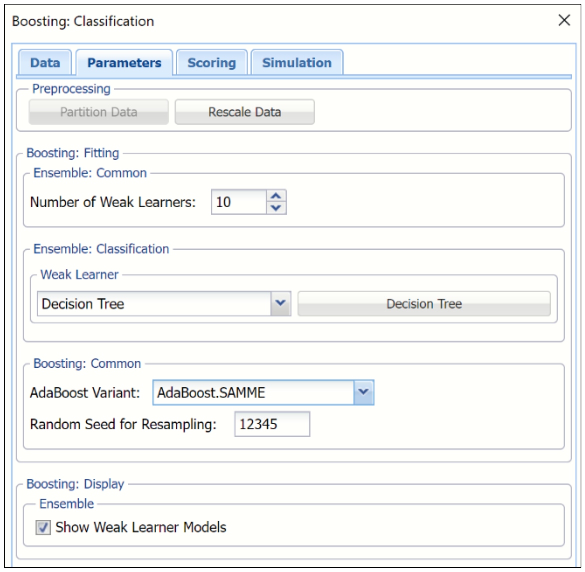
☐ Records in Terminal Nodes 1

All options will be left at their default values. For more information on these options, see the Classification Tree chapter that occurs earlier in this guide.

8. Under *Boosting: Common*, leave **AdaBoost.M1(Freund)** for *AdaBoost Variant*. The difference between AdaBoost.M1 (Freund), AdaBoost.M1 (Breiman) and Adaboost.SAMME is the way in which the weights assigned to each observation or record are updated. In AdaBoost.M1 (Freund), the constant is calculated as: $\alpha_b = \ln((1-e_b)/e_b)$. (Please refer to the section *Ensemble Methods* in the *Introduction* to the chapter for more information.)

9. Leave *Random Seed for Resampling* at the default of "12345". If an integer value appears for *Random Seed for Resampling*, Analytic Solver Data Science will use this value to specify the seed for random resampling of the training data for each weak learner. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier.

10. To display the weak learner models in the output, select **Show Weak Learner Models**.



Boosting: Classification

Preprocessing

Boosting: Fitting

Ensemble: Common

Number of Weak Learners: 10

Ensemble: Classification

Weak Learner

Decision Tree

Boosting: Common

AdaBoost Variant: AdaBoost.SAMME

Random Seed for Resampling: 12345

Boosting: Display

Ensemble

☑ Show Weak Learner Models

11. Click Next to advance to the Boosting Classification Scoring tab. Summary Report is selected by default under both *Score Training Data* and *Score Validation Data*.

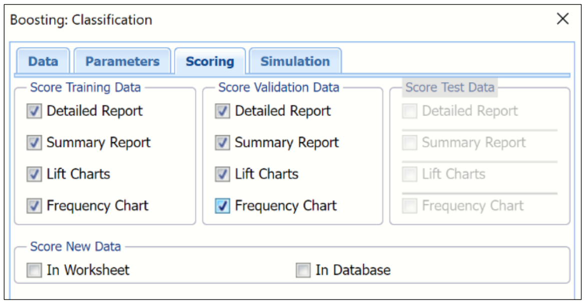
    - Select **Detailed Report** under both *Score Training Data* and *Score Validation Data* to produce a detailed assessment of the performance of the tree in both sets.

- Select **Lift Charts** to include Lift Charts, ROC Curves and Decile charts for both the Training and Validation datasets.

- Select **Frequency Chart** under *Score Training/Validation Data* to display a frequency chart on both the CBoosting_TrainingScore and CBoosting_ValidationScore worksheets. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

- Since we did not create a test partition, the options for *Score test data* are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

- See the "Scoring New Data" chapter within the Analytic Solver Data Science User Guide for information on the *Score new data* options.



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

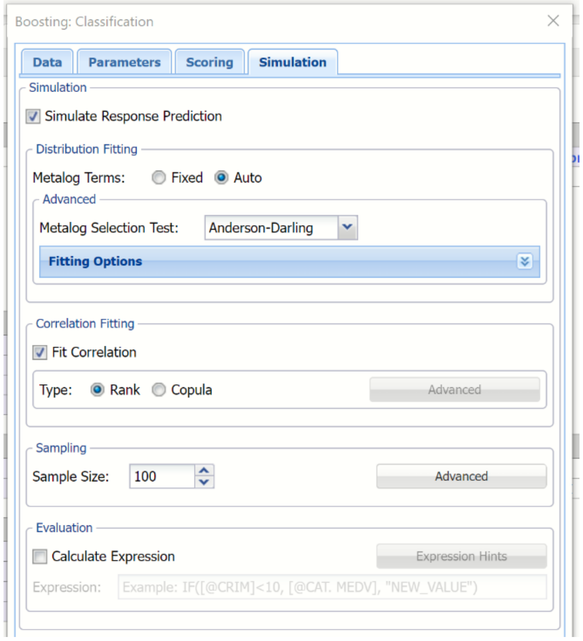
Score New Data
□ In Worksheet
□ In Database

12. Click **Next** to advance to the Simulation tab.

    Select **Simulation Response Prediction** to enable all options on the Simulation tab. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

    Simulation tab: All supervised algorithms include a new Simulation. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CBoosting_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Boosting Classification dialog, Simulation tab



Boosting: Classification

Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

□ Calculate Expression

Expression: Example: IF([@CRIM]<10, [@CAT. MEDV], "NEW_VALUE")

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the CBoosting_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog. Leave Calculate Expression unchecked. See the example above to see this feature in use.

For more information on the rptions shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

13. Click **Finish** to run the ensemble method.

Output

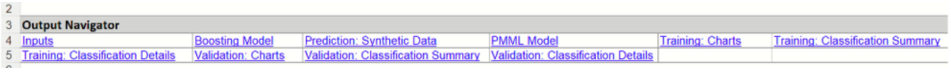
Output from the Ensemble Methods algorithm will be inserted to the right.

CBoosting_Output

This result worksheet includes 4 segments: Output Navigator, Inputs and Boosting Model.

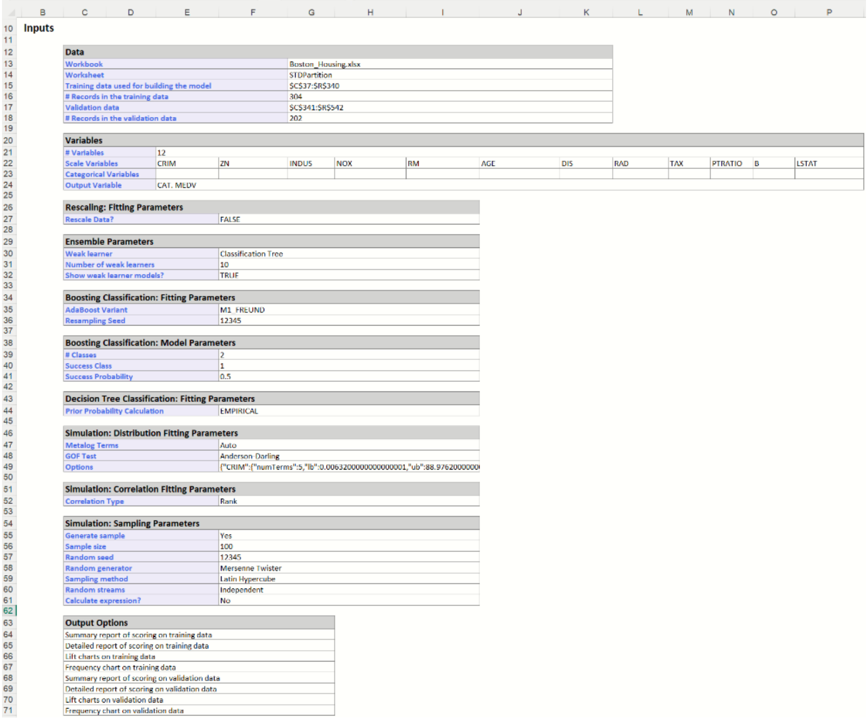
- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

CBoosting_Output: Output Navigator



Output Navigator					
Inputs	Boosting Model	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Classification Summary
Training: Classification Details	Validation: Charts	Validation: Classification Summary	Validation: Classification Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Bagging Classification dialog.



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	CAT. MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Ensemble Parameters	
Weak learner	Classification Tree
Number of weak learners	10
Show weak learner models?	TRUE

Boosting Classification: Fitting Parameters	
AdaBoost Variant	M1_FREUND
Resampling Seed	12345

Boosting Classification: Model Parameters	
# Classes	2
Success Class	1
Success Probability	0.5

Decision Tree Classification: Fitting Parameters	
Prior Probability Calculation	EMPIRICAL

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.0063200000000000001,"ub":88.97620000000

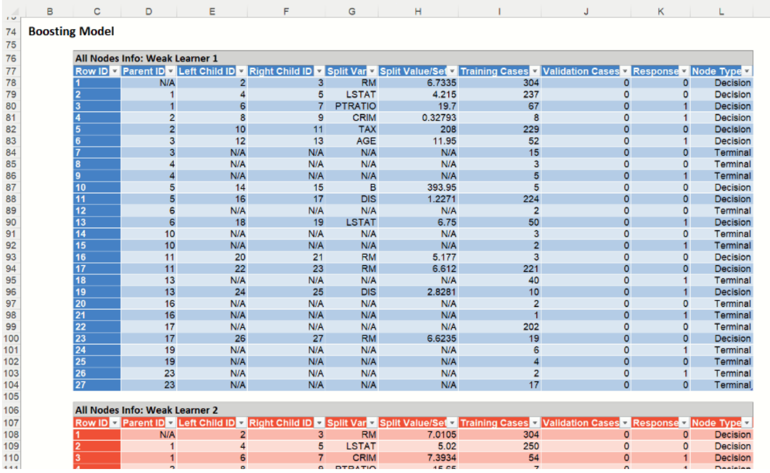
Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Boosting Model:** The number of Weak Learners in the output is equal to 10 which matches our input on the Parameters tab for the Number of Weak Learners option.

  The Importance percentage for each Variable in each Learner is listed in each table. This percentage measures the variable's contribution in reducing the total misclassification error.



Boosting Model

All Nodes Info: Weak Learner 1

Row ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	RM	6.7335	304	0	0	Decision
2	1	4	5	LSTAT	4.215	237	0	0	Decision
3	1	6	7	PTRATIO	19.7	67	0	1	Decision
4	2	8	9	CRIM	0.32793	8	0	1	Decision
5	2	10	11	TAX	208	229	0	0	Decision
6	3	12	13	AGE	11.95	52	0	1	Decision
7	3	N/A	N/A	N/A	N/A	15	0	0	Terminal
8	4	N/A	N/A	N/A	N/A	3	0	0	Terminal
9	4	N/A	N/A	N/A	N/A	5	0	1	Terminal
10	5	14	15	B	393.95	5	0	0	Decision
11	5	16	17	DIS	1.2271	224	0	0	Decision
12	6	N/A	N/A	N/A	N/A	2	0	0	Terminal
13	6	18	19	LSTAT	6.75	50	0	1	Decision
14	10	N/A	N/A	N/A	N/A	3	0	0	Terminal
15	10	N/A	N/A	N/A	N/A	2	0	1	Terminal
16	11	20	21	RM	5.177	3	0	0	Decision
17	11	22	23	RM	6.612	221	0	0	Decision
18	13	N/A	N/A	N/A	N/A	40	0	1	Terminal
19	13	24	25	DIS	2.8281	10	0	1	Decision
20	16	N/A	N/A	N/A	N/A	2	0	0	Terminal
21	16	N/A	N/A	N/A	N/A	1	0	1	Terminal
22	17	N/A	N/A	N/A	N/A	202	0	0	Terminal
23	17	26	27	RM	6.6235	19	0	0	Decision
24	19	N/A	N/A	N/A	N/A	6	0	1	Terminal
25	19	N/A	N/A	N/A	N/A	4	0	0	Terminal
26	23	N/A	N/A	N/A	N/A	2	0	1	Terminal
27	23	N/A	N/A	N/A	N/A	17	0	0	Terminal

All Nodes Info: Weak Learner 2

Row ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	RM	7.0105	304	0	0	Decision
2	1	4	5	LSTAT	5.02	250	0	0	Decision
3	1	6	7	CRIM	7.3934	54	0	1	Decision
4	2	8	9	PTRATIO	15.65	7	0	1	Decision

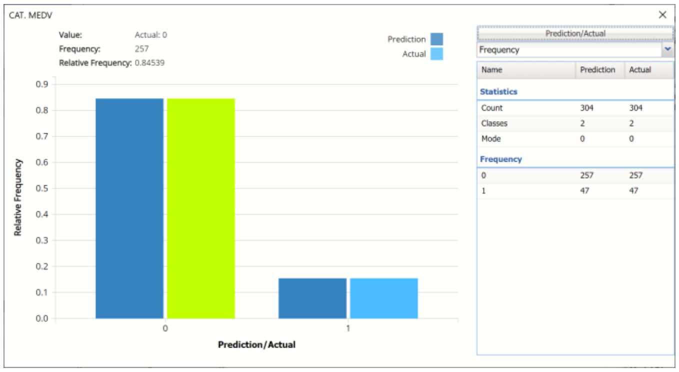
CBoosting_TrainingScore

Click the CBoosting_TrainingScore Scroll down to view the Classification Summary and Classification Details Reports for the Training partition as well as the Frequency charts. For detailed information on each of these components, see the Classification Tree chapter that appears earlier in this guide.

- **Frequency Chart:** This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

  Note: To view this frequency charts in the Cloud app, click the Charts icon on the Ribbon, select *CBoosting_TrainingScore* for Worksheet and *Frequency* for Chart.

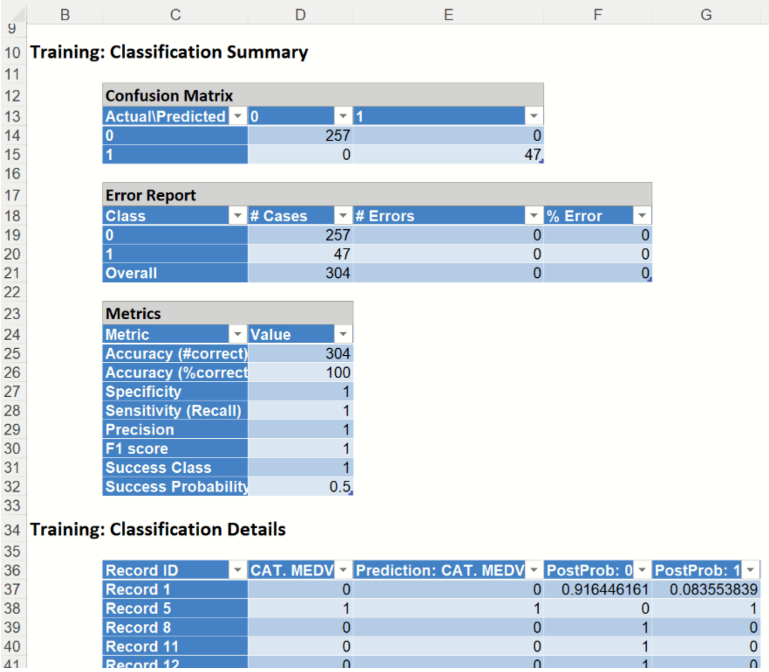
  Frequency Chart for Training Partition



Name	Prediction	Actual
Statistics		
Count	304	304
Classes	2	2
Mode	0	0
Frequency		
0	257	257
1	47	47

- **Classification Summary and Classification Details**: In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the classification method.

  Classification Summary and Classification Details Reports



Training: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	257	0
1	0	47

Error Report

Class	# Cases	# Errors	% Error
0	257	0	0
1	47	0	0
Overall	304	0	0

Metrics

Metric	Value
Accuracy (#correct)	304
Accuracy (%correct	100
Specificity	1
Sensitivity (Recall)	1
Precision	1
F1 score	1
Success Class	1
Success Probability	0.5

Training: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 0	PostProb: 1
Record 1	0	0	0.916446161	0.083553839
Record 5	1	1	0	1
Record 8	0	0	1	0
Record 11	0	0	1	0
Record 12	0	0	1	0

The Classification Summary displays the confusion matrix for the Training Partition.

- True Positive: 47 records belonging to the Success class were correctly assigned to that class.
- False Negative: 0 records belonging to the Success class were incorrectly assigned to the Failure class.

- True Negative: 257 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 0 records belonging to the Failure class were incorrectly assigned to the Success class.

There were no misclassified records. The Boosting model was able to correctly classify each record in the training partition.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 100.00% **-** Refers to the ability of the classifier to predict a class label correctly.
- Specificity: 1.0 **-** Also called the true negative rate, measures the percentage of failures correctly identified as failures

  Specificity (SPC) or True Negative Rate =TN / (FP + TN)
- Recall (or Sensitivity): 1.0 **-** Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)
- Precision: 1.0 **-** The probability of correctly identifying a randomly selected record as one belonging to the Success class

  Precision = TP/(TP+FP)
- F-1 Score: 1.0 **-** Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP / (2 * TP + FP + FN)
- Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

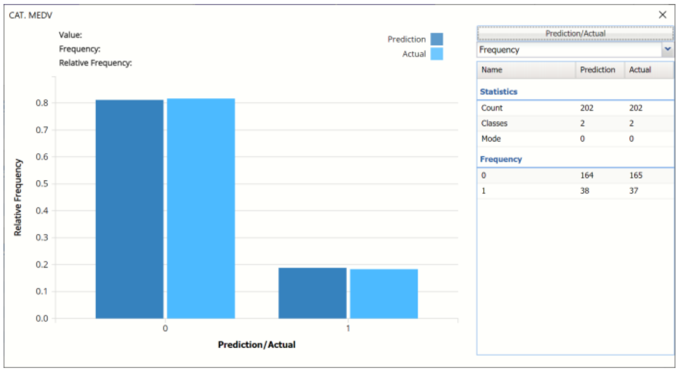
- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

CBoosting_ValidationScore

Click the CBagging_ValidationScore Scroll down to view the Classification Summary and Classification Details Reports for the Validation partition as well as the Frequency charts. For detailed information on each of these components, see the Classification Tree chapter that appears earlier in this guide.

- **Frequency Chart:** This chart shows the frequency for both the predicted and actual values of the output variable, along with various statistics such as count, number of classes and the mode.

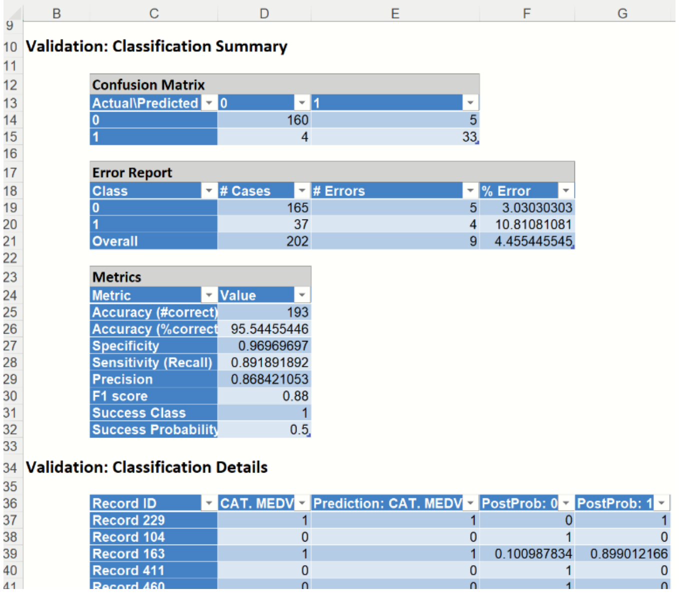
Frequency Chart for Validation Partition



Name	Prediction	Actual
Statistics		
Count	202	202
Classes	2	2
Mode	0	0
Frequency		
0	164	165
1	38	37

- **Classification Summary and Classification Details**: In the Classification Summary report, a Confusion Matrix is used to evaluate the performance of the fitted classification model on the validation partition.

Classification Summary and Classification Details Reports



Validation: Classification Summary

Confusion Matrix

Actual\Predicted	0	1
0	160	5
1	4	33

Error Report

Class	# Cases	# Errors	% Error
0	165	5	3.03030303
1	37	4	10.81081081
Overall	202	9	4.455445545

Metrics

Metric	Value
Accuracy (#correct)	193
Accuracy (%correct	95.54455446
Specificity	0.96969697
Sensitivity (Recall)	0.891891892
Precision	0.868421053
F1 score	0.88
Success Class	1
Success Probability	0.5

Validation: Classification Details

Record ID	CAT. MEDV	Prediction: CAT. MEDV	PostProb: 0	PostProb: 1
Record 229	1	1	0	1
Record 104	0	0	1	0
Record 163	1	1	0.100987834	0.899012166
Record 411	0	0	1	0
Record 460	0	0	1	0

The Classification Summary displays the confusion matrix for the Training Partition.

- True Positive: 33 records belonging to the Success class were correctly assigned to that class.
- False Negative: 4 records belonging to the Success class were incorrectly assigned to the Failure class.
- True Negative: 160 records belonging to the Failure class were correctly assigned to this same class
- False Positive: 5 records belonging to the Failure class were incorrectly assigned to the Success class.

The total number of misclassified records was 9 (5 + 4) which results in an error equal to 4.46%.

Metrics

The following metrics are computed using the values in the confusion matrix.

- Accuracy (#Correct and %Correct): 95.5% **-** Refers to the ability of the classifier to predict a class label correctly.
- Specificity: 0.970 **-** Also called the true negative rate, measures the percentage of failures correctly identified as failures

  Specificity (SPC) or True Negative Rate =TN / (FP + TN)
- Recall (or Sensitivity): 0.892 **-** Measures the percentage of actual positives which are correctly identified as positive (i.e. the proportion of people who experienced catastrophic heart failure who were predicted to have catastrophic heart failure).

  Sensitivity or True Positive Rate (TPR) = TP/(TP + FN)
- Precision: 0.868 **-** The probability of correctly identifying a randomly selected record as one belonging to the Success class

  Precision = TP/(TP+FP)
- F-1 Score: 0.88 **-** Fluctuates between 1 (a perfect classification) and 0, defines a measure that balances precision and recall.

  F1 = 2 * TP / (2 * TP + FP + FN)
- Success Class and Success Probability: Selected on the Data tab of the Discriminant Analysis dialog.

- **Classification Details**: This table displays how each observation in the training data was classified. The probability values for success in each record are shown after the predicted class and actual class columns. Records assigned to a class other than what was predicted are highlighted in red.

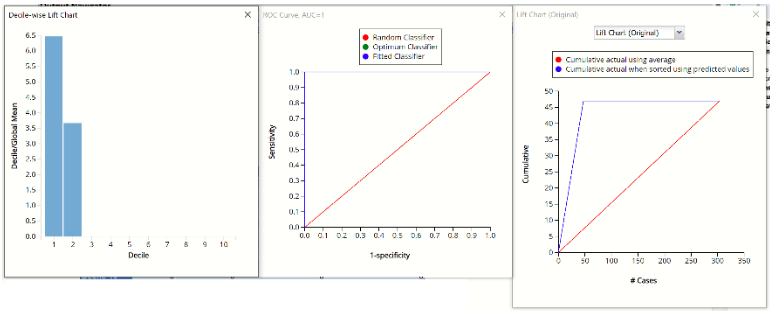
CBoosting_TrainingLiftCharts & Boosting_ValidationLiftCharts

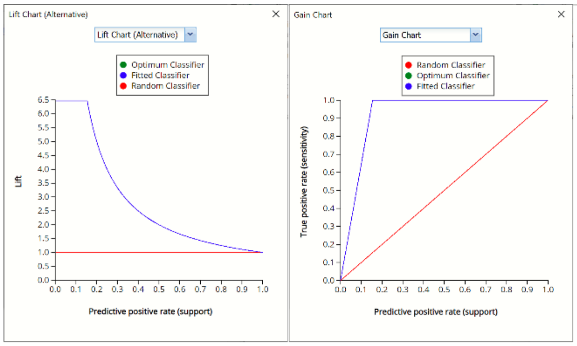
Click the **CBoosting_TrainingLiftChart and CBoosting_ValidationLiftChart** to navigate to the Lift Charts, shown below. For more information on lift charts, ROC curves, and Decile charts, please see the Classification Tree chapter that appears previously in this guide.

Lift Charts and ROC Curves are visual aids that help users evaluate the performance of their fitted models. Charts found on the CBoosting_TrainingLiftChart tab were calculated using the Training Data Partition. Charts found on the CBoosting_ValidationLiftChart tab were calculated using the Validation Data Partition. It is good practice to look at both sets of charts to assess model performance on both the Training and Validation partitions.

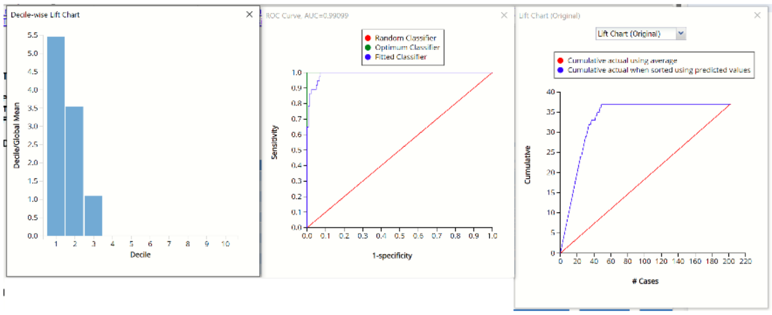
Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select *CBoosting_TrainingLiftChart* or *CBoosting_ValidationLiftChart* for *Worksheet* and *Decile Chart, ROC Chart* or *Gain Chart* for *Chart*.

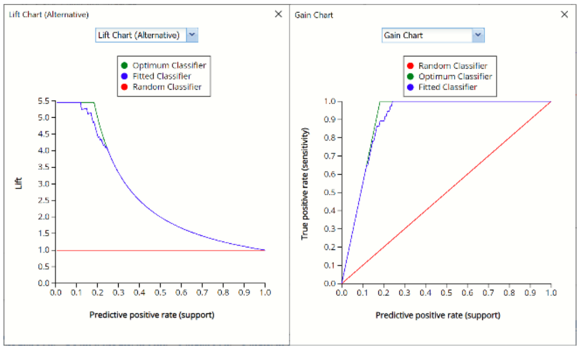
Boosting Training Partition Decile-wise, ROC, Lift and Gain Charts





Boosting Validation Partition Decile-wise, ROC, Lift and Gain Charts



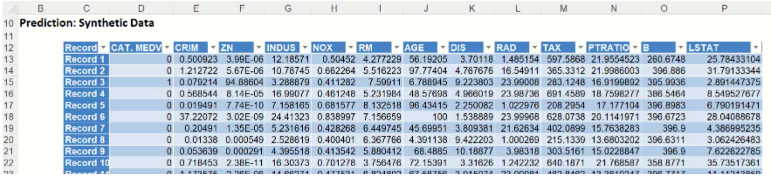


CBoosting_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, CBoosting_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Boosting Classification dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predictions for the training partition (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



Prediction: Synthetic Data

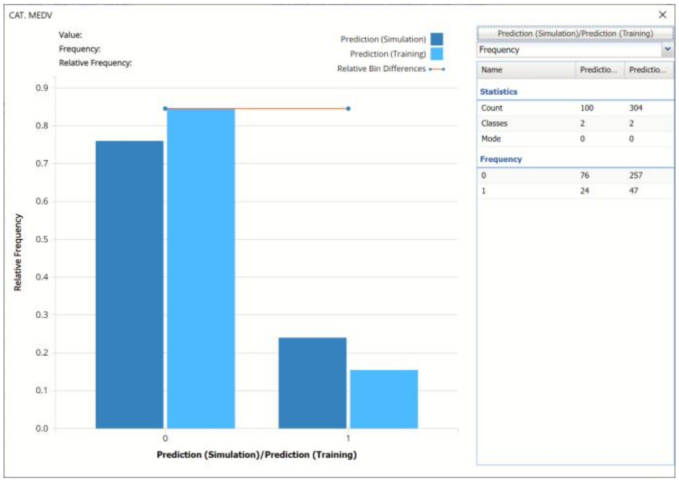
Record	CAT. MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	0	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.70118	1.485154	597.5868	21.9554523	260.6748	25.78433104
Record 2	0	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767676	16.54911	365.3312	21.9986003	396.886	31.79133344
Record 3	1	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223803	23.99008	283.1248	16.9199892	395.9936	2.891447375
Record 4	0	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966019	23.98736	691.4589	18.7598277	386.5464	8.549527677
Record 5	0	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082	1.022976	208.2954	17.177104	396.8983	6.790191471
Record 6	0	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538889	23.99968	628.0738	20.1141971	396.6723	28.04088678
Record 7	0	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809381	21.62634	402.0899	15.7638283	396.9	4.386995235
Record 8	0	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422203	1.000269	215.1339	13.6803202	396.6311	3.062426483
Record 9	0	0.053639	0.000291	4.395518	0.413542	5.880412	68.4885	10.18877	3.98318	303.5161	15.0228847	396.9	7.622622785
Record 10	0	0.718453	2.38E-11	16.30373	0.701278	3.756476	72.15391	3.31626	1.242232	640.1871	21.768587	358.8771	35.73517361

This data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the predictions for the output variable in the training partition, the synthetic data and the expression, if it exists.

In the screenshot below, the bars in the darker shade of blue are based on the Prediction, or synthetic, data as generated in the table above for the CAT. MEDV variable. The bars in the lighter shade of blue display the frequency of the predictions for the CAT. MEDV variable in the training partition.

Frequency Chart for CBoosting_Simulation output



Name	Predictio...	Predictio...
Statistics		
Count	100	304
Classes	2	2
Mode	0	0
Frequency		
0	76	257
1	24	47

Click the down arrow next to Frequency to change the chart view to Relative Frequency, to change the look by clicking Chart Options or to see details of each bin listed in the chart. Statistics on the right of the chart dialog are discussed earlier in the Classification Tree chapter. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

Analytic Solver Data Science generates *CBoosting_Stored* along with the other output worksheets. Please refer to the "Scoring New Data" chapter in the Analytic Solver Data Science User Guide for details.

Random Trees Ensemble Method Example

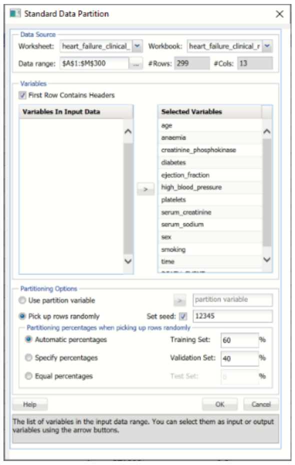
To further emulate the results of the journal article discussed in the Feature Selection Example in the Data Science User Guide, the Random Trees Ensemble

Classification Methods will be used to investigate if a machine learning algorithm can predict a patient's survival using the top two or three ranked features as found by the Feature Selection tool.

Inputs

1. First, click **Partition – Standard Partition** to partition the dataset into Training, Validation and Test Sets using the default percentages of 60% allocated to the Training Set and 40% allocate to the Validation Set.

   Figure 1: Standard Data Partition dialog



Standard Data Partition

Data Source
Worksheet: heart_failure_clinical_ Workbook: heart_failure_clinical_r
Data range: $A$1:$M$300 #Rows: 299 #Cols: 13

Variables
☑ First Row Contains Headers

Variables In Input Data

Selected Variables
age
anaemia
creatinine_phosphokinase
diabetes
ejection_fraction
high_blood_pressure
platelets
serum_creatinine
serum_sodium
sex
smoking
time

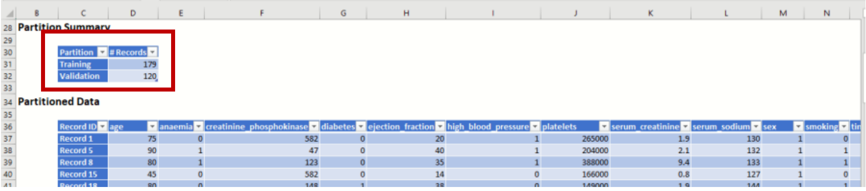
Partitioning Options
○ Use partition variable — partition variable
● Pick up rows randomly — Set seed: ☑ 12345
Partitioning percentages when picking up rows randomly
● Automatic percentages — Training Set: 60 %
○ Specify percentages — Validation Set: 40 %
○ Equal percentages — Test Set: 0 %

The list of variables in the input data range. You can select them as input or output variables using the arrow buttons.

2. Click **OK** to create the two partitions.

   A new worksheet *STDPartition* is inserted to the right of the dataset. The number of records allocated to the Training partition is 179 and the number of records allocated to the Validation partition is 120.

Figure 2: Standard Data Partitioning results



Partition Summary

Partition	# Records
Training	179
Validation	120

Partitioned Data

Record ID	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking
Record 1	75	0	582	0	20	1	265000	1.9	130	1	0
Record 5	90	1	47	0	40	1	204000	2.1	132	1	1
Record 8	80	1	123	0	35	1	388000	9.4	133	1	1
Record 15	45	0	582	0	14	0	166000	0.8	127	1	0

The first time that the model is fit, only two features (ejection_fraction and serum_creatinine) will be utilized.

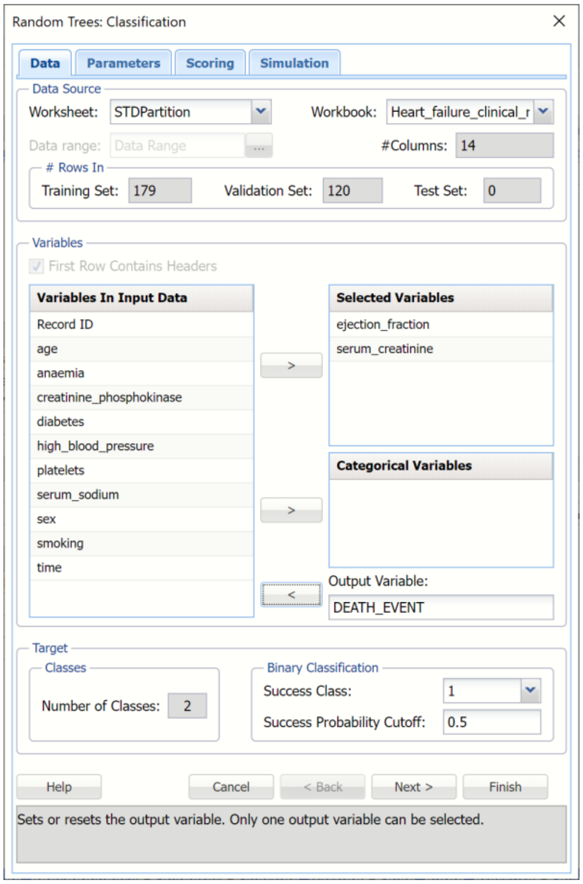
3. With the StdPartition workbook selected, click **Classify – Ensemble – Random Trees** to open the *Random Trees: Classification* dialog.

4. Select the two Variables from *Variables In Input Data* (ejection_fraction and serum_creatinine) and click the right pointing arrow to the left of *Selected Variables* to add these two variables to the model. Then take similar steps to select DEATH_EVENT as the *Output Variable*.

5. Leave Success Class as "1" and Success Probability Cutoff at 0.5 under Binary Classification.

   The Random Trees: Classification dialog should be similar to the one pictured in the Figure 3 below.

Figure 3: Random Trees: Classification dialog with Selection Variables (serum_creatinine and ejection_fraction) and Output Variables (DEATH_EVENT) selected.



Random Trees: Classification

Data | Parameters | Scoring | Simulation

Data Source

Worksheet: STDPartition

Workbook: Heart_failure_clinical_r

Data range: Data Range

#Columns: 14

Rows In

Training Set: 179

Validation Set: 120

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
age
anaemia
creatinine_phosphokinase
diabetes
high_blood_pressure
platelets
serum_sodium
sex
smoking
time

Selected Variables
ejection_fraction
serum_creatinine

Categorical Variables

Output Variable: DEATH_EVENT

Target

Classes

Number of Classes: 2

Binary Classification

Success Class: 1

Success Probability Cutoff: 0.5

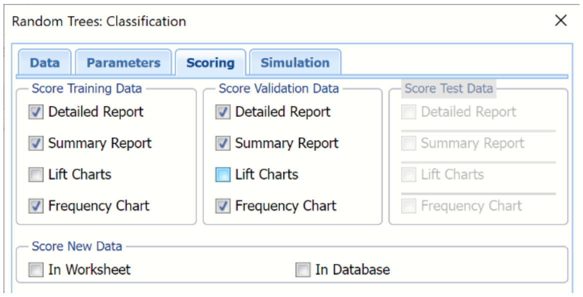
Sets or resets the output variable. Only one output variable can be selected.

6. Click the **Scoring tab** to advance to the *Random Trees: Classification Scoring* tab.

   For more information on Random Trees parameters, see the Random Trees Classification Options section below.

7. Summary Report is selected by default. Select **Detailed Report** and **Frequency Chart** for both *Score Training Data* and *Score Validation Data* and then click **Finish**.

   Figure 4: Random Trees: Classification dialog with output choices selected



Random Trees: Classification

Data | Parameters | Scoring | Simulation

Score Training Data

☑ Detailed Report

☑ Summary Report

☐ Lift Charts

☑ Frequency Chart

Score Validation Data

☑ Detailed Report

☑ Summary Report

☐ Lift Charts

☑ Frequency Chart

Score Test Data

☐ Detailed Report

☐ Summary Report

☐ Lift Charts

☐ Frequency Chart

Score New Data

☐ In Worksheet

☐ In Database

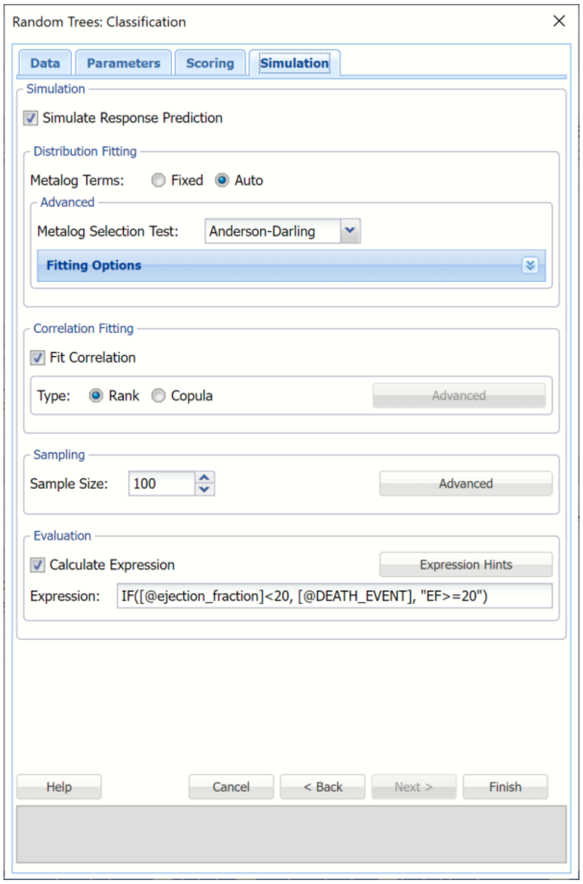
8. Click **Next** to advance to the Simulation tab.

9. Select **Simulation Response Prediction** to enable all options on the Simulation tab of the Random Trees Classification dialog. This tab is

disabled in Anlaytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, CRandTrees_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Figure 5: Random Trees Classification dialog, Simulation tab



Random Trees: Classification

Data | Parameters | Scoring | Simulation

Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

☑ Calculate Expression

Expression: IF([@ejection_fraction]<20, [@DEATH_EVENT], "EF>=20")

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the CRandTrees_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting and Sampling sections of the dialog. For Expression, enter the following formula to display if the patient suffered catastrophic heart failure (@DEATH_EVENT) when his/her Ejection_Fraction was less than or equal to 20.

IF([@ejection_fraction]<=20, [@DEATH_EVENT], “EF>20”)

Note that variable names are case sensitive.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

10. Click **Finish** to run Random Trees Classification on the example dataset.

Outputs

Five worksheets are inserted to the right of the STDPartition tab: *CRandTrees_Output*, *CRandTrees_TrainingScore*, *CRandTrees_ValidationScore*, *CRandTrees_Simulation* and *CRandTrees_Stored*.

CRandTrees_Output reports the input data, output data, and parameter settings.

CRandTrees_TrainingScore reports the confusion matrix, calculated metrics and the actual classification by row for the training partition.

CRandTrees_ValidationScore reports the confusion matrix, calculated metrics and the actual classification by row for the validation partition.

CRandTrees_Simulation contains the automated risk analysis simulation results.

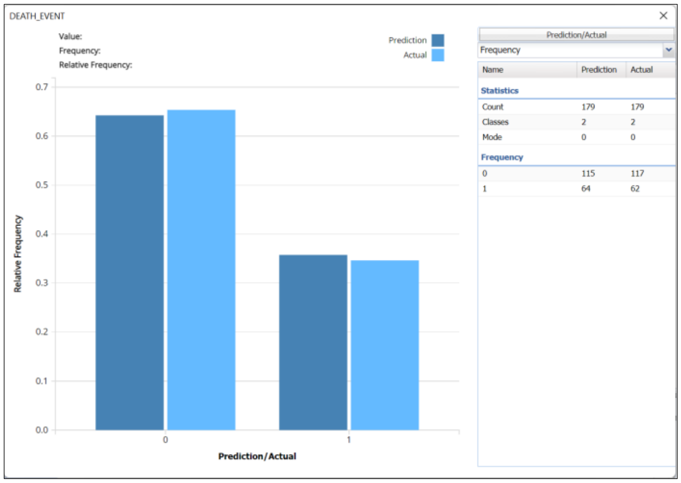
CRandTrees_Stored contains the stored model which can be used to apply the fitted model to new data. See the Scoring chapter within the Analytic Solver Data Science User Guide for an example of scoring new data using the stored model.

CRandTrees_TrainingScore

Click **CRandTrees_TrainingScore** to view the Classification Summary and then new output variable frequency chart for the Training partition.

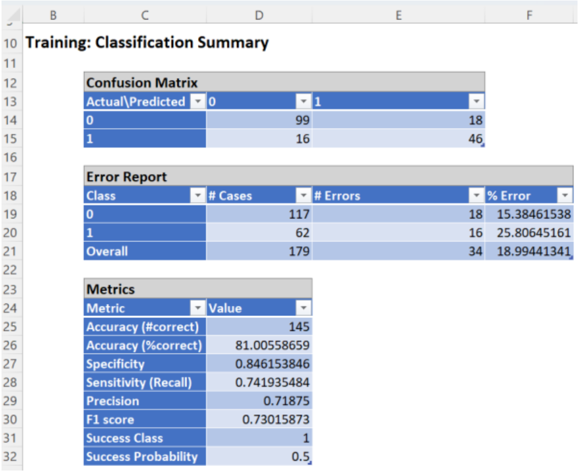
Since Frequency Chart was selected on the Scoring tab of the Random Trees dialog, a frequency chart is displayed upon opening of the worksheet.

Click *Prediction* in the upper right of the dialog, and select **Prediction** and **Actual** checkboxes to display frequency information between the Actual (Training) partition and the predicted values (Prediction). This chart quickly displays the Frequency of records labeled as 0 (survivors) and 1 (patients who succumbed to the complications of heart disease). Click the down arrow next to Frequency to view the Relative Frequency chart.



Name	Prediction	Actual
Statistics		
Count	179	179
Classes	2	2
Mode	0	0
Frequency		
0	115	117
1	64	62

Figure 6: Training: Classification Summary



Training: Classification Summary

Confusion Matrix		
Actual\Predicted	0	1
0	99	18
1	16	46

Error Report			
Class	# Cases	# Errors	% Error
0	117	18	15.38461538
1	62	16	25.80645161
Overall	179	34	18.99441341

Metrics	
Metric	Value
Accuracy (#correct)	145
Accuracy (%correct)	81.00558659
Specificity	0.846153846
Sensitivity (Recall)	0.741935484
Precision	0.71875
F1 score	0.73015873
Success Class	1
Success Probability	0.5

The overall error for the training partition was 18.99 with 18 surviving patients reported as deceased and 16 deceased patients reported as survivors.

- Accuracy: 81.01% -- Accuracy refers to the ability of the classifier to predict a class label correctly.
- Specificity: 0.846 – (True Negative)/(True Negative + False Positives)

  Specificity is defined as the proportion of negative classifications that were actually negative, or the fraction of survivors that actually survived. In this model, 99 actual surviving patients were classified correctly as survivors. There were 18 false positives or 18 actual survivors classified incorrectly as deceased.
- Sensitivity or Recall: 0.742 – (True Positive)/(True Positive + False Negative)

  Sensitivity is defined as the proportion of positive cases there were classified correctly as positive, or the proportion of actually deceased patients there were classified as deceased. In this model, 46 actual deceased patients were correctly classified as deceased. There were 16 false negatives or 16 actual deceased patients were incorrectly classified as survivors.

  Note: Since the object of this model is to correctly classify which patients will succumb to heart failure, this is an important statistic as it is very important for a physician to be able to accurately predict which patients require mitigation.
- Precision: 0.719 – (True Positives)/(True Positives + False Positives)

  Precision is defined as the proportion of positive results that are true positive. In this model, 46 actual deceased patients were classified correctly as deceased. There were 18 false positives or 18 actual survivors classified incorrectly as deceased.
- F-1 Score: 0.730 –2 x (Precision * Sensitivity)/(Precision + Sensitivity)

  The F-1 Score provides a statistic to balance between Precision and Sensitivity, especially if an uneven class distribution exists, as in this example, (99 survivors vs 46 deceased). The closer the F-1 score is to 1 (the upper bound) the better the precision and recall.

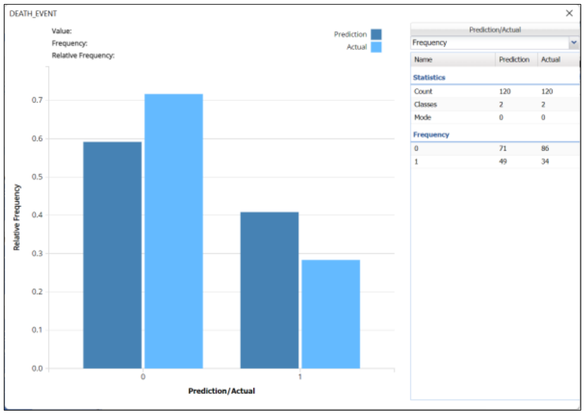
- Success Class and Success Probability simply reports the settings for these two values as input on the Random Trees: Classification Data tab.

View individual records and their classifications beneath Training: Classification Details.

CRandTrees_ValidationScore

Click the **CRandTrees_ValidationScore** tab to view the Summary Results for the Validation partition.

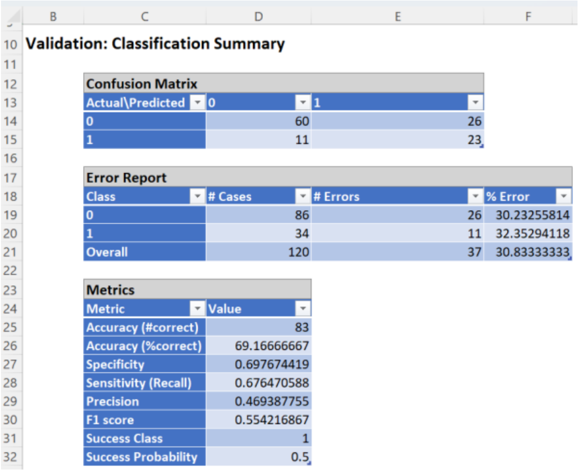
The Frequency Chart quickly displays how the fitted model performed on the validation partition.



Name	Prediction	Actual
Statistics		
Count	120	120
Classes	2	2
Mode	0	0
Frequency		
0	71	86
1	49	34

The overall error for the validation partition was 24.17 with 19 false positives (surviving patients reported as deceased) and 10 false negatives (deceased patients reported as survivors).

Figure 7: Validation: Classification Summery



Validation: Classification Summary

Confusion Matrix		
Actual\Predicted	0	1
0	60	26
1	11	23

Error Report			
Class	# Cases	# Errors	% Error
0	86	26	30.23255814
1	34	11	32.35294118
Overall	120	37	30.83333333

Metrics	
Metric	Value
Accuracy (#correct)	83
Accuracy (%correct)	69.16666667
Specificity	0.697674419
Sensitivity (Recall)	0.676470588
Precision	0.469387755
F1 score	0.554216867
Success Class	1
Success Probability	0.5

The overall error for the validation partition was 30.83 with 26 false positives (surviving patients reported as deceased) and 11 false negatives (deceased patients reported as survivors).

Note the following metrics:

- Accuracy: 69.17
- Specificity: .698
- Sensitivity or Recall: 0.676

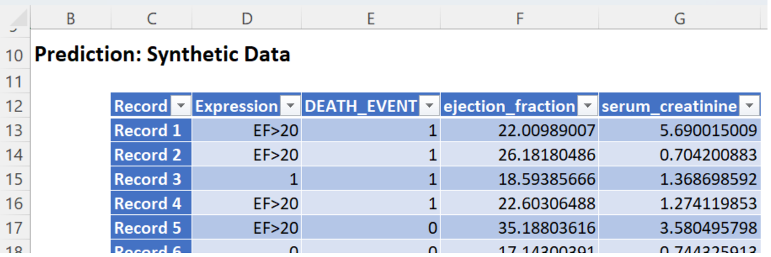
- Precision: 0.469
- F1 Score: 0.554

CRandTrees_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, CRandTrees_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Random Trees dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the prediction (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



Prediction: Synthetic Data

Record	Expression	DEATH_EVENT	ejection_fraction	serum_creatinine
Record 1	EF>20	1	22.00989007	5.690015009
Record 2	EF>20	1	26.18180486	0.704200883
Record 3	1	1	18.59385666	1.368698592
Record 4	EF>20	1	22.60306488	1.274119853
Record 5	EF>20	0	35.18803616	3.580495798
Record 6	0	0	17.14300391	0.744325913

Note the first column in the output, Expression. This column was inserted into the Synthetic Data results because Calculate Expression was selected and an Excel function was entered into the Expression field, on the Simulation tab of the Discriminant Analysis dialog

IF([@ejection_fraction]<=20, [@DEATH_EVENT], “EF>20”)

The results in this column are either 0, 1, or EF > 20.

- DEATH_EVENT = 0 indicates that the patient had an ejection_fraction <= 20 but did not suffer catastrophic heart failure.
- DEATH_EVENT = 1 in this column indicates that the patient had an ejection_fraction <= 20 and did suffer catastrophic heart failure.
- EF>20 indicates that the patient had an ejection fraction of greater than 20.

The remainder of the data in this report is synthetic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

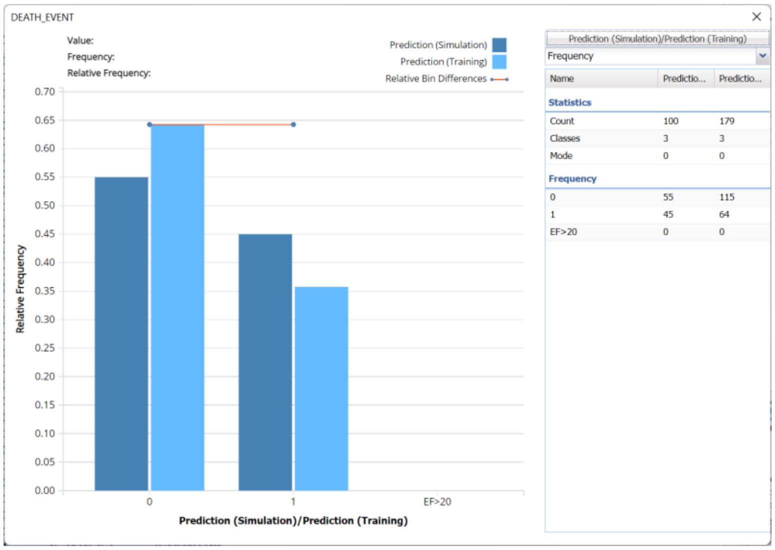
The chart that is displayed once this tab is selected, contains frequency information pertaining to the predictions of the output variable in the training partition, the synthetic data and the expression, if it exists.

The bars in the darker shade of blue display the frequency of labels in the Simulation, or synthetic, data. In the synthetic data, 55 “patients” are predicted to survive and the remaining are not.

The bars in the lighter shade display the frequency information for the training partition’s predicted values where 115 records were labeled as 0 (survivors) and 64 records were labeled as 1 (non-survivors).

The Relative Bin Differences curve indicates that the absolute differences in each bin are equal.

Frequency Chart for CRandTrees_Simulation output



Name	Predictio...	Predictio...
Statistics		
Count	100	179
Classes	3	3
Mode	0	0
Frequency		
0	55	115
1	45	64
EF>20	0	0

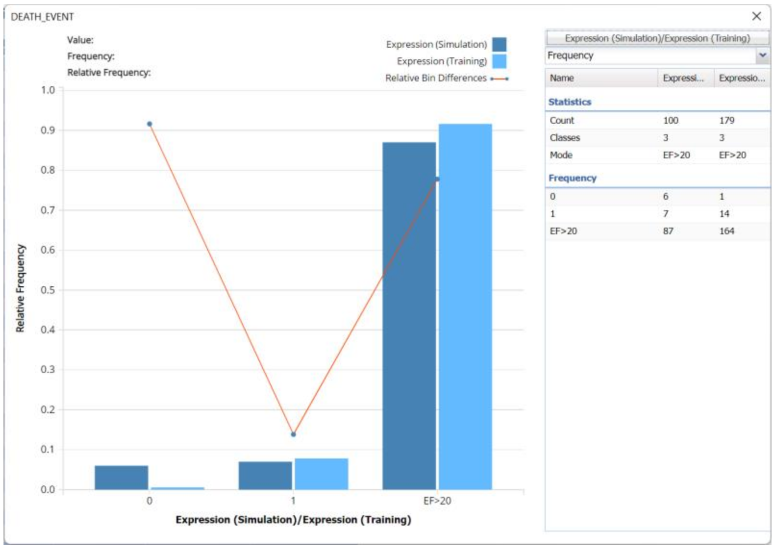
The chart below reveals the results of the expression as applied to each dataset.

The bars in the darker shade of blue display the frequency of labels in the Simulation, or synthetic, data. In the synthetic data, 6 *surviving* “patients” have an ejection fraction less than or equal to 20 while 7 “patients” with an ejection_fraction less than or equal to 20 did not survive.

The bars in the lighter shade display the frequency information for the training partition’s predicted values where 15 “patients”, or records, had ejection fractions less than 20; 1 patient was predicted to survive and 14 were not.

Columns labeled as “EF>20” contain the remainder of the records where the ejection fraction for each patient is larger than 20.

Frequency Chart for CRandTrees_Simulation output



Name	Expressi...	Expressio...
Statistics		
Count	100	179
Classes	3	3
Mode	EF>20	EF>20
Frequency		
0	6	1
1	7	14
EF>20	87	164

For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

The input steps were performed multiple times while adding additional Selected Variables according to the variable's importance or significance found by Feature Selection. The results are summarized in the table below.

The lowest Overall Error in the Validation Partition for any of the variable combinations occurs when just four variables, ejection_fraction, age, serum_sodium and serum_creatinine, are present in the fitted model. In addition, this fitted model also exhibits the highest Accuracy, Sensitivity, Precision and F1 Score metrics in the validation partition.   These results suggest that by obtaining these four measurements for a patient, a physician can determine whether the patient should undergo some type of mitigation for their heart failure diagnosis.

	Training Partition						Validation Partition					
Variables	Overall Error %	Accuracy (% Correct)	Specificity	Sensitivity (Recall)	Precision	F1 Score	Overall Error %	Accuracy	Specificity	Sensitivity (Recall)	Precision	F1 Score
ejection_fraction, serum_creatinine	18.99	81.01	0.846	0.742	0.719	0.730	30.83	69.17	0..698	0..676	0..469	0.554
+ age	2.23	97.765	0.974	0.984	0.953	0.68	32.5	67.5	0.686	0.647	0.449	0.530
+ serum_sodium	.5587	99.44	0.991	1.00	0.984	0.992	0.5587	99.44	0.991	1.00	0.984	0.992
+ high_blood_pressure (added as categorical variable)	3.35	96.65	0..966	0.968	0.938	0.952	27.50	72.5	0.767	0..618	0.512	0.56
+ anaemia (added as categorical variable)	2.79	97.21	0.983	0.952	0.967	0.959	31.67	68.33	0.744	0.529	0.45	0.486
+ serum_phosphokinase	1.676	98.324	0.974	1.00	0.954	0.976	34.167	65.83	0.721	0.50	0.414	0.453
+ platelets	2.235	97.765	0.967	1.00	0.939	0.969	29.167	70.833	0.674	0.794	0.491	0.607
+ smoking (added as categorical variable)	1.678	98.324	0.974	1.00	0.953	0.976	30.833	69.167	0.686	0.706	0.471	0.565
+ sex (added as categorical variable)	1.117	98.88	0.982	1.00	0.969	0.984	29.177	70.833	0.709	0.706	0.490	0.578
+ diabetes (added as categorical variable)	2.23	97.77	0.966	0.1.00	0.939	0.969	29.17	70.8333	0.779	0.529	0.486	0.507

Classification Ensemble Methods Options

The following options appear on the Bagging, Boosting, and Random Trees Data tabs.

Ensemble Method Classification Dialog, Data tab

Please see below for options appearing on the *Ensemble Methods- Data* tab.

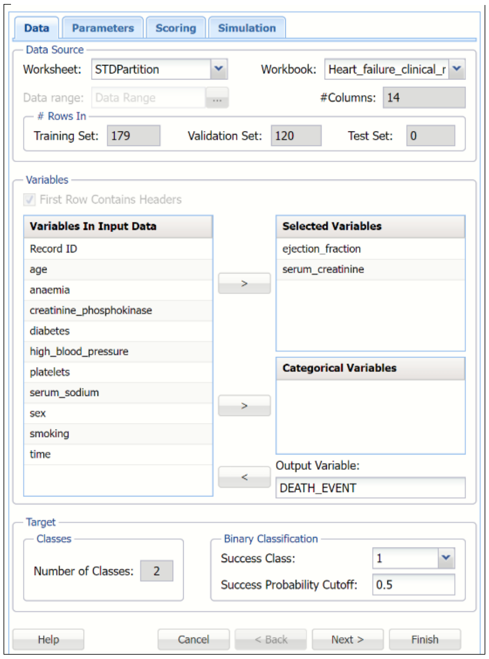
Variables In Input Data

The variables included in the dataset appear here.

Selected Variables

Variables selected to be included in the output appear here.

Ensemble Methods dialog, Data tab



Worksheet: STDPartition

Workbook: Heart_failure_clinical_r

#Columns: 14

Training Set: 179

Validation Set: 120

Test Set: 0

Variables In Input Data
Record ID
age
anaemia
creatinine_phosphokinase
diabetes
high_blood_pressure
platelets
serum_sodium
sex
smoking
time

Selected Variables
ejection_fraction
serum_creatinine

Categorical Variables

Output Variable: DEATH_EVENT

Number of Classes: 2

Success Class: 1

Success Probability Cutoff: 0.5

Categorical Variables

Place categorical variables from the Variables listbox to be included in the model by clicking the > command button. Ensemble Methods will accept non-numeric categorical variables.

Output Variable

The dependent variable or the variable to be classified appears here.

Number of Classes

Displays the number of classes in the Output variable.

Success Class

This option is selected by default. Click the drop down arrow to select the value to specify a “success”. This option is only enabled when the # of classes is equal to 2.

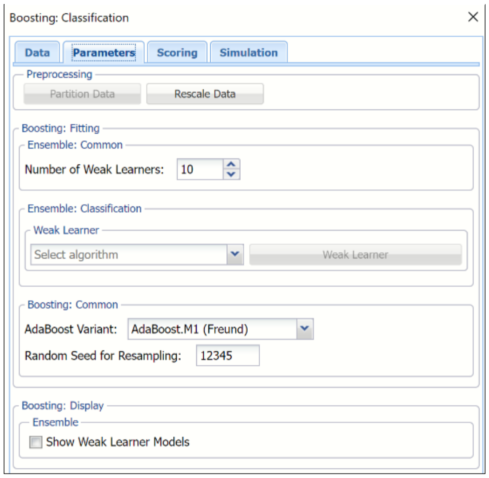
Success Probability Cutoff

Enter a value between 0 and 1 here to denote the cutoff probability for success. If the calculated probability for success for an observation is greater than or equal to this value, than a “success” (or a 1) will be predicted for that observation. If the calculated probability for success for an observation is less than this value, then a “non-success” (or a 0) will be predicted for that observation. The default value is 0.5. This option is only enabled when the # of classes is equal to 2.

Boosting Classification, Parameters tab

Please see below for options appearing on the *Boosting – Parameters* tab.

Boosting Ensemble Methods dialog, Parameters tab



Number of Weak Learners: 10

Select algorithm

AdaBoost Variant: AdaBoost.M1 (Freund)

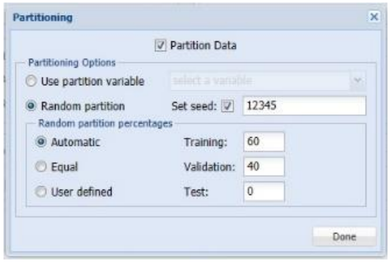
Random Seed for Resampling: 12345

□ Show Weak Learner Models

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Click **Partition Data** to open the Partitioning dialog. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the classification method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

On-the-fly Partitioning dialog



Partitioning

☑ Partition Data

Partitioning Options

○ Use partition variable

● Random partition   Set seed: ☑ 12345

Random partition percentages

● Automatic   Training: 60

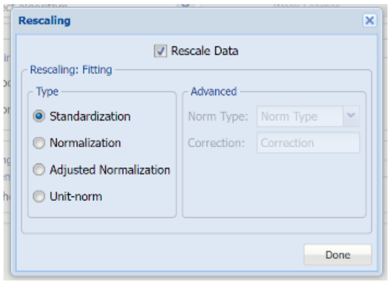
○ Equal   Validation: 40

○ User defined   Test: 0

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

On-the-fly Rescaling dialog



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

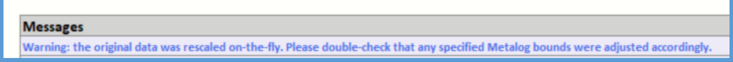
Advanced

Norm Type: Norm Type

Correction: Correction

Notes on Rescaling and the Simulation functionality

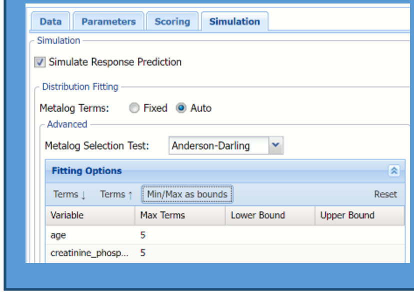
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then "Min/Max as bounds" on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the *EnsembleMethod*_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the "Min/Max as bounds" button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the "Min/Max as bounds" feature is available for the user's convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Number of Weak Learners

This option controls the number of "weak" classification models that will be created. The ensemble method will stop when the number or classification models created reaches the value set for this option. The algorithm will then compute the weighted sum of votes for each class and assign the "winning" classification to each record.

Weak Learner

Under Ensemble: Classification click the down arrow beneath Weak Leaner to select one of the six featured classifiers: Discriminant Analysis, Logistic Regression, k-NN, Naïve Bayes, Neural Networks, or Decision Trees. After a weak learner is chosen, the command button to the right will be enabled. Click this command button to control various option settings for the weak leaner.

AdaBoost Variant

The difference in the algorithms is the way in which the weights assigned to each observation or record are updated. (Please refer to the section *Ensemble Methods* in the *Introduction* to the chapter.)

In AdaBoost.M1 (Freund), the constant is calculated as:

$$\alpha_b = \ln((1-e_b)/e_b)$$

In AdaBoost.M1 (Breiman), the constant is calculated as:

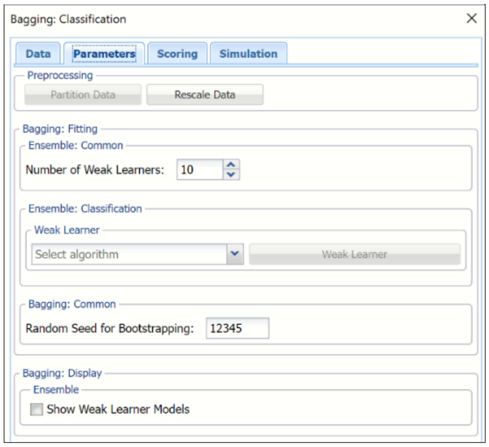
$$\alpha_b = 1/2\ln((1-e_b)/e_b)$$

In SAMME, the constant is calculated as:

$$\alpha_b = 1/2\ln((1-e_b)/e_b + \ln(k-1) \text{ where k is the number of classes}$$

(When the number of categories is equal to 2, SAMME behaves the same as AdaBoost Breiman.)

Bagging Ensemble Methods dialog, Parameters tab



Number of Weak Learners: 10

Random Seed for Bootstrapping: 12345

☐ Show Weak Learner Models

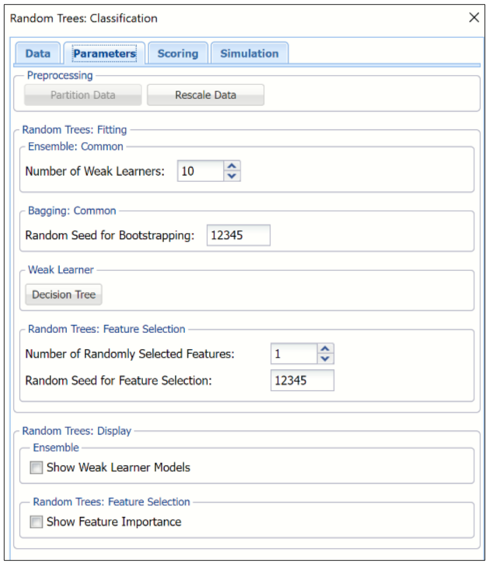
Random Seed for Resampling

If an integer value appears for *Random Seed for Resampling*, Analytic Solver Data Science will use this value to specify the seed for random resampling of the training data for each weak learner. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is "12345". If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Show Weak Learner

To display the weak learner models in the output, select **Show Weak Learner Models**.

Random Trees Ensemble Methods dialog, Parameters tab



Random Trees: Classification

Data | Parameters | Scoring | Simulation

Preprocessing
- Partition Data
- Rescale Data

Random Trees: Fitting

Ensemble: Common
- Number of Weak Learners: 10

Bagging: Common
- Random Seed for Bootstrapping: 12345

Weak Learner
- Decision Tree

Random Trees: Feature Selection
- Number of Randomly Selected Features: 1
- Random Seed for Feature Selection: 12345

Random Trees: Display

Ensemble
- □ Show Weak Learner Models

Random Trees: Feature Selection
- □ Show Feature Importance

Bagging Classification Dialog, Parameters tab

Please see below for options unique to the *Bagging – Parameters* tab.

Random Seed for Bootstrapping

If an integer value appears for *Bootstrapping Random seed*, Analytic Solver Data Science will use this value to set the bootstrapping random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Random Trees Classification, Parameters tab

Please see below for options unique to the *Random Trees – Parameters* tab.

Number of Randomly Selected Features

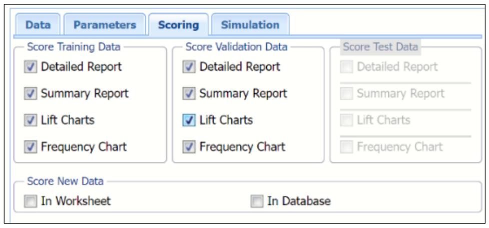
The Random Trees ensemble method works by training multiple “weak” classification trees using a fixed number of randomly selected features then taking the mode of each class to create a “strong” classifier. The option *Number of randomly selected features* controls the fixed number of randomly selected features in the algorithm. The default setting is **3**.

Random Seed for Featured Selection

If an integer value appears for *Feature Selection Random seed*, Analytic Solver Data Science will use this value to set the feature selection random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Please see below for options that are unique to the *Ensemble Methods Scoring tab*.

Random Trees Ensemble Methods dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
- ☑ Detailed Report
- ☑ Summary Report
- ☑ Lift Charts
- ☑ Frequency Chart

Score Validation Data
- ☑ Detailed Report
- ☑ Summary Report
- ☑ Lift Charts
- ☑ Frequency Chart

Score Test Data
- □ Detailed Report
- □ Summary Report
- □ Lift Charts
- □ Frequency Chart

Score New Data
- □ In Worksheet
- □ In Database

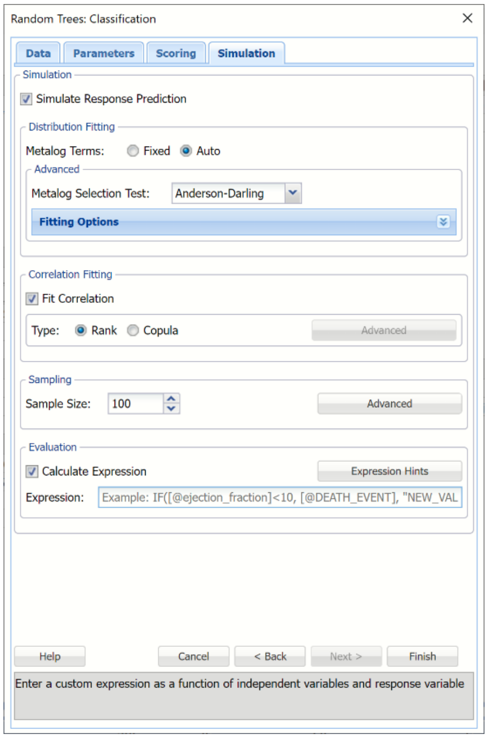
Classification Ensemble Dialog, Scoring Tab

Score Training Data

Select these options to show an assessment of the performance of the Ensemble Methods in classifying the training data. The report is displayed according to your specifications - Detailed, Summary, Frequency and Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories.

When Frequency Chart is selected, a frequency chart will be displayed when the <EnsembleMethod>_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Random Trees Ensemble Methods dialog, Simulate tab


Random Trees: Classification

Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

☑ Calculate Expression

Expression: Example: IF([@ejection_fraction]<10, [@DEATH_EVENT], "NEW_VAL

Enter a custom expression as a function of independent variables and response variable

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Ensemble Method algorithm in classifying the validation data. The report is displayed according to your specifications - Detailed, Summary, Frequency or Lift charts. Lift charts are only available when the *Output Variable* contains 2 categories. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the <EnsembleMethod>_ValidationScore worksheet is selected.

Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Ensemble Method in classifying the test data. The report is displayed according to your specifications - Detailed, Summary, Frequency and Lift charts Lift charts are only available when the *Output Variable* contains 2 categories. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the <EnsembleMethod>_TestScore worksheet is selected.

Score New Data

See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Ensemble Classification Dialog, Simulate tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, <EnsembleMethod>_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the <EnsembleMethod>_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Linear Regression Method

Introduction

Linear regression is performed on a dataset either to predict the response variable based on the predictor variable, or to study the relationship between the response variable and predictor variables. For example, using linear regression, the crime rate of a state can be explained as a function of demographic factors such as population, education, male to female ratio etc.

This procedure performs linear regression on a selected dataset that fits a linear model of the form

$$\mathbf{Y= b_0 + b_1X_1 + b_2X_2+ .... + b_kX_k+ e}$$

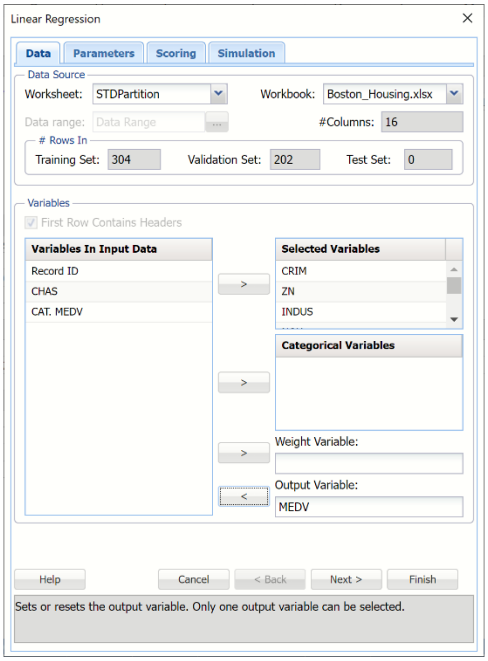
where Y is the dependent variable (response), X_1, X_2,.. .,X_k are the independent variables (predictors) and e is the random error. b_0 , b_1, b_2, ... b_k are known as the regression coefficients, which are estimated from the data. The multiple linear regression algorithm in Analytic Solver Data Science chooses regression coefficients to minimize the difference between the predicted and actual values.

See the Analytic Solver Data Science User Guide for a step-by-step example on how to use Linear Regression to predict housing prices using the example dataset, Boston_Housing.xlsx.

Linear Regression Options

The following options appear on the four Linear Regression dialogs: Data, Parameters, Scoring and Simulation.

Linear Regression Dialog, Data tab



Linear Regression

Data Source

Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Rows In

Training Set: 304

Validation Set: 202

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS

Categorical Variables

Weight Variable:

Output Variable: MEDV

Sets or resets the output variable. Only one output variable can be selected.

Linear Regression Dialog, Data tab

See below, for option explanations included on the Linear Regression Data tab.

Variables Input Data

All variables in the dataset are listed here.

Selected Variables

Variables listed here will be utilized in the Analytic Solver Data Science output.

Weight Variable

One major assumption of Linear Regression is that each observation provides equal information. Analytic Solver Data Science offers an opportunity to provide a Weight variable. Using a Weight variable allows the user to allocate a weight to each record. A record with a large weight will influence the model more than a record with a smaller weight.

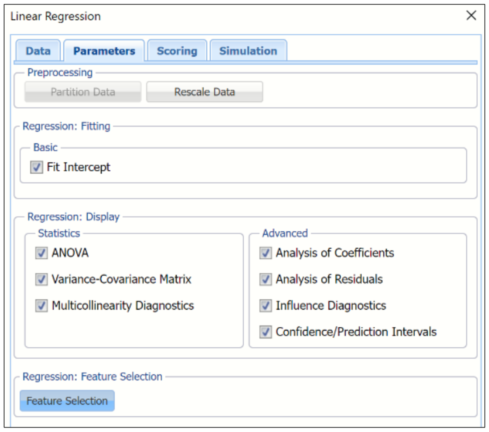
Output Variable

Select the variable whose outcome is to be predicted here.

Linear Regression Dialog, Parameters tab

See below, for option explanations included on the Linear Regression Parameters tab.

Linear Regression Dialog, Parameters tab



☑ Fit Intercept

☑ ANOVA

☑ Variance-Covariance Matrix

☑ Multicollinearity Diagnostics

☑ Analysis of Coefficients

☑ Analysis of Residuals

☑ Influence Diagnostics

☑ Confidence/Prediction Intervals

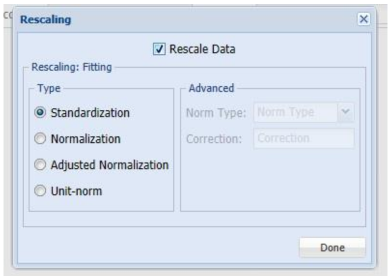
Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the regression method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Note: Rescaling has minimal effect in Regression methods. The coefficient estimates will be scaled proportionally with the data resulting in the same results with or without scaling. This feature is included on this dialog for consistency.



☑ Rescale Data

● Standardization

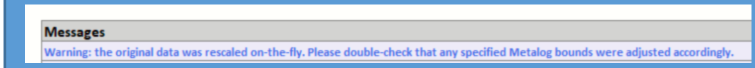
○ Normalization

○ Adjusted Normalization

○ Unit-norm

Notes on Rescaling and the Simulation functionality

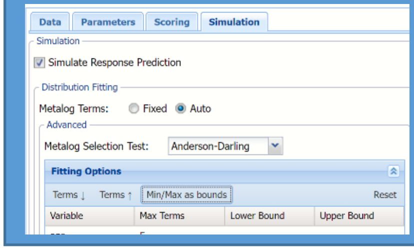
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the LinReg_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound

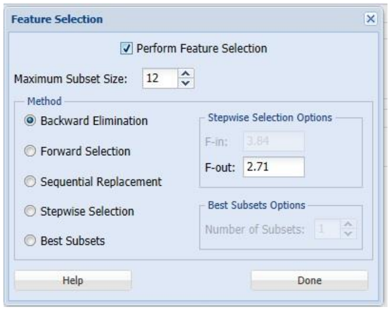
Fit Intercept

When this option is selected, the default setting, Analytic Solver Data Science will fit the Linear Regression intercept. If this option is not selected, Analytic Solver Data Science will force the intercept term to 0.

Feature Selection

When you have a large number of predictors and you would like to limit the model to only significant variables, click **Feature Selection** to open the *Feature Selection* dialog and select **Perform Feature Selection** at the top of the dialog.

Maximum Subset Size can take on values of 1 up to N where N is the number of Selected Variables. If no Categorical Variables exist, the default for this option is N. If one or more Categorical Variables exist, the default is "15".



☑ Perform Feature Selection

Maximum Subset Size: 12

● Backward Elimination

○ Forward Selection

○ Sequential Replacement

○ Stepwise Selection

○ Best Subsets

F-in: 3.84

F-out: 2.71

Number of Subsets: 1

Analytic Solver Data Science offers five different selection procedures for selecting the best subset of variables.

- *Backward Elimination* in which variables are eliminated one at a time, starting with the least significant. If this procedure is selected, FOUT is enabled. A statistic is calculated when variables are eliminated. For a variable to leave the regression, the statistic's value must be less than the value of FOUT (default = 2.71).

- *Forward Selection* in which variables are added one at a time, starting with the most significant. If this procedure is selected, FIN is enabled. On each iteration of the Forward Selection procedure, each variable is examined for the eligibility to enter the model. The significance of variables is measured as a partial F-statistic. Given a model at a current iteration, we perform an F Test, testing the null hypothesis stating that the regression coefficient would be zero if added to the existing set if variables and an alternative hypothesis stating otherwise. Each variable is examined to find the one with the largest partial F-Statistic. The decision rule for adding this variable into a model is: Reject the null hypothesis if the F-Statistic for this variable exceeds the critical value chosen as a threshold for the F Test (FIN value), or Accept the null hypothesis if the F-Statistic for this variable is less than a threshold. If the null hypothesis is rejected, the variable is added to the model and selection continues in the same fashion, otherwise the procedure is terminated.
- *Sequential Replacement* in which variables are sequentially replaced and replacements that improve performance are retained.
- *Stepwise selection* is similar to Forward selection except that at each stage, Analytic Solver Data Science considers dropping variables that are not statistically significant. When this procedure is selected, the Stepwise selection options FIN and FOUT are enabled. In the stepwise selection procedure a statistic is calculated when variables are added or eliminated. For a variable to come into the regression, the statistic's value must be greater than the value for FIN (default = 3.84). For a variable to leave the regression, the statistic's value must be less than the value of FOUT (default = 2.71). The value for FIN must be greater than the value for FOUT.
- *Best Subsets* where searches of all combinations of variables are performed to observe which combination has the best fit. (This option can become quite time consuming depending on the number of input variables.) If this procedure is selected, Number of best subsets is enabled.

Regression Display

Under *Regression: Display*, select all desired display options to include each in the output.

Under Statistics, the following display options are present.

- ANOVA
- Variance-Covariance Matrix
- Multicollinearity Diagnostics

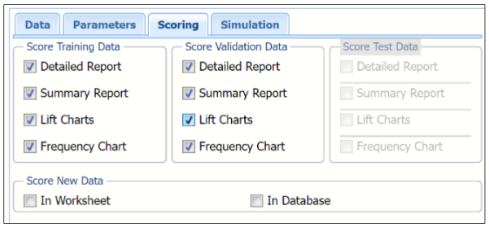
Under Advanced, the following display options are present.

- Analysis of Coefficients
- Analysis of Residuals
- Influence Diagnostics

- Confidence/Prediction Intervals

Linear Regression Dialog, Scoring tab

Linear Regression Dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

See below, for option explanations included on the Linear Regression Scoring tab.

Score Training Data

Select these options to show an assessment of the performance of the Linear Analysis algorithm in predicting the output variable value in the training partition. The report is displayed according to your specifications - Detailed, Summary, and Lift charts.

When Frequency Chart is selected, a frequency chart will be displayed when the LinReg_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Linear Analysis algorithm in predicting the output variable value in the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the LinReg_ValidationScore worksheet is selected.

Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Linear Regression algorithm in predicting the value of the output variable in the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the LinReg_TestScore worksheet is selected.

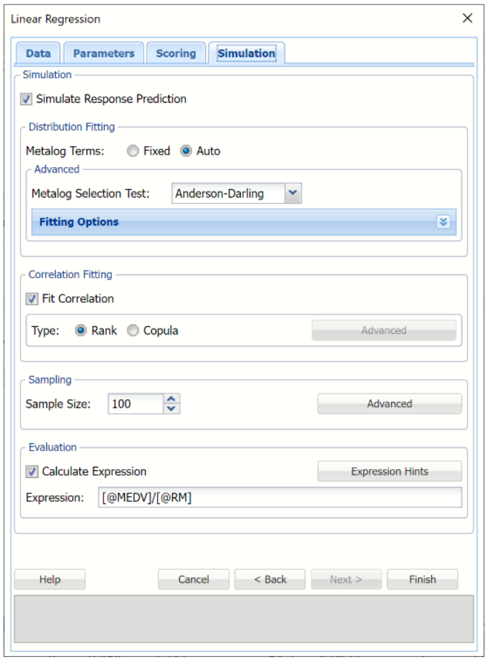
Score New Data

See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Linear Regression Dialog, Simulation tab

See below, for option explanations included on the Linear Regression Simulation tab.

Linear Regression Dialog, Simulation tab



Linear Regression

Data | Parameters | Scoring | Simulation

Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula

Sampling
Sample Size: 100

Evaluation
☑ Calculate Expression
Expression: [@MEDV]/[@RM]

Simulation Tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, LinReg_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the LinReg_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

k-Nearest Neighbors Regression Method

Introduction

In the k-nearest-neighbor regression method, the training data set is used to predict the value of a variable of interest for each member of a "target" data set. The structure of the data generally consists of a variable of interest ("amount purchased," for example), and a number of additional predictor variables (age, income, location, etc.).

1. For each row (case) in the target data set (the set to be predicted), locate the k closest members (the k nearest neighbors) of the training data set. A Euclidean Distance measure is used to calculate how close each member of the training set is to the target row that is being examined.
2. Find the weighted sum of the variable of interest for the k nearest neighbors (the weights are the inverse of the distances).
3. Repeat this procedure for the remaining rows (cases) in the target set.
4. Additionally, Analytic Solver Data Science also allows the user to select a maximum value for k, builds models in parallel on all values of k (up to the maximum specified value) and performs scoring on the best of these models.

Computing time increases as k increases, but the advantage is that higher values of k provide “smoothing” that reduces vulnerability to noise in the training data. Typically, k is in units of tens rather than in hundreds or thousands of units.

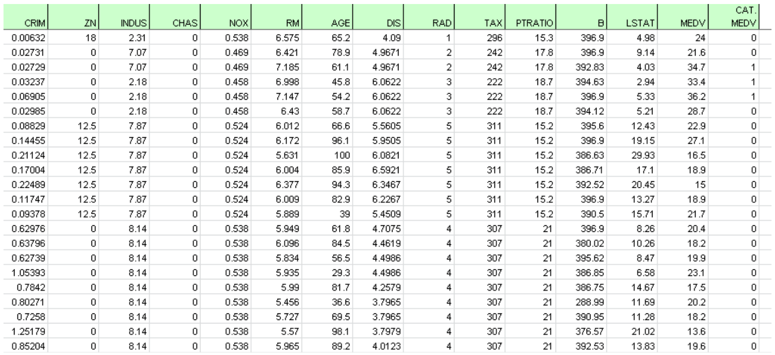
k-Nearest Neighbors Regression Method Example

The example below illustrates the use of Analytic Solver Data Science’s k-Nearest Neighbors Regression method using the Boston_Housing.xlsx dataset.

Input

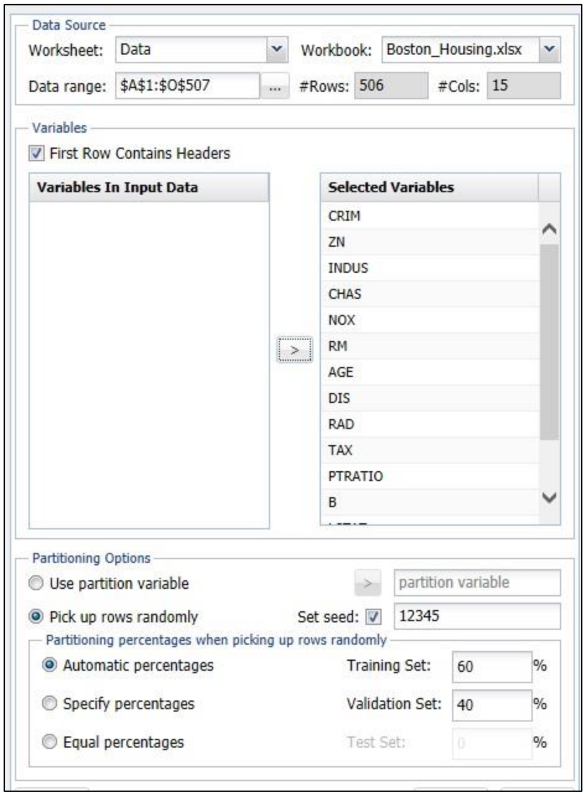
1. Click **Help – Example Models** on the Data Science ribbon, then click **Forecasting/Data Science Examples.** Click the **Boston Housing** link to open **Boston_Housing.xlsx**. This dataset contains 14 variables, the description of each is given in the Description worksheet included within the example workbook. The dependent variable MEDV is the median value of a dwelling. The objective of this example is to predict the value of this variable. A portion of the dataset is shown below.

   All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described in the Generate Data section appearing earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, KNNP_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. Since this new functionality does not support categorical variables, the CHAS variable will not be included in the k-Nearest Neighbors prediction model. The last variable, CAT. MEDV, is a discrete classification of the MEDV variable and will also not be used in this example.



CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT. MEDV
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24	0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	0
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	1
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	1
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	1
0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7	0
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	0
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	0
0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	0
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	0
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	0
0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	0
0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	0
0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	0
0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	0
0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	0
1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	0
0.7842	0	8.14	0	0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5	0
0.80271	0	8.14	0	0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2	0
0.7258	0	8.14	0	0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2	0
1.25179	0	8.14	0	0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6	0
0.85204	0	8.14	0	0.538	5.965	89.2	4.0123	4	307	21	392.53	13.83	19.6	0

2. Partition the data into training and validation sets using the Standard Data Partition defaults with percentages of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.



Data Source

Worksheet: Data

Workbook: Boston_Housing.xlsx

Data range: $A$1:$O$507

#Rows: 506

#Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables: CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B

Partitioning Options

○ Use partition variable

● Pick up rows randomly

Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages

○ Specify percentages

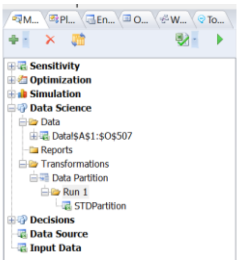
○ Equal percentages

Training Set: 60 %

Validation Set: 40 %

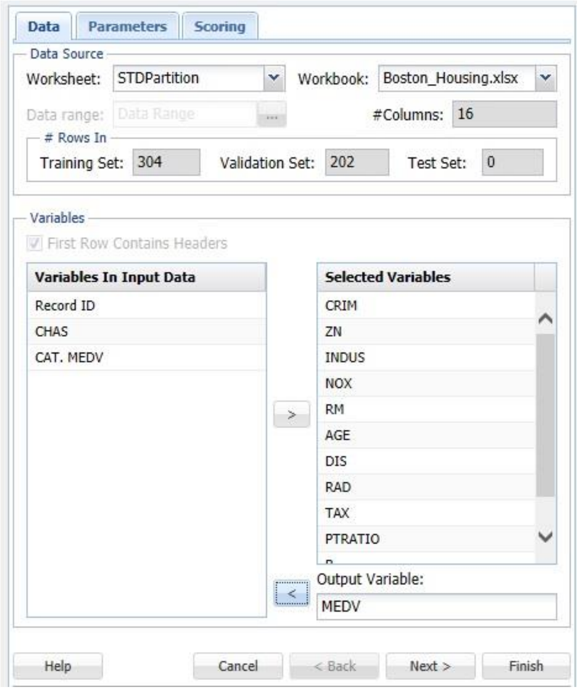
Test Set: 0 %

Note: If using Analytic Solver Desktop, the STDPartition worksheet is inserted into the Model tab of the Analytic Solver task pane under Transformations -- Data Partition and the data used in the partition will appear under Data, as shown in the screenshot below.



3. Click **Predict – k-Nearest Neighbors** to open the k-Nearest Neighbors regression dialog.

4. Select **MEDV** as the *Output Variable*, and the remaining variables (except CAT. MEDV, CHAS, and Record ID) as *Selected Variables*.



Worksheet: STDPartition
Workbook: Boston_Housing.xlsx
Data range: Data Range
#Columns: 16
Rows In
Training Set: 304
Validation Set: 202
Test Set: 0

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM
AGE
DIS
RAD
TAX
PTRATIO

Output Variable: MEDV

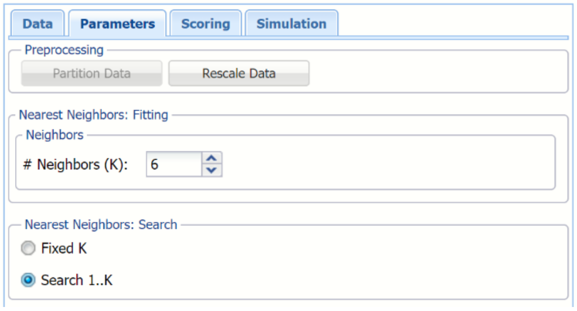
5. Click **Next** to advance to the *Parameters* tab.

Analytic Solver Data Science includes the ability to partition and rescale a dataset from within a classification or regression method by selecting Partition Data and/or Rescale Data on the Parameters tab. If both or either of these options are selected, Analytic Solver Data Science will partition and/or rescale your dataset (according to the partition and rescaling options you set) immediately before running the regression method. If partitioning has already occurred on the dataset, the Partition Data button will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter. For more information on rescaling, please see the Transforming Continuous Data chapter.

6. Enter **6** for *# Neighbors (K).* (This number is based on standard practice from the literature.) This is the parameter k in the k-Nearest Neighbor algorithm.

   If the number of observations (rows) is less than 50 then the value of k should be between 1 and the total number of observations (rows). If the number of rows is greater than 50, then the value of k should be between 1 and 50. Note that if k is chosen as the total number of observations in the training set, then for any new observation, all the observations in the training set become nearest neighbors. The default value for this option is 1.

7. Select **Search 1..K** under *Nearest Neighbors Search*. When this option is selected, Analytic Solver Data Science will display the output for the best k between 1 and the value entered for *# Neighbors*. If *Fixed K* is selected, the output will be displayed for the specified value of k.



# Neighbors (K): 6

○ Fixed K

● Search 1..K

8. Click **Next** to advance to the Scoring tab.

9. Select all four options for **Score Training/Validation data**.

   When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the k-Nearest Neighbors Regression output.

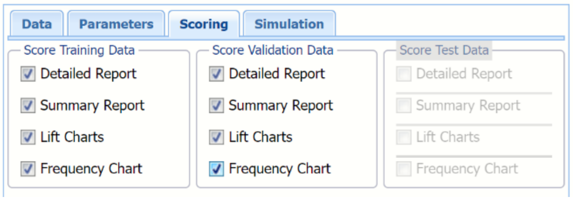
   When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the k-Nearest Neighbors Regression output.

   When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and RROC Curve plots in the output.

   When *Frequency Chart* is selected, a frequency chart will be displayed when the KNNP_TrainingScore and KNNP_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

   Since we did not create a test partition, the options for Score test data are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

   See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

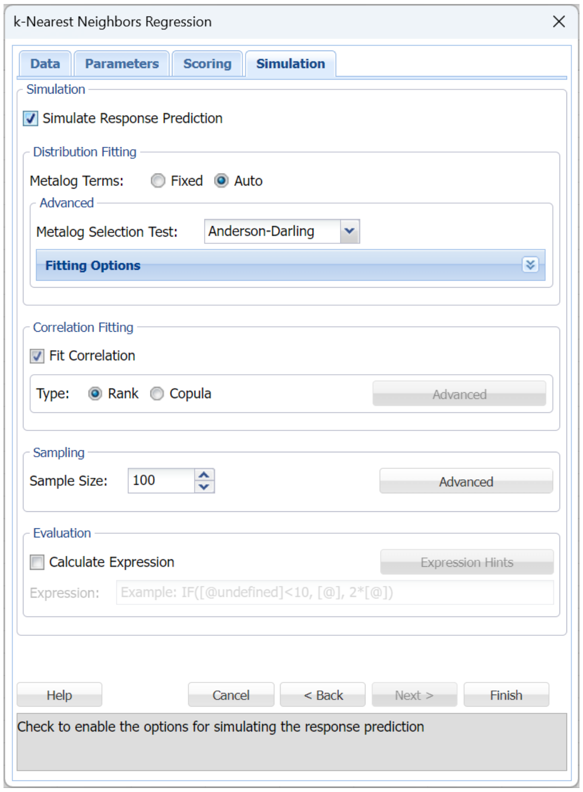
Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

10. Click **Next** to advance to the Simulation tab. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

11. Select *Simulation Response Prediction* to enable all options on the Simulation tab of the *k-Nearest Neighbors Regression* dialog.

    Simulation tab: As mentioned above, all supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, KNNP_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

    k-Nearest Neighbors Prediction dialog, Simulation tab



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

☑ Fit Correlation

Type: ● Rank ○ Copula

Sample Size: 100

☐ Calculate Expression

Expression: Example: IF([@undefined]<10, [@], 2*[@])

Check to enable the options for simulating the response prediction

    Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the KNNP_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression. Note that variable names are case sensitive.

    For the purposes of this example, leave all options at their defaults in the Distribution Fitting, Correlation Fitting, Sampling and Evaluation sections of the dialog. For more information on these options, see the Generate Data chapter that appears earlier in this guide.

12. Click Finish to run k-Nearest Neighbors Prediction on the example dataset.

Output

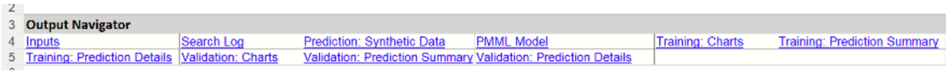
Output sheets containing the results of the k-Nearest Neighbors Prediction model will be inserted into your active workbook to the right of the STDPartition worksheet.

KNNP_Output

This result worksheet includes 3 segments: Output Navigator, Inputs and the Search Log.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

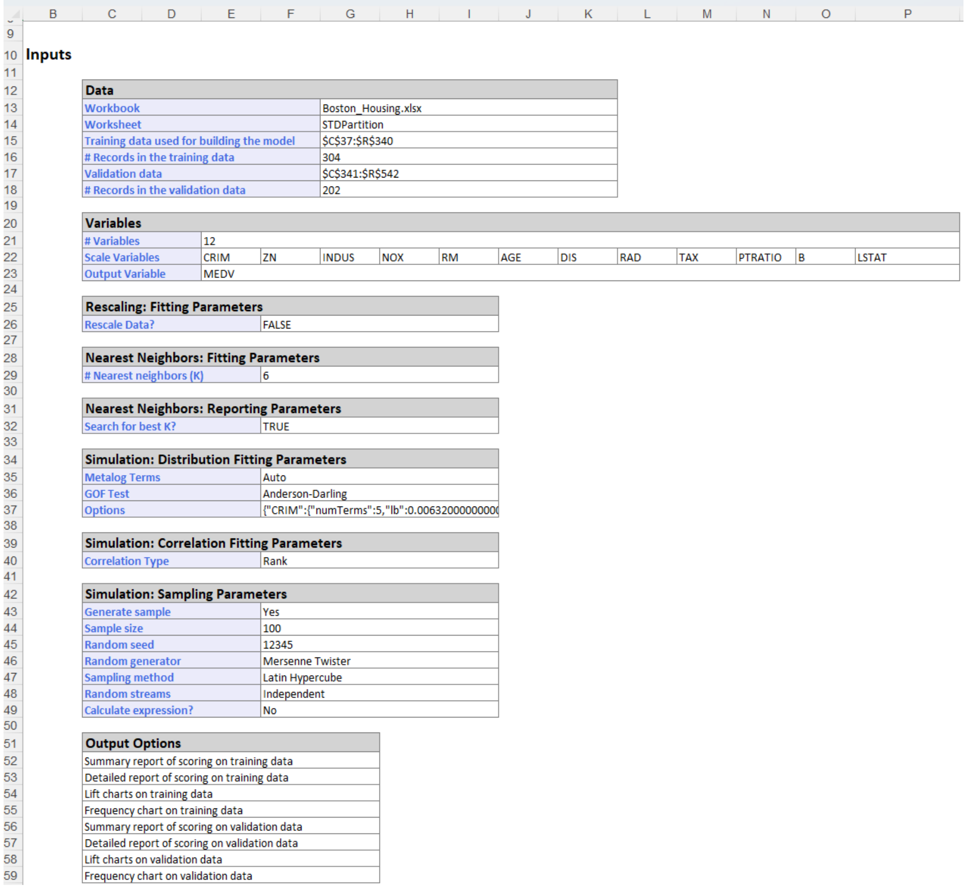
  KNNP_Output: Output Navigator



Output Navigator					
Inputs	Search Log	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Prediction Summary
Training: Prediction Details	Validation: Charts	Validation: Prediction Summary	Validation: Prediction Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the k-Nearest Neighbors Prediction dialog.

KNNP_Output: Inputs



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Output Variable	MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Nearest Neighbors: Fitting Parameters	
# Nearest neighbors (K)	6

Nearest Neighbors: Reporting Parameters	
Search for best K?	TRUE

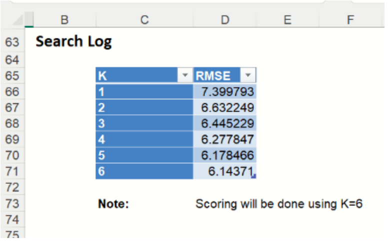
Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.00632000000000

Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Search Log:** Scroll down *KNNP_Output* to the *Search Log* report (shown below). As per our specifications, Analytic Solver Data Science has calculated the RMS error for all values of k and denoted the value of k with the smallest RMS Error. The validation partition will be scored using this value of k.



Search Log

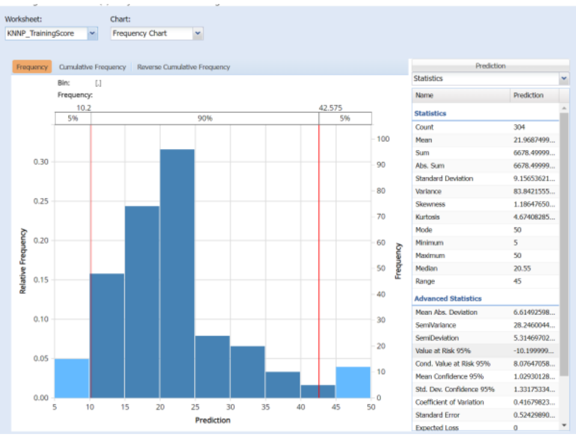
K	RMSE
1	7.399793
2	6.632249
3	6.445229
4	6.277847
5	6.178466
6	6.14371

Note: Scoring will be done using K=6

KNNP_TrainingScore

Click the *KNNP_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Prediction Summary and the Training: Prediction Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

> Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: KNNP_TrainingScore

Chart: Frequency Chart

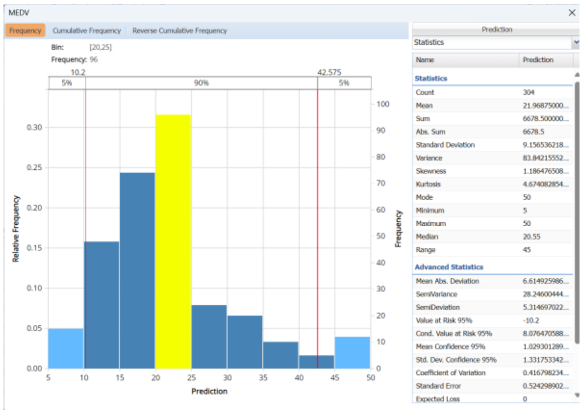
Bin: [,]

Frequency:

Name	Prediction
Statistics	
Count	304
Mean	21.9687499...
Sum	6678.49999...
Abs. Sum	6678.49999...
Standard Deviation	9.15653621...
Variance	83.8421555...
Skewness	1.18647650...
Kurtosis	4.67408285...
Mode	50
Minimum	5
Maximum	50
Median	20.55
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.61492598...
SemiVariance	28.2460044...
SemiDeviation	5.31469702...
Value at Risk 95%	-10.199999...
Cond. Value at Risk 95%	8.07647058...
Mean Confidence 95%	1.02930128...
Std. Dev. Confidence 95%	1.33175334...
Coefficient of Variation	0.41679823...
Standard Error	0.52429890...
Expected Loss	0

- **Frequency Charts:** The output variable frequency chart for the training partition opens automatically once the *KNNP_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *KNNP_TrainingScore* tab.   To move the dialog to a new location on the screen, simply grab the title bar and drag the dialog to the desired location. This chart displays a detailed, interactive frequency chart for the Actual variable data and the Predicted data, for the training partition.

  Predicted values for training partition, MEDV variable



MEDV

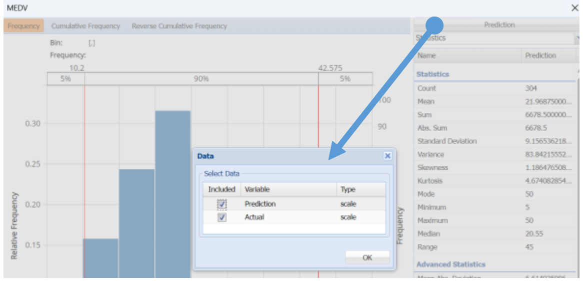
Bin: [20,25]

Frequency: 96

Name	Prediction
Statistics	
Count	304
Mean	21.96875000...
Sum	6678.500000...
Abs. Sum	6678.5
Standard Deviation	9.156536218...
Variance	83.84215552...
Skewness	1.186476508...
Kurtosis	4.674082854...
Mode	50
Minimum	5
Maximum	50
Median	20.55
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.614925986...
SemiVariance	28.24600444...
SemiDeviation	5.314697022...
Value at Risk 95%	-10.2
Cond. Value at Risk 95%	8.076470588...
Mean Confidence 95%	1.029301289...
Std. Dev. Confidence 95%	1.331753342...
Coefficient of Variation	0.416798234...
Standard Error	0.524298902...
Expected Loss	0

To display the predicted values and the actual data at the same time, click Prediction in the upper right hand corner and select both checkboxes in the Data dialog.

Click Prediction, then select Actual to display original data in the interactive chart



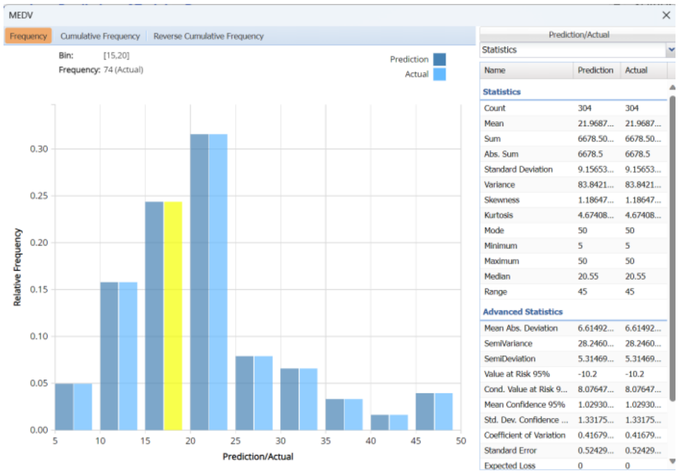
Data

Select Data

Included	Variable	Type
☑	Prediction	scale
☑	Actual	scale

Name	Prediction
Statistics	
Count	304
Mean	21.96875000...
Sum	6678.500000...
Abs. Sum	6678.5
Standard Deviation	9.156536218...
Variance	83.84215552...
Skewness	1.186476508...
Kurtosis	4.674082854...
Mode	50
Minimum	5
Maximum	50
Median	20.55
Range	45
Advanced Statistics	

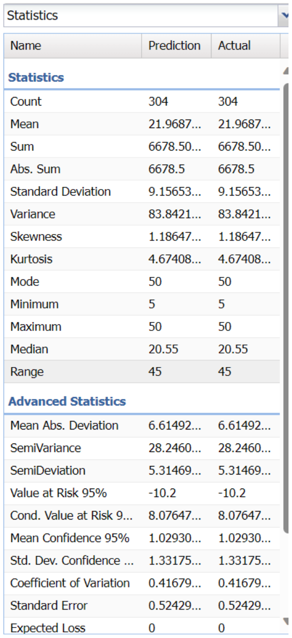
Actual vs Predicted values for Training Partition, MEDV variable



Name	Prediction	Actual
Statistics		
Count	304	304
Mean	21.9687...	21.9687...
Sum	6678.50...	6678.50...
Abs. Sum	6678.5	6678.5
Standard Deviation	9.15653...	9.15653...
Variance	83.8421...	83.8421...
Skewness	1.18647...	1.18647...
Kurtosis	4.67408...	4.67408...
Mode	50	50
Minimum	5	5
Maximum	50	50
Median	20.55	20.55
Range	45	45
Advanced Statistics		
Mean Abs. Deviation	6.61492...	6.61492...
SemiVariance	28.2460...	28.2460...
SemiDeviation	5.31469...	5.31469...
Value at Risk 95%	-10.2	-10.2
Cond. Value at Risk 9...	8.07647...	8.07647...
Mean Confidence 95%	1.02930...	1.02930...
Std. Dev. Confidence ...	1.33175...	1.33175...
Coefficient of Variation	0.41679...	0.41679...
Standard Error	0.52429...	0.52429...
Expected Loss	0	0

Notice in the screenshot below that both the Actual and Prediction data appear in the chart together, and statistics for both data appear on the right.

Statistics Pane



Statistics

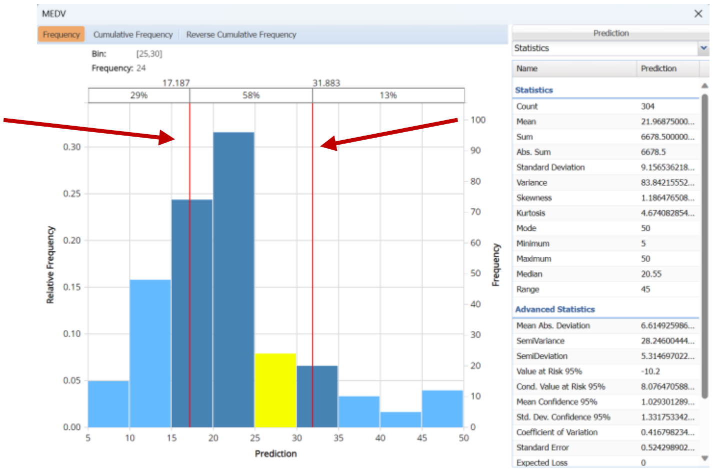
Name	Prediction	Actual
Statistics		
Count	304	304
Mean	21.9687...	21.9687...
Sum	6678.50...	6678.50...
Abs. Sum	6678.5	6678.5
Standard Deviation	9.15653...	9.15653...
Variance	83.8421...	83.8421...
Skewness	1.18647...	1.18647...
Kurtosis	4.67408...	4.67408...
Mode	50	50
Minimum	5	5
Maximum	50	50
Median	20.55	20.55
Range	45	45
Advanced Statistics		
Mean Abs. Deviation	6.61492...	6.61492...
SemiVariance	28.2460...	28.2460...
SemiDeviation	5.31469...	5.31469...
Value at Risk 95%	-10.2	-10.2
Cond. Value at Risk 9...	8.07647...	8.07647...
Mean Confidence 95%	1.02930...	1.02930...
Std. Dev. Confidence ...	1.33175...	1.33175...
Coefficient of Variation	0.41679...	0.41679...
Standard Error	0.52429...	0.52429...
Expected Loss	0	0

To remove either the Actual or Prediction data from the chart, click Prediction/Actual in the top right and then uncheck the data type to be removed.

This chart behaves the same as the interactive chart in the Analyze Data feature found on the Explore menu (and explained in depth in the Data Science Reference Guide).

- Use the mouse to hover over any of the bars in the graph to populate the Bin and Frequency headings at the top of the chart.
- When displaying either Original or Synthetic data (not both), red vertical lines will appear at the 5% and 95% percentile values in all three charts (Frequency, Cumulative Frequency and Reverse Cumulative Frequency) effectively displaying the 90th confidence interval. The middle percentage is the percentage of all the variable values that lie within the 'included' area, i.e. the darker shaded area. The two percentages on each end are the percentage of all variable values that lie outside of the 'included' area or the "tails". i.e. the lighter shaded area. Percentile values can be altered by moving either red vertical line to the left or right.

Frequency Chart for MEDV Prediction with red percentile lines moved



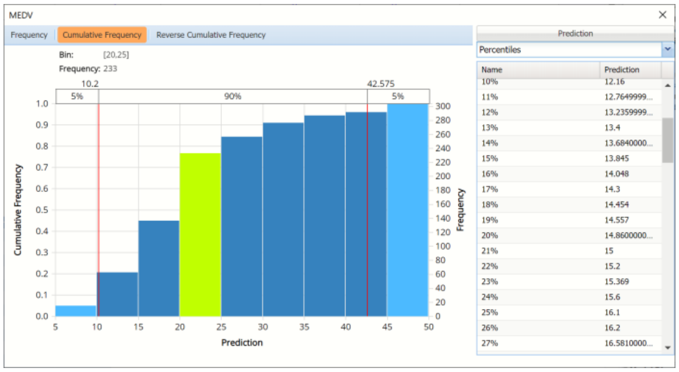
Prediction

Statistics

Name	Prediction
Statistics	
Count	304
Mean	21.96875000...
Sum	6678.500000...
Abs. Sum	6678.5
Standard Deviation	9.156536218...
Variance	83.84215552...
Skewness	1.186476508...
Kurtosis	4.674082854...
Mode	50
Minimum	5
Maximum	50
Median	20.55
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.614925986...
SemiVariance	28.24600444...
SemiDeviation	5.314697022...
Value at Risk 95%	-10.2
Cond. Value at Risk 95%	8.076470588...
Mean Confidence 95%	1.029301289...
Std. Dev. Confidence 95%	1.331753342...
Coefficient of Variation	0.416798234...
Standard Error	0.524298902...
Expected Loss	0

- Click Cumulative Frequency and Reverse Cumulative Frequency tabs to see the Cumulative Frequency and Reverse Cumulative Frequency charts, respectively. Select Percentiles from the drop down menu to view Percentile values.

Cumulative Frequency chart with Percentiles displayed



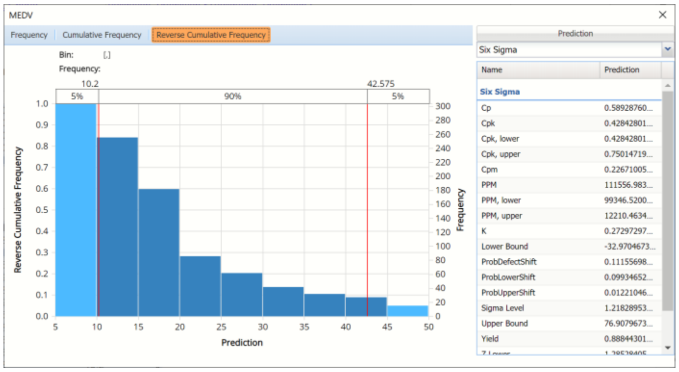
Prediction

Percentiles

Name	Prediction
10%	12.16
11%	12.7649999...
12%	13.2359999...
13%	13.4
14%	13.6840000...
15%	13.845
16%	14.048
17%	14.3
18%	14.454
19%	14.557
20%	14.8600000...
21%	15
22%	15.2
23%	15.369
24%	15.6
25%	16.1
26%	16.2
27%	16.5810000...

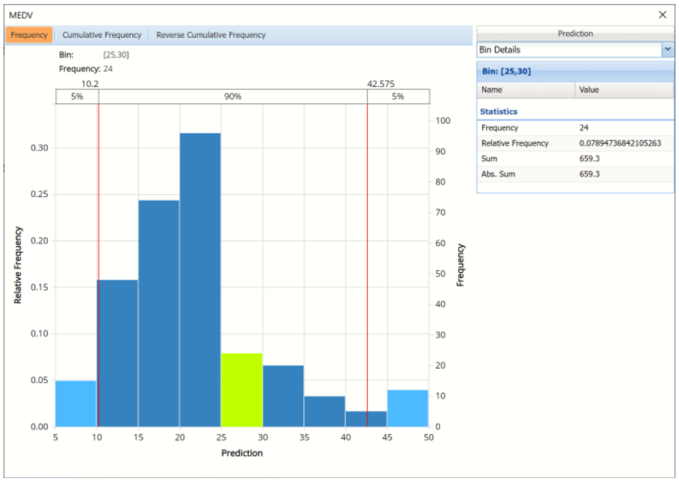
- Select Six Sigma from the drop down menu to view the Six Sigma indices.

  Reverse Cumulative Frequency chart with Six Sigma indices displayed

Name	Prediction
Six Sigma	
Cp	0.58928760...
Cpk	0.42842801...
Cpk, lower	0.42842801...
Cpk, upper	0.75014719...
Cpm	0.22671005...
PPM	111556.983...
PPM, lower	99346.5200...
PPM, upper	12210.4634...
K	0.27297297...
Lower Bound	-32.9704673...
ProbDefectShift	0.11155698...
ProbLowerShift	0.09934652...
ProbUpperShift	0.01221046...
Sigma Level	1.21828953...
Upper Bound	76.9079673...
Yield	0.88844301...
Z Lower	1.28529405...

  Select Bin Details from the drop down menu to view Bin Details for each bin in the chart. Use the Chart Options view to manually select the number of bins to use in the chart, as well as to set personalization options.

  Bin: [25,30]

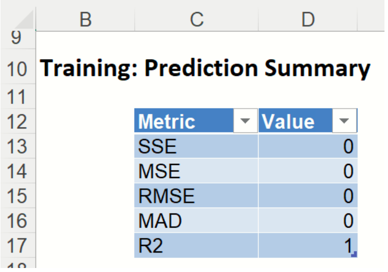
Name	Value
Statistics	
Frequency	24
Relative Frequency	0.07894736842105263
Sum	659.3
Abs. Sum	659.3

  As discussed above, see the Analyze Data section of the Exploring Data chapter in the Data Science Reference Guide for an in-depth discussion of this chart as well as descriptions of all statistics, percentiles, bin details and six sigma indices.

- **Prediction Summary:** A key interest in a data-mining context will be the predicted and actual values for the MEDV variable along with the residual (difference) for each predicted value in the Training partition.

  The *Training: Prediction Summary* report summarizes the prediction error. The first number, the total sum of squared errors, is the sum of the squared deviations (residuals) between the predicted and actual values. The second is the average of the squared residuals, the third is the square root of the average of the squared residuals and the fourth is the average deviation. All these values are calculated for the best k, i.e. k=6. Note that the algorithm perfectly predicted the correct median selling price for each census tract in the training partition.

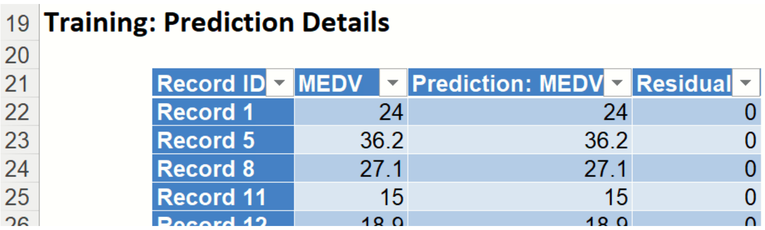
Training Prediction Summary



Training: Prediction Summary

Metric	Value
SSE	0
MSE	0
RMSE	0
MAD	0
R2	1

- **Prediction Details** displays the predicted value, the actual value and the difference between them (the residuals), for each record.



Training: Prediction Details

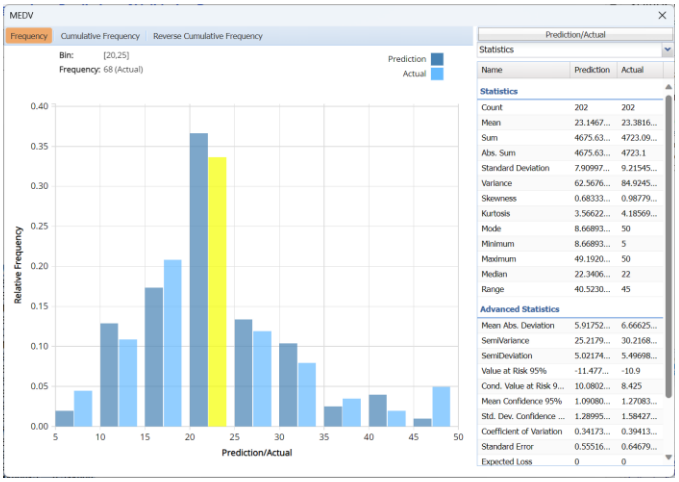
Record ID	MEDV	Prediction: MEDV	Residual
Record 1	24	24	0
Record 5	36.2	36.2	0
Record 8	27.1	27.1	0
Record 11	15	15	0
Record 12	18.9	18.9	0

KNNP_ValidationScore

A key interest in a data-mining context will be the predicted and actual values for the MEDV variable along with the residual (difference) for each predicted value in the Validation partition.

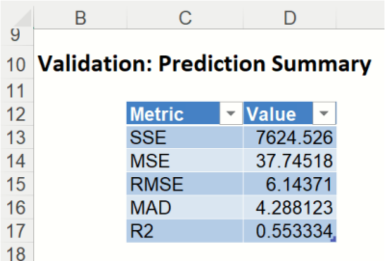
KNNP_ValidationScore displays the newly added Output Variable frequency chart, the Validation: Prediction Summary and the Validation: Prediction Details report. All calculations, charts and predictions on the KNNP_ValidationScore output sheet apply to the Validation partition.

- **Frequency Charts:** The output variable frequency chart for the validation partition opens automatically once the *KNNP_ValidationScore* worksheet is selected. This chart displays a detailed, interactive frequency chart for the Actual variable data and the Predicted data, for the validation partition. For more information on this chart, see the KNNP_TrainingScore explanation above.



Name	Prediction	Actual
Statistics		
Count	202	202
Mean	23.1467...	23.3816...
Sum	4675.63...	4723.09...
Abs. Sum	4675.63...	4723.1
Standard Deviation	7.90997...	9.21545...
Variance	62.5676...	84.9245...
Skewness	0.68333...	0.98779...
Kurtosis	3.56622...	4.18569...
Mode	8.66893...	50
Minimum	8.66893...	5
Maximum	49.1920...	50
Median	22.3406...	22
Range	40.5230...	45
Advanced Statistics		
Mean Abs. Deviation	5.91752...	6.66625...
SemiVariance	25.2179...	30.2168...
SemiDeviation	5.02174...	5.49698...
Value at Risk 95%	-11.477...	-10.9
Cond. Value at Risk 9...	10.0802...	8.425
Mean Confidence 95%	1.09080...	1.27083...
Std. Dev. Confidence ...	1.28995...	1.58427...
Coefficient of Variation	0.34173...	0.39413...
Standard Error	0.55516...	0.64679...
Expected Loss	0	0

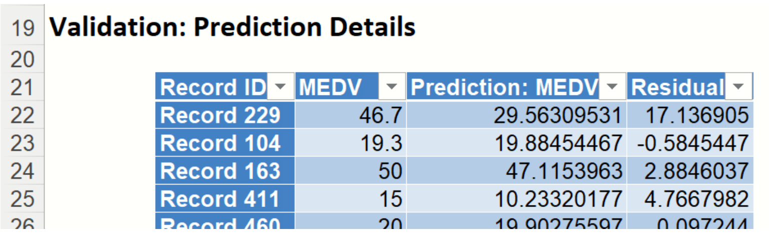
- **Prediction Summary:** In the Prediction Summary report, Analytic Solver Data Science displays the total sum of squared errors summaries for the Validation partition.



Validation: Prediction Summary

Metric	Value
SSE	7624.526
MSE	37.74518
RMSE	6.14371
MAD	4.288123
R2	0.553334

- **Prediction Details:** Scroll down to the Validation: Prediction Details report to find the Prediction value for the MEDV variable for each record, as well as the Residual value in the Validation partition.



Validation: Prediction Details

Record ID	MEDV	Prediction: MEDV	Residual
Record 229	46.7	29.56309531	17.136905
Record 104	19.3	19.88454467	-0.5845447
Record 163	50	47.1153963	2.8846037
Record 411	15	10.23320177	4.7667982
Record 460	20	19.90275597	0.097244

KNNP_TrainingLiftChart and KNNP_ValidationLiftChart

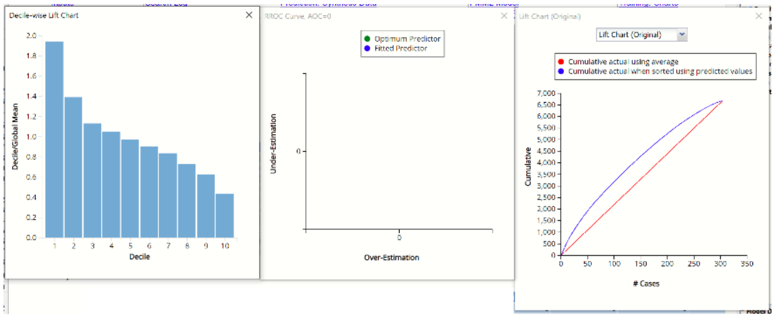
Lift charts and RROC Curves (on the *KNNP_TrainingLiftChart* and *KNNP_ValidationLiftChart* tabs, respectively) are visual aids for measuring model performance. Lift Charts consist of a lift curve and a baseline. The greater the area between the lift curve and the baseline, the better the model. RROC (regression receiver operating characteristic) curves plot the performance of regressors by graphing over-estimations (or predicted values that are too high) versus underestimations (or predicted values that are too low.) The closer the curve is to the top left corner of the graph (in other words, the smaller the area above the curve), the better the performance of the model.

After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted in descending order using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the number of cases versus the cumulated value. The baseline (red line connecting the origin to the end point of the blue line) is drawn as the number of cases versus the average of actual output variable values multiplied by the number of cases.

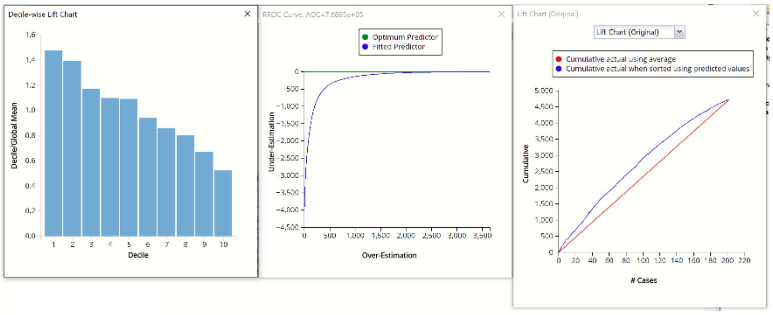
The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. The bars in this chart indicate the factor by which the kNNP model outperforms a random assignment, one decile at a time. Refer to the validation graph below. In the first decile in both the training and validation datasets, taking the most expensive predicted housing prices in the dataset, the predictive performance of the model is about 1.8 times better as simply assigning a random predicted value.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select KNNP_TrainingLiftChart or KNNP_ValidationLiftChart for Worksheet and Decile Chart, RROC Chart or Gain Chart for Chart.

Decile-Wise Lift Chart, RROC Curve and Lift Chart from Training Partition

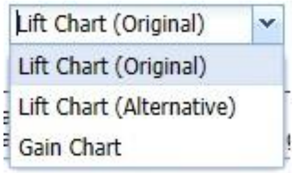


Decile-Wise Lift Chart, RROC Curve and Lift Chart from Validation Partition



In an RROC curve, we can compare the performance of a regressor with that of a random guess (red line) for which under estimations are equal to over-estimations shifted to the minimum under estimate. Anything to the left of this line signifies a better prediction and anything to the right signifies a worse prediction. The best possible prediction performance would be denoted by a point at the top left of the graph at the intersection of the x and y axis. Area Over the Curve (AOC) is the space in the graph that appears above the RROC curve and is calculated using the formula: sigma^2 * n^2/2 where n is the number of records The smaller the AOC, the better the performance of the model. The RROC Curve for the Training Partition is blank. This is because the KNN algorithm perfectly predicted the selling price in the training partition.

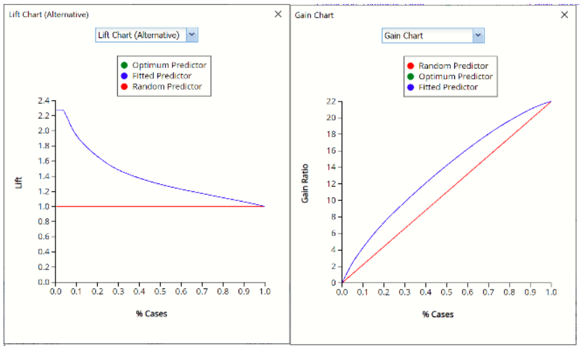
In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.



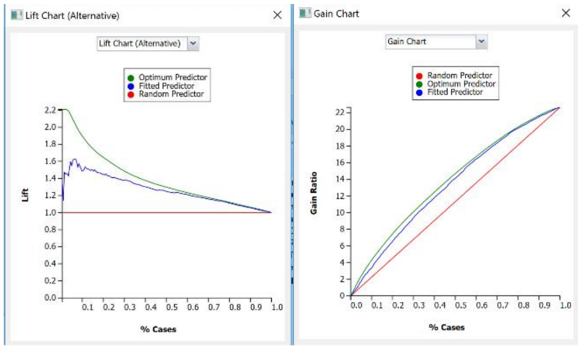
Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Predictor curve, a Fitted Predictor curve, and a Random Predictor curve. The Optimum Predictor curve plots a hypothetical model that would provide a perfect fit to the data. The Fitted Predictor curve plots the fitted model and the Random Predictor curve plots the results from using no model or by using a random guess.

The Alternative Lift Chart plots Lift against % Cases. The Gain Chart plots the Gain Ratio against % Cases.

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition

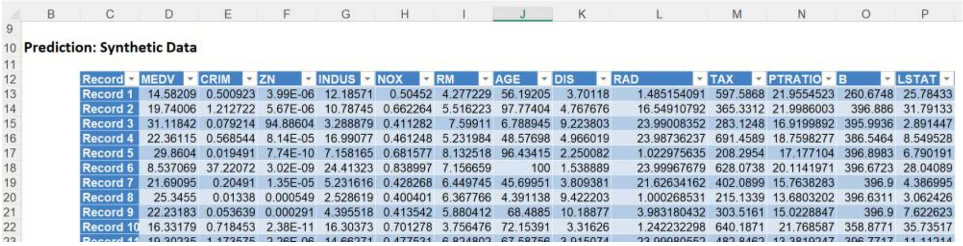


KNNP_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, KNNP_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the k-Nearest Neighbors dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predicted values for the training partition (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



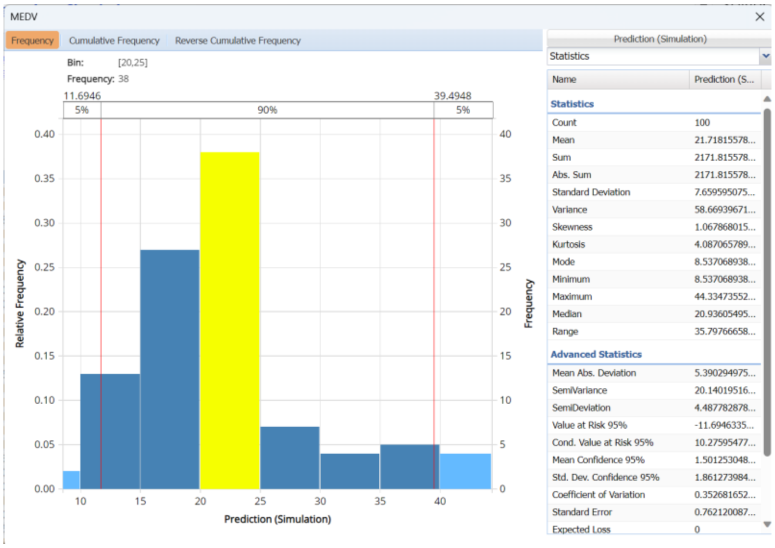
Prediction: Synthetic Data

Record	MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	14.58209	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.70118	1.485154091	597.5868	21.9554523	260.6748	25.78433
Record 2	19.74006	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767676	16.54910792	365.3312	21.9986003	396.886	31.79133
Record 3	31.11842	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223803	23.99008352	283.1248	16.9199892	395.9936	2.891447
Record 4	22.36115	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966019	23.98736237	691.4589	18.7598277	386.5464	8.549528
Record 5	29.8604	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082	1.022975635	208.2954	17.177104	396.8983	6.790191
Record 6	8.537069	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538889	23.99967679	628.0738	20.1141971	396.6723	28.04089
Record 7	21.69095	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809381	21.62634162	402.0899	15.7638283	396.9	4.386995
Record 8	25.3455	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422203	1.000268531	215.1339	13.6803202	396.6311	3.062426
Record 9	22.23183	0.053639	0.000291	4.395518	0.413542	5.880412	68.4885	10.18877	3.983180432	303.5161	15.0228847	396.9	7.622623
Record 10	16.33179	0.718453	2.38E-11	16.30373	0.701278	3.756476	72.15391	3.31626	1.242232298	640.1871	21.768587	358.8771	35.73517

The data contained in the Synthetic Data report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training data, the synthetic data and the expression, if it exists. (Recall that no expression was entered in this example.)

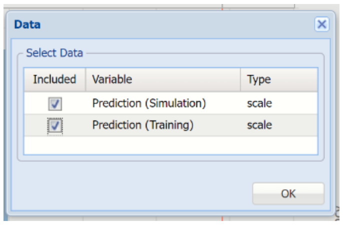
Frequency Chart for Prediction (Simulation) data



Name	Prediction (S...
Statistics	
Count	100
Mean	21.71815578...
Sum	2171.815578...
Abs. Sum	2171.815578...
Standard Deviation	7.659595075...
Variance	58.66939671...
Skewness	1.067868015...
Kurtosis	4.087065789...
Mode	8.537068938...
Minimum	8.537068938...
Maximum	44.33473552...
Median	20.93605495...
Range	35.79766658...
Advanced Statistics	
Mean Abs. Deviation	5.390294975...
SemiVariance	20.14019516...
SemiDeviation	4.487782878...
Value at Risk 95%	-11.6946335...
Cond. Value at Risk 95%	10.27595477...
Mean Confidence 95%	1.501253048...
Std. Dev. Confidence 95%	1.861273984...
Coefficient of Variation	0.352681652...
Standard Error	0.762120087...
Expected Loss	0

To change the data view, click the *Prediction (Simulation)* button. Select *Prediction (Training) and Prediction (Simulation)* to add the training data to the chart.

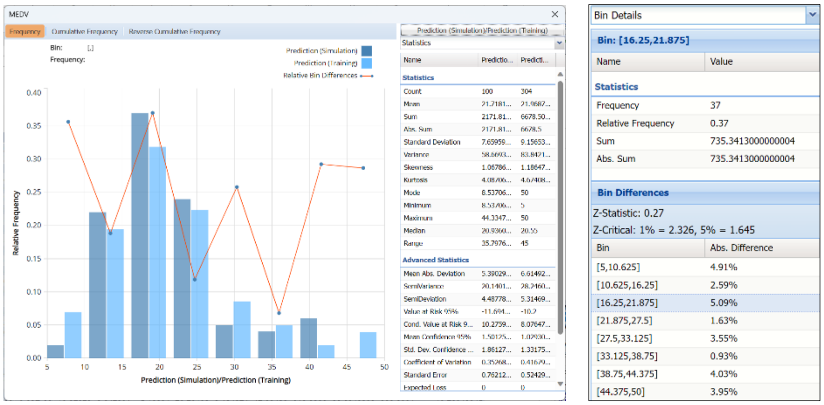
Data Dialog



Included	Variable	Type
☑	Prediction (Simulation)	scale
☑	Prediction (Training)	scale

In the chart below, the darker blue bars represent the predictions for the synthetic data while the lighter blue bars represent the predictions for the training data.

Prediction (Simulation) and Prediction (Training) Frequency chart for MEDV variable



Name	Predictio...	Predicti...
Statistics		
Count	100	304
Mean	21.7181...	21.9687...
Sum	2171.81...	6678.50...
Abs. Sum	2171.81...	6678.5
Standard Deviation	7.65959...	9.15653...
Variance	58.6693...	83.8421...
Skewness	1.06786...	1.18647...
Kurtosis	4.08706...	4.67408...
Mode	8.53706...	50
Minimum	8.53706...	5
Maximum	44.3347...	50
Median	20.9360...	20.55
Range	35.7976...	45
Advanced Statistics		
Mean Abs. Deviation	5.39029...	6.61492...
SemiVariance	20.1401...	28.2460...
SemiDeviation	4.48778...	5.31469...
Value at Risk 95%	-11.694...	-10.2
Cond. Value at Risk 9...	10.2759...	8.07647...
Mean Confidence 95%	1.50125...	1.02930...
Std. Dev. Confidence ...	1.86127...	1.33175...
Coefficient of Variation	0.35268...	0.41679...
Standard Error	0.76212...	0.52429...
Expected Loss	0	0

Bin: [16.25,21.875]

Name	Value
Statistics	
Frequency	37
Relative Frequency	0.37
Sum	735.3413000000004
Abs. Sum	735.3413000000004

Bin Differences

Z-Statistic: 0.27

Z-Critical: 1% = 2.326, 5% = 1.645

Bin	Abs. Difference
[5,10.625]	4.91%
[10.625,16.25]	2.59%
[16.25,21.875]	5.09%
[21.875,27.5]	1.63%
[27.5,33.125]	3.55%
[33.125,38.75]	0.93%
[38.75,44.375]	4.03%
[44.375,50]	3.95%

The Relative Bin Differences curve charts the absolute differences between the data in each bin. Click the down arrow next to Statistics to view the Bin Details pane to display the calculations.

Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

KNNP_Stored

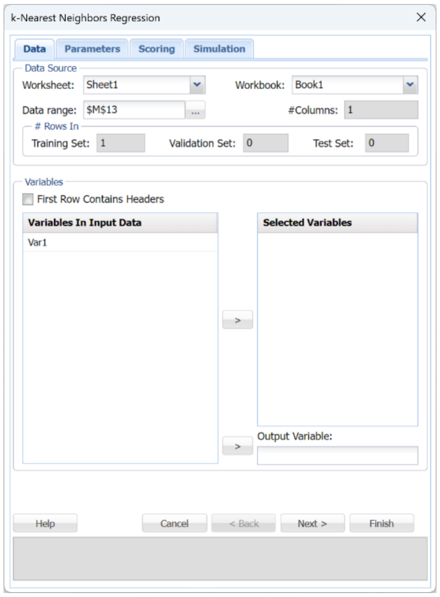
For information on Stored Model Sheets, in this example *KNNP_Stored*, please refer to the “Scoring New Data” chapter that apperas later in this guide.

k-Nearest Neighbors Regression Method Options

The following options appear on the four k-Nearest Neighbors dialog tabs.

k-Nearest Neighbors Regression Dialog, Data tab

k-Nearest Neighbors Regression Dialog, Data tab



k-Nearest Neighbors Regression

Data Source

Worksheet: Sheet1 Workbook: Book1

Data range: $M$13 #Columns: 1

Rows In

Training Set: 1 Validation Set: 0 Test Set: 0

Variables

□ First Row Contains Headers

Variables In Input Data: Var1

Selected Variables

Output Variable:

Variables In Input Data

All variables in the dataset are listed here.

Selected Variables

Select variables to be included in the model here.

Output Variable

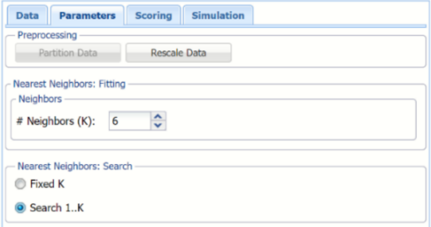
Select the continous variable whose outcome is to be predicted here.

k-Nearest Neighbors Regression Dialog, Parameters tab

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by selecting Partition Options on the Parameters tab. If this option is selected, Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the prediction method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

k-Nearest Neighbors Regression Dialog, Parameters tab



Preprocessing

Nearest Neighbors: Fitting

Neighbors

Neighbors (K): 6

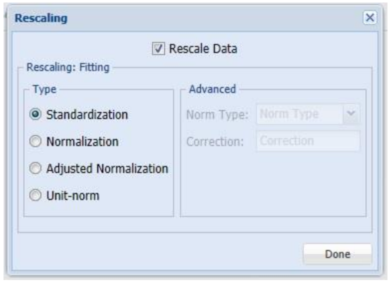
Nearest Neighbors: Search

○ Fixed K

● Search 1..K

Rescale Data

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

Advanced

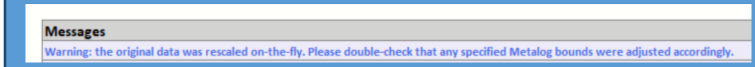
Norm Type: Norm Type

Correction: Correction

Done

Notes on Rescaling and the Simulation functionality

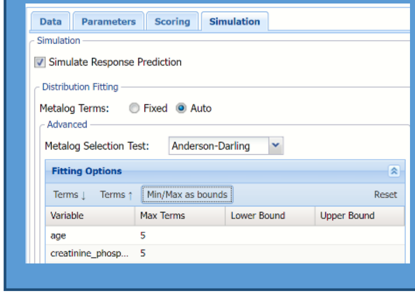
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the KNNP_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

# Neighbors (k)

This is the parameter k in the k-nearest neighbor algorithm. If the number of observations (rows) is less than 50 then the value of k should be between 1 and the total number of observations (rows). If the number of rows is greater than 50, then the value of k should be between 1 and 50. The default value is 1.

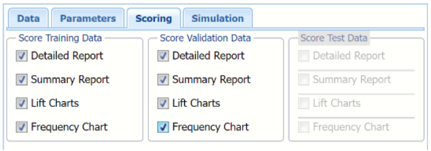
Nearest Neighbors Search

If *Search 1..K* is selected, Analytic Solver Data Science will display the output for the best k between 1 and the value entered for *# Neighbors (k).*

If *Fixed K* selected, the output will be displayed for the specified value of k. This is the default setting.

k-Nearest Neighbors Regression Dialog, Scoring tab

k-Nearest Neighbors Regression Dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
- ☑ Detailed Report
- ☑ Summary Report
- ☑ Lift Charts
- ☑ Frequency Chart

Score Validation Data
- ☑ Detailed Report
- ☑ Summary Report
- ☑ Lift Charts
- ☑ Frequency Chart

Score Test Data
- □ Detailed Report
- □ Summary Report
- □ Lift Charts
- □ Frequency Chart

Score Training Data

Select these options to show an assessment of the performance of the k-Nearest Neighbors Prediction algorithm in predicting the training data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts.

When Frequency Chart is selected, a frequency chart will be displayed when the KNNP_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the k-Nearest Neighbors Prediction algorithm in predicting the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the KNNP_ValidationScore worksheet is selected.

Score Test Data

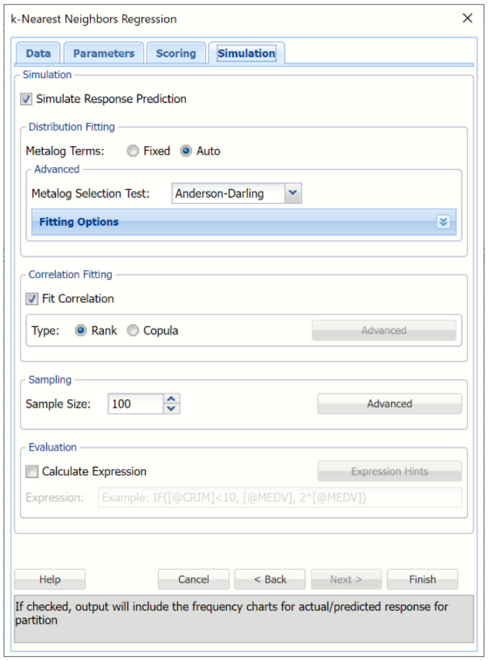
These options are enabled when a test set is present. Select these options to show an assessment of the performance of the k-Nearest Neighbors Prediction algorithm in predicting the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the KNNP_TestScore worksheet is selected.

Score New Data

See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score New Data* groups.

k-Nearest Neighbors Regression Dialog, Simulation tab

k-Nearest Neighbors Regression Dialog, Simulation tab



k-Nearest Neighbors Regression

Data | Parameters | Scoring | Simulation

Simulation
- ☑ Simulate Response Prediction

Distribution Fitting
- Metalog Terms: ○ Fixed ● Auto
- Advanced
  - Metalog Selection Test: Anderson-Darling
  - Fitting Options

Correlation Fitting
- ☑ Fit Correlation
- Type: ● Rank ○ Copula

Sampling
- Sample Size: 100

Evaluation
- □ Calculate Expression
- Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

If checked, output will include the frequency charts for actual/predicted response for partition

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, KNNP_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the KNNP_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Regression Tree Method

Introduction

As with all regression techniques, Analytic Solver Data Science assumes the existence of a single output (response) variable and one or more input (predictor) variables. The output variable is numerical. The general regression tree building methodology allows input variables to be a mixture of continuous and categorical variables. A decision tree is generated where each decision node in the tree contains a test on some input variable's value. The terminal nodes of the tree contain the predicted output variable values.

A Regression tree may be considered as a **variant of decision trees**, designed to **approximate real-valued functions** instead of being used for classification methods.

Methodology

A Regression tree is built through a process known as binary recursive partitioning. This is an iterative process that splits the data into partitions or “branches”, and then continues splitting each partition into smaller groups as the method moves up each branch.

Initially, all records in the training set (the pre-classified records that are used to determine the structure of the tree) are grouped into the same partition. The algorithm then begins allocating the data into the first two partitions or “branches”, using every possible binary split on every field. The algorithm selects the split that minimizes the sum of the squared deviations from the mean in the two separate partitions. This splitting “rule” is then applied to each of the new branches. This process continues until each node reaches a user-specified minimum node size and becomes a terminal node. (If the sum of squared deviations from the mean in a node is zero, then that node is considered a terminal node even if it has not reached the minimum size.)

Pruning the tree

Since the tree is grown from the training data set, a fully developed tree typically suffers from over-fitting (i.e. it is "explaining" random elements of the training data that are not likely to be features of the larger population). This over-fitting results in poor performance on “real life” data. Therefore, the tree must be “pruned” using the validation data set. Analytic Solver Data Science calculates the cost complexity factor at each step during the growth of the tree and decides the number of decision nodes in the pruned tree. The cost complexity factor is the multiplicative factor that is applied to the size of the tree (which is measured by the number of terminal nodes).

The tree is pruned to minimize the sum of (1) the output variable variance in the validation data, taken one terminal node at a time, and (2) the product of the cost complexity factor and the number of terminal nodes. If the cost complexity factor is specified as zero then pruning is simply finding the tree that performs best on validation data in terms of total terminal node variance. Larger values of the cost complexity factor result in smaller trees. Pruning is performed on a “last in first out" basis meaning the last grown node is the first to be subject to elimination.

Single Tree Regression Tree Example

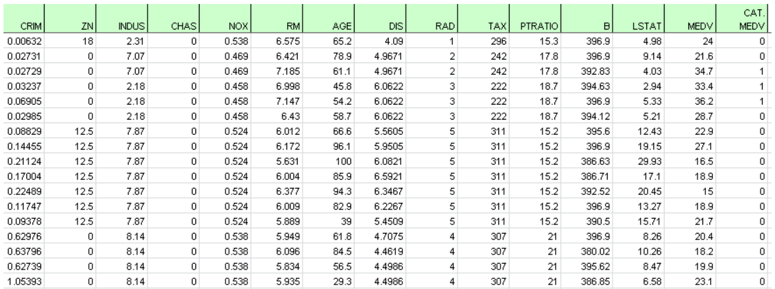
Analytic Solver Data Science includes four different methods for creating regression trees: boosting, bagging, random trees, and single tree. The first three (boosting, bagging, and random trees) are ensemble methods that are used to generate one powerful model by combining several "weaker" tree models. For information on these methods, please see the Ensemble Methods chapter that occurs later in this guide.

This example illustrates how to use the Regression Tree algorithm using a single tree. We will use the Boston_Housing.xlsx dataset to illustrate this method. See below for examples using bagging, random trees and single trees.

1. Click **Help – Example Models**, then **Forecasting/Data Science Examples** to open the **Boston_Housing.xlsx** dataset. This dataset includes fourteen variables pertaining to housing prices from census tracts in the Boston area. This dataset was collected by the US Census Bureau in the 1940's. See the Data sheet, within the example worksheet, for a description of all variables.

   The figure below displays a portion of the data; observe the last column (CAT. MEDV). This variable has been derived from the MEDV variable by assigning a 1 for MEDV levels above 30 (>= 30) and a 0 for levels below 30 (<30) and will not be used in these examples. The variable will not be used in this example.

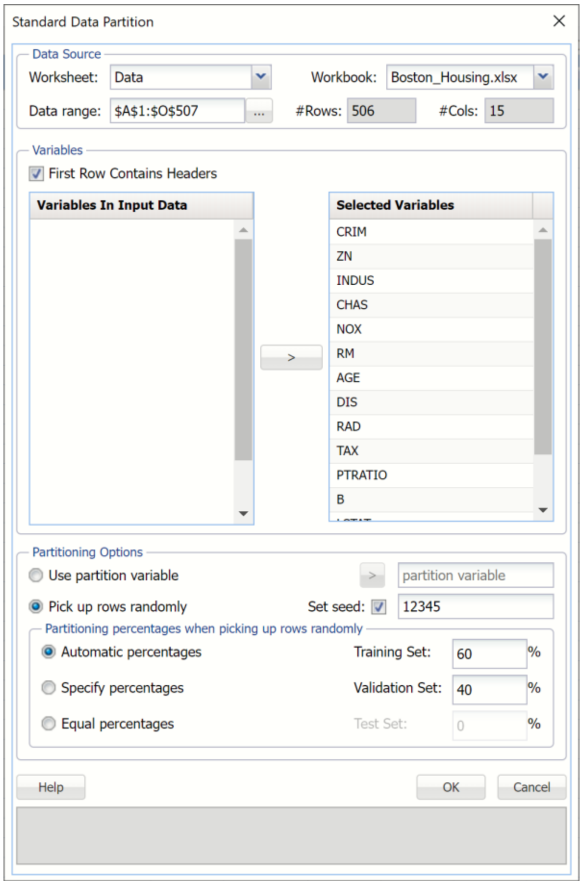
   Boston Housing example dataset



CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT. MEDV
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24	0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	0
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	1
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	1
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	1
0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7	0
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	0
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	0
0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	0
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	0
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	0
0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	0
0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	0
0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	0
0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	0
0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	0
1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	0

2. Partition the data into training and validation sets using the Standard Data Partition defaults of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.

Standard Data Partition dialog



Standard Data Partition

Data Source

Worksheet: Data

Workbook: Boston_Housing.xlsx

Data range: $A$1:$O$507

#Rows: 506

#Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data	Selected Variables
	CRIM
	ZN
	INDUS
	CHAS
	NOX
	RM
	AGE
	DIS
	RAD
	TAX
	PTRATIO
	B

Partitioning Options

○ Use partition variable

● Pick up rows randomly

Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages — Training Set: 60 %

○ Specify percentages — Validation Set: 40 %

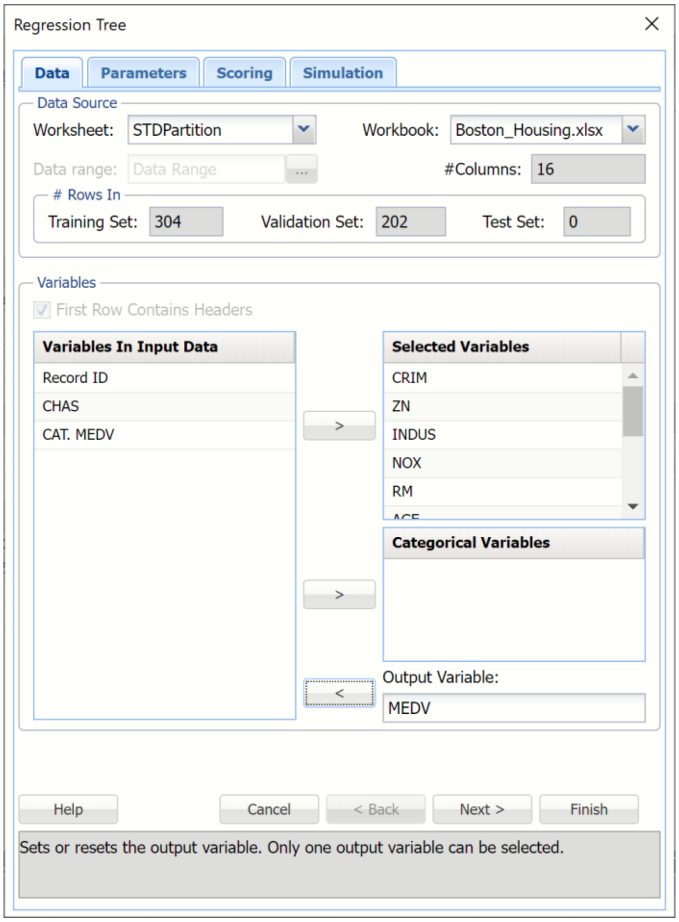
○ Equal percentages — Test Set: 0 %

3. Click **Predict – Regression Tree - Single Tree** to open the *Regression Tree – Data* tab.

Select **MEDV** as the *Output Variable*, then select the remaining variables (except *CAT.MEDV, RecordID, and CHAS*) as *Selected Variables.*

All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described in the What's New section of th Data Science User Guide and then more in depth earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. Since this new functionality does not support categorical variables, the CHAS variable will not be included in the model.

Regression Tree dialog, Data tab



Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Training Set: 304

Validation Set: 202

Test Set: 0

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM
AGE

Categorical Variables

Output Variable: MEDV

Sets or resets the output variable. Only one output variable can be selected.

4. Click **Next** to advance to the *Regression Tree – Parameters* tab.

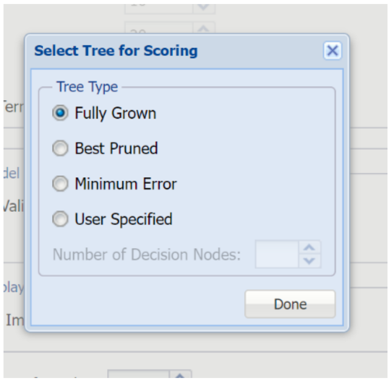
   As discussed in previous sections, Analytic Solver Data Science includes the ability to partition and rescale a dataset "on-the-fly" from within a classification or prediction method by clicking Partition Data and/or Rescale Data on the Parameters tab. Analytic Solver Data Science will partition or rescale your dataset (according to the partition and rescaling options you set) immediately before running the regression method. If partitioning or rescaling has already occurred on the dataset, these options will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter. For more information on rescaling your data, please see the Transform Continuous Data chapter.

5. In the *Tree Growth* section, leave all selections at their default settings. Values entered for these options limit tree growth, i.e. if 10 is entered for Levels, the tree will be limited to 10 levels.

6. Select **Prune (Using Validation Set).** (This option is enabled when a Validation Dataset exists.) Analytic Solver Data Science will prune the tree using the validation set when this option is selected. (Pruning the tree using the validation set reduces the error from over-fitting the tree to the training data.)

7. Click **Tree for Scoring** and select **Fully Grown**.

Select the tree used for scoring (enabled when a validation partition exists)



Select Tree for Scoring

Tree Type

● Fully Grown

○ Best Pruned

○ Minimum Error

○ User Specified

Number of Decision Nodes:

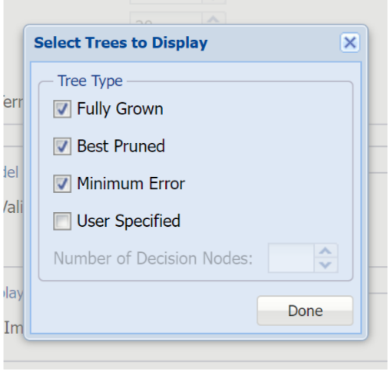
8. Select **Show Feature Importance**. This table shows the relative importance of the feature measured as the reduction of the error criterion during the tree growth.

9. Leave *Maximum Number of Levels* at the default setting of 7. This option specifies the maximum number of levels in the tree to be displayed in the output. Select **Trees to Display** to select the types of trees to display: Fully Grown, Best Pruned, Minimum Error or User Specified.

   - Select *Fully Grown* to "grow" a complete tree using the training data.
   - Select *Best Pruned* to create a tree with the fewest number of nodes, subject to the constraint that the error be kept below a specified level (minimum error rate plus the standard error of that error rate).
   - Select *Minimum error* to produce a tree that yields the minimum classification error rate when tested on the validation data.
   - To create a tree with a specified number of decision nodes select *User Specified* and enter the desired number of nodes.

   Select **Fully Grown**, **Best Pruned**, and **Minimum Error**.

   Select the tree(s) to display in the output.



Select Trees to Display

Tree Type

☑ Fully Grown

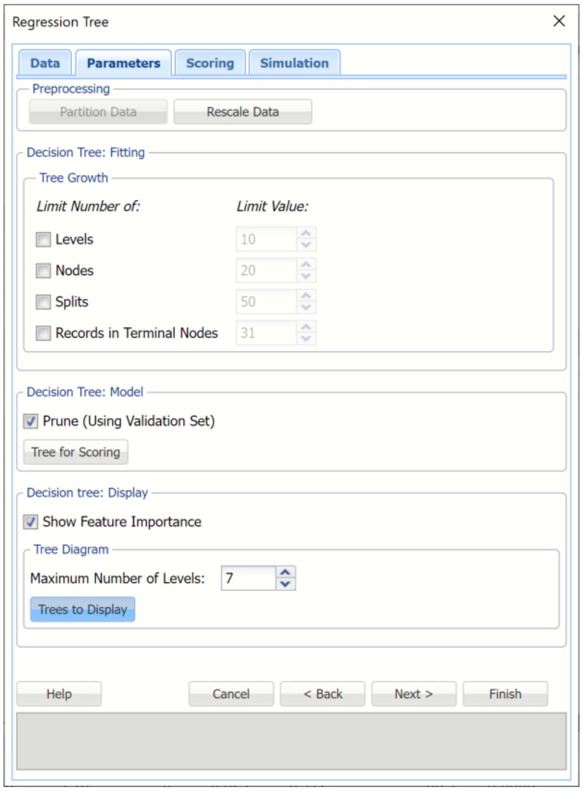
☑ Best Pruned

☑ Minimum Error

□ User Specified

Number of Decision Nodes:

Regression Tree dialog, Parameters tab



Regression Tree

Data | Parameters | Scoring | Simulation

Preprocessing

Partition Data | Rescale Data

Decision Tree: Fitting

Tree Growth

Limit Number of: *Limit Value:*

- □ Levels 10
- □ Nodes 20
- □ Splits 50
- □ Records in Terminal Nodes 31

Decision Tree: Model

☑ Prune (Using Validation Set)

Tree for Scoring

Decision tree: Display

☑ Show Feature Importance

Tree Diagram

Maximum Number of Levels: 7

Trees to Display

Help | Cancel | < Back | Next > | Finish

10. Select **Next** to advance to the *Scoring* tab.

18. Select all four options for **Score Training/Validation data**.

When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the Regression Trees output.

When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the Regression Trees output.

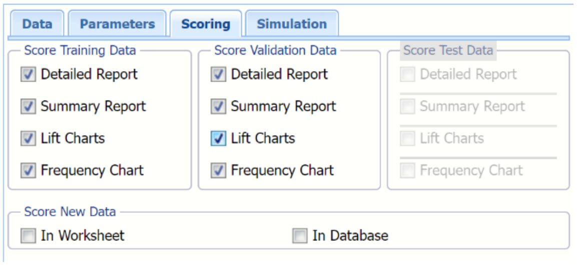
When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and ROC Curve plots in the output.

When Frequency Chart is selected, a frequency chart will be displayed when the RT_TrainingScore and RT_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Since we did not create a test partition, the options for Score test data are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.

Regression Tree dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

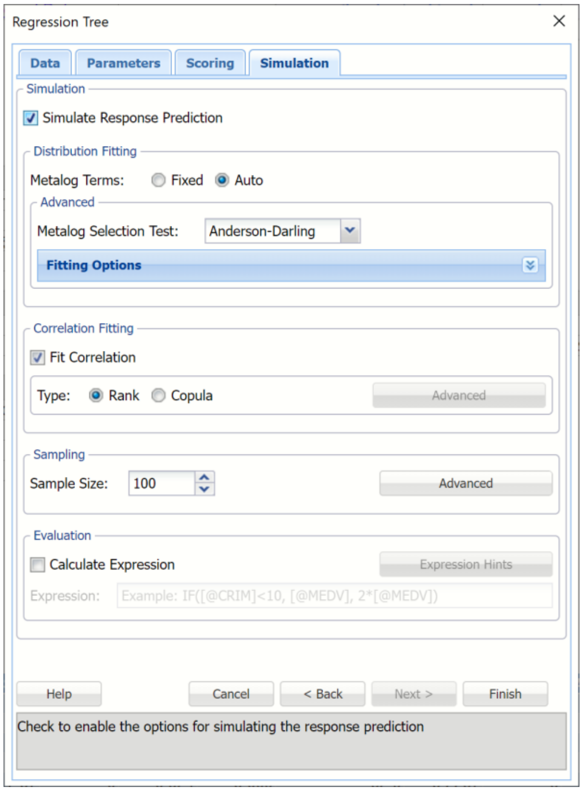
Score New Data
□ In Worksheet
□ In Database

19. Click **Next** to advance to the Simulation tab.

20. Select **Simulation Response Prediction** to enable all options on the Simulation tab of the Regression Tree dialog. This tab is disabled in Analtyic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, RT_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

    Regression Tree *dialog, Simulation tab*



Regression Tree

Data | Parameters | Scoring | Simulation

Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula
Advanced

Sampling
Sample Size: 100
Advanced

Evaluation
□ Calculate Expression
Expression Hints
Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

Help | Cancel | < Back | Next > | Finish

Check to enable the options for simulating the response prediction

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RT_Simulation output tab.

Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression. Note that variable names are case sensitive. See any of the prediction methods to see the Expressison field in use.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

21. Click **Finish** to run Regression Tree on the example dataset.

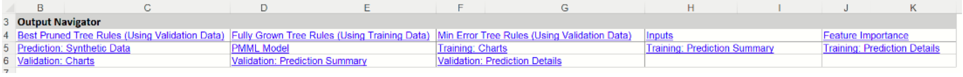
Output Worksheets

Output sheets containing the results for Regression Tree will be inserted into your active workbook to the right of the STDPartition worksheet.

RT_Output

Output from prediction method will be inserted to the right of the workbook. *RT_Output* includes 4 segments: Output Navigator, Inputs, Training Log and Feature Importance.

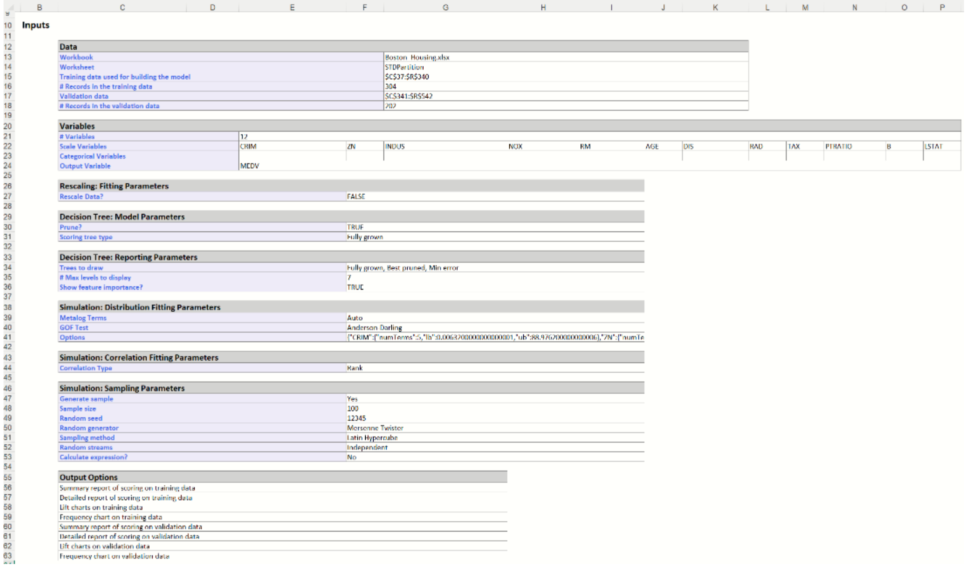
- **Output Navigator:** The Output Navigator appears at the top of each output worksheet. Use this feature to quickly navigate to all reports included in the output.



Output Navigator

Best Pruned Tree Rules (Using Validation Data)	Fully Grown Tree Rules (Using Training Data)	Min Error Tree Rules (Using Validation Data)	Inputs	Feature Importance
Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Prediction Summary	Training: Prediction Details
Validation: Charts	Validation: Prediction Summary	Validation: Prediction Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Regression Tree dialog.



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Decision Tree: Model Parameters	
Prune?	TRUE
Scoring tree type	Fully grown

Decision Tree: Reporting Parameters	
Trees to draw	Fully grown, Best pruned, Min error
# Max levels to display	7
Show feature importance?	TRUE

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson Darling
Options	{"CRIM":{"numTerms":5,"lb":0.0063200000000000001,"ub":88.976200000000006},"ZN":{"numTe

Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

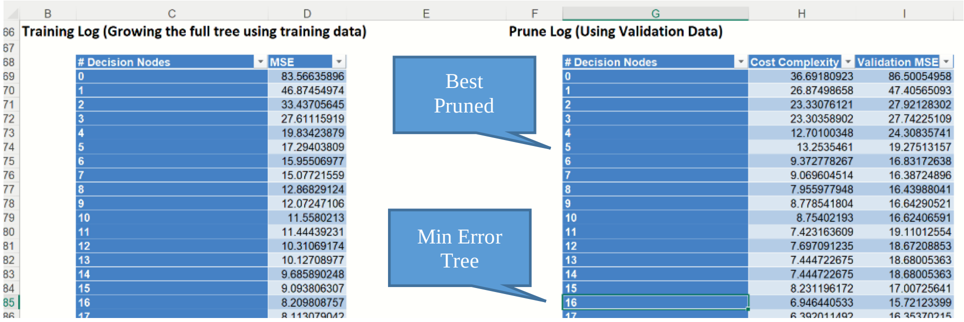
Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Training Log:** Scroll down to the **Training log** (shown below) to see the mean-square error (MSE) at each stage of the tree for both the training and validation data sets. The MSE value is the average of the squares of the errors between the predicted and observed values in the sample. The training log shows that the training MSE continues reducing as the tree continues to split.

Analytic Solver Data Science chooses the number of decision nodes for the pruned tree and the minimum error tree from the values of Validation MSE. In the Prune log shown below, the smallest Validation MSE error belongs to the tree with 16 decision nodes (MSE=15.72). This is the Minimum Error Tree – the tree with the smallest misclassification error in the validation dataset. The Best Pruned Tree is the smallest tree with an error within one standard error of the minimum error tree.

In this example, the Minimum Error Tree has a Cost Complexity = 6.95 and a Validation MSE = 15.72 while the Best Pruned Tree has 5 decision nodes.



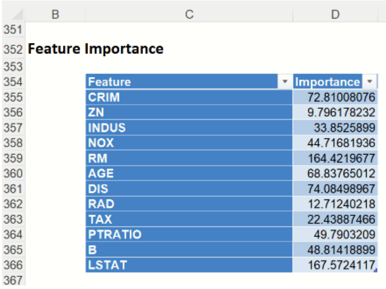
Training Log (Growing the full tree using training data)

# Decision Nodes	MSE
0	83.56635896
1	46.87454974
2	33.43705645
3	27.61115919
4	19.83423879
5	17.29403809
6	15.95506977
7	15.07721559
8	12.86829124
9	12.07247106
10	11.5580213
11	11.44439231
12	10.31069174
13	10.12708977
14	9.685890248
15	9.093806307
16	8.209808757
17	8.113079042

Prune Log (Using Validation Data)

# Decision Nodes	Cost Complexity	Validation MSE
0	36.69180923	86.50054958
1	26.87498658	47.40565093
2	23.33076121	27.92128302
3	23.30358902	27.74225109
4	12.70100348	24.30835741
5	13.2535461	19.27513157
6	9.372778267	16.83172638
7	9.069604514	16.38724896
8	7.955977948	16.43988041
9	8.778541804	16.64290521
10	8.75402193	16.62406591
11	7.423163609	19.11012554
12	7.697091235	18.67208853
13	7.444722675	18.68005363
14	7.444722675	18.68005363
15	8.231196172	17.00725641
16	6.946440533	15.72123399
17	6.392011492	16.35370215

Feature Importance: This table displays the variables that are included in the model along with their Importance value. The larger the Importance value, the bigger the influence the variable has on the predicted classification. In this instance, the census tracts with homes with many rooms will be predicted as having a larger selling price.

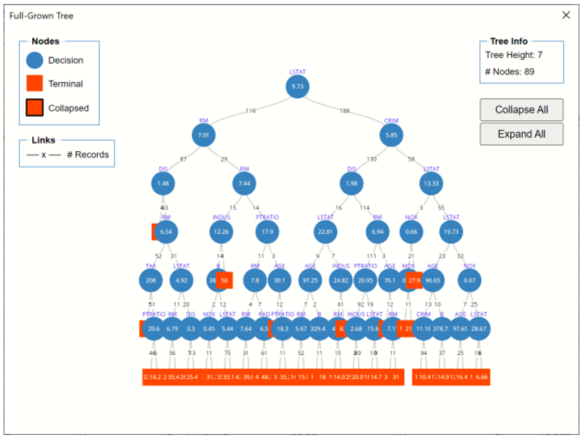


Feature Importance

Feature	Importance
CRIM	72.81008076
ZN	9.796178232
INDUS	33.8525899
NOX	44.71681936
RM	164.4219677
AGE	68.83765012
DIS	74.08498967
RAD	12.71240218
TAX	22.43887466
PTRATIO	49.7903209
B	48.81418899
LSTAT	167.5724117

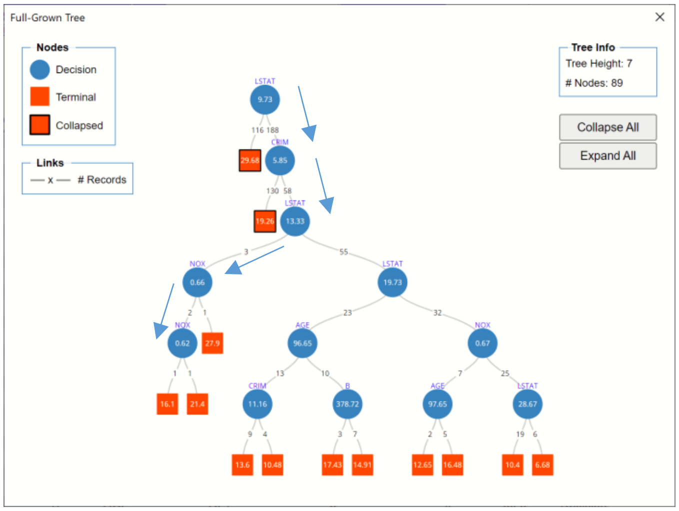
RT_FullTree

To view the Full Grown Tree, either click the Fully Grown Tree link in the Output Navigator or click the RT_FullTree worksheet tab. Recall that the Fully Grown Tree is the tree used to fit the Regression Tree model (using the Training data) and the tree used to score the validation partition.



Nodes may be collapsed for easier reading.

Here is an example of a path down the Fully Grown Tree fitted on the Training partition.



- LSTAT (% Lower Status of the Population) is chosen as the first splitting variable; if this value is >= 9.73 (188 cases), then CRIM (crime rate per capita) is chosen for splitting.
- If CRIM >= 5.85 (58 cases) then the LSTAT variable is again selected for splitting.
- If LSTAT < 13.33 (3 cases), then the NOX variable (concentration of nitric oxide) is selected as the splitting variable.
- If NOX is less than .66 (2 cases), then NOX is again selected for splitting. One record (record #433) in the training partition has a NOX variable value < 0.62 with an MEDV value = 16.1 and the other training partition record has a NOX variable value >= 0.62 with an MEDV variable = 21.4 (record #126). (Note record IDs found by looking up MEDV values of 16.1 and 21.4, respectively in STDPartition.)

This same path can be followed in the Tree Rules using the records under Training Cases.

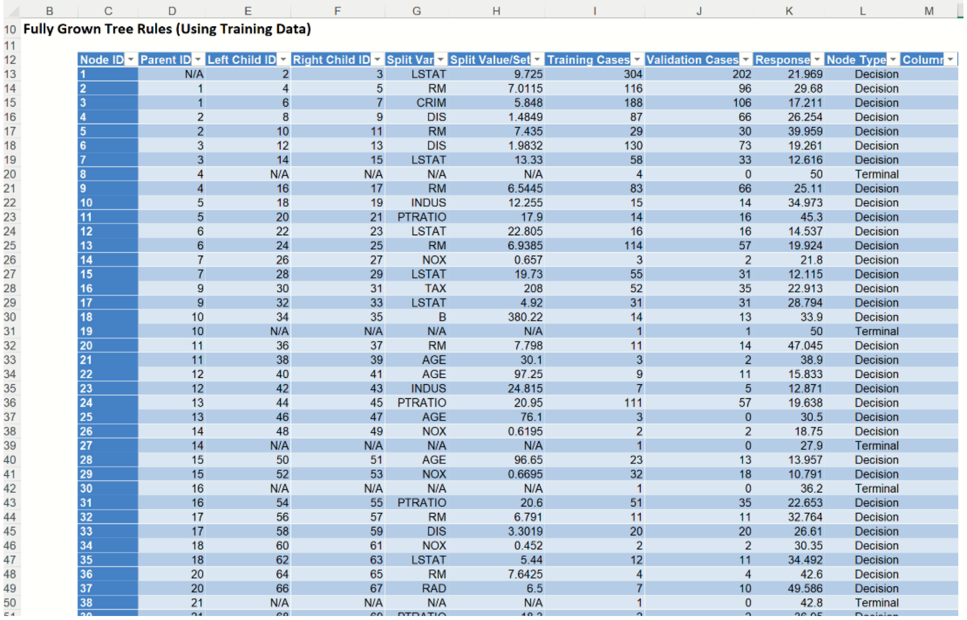
Node 1: All 304 cases in the training partition were split on the LSTAT variable using a splitting value of 9.725. The average value of the response variable (MEDV) for all 304 records is 21.969. From here, 116 cases were assigned to Node 2 and 188 cases were assigned to Node 3.

Node 3: 188 cases were assigned from Node 1. These cases were split on the CRIM variable using a splitting value of 5.848. The average value of the response variable for these 188 cases is 17.211 From here, 130 cases were assigned Node 6 and 58 cases were assigned Node 7.

Node 7: 58 cases were assigned to Node 7 from Node 3. These cases were split on the LSTAT variable (again) using a splitting value of 13.33. The average value of the response variable for these 58 cases is 12.616. From here, 3 cases were assigned to Node 14 and 55 were assigned to Node 15.

Node 14: 3 cases were assigned to Node 14 from Node 7. These cases were split on the NOX variable using a splitting value of 0.657. The average value of the response variable for these 3 cases is 21.8 From here, 2 records were assigned to Node 26 and 1 record was assigned to Node 27.

Node 26. 2 cases were assigned to Node 26 from Node 14. These cases were split on the NOX variable (again) using a splitting value of 0.6195. Both records assigned to this node have a tentative predicted value of 18.75.



Fully Grown Tree Rules (Using Training Data)

Node ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type	Column
1	N/A	2	3	LSTAT	9.725	304	202	21.969	Decision	
2	1	4	5	RM	7.0115	116	96	29.68	Decision	
3	1	6	7	CRIM	5.848	188	106	17.211	Decision	
4	2	8	9	DIS	1.4849	87	66	26.254	Decision	
5	2	10	11	RM	7.435	29	30	39.959	Decision	
6	3	12	13	DIS	1.9832	130	73	19.261	Decision	
7	3	14	15	LSTAT	13.33	58	33	12.616	Decision	
8	4	N/A	N/A	N/A	N/A	4	0	50	Terminal	
9	4	16	17	RM	6.5445	83	66	25.11	Decision	
10	5	18	19	INDUS	12.255	15	14	34.973	Decision	
11	5	20	21	PTRATIO	17.9	14	16	45.3	Decision	
12	6	22	23	LSTAT	22.805	16	16	14.537	Decision	
13	6	24	25	RM	6.9385	114	57	19.924	Decision	
14	7	26	27	NOX	0.657	3	2	21.8	Decision	
15	7	28	29	LSTAT	19.73	55	31	12.115	Decision	
16	9	30	31	TAX	208	52	35	22.913	Decision	
17	9	32	33	LSTAT	4.92	31	31	28.794	Decision	
18	10	34	35	B	380.22	14	13	33.9	Decision	
19	10	N/A	N/A	N/A	N/A	1	1	50	Terminal	
20	11	36	37	RM	7.798	11	14	47.045	Decision	
21	11	38	39	AGE	30.1	3	2	38.9	Decision	
22	12	40	41	AGE	97.25	9	11	15.833	Decision	
23	12	42	43	INDUS	24.815	7	5	12.871	Decision	
24	13	44	45	PTRATIO	20.95	111	57	19.638	Decision	
25	13	46	47	AGE	76.1	3	0	30.5	Decision	
26	14	48	49	NOX	0.6195	2	2	18.75	Decision	
27	14	N/A	N/A	N/A	N/A	1	0	27.9	Terminal	
28	15	50	51	AGE	96.65	23	13	13.957	Decision	
29	15	52	53	NOX	0.6695	32	18	10.791	Decision	
30	16	N/A	N/A	N/A	N/A	1	0	36.2	Terminal	
31	16	54	55	PTRATIO	20.6	51	35	22.653	Decision	
32	17	56	57	RM	6.791	11	11	32.764	Decision	
33	17	58	59	DIS	3.3019	20	20	26.61	Decision	
34	18	60	61	NOX	0.452	2	2	30.35	Decision	
35	18	62	63	LSTAT	5.44	12	11	34.492	Decision	
36	20	64	65	RM	7.6425	4	4	42.6	Decision	
37	20	66	67	RAD	6.5	7	10	49.586	Decision	
38	21	N/A	N/A	N/A	N/A	1	0	42.8	Terminal	

Now these rules are used to score on the validation partition.

Node 1: All 202 cases in the training partition were split on the LSTAT variable using a splitting value of 9.725. From here, 96 cases were assigned to Node 2 and 106 cases were assigned to Node 3.

Node 3: 106 cases were assigned to Node 3 from Node 1. These cases were split on the CRIM variable using a splitting value of 5.848. From here, 73 cases were assigned to Node 6 and 33 cases were assigned Node 7.

Node 7: 33 cases were assigned to Node 7 from Node 3. These cases were split on the LSTAT variable (again) using a splitting value of 13.33. From here, 2 cases were assigned to Node 14 and 31 were assigned to Node 15.

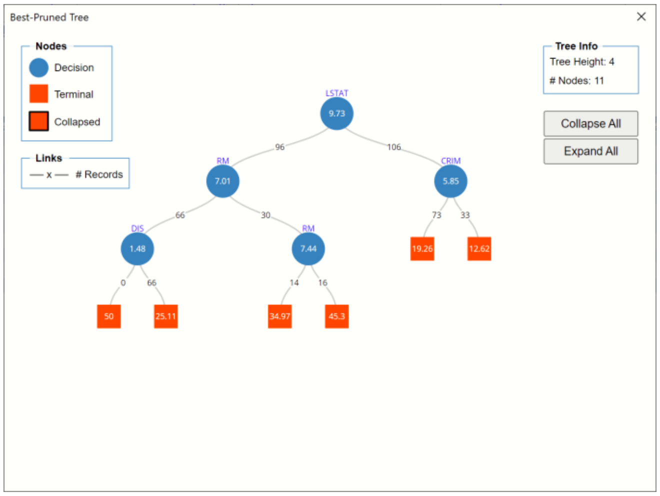
Node 14: 2 cases were assigned to Node 14 from Node 7. These cases were split on the NOX variable using a splitting value of 0.657. From here, 2 records were assigned to Node 26 and 1 record was assigned to Node 27.

Node 26. 2 cases were assigned to Node 26 from Node 14. These cases were split on the NOX variable (again) using a splitting value of 0.6195. Both cases were assigned to Node 48.

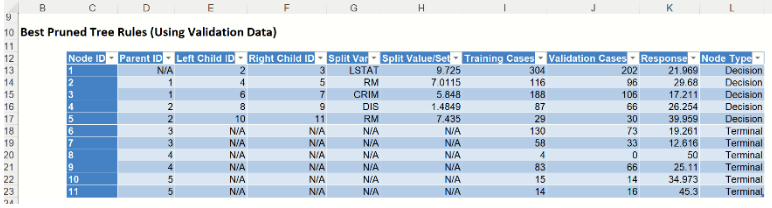
Node 48: This is a terminal node; no other splitting occurs.

RT_BestTree

To view the Best Pruned Tree, either click the Best Pruned Tree link in the Output Navigator or click the RT_BestTree worksheet tab. Recall that the Best Pruned Tree is the smallest tree that has an error within one standard error of the minimum error tree.



LSTAT (% Lower Status of the Population) is chosen as the first splitting variable; if this value is >= 9.73 (106 cases), then CRIM (crime rate per capita) is chosen for splitting; if CRIM is >= 5.85 (33 cases) then the predicted value equals 12.62. If CRIM is < 5.85 (73 cases) then the predicted value equals 19.26.



Best Pruned Tree Rules (Using Validation Data)

Node ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	LSTAT	9.725	304	202	21.969	Decision
2	1	4	5	RM	7.0115	116	96	29.68	Decision
3	1	6	7	CRIM	5.848	188	106	17.211	Decision
4	2	8	9	DIS	1.4849	87	66	26.254	Decision
5	2	10	11	RM	7.435	29	30	39.959	Decision
6	3	N/A	N/A	N/A	N/A	130	73	19.261	Terminal
7	3	N/A	N/A	N/A	N/A	58	33	12.616	Terminal
8	4	N/A	N/A	N/A	N/A	4	0	50	Terminal
9	4	N/A	N/A	N/A	N/A	83	66	25.11	Terminal
10	5	N/A	N/A	N/A	N/A	15	14	34.973	Terminal
11	5	N/A	N/A	N/A	N/A	14	16	45.3	Terminal

The path from above can be followed through the Best Pruned Tree Rules table.

Node 1: 202 cases from the validation partition are assigned to nodes 2 (96 cases) and 3 (106 cases) using the LSTAT variable with a split value of 9.725.

Node 3: 106 cases from the validation partition are assigned to nodes 6 (66 cases) and 7 (30 cases) using the RM variable with a split value of 7.011.

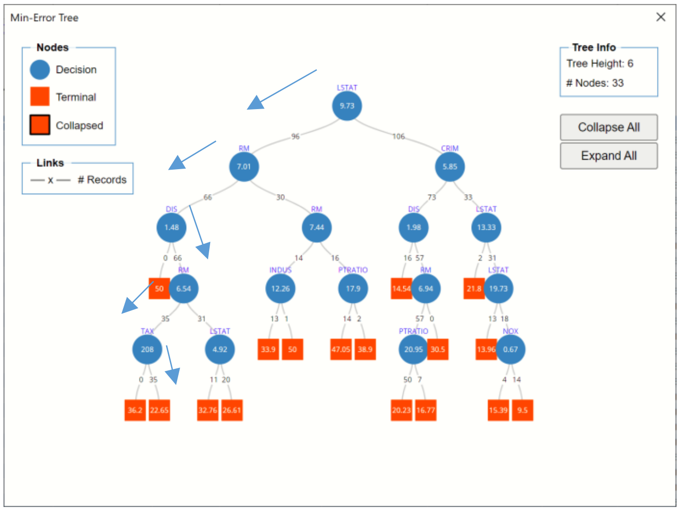
Node 6: 73 cases from the validation partition are assigned to this terminal node. The predicted response is equal to 19.261.

Node 7: 33 cases from the validation partition are assigned to this termional node. The predicted response is equal to 12.616.

RT_MinErrorTree

To view the Min-Error Tree, either click the **Min-Error Rules (Using Validation Data** link in the Output Navigator or click the RT_MinErrorTree worksheet tab. Recall that the Minimum Error tree is the tree with the minimum error on the validation dataset.

An example of a path down the tree is shown below.



LSTAT (% Lower Status of the Population) is chosen as the first splitting variable; if this value is < 9.73 (96 cases), then RM (# of Rooms) is chosen for splitting; if RM <7.01 (66 cases) then DIS (weighted distances to 5 employment centers) is chosen for splitting.   If DIS is >=1.48 (66 cases) then the RM variable is (again) chosen as the next divider. If RM <6.54 (35 cases), then TAX is chosen for splitting. All 35 cases assigned to this node have a TAX value >=208. The predicted value for all 35 cases is 22.65. No cases are assigned a predicted value of 36.2.

Click the **Min- Error Tree Rules** link to navigate to Tree rules for the Min-Error tree.   The path from above can be followed through the table.

Node 1: 202 cases from the validation partition are assigned to nodes 2 (96 cases) and 3 (106 cases) using the LSTAT variable with a split value of 9.725.

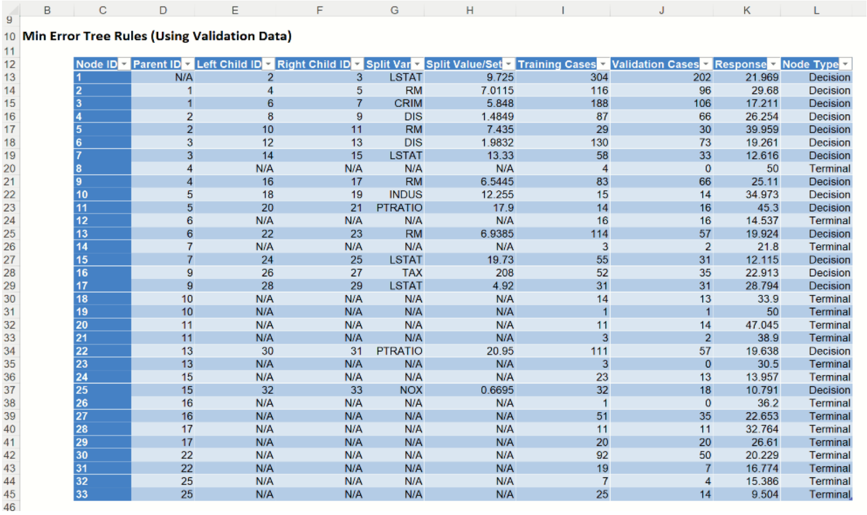
Node 2: 96 cases from the validation partition are assigned to nodes 4 (66 cases) and 5 (30 cases) using the RM variable with a split value of 7.011.

Node 4: 66 cases from the validation partition are assigned to nodes 8 (0 cases) and 9 (66 cases) using the DIS variable with a split value of 1.49.

Node 9: 66 cases from the validation partition are assigned to nodes 16 (35 cases) and 17 (31 cases) using the RM variable with a split value of 6.54.

Node 16: 35 cases from the validation partition are assigned to nodes 26 (0 cases) and 27 (35 cases) using the TAX variable with a split value of 208.

Node 27: 35 cases from the validation partition are assigned to this terminal node. All are assigned a predicted value of 22.653.



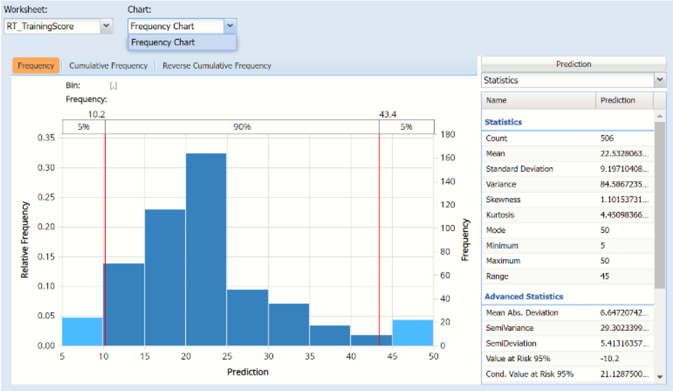
Min Error Tree Rules (Using Validation Data)

Node ID	Parent ID	Left Child ID	Right Child ID	Split Var	Split Value/Set	Training Cases	Validation Cases	Response	Node Type
1	N/A	2	3	LSTAT	9.725	304	202	21.969	Decision
2	1	4	5	RM	7.0115	116	96	29.68	Decision
3	1	6	7	CRIM	5.848	188	106	17.211	Decision
4	2	8	9	DIS	1.4849	87	66	26.254	Decision
5	2	10	11	RM	7.435	29	30	39.959	Decision
6	3	12	13	DIS	1.9832	130	73	19.261	Decision
7	3	14	15	LSTAT	13.33	58	33	12.616	Decision
8	4	N/A	N/A	N/A	N/A	4	0	50	Terminal
9	4	16	17	RM	6.5445	83	66	25.11	Decision
10	5	18	19	INDUS	12.255	15	14	34.973	Decision
11	5	20	21	PTRATIO	17.9	14	16	45.3	Decision
12	6	N/A	N/A	N/A	N/A	16	16	14.537	Terminal
13	6	22	23	RM	6.9385	114	57	19.924	Decision
14	7	N/A	N/A	N/A	N/A	3	2	21.8	Terminal
15	7	24	25	LSTAT	19.73	55	31	12.115	Decision
16	9	26	27	TAX	208	52	35	22.913	Decision
17	9	28	29	LSTAT	4.92	31	31	28.794	Decision
18	10	N/A	N/A	N/A	N/A	14	13	33.9	Terminal
19	10	N/A	N/A	N/A	N/A	1	1	50	Terminal
20	11	N/A	N/A	N/A	N/A	11	14	47.045	Terminal
21	11	N/A	N/A	N/A	N/A	3	2	38.9	Terminal
22	13	30	31	PTRATIO	20.95	111	57	19.638	Decision
23	13	N/A	N/A	N/A	N/A	3	0	30.5	Terminal
24	15	N/A	N/A	N/A	N/A	23	13	13.957	Terminal
25	15	32	33	NOX	0.6695	32	18	10.791	Decision
26	16	N/A	N/A	N/A	N/A	1	0	36.2	Terminal
27	16	N/A	N/A	N/A	N/A	51	35	22.653	Terminal
28	17	N/A	N/A	N/A	N/A	11	11	32.764	Terminal
29	17	N/A	N/A	N/A	N/A	20	20	26.61	Terminal
30	22	N/A	N/A	N/A	N/A	92	50	20.229	Terminal
31	22	N/A	N/A	N/A	N/A	19	7	16.774	Terminal
32	25	N/A	N/A	N/A	N/A	7	4	15.386	Terminal
33	25	N/A	N/A	N/A	N/A	25	14	9.504	Terminal

RT_TrainingScore

Click the *RT_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Prediction Summary and the Training: Prediction Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



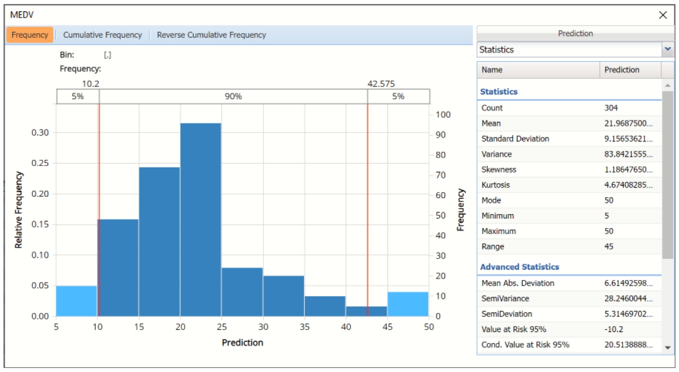
Worksheet: RT_TrainingScore

Chart: Frequency Chart

Name	Prediction
Statistics	
Count	506
Mean	22.5328063...
Standard Deviation	9.19710408...
Variance	84.5867235...
Skewness	1.10153731...
Kurtosis	4.45098366...
Mode	50
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.64720742...
SemiVariance	29.3023399...
SemiDeviation	5.41316357...
Value at Risk 95%	-10.2
Cond. Value at Risk 95%	21.1287500...

- **Frequency Charts:** The output variable frequency chart opens automatically once the *CT_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *CT_TrainingScore* tab. To move this chart to another location on the screen, grab the title bar of the dialog and then drag.

Frequency Chart for Prediction data



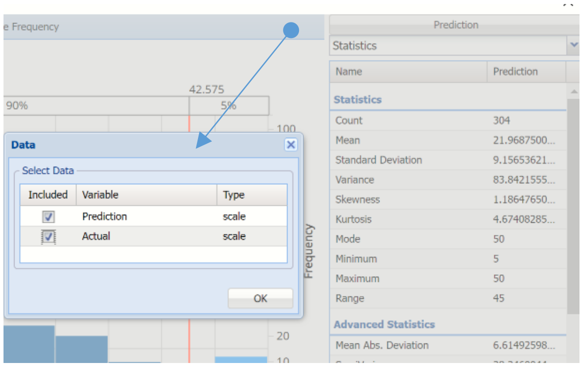
Prediction

Statistics

Name	Prediction
Statistics	
Count	304
Mean	21.9687500...
Standard Deviation	9.15653621...
Variance	83.8421555...
Skewness	1.18647650...
Kurtosis	4.67408285...
Mode	50
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.61492598...
SemiVariance	28.2460044...
SemiDeviation	5.31469702...
Value at Risk 95%	-10.2
Cond. Value at Risk 95%	20.5138888...

To add the actual data, click Prediction, then select both Prediction and Actual.

Click to add Actual data



Data

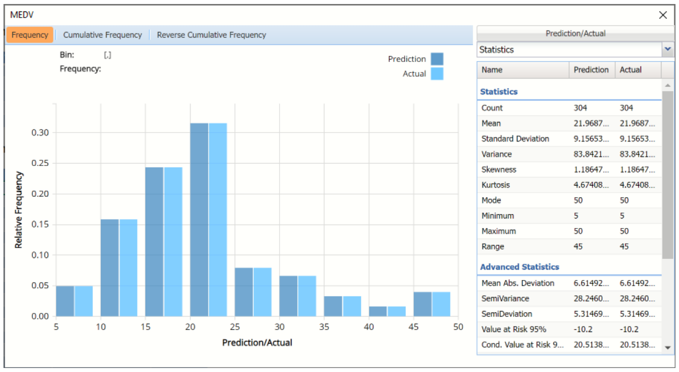
Select Data

Included	Variable	Type
☑	Prediction	scale
☑	Actual	scale

OK

Name	Prediction
Statistics	
Count	304
Mean	21.9687500...
Standard Deviation	9.15653621...
Variance	83.8421555...
Skewness	1.18647650...
Kurtosis	4.67408285...
Mode	50
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.61492598...

Prediction and Actual Frequency Chart



Prediction/Actual

Statistics

Name	Prediction	Actual
Statistics		
Count	304	304
Mean	21.9687...	21.9687...
Standard Deviation	9.15653...	9.15653...
Variance	83.8421...	83.8421...
Skewness	1.18647...	1.18647...
Kurtosis	4.67408...	4.67408...
Mode	50	50
Minimum	5	5
Maximum	50	50
Range	45	45
Advanced Statistics		
Mean Abs. Deviation	6.61492...	6.61492...
SemiVariance	28.2460...	28.2460...
SemiDeviation	5.31469...	5.31469...
Value at Risk 95%	-10.2	-10.2
Cond. Value at Risk 9...	20.5138...	20.5138...

Notice in the screenshot below that both the Original and Synthetic data appear in the chart together, and statistics for both data appear on the right. As you can see from this chart, the fitted regression model perfectly predicted the values for the output variable, MEDV, in the training partition.

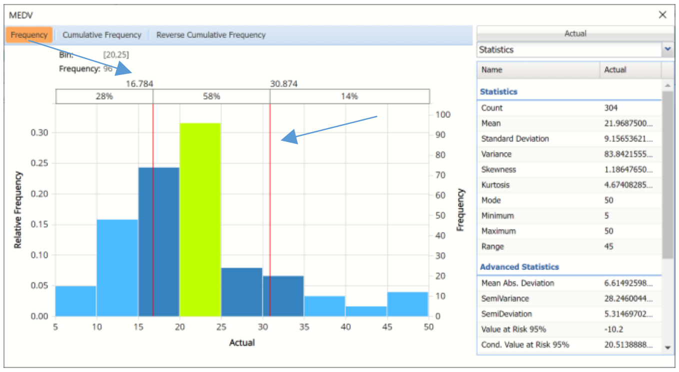
To remove either the Prediction or Actual data from the chart, click Prediction/Actual in the top right and then uncheck the data type to be removed.

This chart behaves the same as the interactive chart in the Analyze Data feature found on the Explore menu (described in the Analyze Data chapter that appears earlier in this guide).

- Use the mouse to hover over any of the bars in the graph to populate the Bin and Frequency headings at the top of the chart.

- When displaying either Prediction or Actual data (not both), red vertical lines will appear at the 5% and 95% percentile values in all three charts (Frequency, Cumulative Frequency and Reverse Cumulative Frequency) effectively displaying the 90th confidence interval. The middle percentage is the percentage of all the variable values that lie within the 'included' area, i.e. the darker shaded area. The two percentages on each end are the percentage of all variable values that lie outside of the 'included' area or the "tails". i.e. the lighter shaded area. Percentile values can be altered by moving either red vertical line to the left or right.

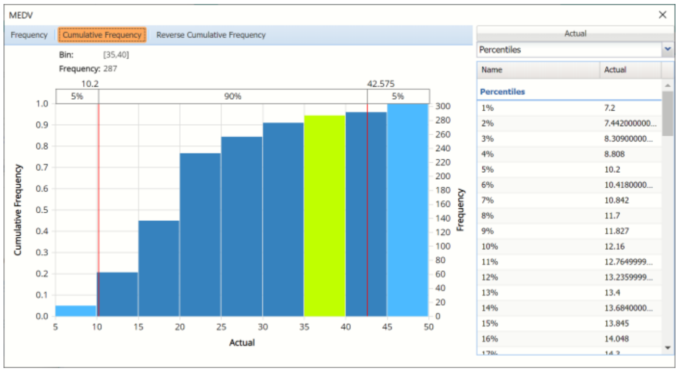
  Frequency Chart, confidence intervals modified



Name	Actual
Statistics	
Count	304
Mean	21.9687500...
Standard Deviation	9.15653621...
Variance	83.8421555...
Skewness	1.18647650...
Kurtosis	4.67408285...
Mode	50
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.61492598...
SemiVariance	28.2460044...
SemiDeviation	5.31469702...
Value at Risk 95%	-10.2
Cond. Value at Risk 95%	20.5138888...

- Click Cumulative Frequency and Reverse Cumulative Frequency tabs to see the Cumulative Frequency and Reverse Cumulative Frequency charts, respectively.

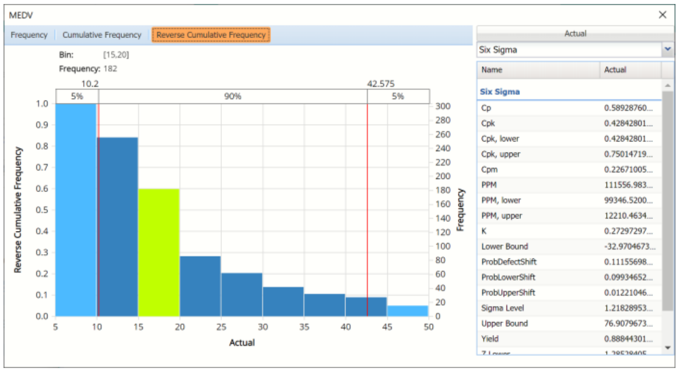
  Cumulative Frequency chart with Percentiles displayed



Name	Actual
Percentiles	
1%	7.2
2%	7.442000000...
3%	8.30900000...
4%	8.808
5%	10.2
6%	10.4180000...
7%	10.842
8%	11.7
9%	11.827
10%	12.16
11%	12.7649999...
12%	13.2359999...
13%	13.4
14%	13.6840000...
15%	13.845
16%	14.048
17%	14.2

- Click the down arrow next to Statistics to view Percentiles for each type of data along with Six Sigma indices.

Reverse Cumulative Frequency chart and Six Sigma indices displayed.



Bin: [15,20]

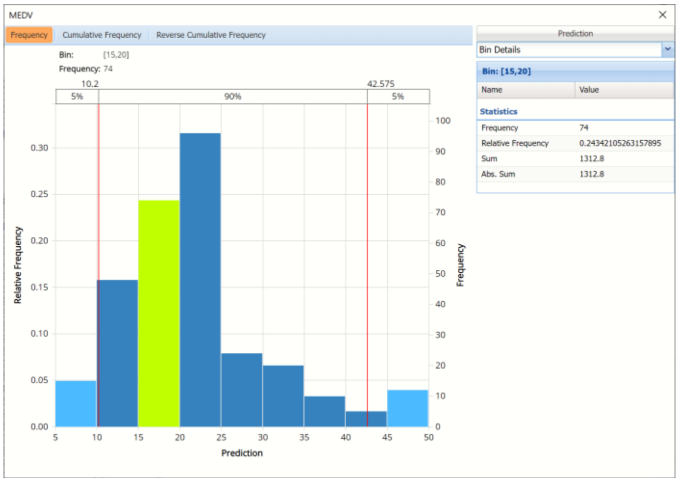
Frequency: 182

Actual

Six Sigma

Name	Actual
Six Sigma	
Cp	0.58928760...
Cpk	0.42842801...
Cpk, lower	0.42842801...
Cpk, upper	0.75014719...
Cpm	0.22671005...
PPM	111556.983...
PPM, lower	99346.5200...
PPM, upper	12210.4634...
K	0.27297297...
Lower Bound	-32.9704673...
ProbDefectShift	0.11155698...
ProbLowerShift	0.09934652...
ProbUpperShift	0.01221046...
Sigma Level	1.21828953...
Upper Bound	76.9079673...
Yield	0.88844301...
Z Lower	1.29529405...

- Click the down arrow next to Statistics to view Bin Details to find information pertaining to each bin in the chart.



Bin: [15,20]

Frequency: 74

Prediction

Bin Details

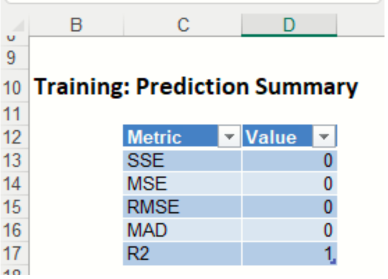
Bin: [15,20]

Name	Value
Statistics	
Frequency	74
Relative Frequency	0.24342105263157895
Sum	1312.8
Abs. Sum	1312.8

- Use the Chart Options view to manually select the number of bins to use in the chart, as well as to set personalization options.

- **Training: Prediction Summary:** The Prediction Summary tables contain summary information for the training partition. These reports contain the total sum of squared errors, the mean squared error, the root mean square error (RMS error, or the square root of the mean squared error), and also the average error (which is much smaller, since errors fall roughly into negative and positive errors and tend to cancel each other out unless squared first.). Small error values in both datasets suggest that the Single Tree method has created a very accurate predictor. However, in general, these errors are not great measures. RROC curves (discussed below) are much more sophisticated and provide more precise information about the accuracy of the predictor.

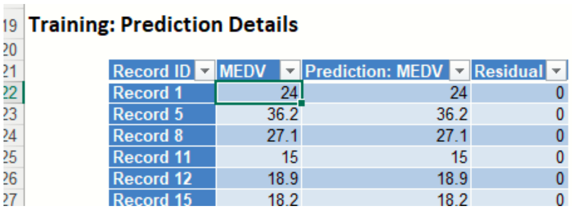
  In this example, we see that the fitted model perfectly predicted the value for the output variable in all training partition records.



Training: Prediction Summary

Metric	Value
SSE	0
MSE	0
RMSE	0
MAD	0
R2	1

- **Training: Prediction Details:** The Prediction Details table displays the predicted value for each record along with the actual value and the residuals for each record.



Training: Prediction Details

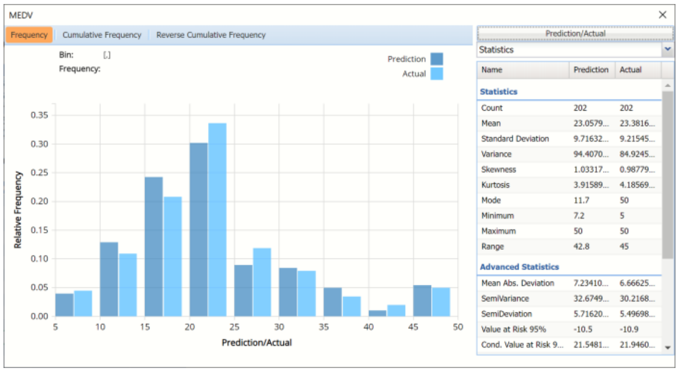
Record ID	MEDV	Prediction: MEDV	Residual
Record 1	24	24	0
Record 5	36.2	36.2	0
Record 8	27.1	27.1	0
Record 11	15	15	0
Record 12	18.9	18.9	0
Record 15	18.2	18.2	0

RT_ValidationScore

Another key interest in a data-mining context will be the predicted and actual values for the MEDV variable along with the residual (difference) for each predicted value in the *Validation* partition.

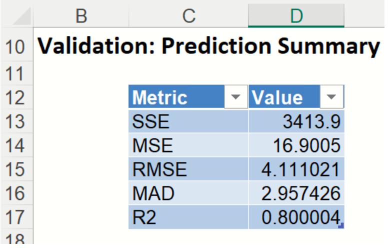
RT_ValidationScore displays the newly added Output Variable frequency chart, the Training: Prediction Summary and the Training: Prediction Details report. All calculations, charts and predictions on the RT_ValidationScore output sheet apply to the Validation partition.

- **Frequency Charts:** The output variable frequency chart for the validation partition opens automatically once the *RT_ValidationScore* worksheet is selected. This chart displays a detailed, interactive frequency chart for the Actual variable data and the Predicted data, for the validation partition. For more information on this chart, see the RT_TrainingLiftChart explanation above.



Name	Prediction	Actual
Statistics		
Count	202	202
Mean	23.0579...	23.3816...
Standard Deviation	9.71632...	9.21545...
Variance	94.4070...	84.9245...
Skewness	1.03317...	0.98779...
Kurtosis	3.91589...	4.18569...
Mode	11.7	50
Minimum	7.2	5
Maximum	50	50
Range	42.8	45
Advanced Statistics		
Mean Abs. Deviation	7.23410...	6.66625...
SemiVariance	32.6749...	30.2168...
SemiDeviation	5.71620...	5.49698...
Value at Risk 95%	-10.5	-10.9
Cond. Value at Risk 9...	21.5481...	21.9460...

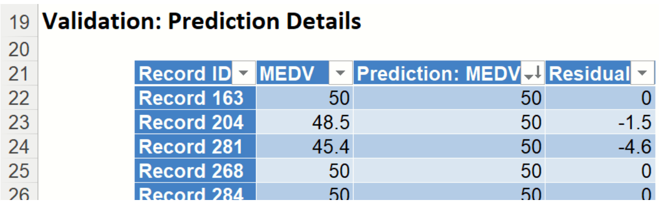
- **Prediction Summary:** In the Prediction Summary report, Analytic Solver Data Science displays the total sum of squared errors summaries for the Validation partition.



Validation: Prediction Summary

Metric	Value
SSE	3413.9
MSE	16.9005
RMSE	4.111021
MAD	2.957426
R2	0.800004

- **Prediction Details:** Scroll down to the Training: Prediction Details report to find the Prediction value for the MEDV variable for each record, as well as the Residual value.



Validation: Prediction Details

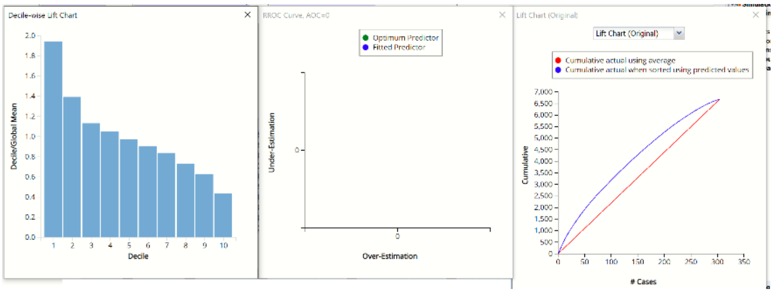
Record ID	MEDV	Prediction: MEDV	Residual
Record 163	50	50	0
Record 204	48.5	50	-1.5
Record 281	45.4	50	-4.6
Record 268	50	50	0
Record 284	50	50	0

RT_TrainingLiftchart and RT_ValidationLiftChart

Click the RT_TrainingLiftChart and RT_ValidationLiftChart tabs top display the lift charts and regression ROC curves. These are visual aids for measuring model performance. Lift Charts consist of a lift curve and a baseline. The greater the area between the lift curve and the baseline, the better the model. RROC (regression receiver operating characteristic) curves plot the performance of regressors by graphing over-estimations (or predicted values that are too high) versus underestimations (or predicted values that are too low.) The closer the curve is to the top left corner of the graph (in other words, the smaller the area above the curve), the better the performance of the model.

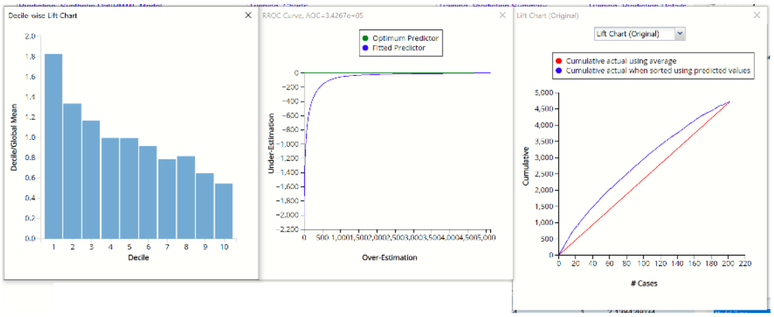
Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select RT_TrainingLiftChart or RT_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, RROC Curve and Lift Chart for Training Partition



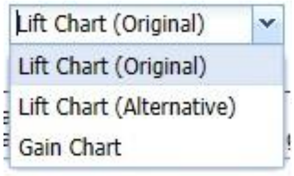
After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted in descending order using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the number of cases versus the cumulated value. The baseline (red line connecting the origin to the end point of the blue line) is drawn as the number of cases versus the average of actual output variable values multiplied by the number of cases. The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. This bars in this chart indicate the factor by which the CT model outperforms a random assignment, one decile at a time. Refer to the validation graph below. In the first decile, taking the most expensive predicted housing prices in the dataset, the predictive performance of the model is almost 2 times better as simply assigning a random predicted value.

Decile-wise Lift Chart, RROC Curve and Lift Chart for Validation Partition



In an Regression ROC curve, we can compare the performance of a regressor with that of a random guess (red line) for which under estimations are equal to over-estimations shifted to the minimum under estimate. Anything to the left of this line signifies a better prediction and anything to the right signifies a worse prediction. The best possible prediction performance would be denoted by a point at the top left of the graph at the intersection of the x and y axis. This point is sometimes referred to as the “perfect prediction”. Area Over the Curve (AOC) is the space in the graph that appears above the ROC curve and is calculated using the formula: sigma^2 * n^2/2 where n is the number of records The smaller the AOC, the better the performance of the model.

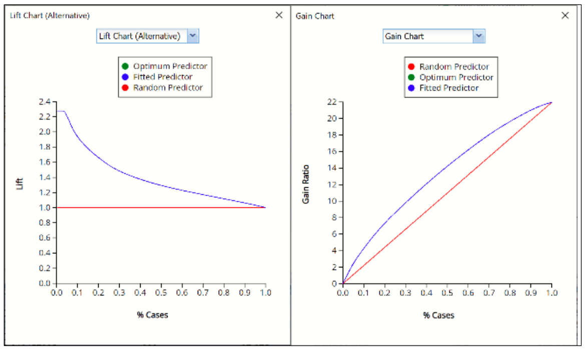
In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.



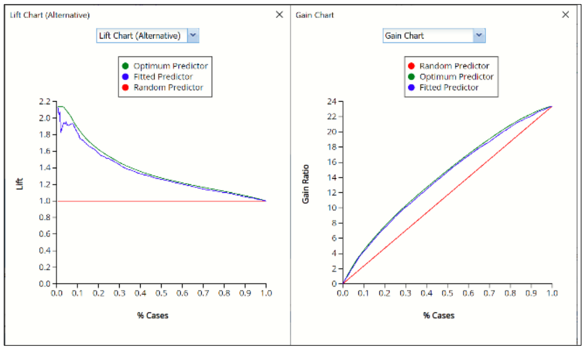
Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Classifier curve, a Fitted Classifier curve, and a Random Classifier curve. The Optimum Classifier curve plots a hypothetical model that would provide perfect classification for our data. The Fitted Classifier curve plots the fitted model and the Random Classifier curve plots the results from using no model or by using a random guess (i.e. for x% of selected observations, x% of the total number of positive observations are expected to be correctly classified).

The Alternative Lift Chart plots Lift against % Cases.

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition



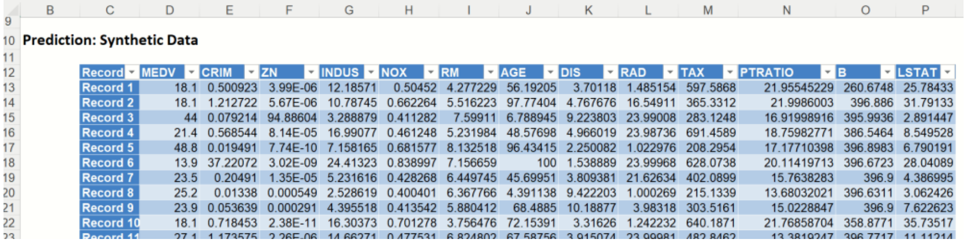
Click the down arrow and select Gain Chart from the menu. In this chart, the Gain Ratio is plotted against the % Cases.

RT_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, RT_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Regression Tree dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predicted values for the training data (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



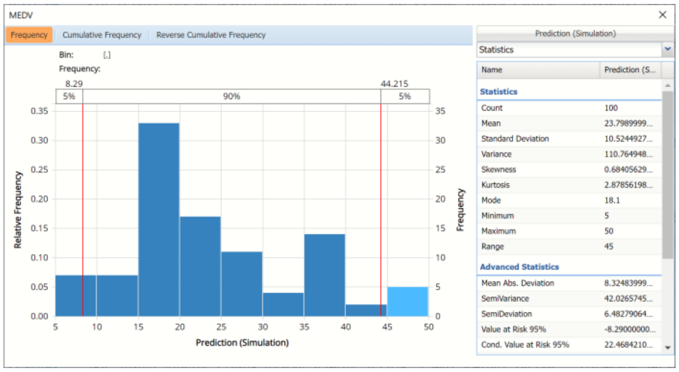
Prediction: Synthetic Data

Record	MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	18.1	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.70118	1.485154	597.5868	21.95545229	260.6748	25.78433
Record 2	18.1	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767676	16.54911	365.3312	21.9986003	396.886	31.79133
Record 3	44	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223803	23.99008	283.1248	16.91998916	395.9936	2.891447
Record 4	21.4	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966019	23.98736	691.4589	18.75982771	386.5464	8.549528
Record 5	48.8	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082	1.022976	208.2954	17.17710398	396.8983	6.790191
Record 6	13.9	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538889	23.99968	628.0738	20.11419713	396.6723	28.04089
Record 7	23.5	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809381	21.62634	402.0899	15.7638283	396.9	4.386995
Record 8	25.2	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422203	1.000269	215.1339	13.68032021	396.6311	3.062426
Record 9	23.9	0.053639	0.000291	4.395518	0.413542	5.880412	68.4885	10.18877	3.98318	303.5161	15.0228847	396.9	7.622623
Record 10	18.1	0.718453	2.38E-11	16.30373	0.701278	3.756476	72.15391	3.31626	1.242232	640.1871	21.76858704	358.8771	35.73517
Record 11	27.1	1.173575	2.26E-06	14.66271	0.477531	6.824802	67.58756	3.915074	23.99981	482.8462	13.3819247	396.7717	11.11214

The data contained in the Synthetic Data report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training data, the synthetic data and the expression, if it exists. (Recall that no expression was entered in this example.)

Frequency Chart for Prediction (Simulation) data



Prediction (Simulation)

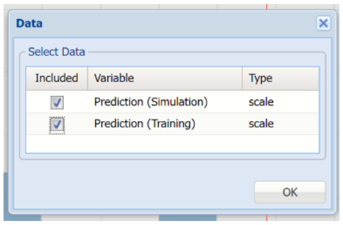
Statistics

Name	Prediction (S...
Statistics	
Count	100
Mean	23.7989999...
Standard Deviation	10.5244927...
Variance	110.764948...
Skewness	0.68405629...
Kurtosis	2.87856198...
Mode	18.1
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	8.32483999...
SemiVariance	42.0265745...
SemiDeviation	6.48279064...
Value at Risk 95%	-8.29000000...
Cond. Value at Risk 95%	22.4684210...

Click *Prediction (Simulation)* to add the training data to the chart.

Click *Prediction(Simulation)* and *Prediction (Training)* to change the Data view.

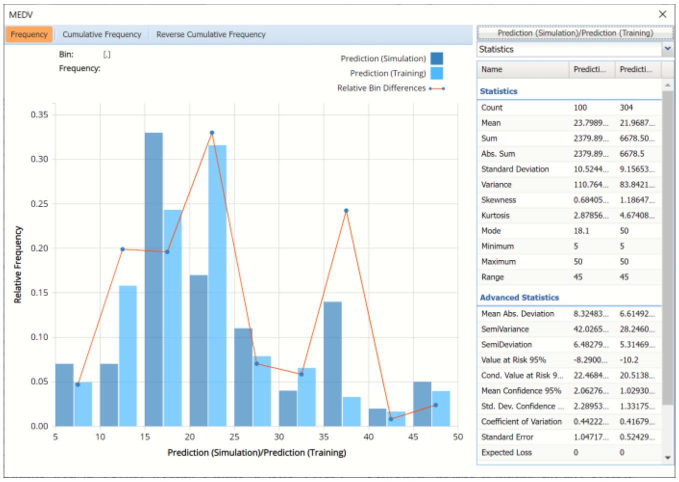
Data Dialog



Included	Variable	Type
☑	Prediction (Simulation)	scale
☑	Prediction (Training)	scale

In the chart below, the dark blue bars display the frequencies for the synthetic data and the light blue bars display the frequencies for the predicted values in the Training partition.

Prediction (Simulation) and Prediction (Training) Frequency chart for MEDV variable



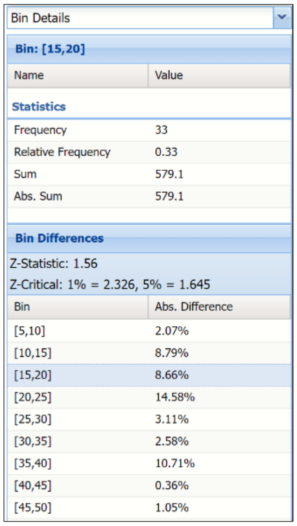
Prediction (Simulation)/Prediction (Training)

Statistics

Name	Predicti...	Predicti...
Statistics		
Count	100	304
Mean	23.7989...	21.9687...
Sum	2379.89...	6678.50...
Abs. Sum	2379.89...	6678.5
Standard Deviation	10.5244...	9.15653...
Variance	110.764...	83.8421...
Skewness	0.68405...	1.18647...
Kurtosis	2.87856...	4.67408...
Mode	18.1	50
Minimum	5	5
Maximum	50	50
Range	45	45
Advanced Statistics		
Mean Abs. Deviation	8.32483...	6.61492...
SemiVariance	42.0265...	28.2460...
SemiDeviation	6.48279...	5.31469...
Value at Risk 95%	-8.2900...	-10.2
Cond. Value at Risk 9...	22.4684...	20.5138...
Mean Confidence 95%	2.06276...	1.02930...
Std. Dev. Confidence ...	2.28953...	1.33175...
Coefficient of Variation	0.44222...	0.41679...
Standard Error	1.04717...	0.52429...
Expected Loss	0	0

The Relative Bin Differences curve charts the absolute differences between the data in each bin. Click the down arrow next to Statistics to view the Bin Details pane to display the calculations.

Bin Details pane



Bin Details

Bin: [15,20]

Name	Value
Statistics	
Frequency	33
Relative Frequency	0.33
Sum	579.1
Abs. Sum	579.1

Bin Differences

Z-Statistic: 1.56

Z-Critical: 1% = 2.326, 5% = 1.645

Bin	Abs. Difference
[5,10]	2.07%
[10,15]	8.79%
[15,20]	8.66%
[20,25]	14.58%
[25,30]	3.11%
[30,35]	2.58%
[35,40]	10.71%
[40,45]	0.36%
[45,50]	1.05%

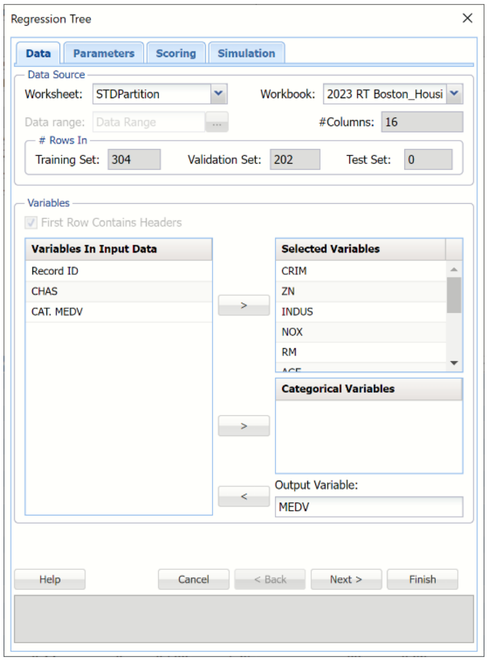
Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

For information on *RT_Stored*, please see the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide.

Regression Tree Options

The options below appear on one of the three Regression Tree dialogs.

Regression Tree Dialog, Data tab



Regression Tree

Data Source
Worksheet: STDPartition
Workbook: 2023 RT Boston_Housi
Data range: Data Range
#Columns: 16

Rows In
Training Set: 304
Validation Set: 202
Test Set: 0

Variables
☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM

Categorical Variables

Output Variable: MEDV

Regression Tree Dialog, Data tab

Variables in input data

All variables in the dataset are listed here.

Selected variables

Variables listed here will be utilized in the Analytic Solver Data Science output.

Output Variable

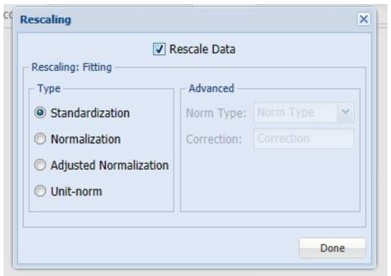
Select the variable whose outcome is to be predicted here.

Regression Tree Dialog, Parameters tab

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the regression method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Rescale Data



Rescaling

☑ Rescale Data

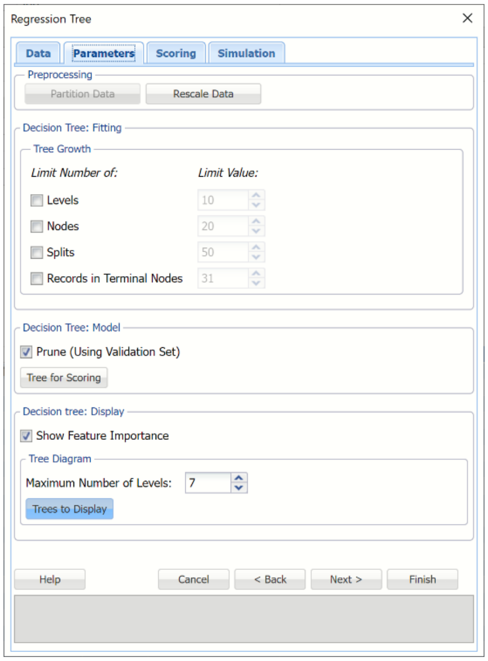
Rescaling: Fitting

Type
● Standardization
○ Normalization
○ Adjusted Normalization
○ Unit-norm

Advanced
Norm Type: Norm Type
Correction: Correction

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Regression Tree Dialog, Parameters tab



Limit Number of: | Limit Value:
□ Levels 10
□ Nodes 20
□ Splits 50
□ Records in Terminal Nodes 31

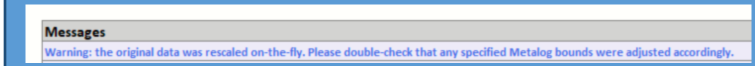
☑ Prune (Using Validation Set)

☑ Show Feature Importance

Maximum Number of Levels: 7

Notes on Rescaling and the Simulation functionality

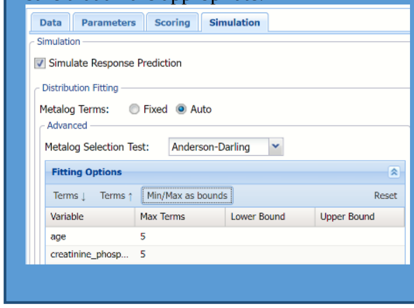
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then "Min/Max as bounds" on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the RT_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the "Min/Max as bounds" button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the "Min/Max as bounds" feature is available for the user's convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Tree Growth

n the *Tree Growth* section, select Levels, Nodes, Splits, and Records in Terminal Nodes. Values entered for these options limit tree growth, i.e. if 10 is entered for Levels, the tree will be limited to 10 levels.

Prune (Using Validation Set)

If a validation partition exists, this option is enabled. When this option is selected, Analytic Solver Data Science will prune the tree. Pruning the tree using the validation set reduces the error from over-fitting the tree to the training data.

Show Feature Importance

Select *Feature Importance* to include the *Features Importance* table in the output. This table displays the variables that are included in the model along with their Importance value.

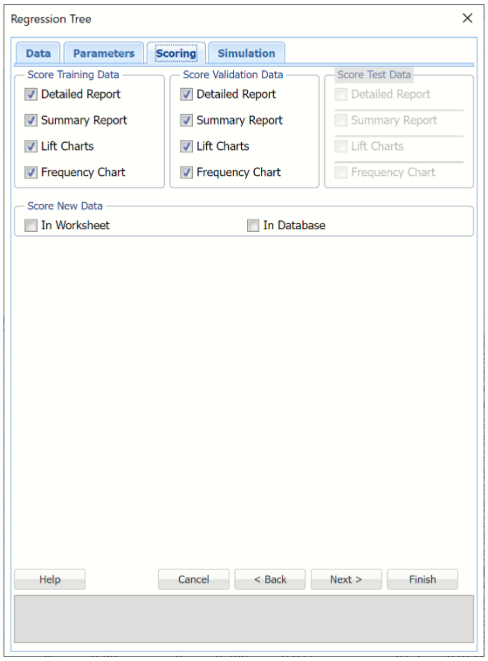
Maximum Number of Levels

This option specifies the maximum number of levels in the tree to be displayed in the output.   Select *Trees to Display* to select the types of trees to display: Fully Grown, Best Pruned, Minimum Error or User Specified.

- Select *Fully Grown* to "grow" a complete tree using the training data.
- Select *Best Pruned* to create a tree with the fewest number of nodes, subject to the constraint that the error be kept below a specified level (minimum error rate plus the standard error of that error rate).
- Select *Minimum error* to produce a tree that yields the minimum classification error rate when tested on the validation data.
- To create a tree with a specified number of decision nodes select *User Specified* and enter the desired number of nodes.

Regression Tree Dialog,Scoring tab

Regression Tree Dialog, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

Score Training Data

Select these options to show an assessment of the performance of the Regression Tree algorithm in predicting the value of the output variable in the training partition.

When Frequency Chart is selected, a frequency chart will be displayed when the RT_TrainingScore worksheet is selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Regression Tree algorithm in predicting the value of the output variable in the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the RT_ValidationScore worksheet is selected.

Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Regression Tree algorithm in predicting the value of the output variable in the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the RT_TestScore worksheet is selected.

Score New Data

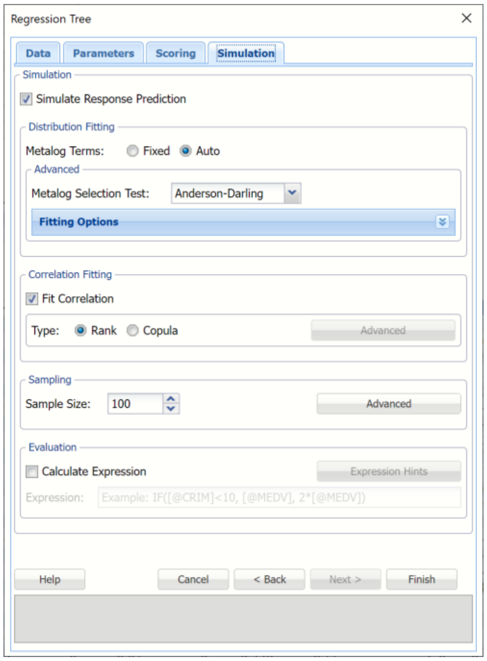
See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Regression Tree Dialog,Simulation tab

Simulation Tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, RT_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RT_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

□ Calculate Expression

Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

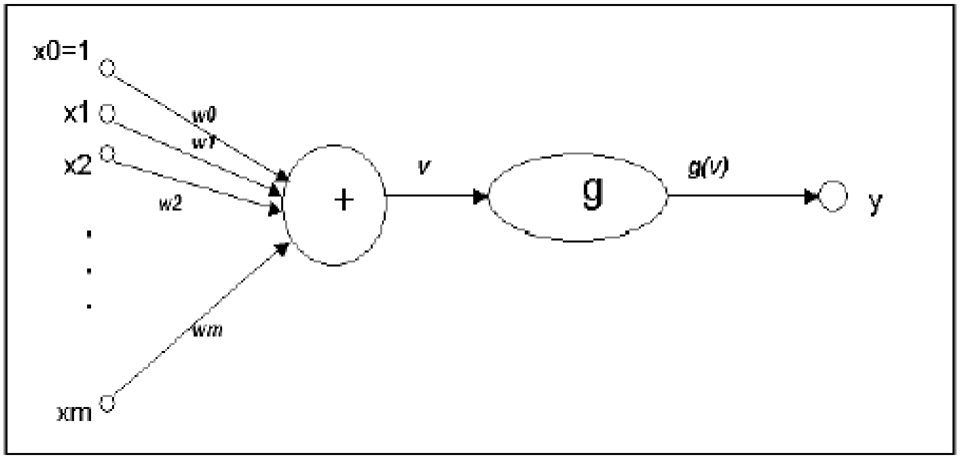
Neural Network Regression Method

Introduction

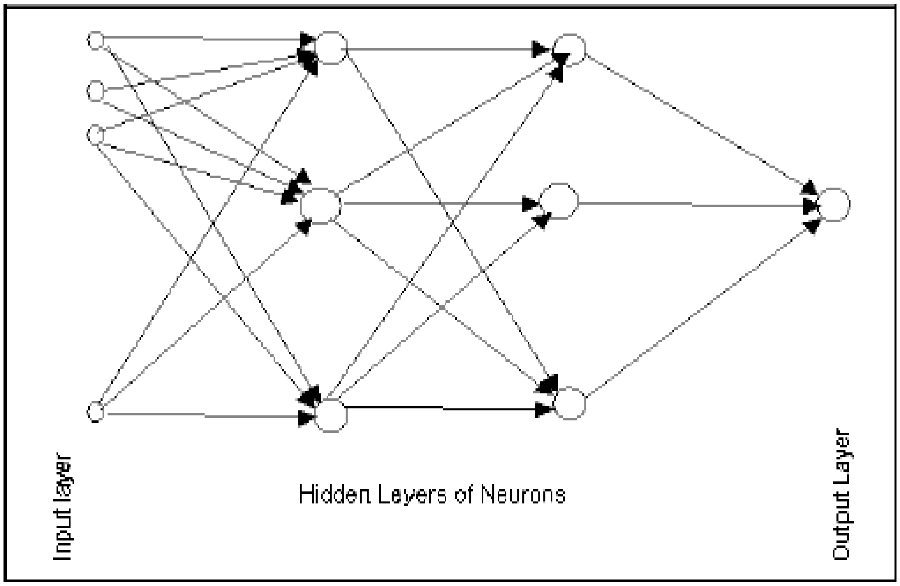
Artificial neural networks are relatively crude electronic networks of "neurons" based on the neural structure of the brain. They process records one at a time, and "learn" by comparing their prediction of the record (which, at the outset, is largely arbitrary) with the known actual record. The errors from the initial prediction of the first record is fed back to the network and used to modify the network's algorithm for the second iteration.   These steps are repeated multiple times.

Roughly speaking, a neuron in an artificial neural network is

1. A set of input values (x_i) with associated weights (w_i)
2. An input function (g) that sums the weights and maps the results to an output function(y).



Neurons are organized into layers: input, hidden and output. The input layer is composed not of full neurons, but simply of the values in a record that are inputs to the next layer of neurons. The next layer is the hidden layer of which there could be several. The final layer is the output layer, where there is one node for each class. A single sweep forward through the network results in the assignment of a value to each output node. The record is assigned to the class node with the highest value.



Training an Artificial Neural Network

In the training phase, the correct class for each record is known (this is termed supervised training), and the output nodes can therefore be assigned "correct" values -- "1" for the node corresponding to the correct class, and "0" for the others. (In practice, better results have been found using values of “0.9” and “0.1”, respectively.) As a result, it is possible to compare the network's calculated values for the output nodes to these "correct" values, and calculate an error term for each node. These error terms are then used to adjust the weights in the hidden layers so that, hopefully, the next time around the output values will be closer to the "correct" values.

The Iterative Learning Process

A key feature of neural networks is an iterative learning process in which records (rows) are presented to the network one at a time, and the weights associated with the input values are adjusted each time. After all cases are presented, the process often starts over again. During this learning phase, the network “trains” by adjusting the weights to predict the correct class label of input samples. Advantages of neural networks include their high tolerance to noisy data, as well as their ability to classify patterns on which they have not been trained. The most popular neural network algorithm is the back-propagation algorithm proposed in the 1980's.

Once a network has been structured for a particular application, that network is ready to be trained. To start this process, the initial weights (described in the next section) are chosen randomly. Then the training, or learning, begins.

The network processes the records in the training data one at a time, using the weights and functions in the hidden layers, then compares the resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights for the next record. This process occurs over and over as the weights are continually tweaked. During the training of a network the same set of data is processed many times as the connection weights are continually refined.

Note that some networks never learn. This could be because the input data do not contain the specific information from which the desired output is derived. Networks also will not converge if there is not enough data to enable complete learning. Ideally, there should be enough data available to create a validation set.

Feedforward, Back-Propagation

The feedforward, back-propagation architecture was developed in the early 1970's by several independent sources (Werbor; Parker; Rumelhart, Hinton and Williams). This independent co-development was the result of a proliferation of articles and talks at various conferences which stimulated the entire industry. Currently, this synergistically developed back-propagation architecture is the most popular, effective, and easy-to-learn model for complex, multi-layered networks. Its greatest strength is in non-linear solutions to ill-defined problems. The typical back-propagation network has an input layer, an output layer, and at least one hidden layer. Theoretically, there is no limit on the number of hidden layers but typically there are just one or two. Some studies have shown that the total number of layers needed to solve problems of any complexity is 5 (one input layer, three hidden layers and an output layer). Each layer is fully connected to the succeeding layer.

As noted above, the training process normally uses some variant of the Delta Rule, which starts with the calculated difference between the actual outputs and the desired outputs. Using this error, connection weights are increased in proportion to the error times, which are a scaling factor for global accuracy. This means that the inputs, the output, and the desired output all must be present at the same processing element. The most complex part of this algorithm is determining which input contributed the most to an incorrect output and how to modify the input to correct the error. (An inactive node would not contribute to the error and would have no need to change its weights.) To solve this problem, training inputs are applied to the input layer of the network, and desired outputs are compared at the output layer. During the learning process, a forward sweep is made through the network, and the output of each element is computed layer by layer. The difference between the output of the final layer and the desired output is back-propagated to the previous layer(s), usually modified by the derivative of the transfer function. The connection weights are normally adjusted using the Delta Rule. This process proceeds for the previous layer(s) until the input layer is reached.

Structuring the Network

The number of layers and the number of processing elements per layer are important decisions. These parameters, to a feedforward, back-propagation topology, are also the most ethereal - they are the "art" of the network designer. There is no quantifiable, best answer to the layout of the network for any particular application. There are only general rules picked up over time and followed by most researchers and engineers applying this architecture to their problems.

Rule One: As the complexity in the relationship between the input data and the desired output increases, the number of the processing elements in the hidden layer should also increase.

Rule Two: If the process being modeled is separable into multiple stages, then additional hidden layer(s) may be required. If the process is not separable into stages, then additional layers may simply enable memorization of the training set, and not a true general solution.

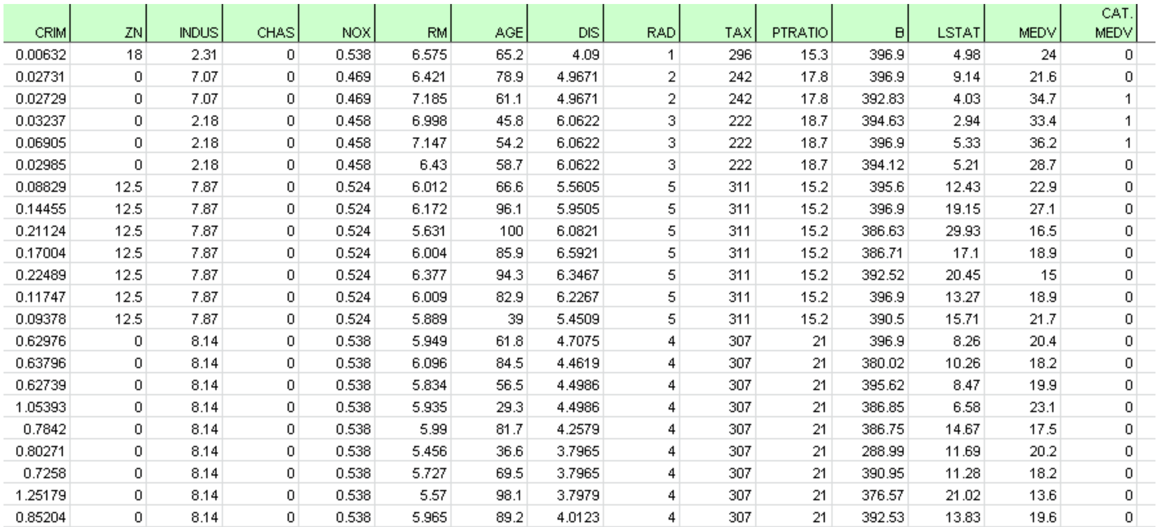
Rule Three: The amount of training data available sets an upper bound for the number of processing elements in the hidden layer(s). To calculate this upper bound, use the number of cases in the training data set and divide that number by the sum of the number of nodes in the input and output layers in the network. Then divide that result again by a scaling factor between five and ten. Larger scaling factors are used for relatively less noisy data. If too many artificial neurons are used the training set will be memorized, not generalized, and the network will be useless on new data sets.

Automated Neural Network Regression Method Example

This example focuses on creating a Neural Network using the Automatic Architecture. See the Ensemble Methods chapter that appears later on in this guide to see an example on creating a Neural Network using the boosting and bagging ensemble methods. See the section below for an example of how to create a single neural network.

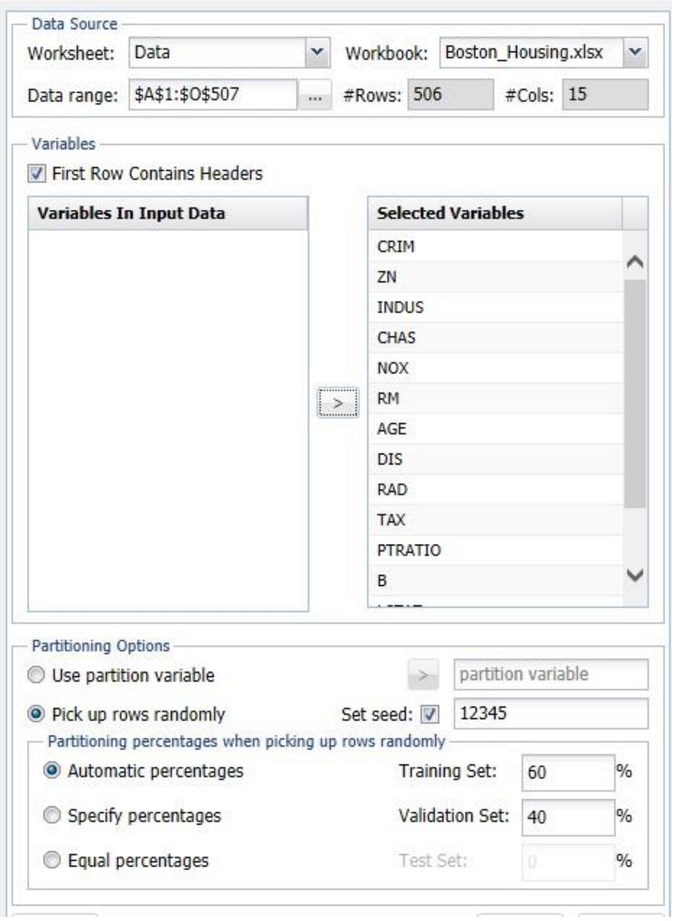
We will use the *Boston_Housing.xlsx* example dataset. This dataset contains 14 variables, the description of each is given in the Description worksheet. The dependent variable MEDV is the median value of a dwelling. This objective of this example is to predict the value of this variable.

A portion of the dataset is shown below. The last variable, CAT.MEDV, is a discrete classification of the MEDV variable and will not be used in this example.



CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT. MEDV
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24	0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	0
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	1
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	1
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	1
0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7	0
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	0
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	0
0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	0
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	0
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	0
0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	0
0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	0
0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	0
0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	0
0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	0
1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	0
0.7842	0	8.14	0	0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5	0
0.80271	0	8.14	0	0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2	0
0.7258	0	8.14	0	0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2	0
1.25179	0	8.14	0	0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6	0
0.85204	0	8.14	0	0.538	5.965	89.2	4.0123	4	307	21	392.53	13.83	19.6	0

First, we partition the data into training and validation sets using the Standard Data Partition defaults with percentages of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.



Data Source

Worksheet: Data

Workbook: Boston_Housing.xlsx

Data range: $A$1:$O$507

#Rows: 506

#Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables

CRIM
ZN
INDUS
CHAS
NOX
RM
AGE
DIS
RAD
TAX
PTRATIO
B

Partitioning Options

○ Use partition variable — partition variable

● Pick up rows randomly — Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

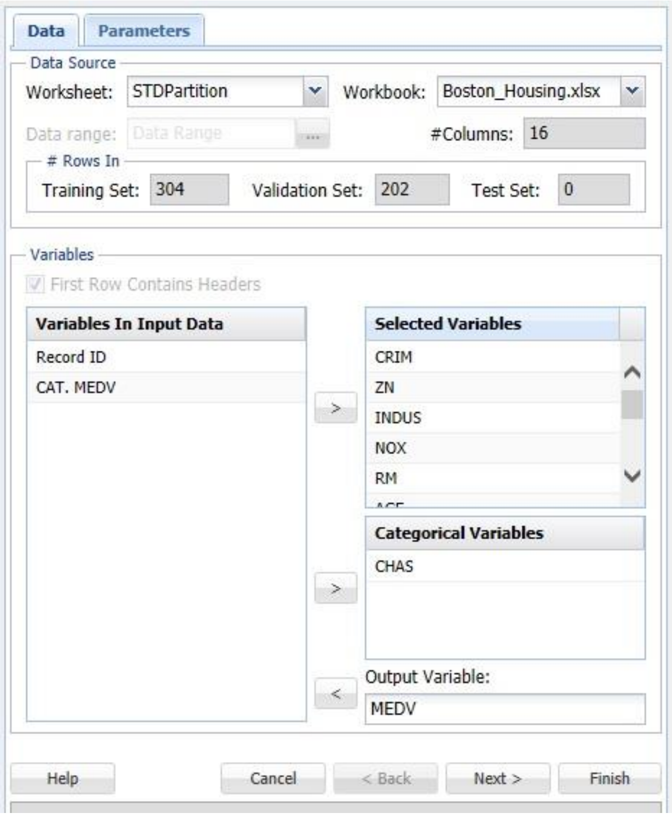
● Automatic percentages — Training Set: 60 %

○ Specify percentages — Validation Set: 40 %

○ Equal percentages — Test Set: 0 %

Click **Predict – Neural Network** – **Automatic** on the Data Science ribbon. The *Neural Network Regression Data* tab appears.

Select **MEDV** as the *Output variable*, ***CHAS** as the Categorical Variable,* and the **remaining variables** as *Selected Variables* (except the CAT.MEDV and Record ID variables). The last variable, CAT.MEDV, is a discrete classification of the MEDV variable and will not be used in this example.



Worksheet: STDPartition
Workbook: Boston_Housing.xlsx
Data range: Data Range
#Columns: 16

Rows In

Training Set: 304
Validation Set: 202
Test Set: 0

☑ First Row Contains Headers

Variables In Input Data
Record ID
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM

Categorical Variables
CHAS

Output Variable: MEDV

Click **Next** to advance to the Parameters tab.

When a neural network with automatic architecture is created, several networks are run with increasing complexity in the architecture. The networks are limited to 2 hidden layers and the number of hidden neurons in each layer is bounded by UB1 = (#features + 1) * 2/3 on the 1st layer and UB2 = (UB1 + 1) * 2/3 on the 2nd layer. For this example, select **Automatic Architecture**. See below for an example on specifying the number of layers when manually defining the network architecture.

First, all networks are trained with 1 hidden layer with the number of nodes not exceeding the UB1 bounds, then a second layer is added and a 2 – layer architecture is tried until the UB2 limit is satisfied.

The limit on the total number of trained networks is the minimum of 100 and (UB1 * (1+UB2)). In this dataset, there are 13 features in the model giving the following bounds:

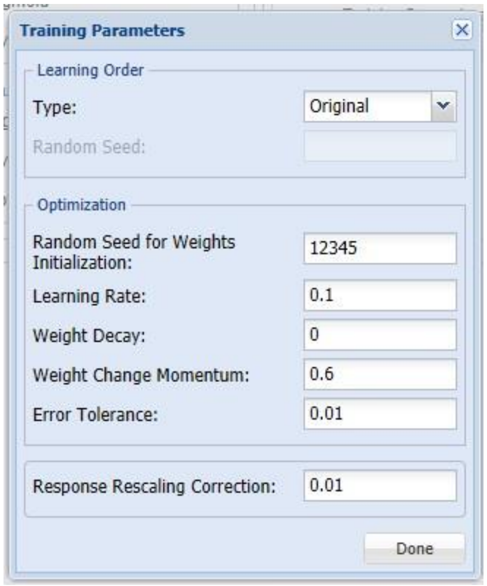
UB1 = Floor ((13 + 1) * 2/3) = 9.33 ~ 9

UB2 = Floor ((9 + 1) * 2/3) = 6.67 ~ 6

(Floor: Rounds a number down to the nearest multiple of significance.)

# Networks Trained = MIN {100, (9 * (1 + 6)} = 63

Users can now change both the Training Parameters and Stopping Rules for the Neural Network. Click Training Parameters to open the Training Parameters dialog. For information on these parameters, please see the Options section that occurs later in this chapter. For now, click Done to accept the default settings and close the dialog.



Training Parameters

Learning Order

Type: Original

Random Seed:

Optimization

Random Seed for Weights Initialization: 12345

Learning Rate: 0.1

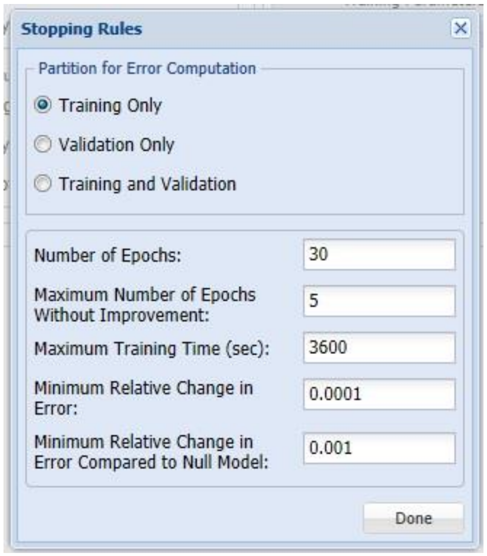
Weight Decay: 0

Weight Change Momentum: 0.6

Error Tolerance: 0.01

Response Rescaling Correction: 0.01

Click Stopping Rules to open the Stopping Rules dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error. For more information on these options, please see the Options section that appears later on in this chapter. For now, click Done to accept the default settings and close the dialog.



Stopping Rules

Partition for Error Computation

● Training Only

○ Validation Only

○ Training and Validation

Number of Epochs: 30

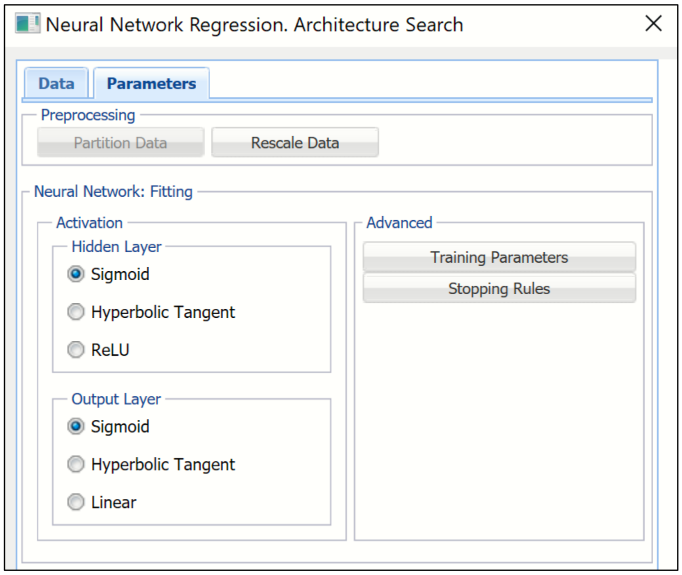
Maximum Number of Epochs Without Improvement: 5

Maximum Training Time (sec): 3600

Minimum Relative Change in Error: 0.0001

Minimum Relative Change in Error Compared to Null Model: 0.001

Keep the defaults for both Hidden Layer and Output Layer. See the Neural Network Regression Options section below for more information on these options.



Neural Network Regression. Architecture Search

Data | Parameters

Preprocessing: Partition Data | Rescale Data

Neural Network: Fitting

Activation

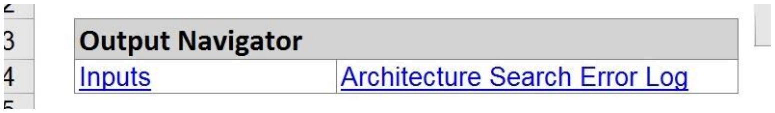
Hidden Layer
- ● Sigmoid
- ○ Hyperbolic Tangent
- ○ ReLU

Output Layer
- ● Sigmoid
- ○ Hyperbolic Tangent
- ○ Linear

Advanced: Training Parameters | Stopping Rules

Click **Finish**.

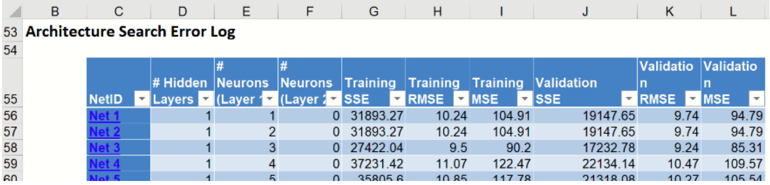
Click the *NNP_Output* tab.



Output Navigator	
Inputs	Architecture Search Error Log

The top section includes the Output Navigator which can be used to quickly navigate to various sections of the output. The Data, Variables, and Parameters/Options sections of the output all reflect inputs chosen by the user.

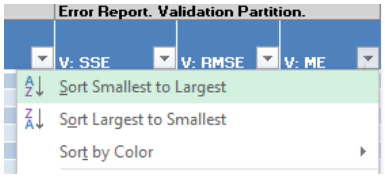
Scroll down to the Error Report, a portion is shown below. This report displays each network created by the Automatic Architecture algorithm and can be sorted by each column by clicking the down arrow next to each column heading.



Architecture Search Error Log

NetID	# Hidden Layers	# Neurons (Layer 1	# Neurons (Layer 2	Training SSE	Training RMSE	Training MSE	Validation SSE	Validation RMSE	Validation MSE
Net 1	1	1	0	31893.27	10.24	104.91	19147.65	9.74	94.79
Net 2	1	2	0	31893.27	10.24	104.91	19147.65	9.74	94.79
Net 3	1	3	0	27422.04	9.5	90.2	17232.78	9.24	85.31
Net 4	1	4	0	37231.42	11.07	122.47	22134.14	10.47	109.57
Net 5	1	5	0	35805.6	10.85	117.78	21318.08	10.27	105.54

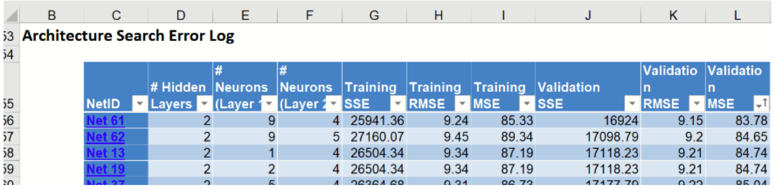
Click the down arrow next to the last column heading, Validation:MSE (Mean Standard Error for the Validation dataset), and select Sort Smallest to Largest from the menu. Note: Sorting is not supported in AnalyticSolver.com.



Error Report. Validation Partition.

V: SSE | V: RMSE | V: ME

Immediately, the records in the table are sorted by smallest value to largest value according to the Validation: MSE values.



Architecture Search Error Log

NetID	# Hidden Layers	# Neurons (Layer 1	# Neurons (Layer 2	Training SSE	Training RMSE	Training MSE	Validation SSE	Validatio n RMSE	Validatio n MSE
Net 61	2	9	4	25941.36	9.24	85.33	16924	9.15	83.78
Net 62	2	9	5	27160.07	9.45	89.34	17098.79	9.2	84.65
Net 13	2	1	4	26504.34	9.34	87.19	17118.23	9.21	84.74
Net 19	2	2	4	26504.34	9.34	87.19	17118.23	9.21	84.74
Net 37	2	5	4	26364.68	9.31	86.73	17177.79	9.22	85.04

Since this column is the average residual error, the error closest to 0 would be the record with the "best" average error, or lowest residual error. Take a look at Net ID 61 which has the lowest average error in the validation dataset. This network contains 2 hidden layers containing 9 neurons in the first hidden layer and 4 neurons in the 2nd hidden layer. The Sum of Squared Error for this network is 25, 941.36 in the Training set and 16924.00 in the Validation set. Note that the number of networks trained is 63 or MIN {100, (9 * (1 + 6)} = 63 (as discussed above). Click the any of the 63 hyperlinks to open the Neural Network Regression Data tab. Click Finish to run the Neural Network Regression method using the input and option settings for the ID selected.

Read on below for the last choice in Neural Network Regression methods, creating a Neural Network, manually.

Manual Neural Network Regression Method Example

This example focuses on creating a Neural Network Manual Architecture. See the Ensemble Methods chapter that appears later on in this guide to see an example on creating a Neural Network using the boosting and bagging ensemble methods.

Inputs

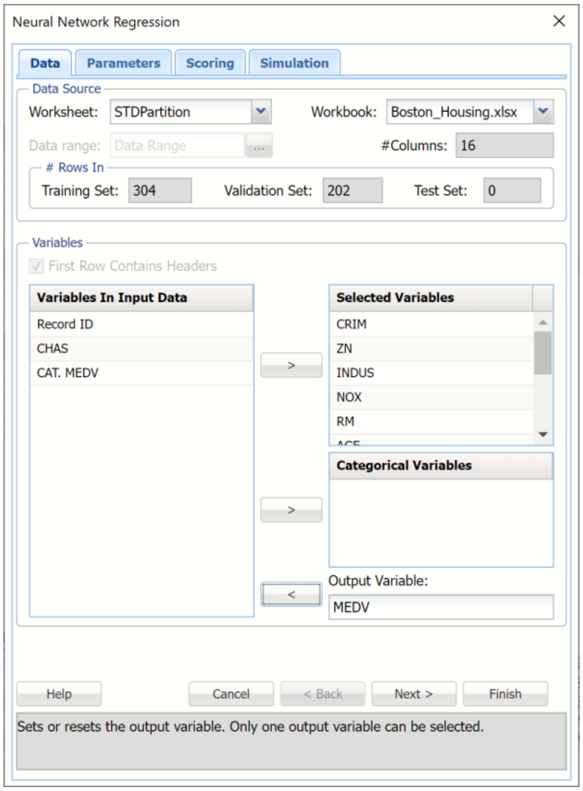
This example will use the same partitioned dataset to illustrate the use of the *Manual Network Architecture* selection.

1. Click back to the STDPartition sheet and then click **Predict – Neural Network** – **Manual Network** on the Data Science ribbon.

2. Select **MEDV** as the *Output variable* and the **remaining variables** as *Selected Variables* (except the CAT.MEDV, CHAS and Record ID variables).

   The last variable, CAT.MEDV, is a discrete classification of the MEDV variable and will not be used in this example. CHAS is a categorial variable which will also not be used in this example.

Neural Network Regression dialog, Data tab



Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

\#Columns: 16

Training Set: 304

Validation Set: 202

Test Set: 0

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM
AGE

Output Variable: MEDV

Sets or resets the output variable. Only one output variable can be selected.

3. Click **Next** to advance to the next tab.

   As discussed in the previous sections, Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the regression method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

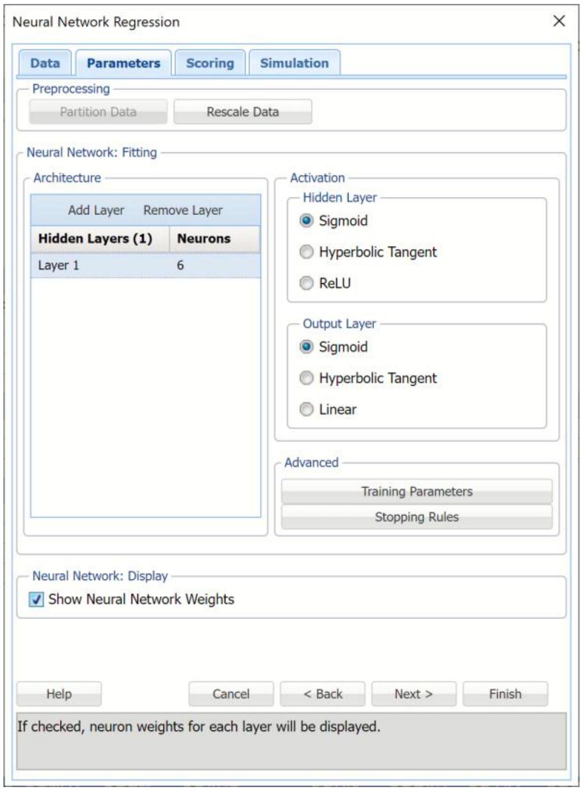
4. Click **Add Layer** to add a hidden layer to the Neural Network. Enter **6** for Neurons for this layer. To remove a layer, select the layer to be removed, then click *Remove Layer*.

   To change the Training Parameters and Stopping Rules for the Neural Network, click Training Parameters and Stopping Rules, respectively. For this example, we will use the defaults. See the Neural Network Regression Options below for more information on these parameters.

5. Leave Sigmoid selected for Hidden Layer and output Layer. See the Neural Network Regression Options section below for more information on these options.

6. Select Show Neural Network Weights to display this information in the output.

Neural Network Regression dialog, Parameters tab



Hidden Layers (1)	Neurons
Layer 1	6

Hidden Layer: ● Sigmoid ○ Hyperbolic Tangent ○ ReLU

Output Layer: ● Sigmoid ○ Hyperbolic Tangent ○ Linear

☑ Show Neural Network Weights

If checked, neuron weights for each layer will be displayed.

7. Click **Next** to advance to the Scoring tab.

8. Select all four options for **Score Training/Validation data**.

   When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the Regression Trees output.

   When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the Regression Trees output.

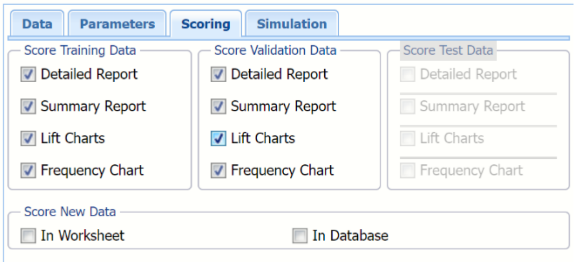
   When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and ROC Curve plots in the output.

   When Frequency Chart is selected, a frequency chart will be displayed when the NNP_TrainingScore and NNP_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

   Since we did not create a test partition, the options for Score test data are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

   See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.

Neural Network Regression dialog, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

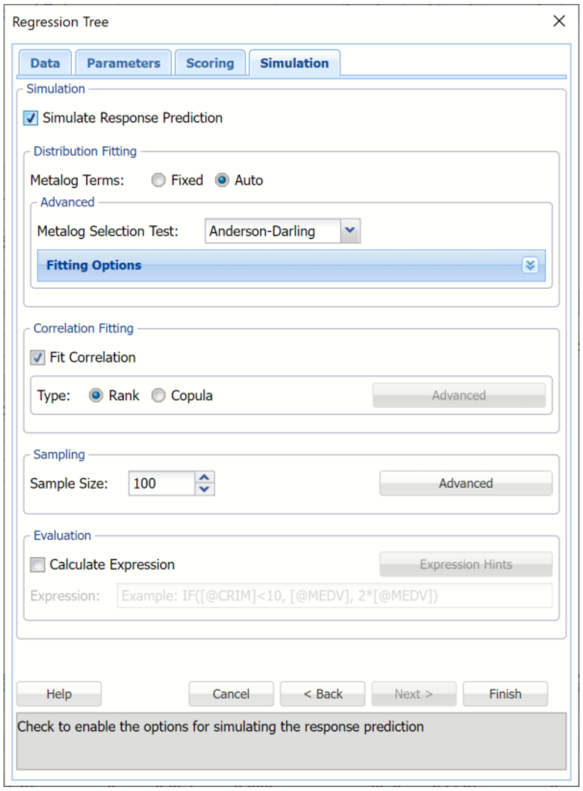
Score New Data
□ In Worksheet
□ In Database

9. Click **Next** to advance to the Simulation tab.

10. Select **Simulation Response Prediction** to enable all options on the Simulation tab of the Regression Tree dialog. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, NNP_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

    Regression Tree *dialog, Simulation tab*



Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula Advanced

Sampling
Sample Size: 100 Advanced

Evaluation
□ Calculate Expression Expression Hints
Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

Check to enable the options for simulating the response prediction

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RT_Simulation output tab.

Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression. Note that variable names are case sensitive. See any of the prediction methods to see the Expressison field in use.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

11. Click **Finish** to run Regression Tree on the example dataset.

Output

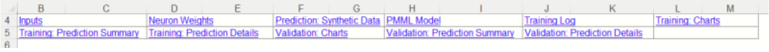
Output sheets containing the results of the Neural Network will be inserted into your active workbook to the right of the STDPartition worksheet.

NNP_Output

This result worksheet includes 3 segments: Output Navigator, Inputs and Nueron Weights.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

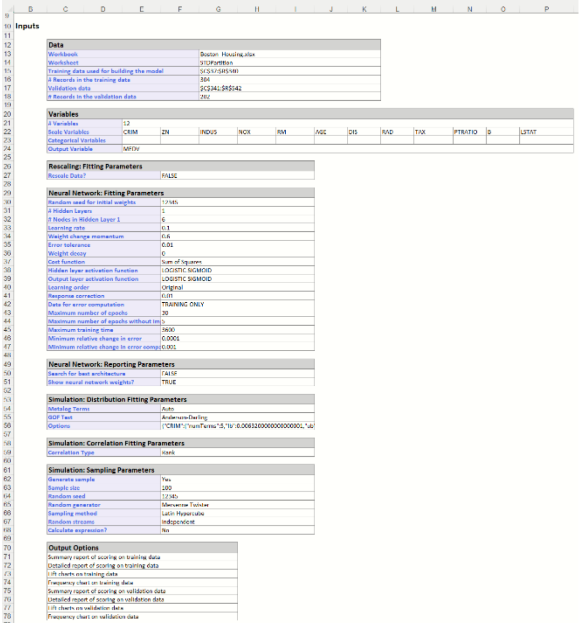
  NNP_Output: Output Navigator



Inputs	Neuron Weights	Prediction: Synthetic Data	PMML Model	Training Log	Training: Charts
Training: Prediction Summary	Training: Prediction Details	Validation: Charts	Validation: Prediction Summary	Validation: Prediction Details	

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Discriminant Analysis dialog.

  NNP_Output, Inputs Report



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Neural Network: Fitting Parameters	
Random seed for initial weights	12345
# Hidden Layers	1
# Nodes in Hidden Layer 1	6
Learning rate	0.1
Weight change momentum	0.6
Error tolerance	0.01
Weight decay	0
Cost function	Sum of Squares
Hidden layer activation function	LOGISTIC SIGMOID
Output layer activation function	LOGISTIC SIGMOID
Learning order	Original
Response correction	0.01
Data for error computation	TRAINING ONLY
Maximum number of epochs	30
Maximum number of epochs without im	5
Maximum training time	3600
Minimum relative change in error	0.0001
Minimum relative change in error comp	0.001

Neural Network: Reporting Parameters	
Search for best architecture	FALSE
Show neural network weights?	TRUE

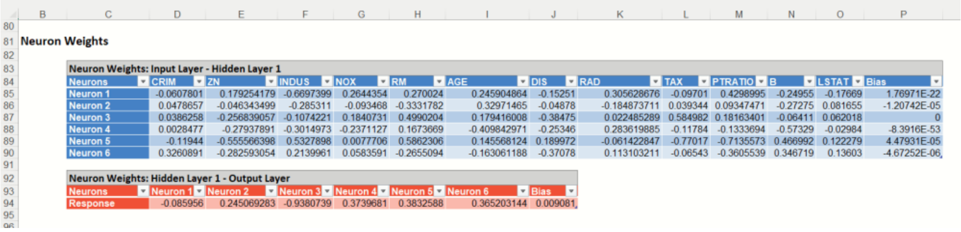
Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.0063200000000000001,"ub"

Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Neuron Weights:** Analytic Solver Data Science provides intermediate information produced during the last pass through the network. Scroll down the *NNP_Output* worksheet to the Interlayer connections' weights table.



Neuron Weights

Neuron Weights: Input Layer - Hidden Layer 1

Neurons	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	Bias
Neuron 1	-0.0607801	0.179254179	-0.6697399	0.2644354	0.270024	0.245904864	-0.15251	0.305628676	-0.09701	0.4298995	-0.24955	-0.17669	1.76971E-22
Neuron 2	0.0478657	-0.046343499	-0.285311	-0.093468	-0.3331782	0.32971465	-0.04878	-0.184873711	0.039344	0.09347471	-0.27275	0.081655	-1.20742E-05
Neuron 3	0.0386258	-0.256839057	-0.1074221	0.1840731	0.4990204	0.179416008	-0.38475	0.022485289	0.584982	0.18163401	-0.06411	0.062018	0
Neuron 4	0.0028477	-0.27937891	-0.3014973	-0.2371127	0.1673669	-0.409842971	-0.25346	0.283619885	-0.11784	-0.1333694	-0.57329	-0.02984	-8.3916E-53
Neuron 5	-0.11944	-0.555566398	0.5327898	0.0077706	0.5862306	0.145568124	0.189972	-0.061422847	-0.77017	-0.7135573	0.466992	0.122279	4.47931E-05
Neuron 6	0.3260891	-0.282593054	0.2139961	0.0583591	-0.2655094	-0.163061188	-0.37078	0.113103211	-0.06543	-0.3605539	0.346719	0.13603	-4.67252E-06

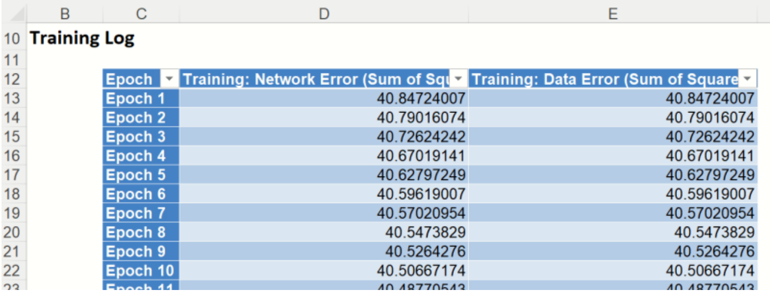
Neuron Weights: Hidden Layer 1 - Output Layer

Neurons	Neuron 1	Neuron 2	Neuron 3	Neuron 4	Neuron 5	Neuron 6	Bias
Response	-0.085956	0.245069283	-0.9380739	0.3739681	0.3832588	0.365203144	0.009081

Recall that a key element in a neural network is the weights for the connections between nodes. In this example, we chose to have one hidden layer with 6 neurons. The Inter-Layer Connections Weights table contains the final values for the weights between the input layer and the hidden layer, between hidden layers, and between the last hidden layer and the output layer. This information is useful at viewing the "insides" of the neural network; however, it is unlikely to be of utility to the data analyst end-user. Displayed above are the final connection weights between the input layer and the hidden layer for our example and also the final weights between the hidden layer and the output layer.

NNP_TrainLog

Click the **Training Log** link on the Output Navigator or the NNP_TrainLog worksheet tab to display the following log.



Training Log

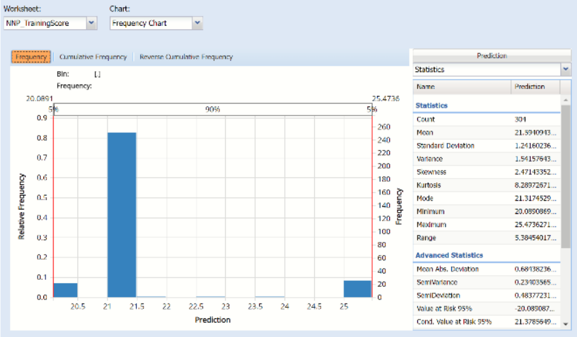
Epoch	Training: Network Error (Sum of Sq	Training: Data Error (Sum of Square
Epoch 1	40.84724007	40.84724007
Epoch 2	40.79016074	40.79016074
Epoch 3	40.72624242	40.72624242
Epoch 4	40.67019141	40.67019141
Epoch 5	40.62797249	40.62797249
Epoch 6	40.59619007	40.59619007
Epoch 7	40.57020954	40.57020954
Epoch 8	40.5473829	40.5473829
Epoch 9	40.5264276	40.5264276
Epoch 10	40.50667174	40.50667174
Epoch 11	40.48770543	40.48770543

During an epoch, each training record is fed forward in the network and classified. The error is calculated and is back propagated for the weights correction. Weights are continuously adjusted during the epoch. The sum of squares error is computed as the records pass through the network but does not report the sum of squares error after the final weight adjustment. Scoring of the training data is performed using the final weights so the training classification error may not exactly match with the last epoch error in the Epoch log.

NNP_TrainingScore

Click the *NNP_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Prediction Summary and the Training: Prediction Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: NNP_TrainingScore

Chart: Frequency Chart

Bin: [,]

Frequency:

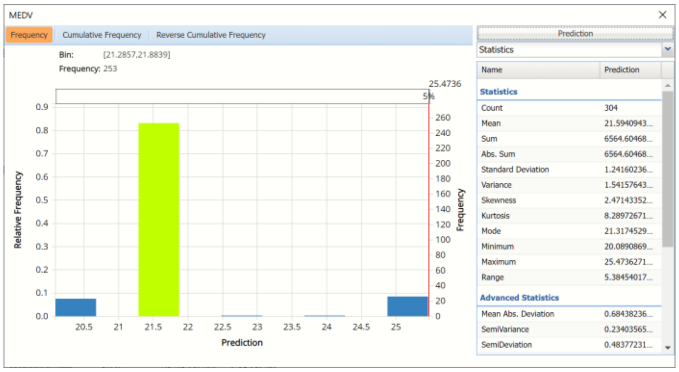
Prediction

Statistics

Name	Prediction
Statistics	
Count	304
Mean	21.5940943...
Standard Deviation	1.24160236...
Variance	1.54157643...
Skewness	2.47143352...
Kurtosis	8.28972671...
Mode	21.3174529...
Minimum	20.0890869...
Maximum	25.4736271...
Range	5.38454017...
Advanced Statistics	
Mean Abs. Deviation	0.68438236...
SemiVariance	0.23403565...
SemiDeviation	0.48377231...
Value at Risk 95%	-20.089087...
Cond. Value at Risk 95%	21.3785649...

- **Frequency Charts:** The output variable frequency chart opens automatically once the *NNP_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *NNP_TrainingScore* tab.

  Frequency chart displaying prediction data



MEDV

Bin: [21.2857,21.8839]

Frequency: 253

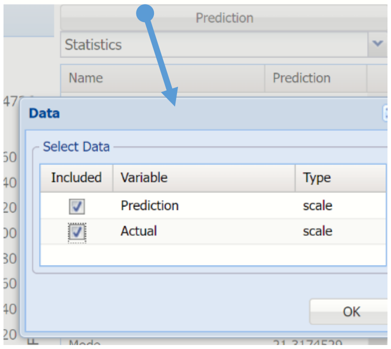
Prediction

Statistics

Name	Prediction
Statistics	
Count	304
Mean	21.5940943...
Sum	6564.60468...
Abs. Sum	6564.60468...
Standard Deviation	1.24160236...
Variance	1.54157643...
Skewness	2.47143352...
Kurtosis	8.28972671...
Mode	21.3174529...
Minimum	20.0890869...
Maximum	25.4736271...
Range	5.38454017...
Advanced Statistics	
Mean Abs. Deviation	0.68438236...
SemiVariance	0.23403565...
SemiDeviation	0.48377231...

To add the Actual data to the chart, click Prediction in the upper right hand corner and select both checkboxes in the Data dialog.

Click Prediction to add Actual data to the interactive chart.

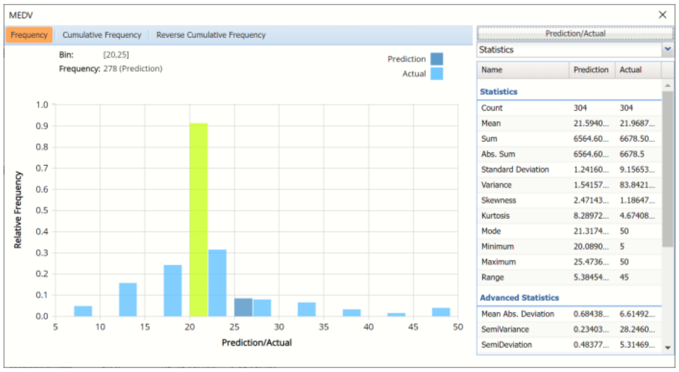


Data

Select Data

Included	Variable	Type
☑	Prediction	scale
☑	Actual	scale

Notice in the screenshot below that both the Prediction and Actual data appear in the chart together, and statistics for both appear on the right.



Bin: [20,25]
Frequency: 278 (Prediction)

Name	Prediction	Actual
Statistics		
Count	304	304
Mean	21.5940...	21.9687...
Sum	6564.60...	6678.50...
Abs. Sum	6564.60...	6678.5
Standard Deviation	1.24160...	9.15653...
Variance	1.54157...	83.8421...
Skewness	2.47143...	1.18647...
Kurtosis	8.28972...	4.67408...
Mode	21.3174...	50
Minimum	20.0890...	5
Maximum	25.4736...	50
Range	5.38454...	45
Advanced Statistics		
Mean Abs. Deviation	0.68438...	6.61492...
SemiVariance	0.23403...	28.2460...
SemiDeviation	0.48377...	5.31469...

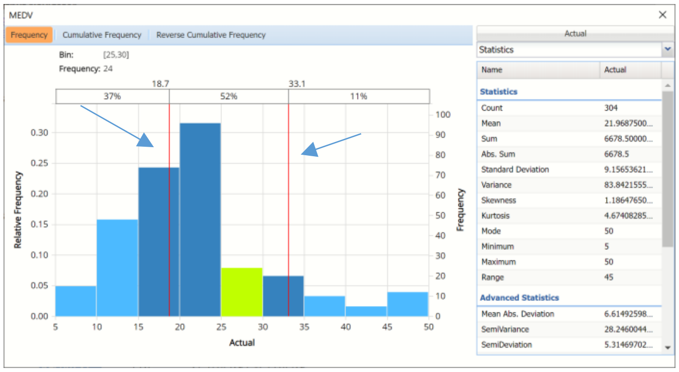
To remove either the Original or the Synthetic data from the chart, click Original/Synthetic in the top right and then uncheck the data type to be removed.

This chart behaves the same as the interactive chart in the Analyze Data feature found on the Explore menu.

- Use the mouse to hover over any of the bars in the graph to populate the Bin and Frequency headings at the top of the chart.

- When displaying either Prediction or Actual data (not both), red vertical lines will appear at the 5% and 95% percentile values in all three charts (Frequency, Cumulative Frequency and Reverse Cumulative Frequency) effectively displaying the 90th confidence interval. The middle percentage is the percentage of all the variable values that lie within the 'included' area, i.e. the darker shaded area. The two percentages on each end are the percentage of all variable values that lie outside of the 'included' area or the "tails". i.e. the lighter shaded area. Percentile values can be altered by moving either red vertical line to the left or right.

  Frequency chart with percentage markers moved

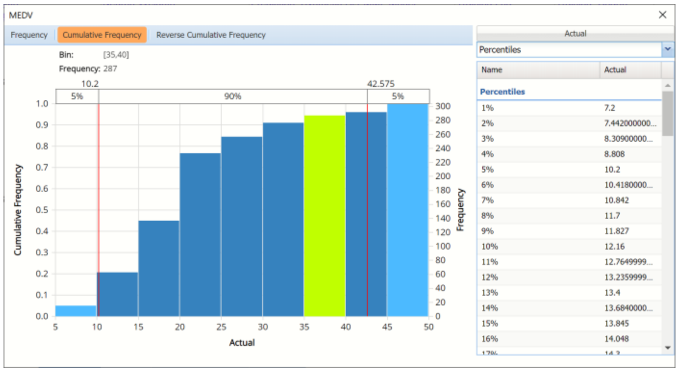
  

  Bin: [25,30]
  Frequency: 24

Name	Actual
Statistics	
Count	304
Mean	21.9687500...
Sum	6678.50000...
Abs. Sum	6678.5
Standard Deviation	9.15653621...
Variance	83.8421555...
Skewness	1.18647650...
Kurtosis	4.67408285...
Mode	50
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.61492598...
SemiVariance	28.2460044...
SemiDeviation	5.31469702...

- Click Cumulative Frequency and Reverse Cumulative Frequency tabs to see the Cumulative Frequency and Reverse Cumulative Frequency charts, respectively.

Cumulative Frequency chart and Percentiles displayed



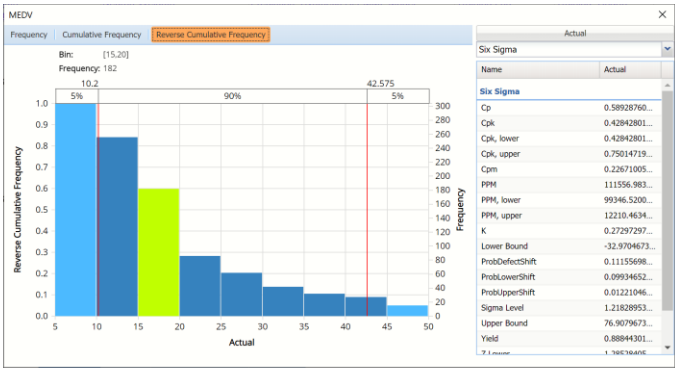
Actual

Percentiles

Name	Actual
Percentiles	
1%	7.2
2%	7.442000000...
3%	8.30900000...
4%	8.808
5%	10.2
6%	10.4180000...
7%	10.842
8%	11.7
9%	11.827
10%	12.16
11%	12.7649999...
12%	13.2359999...
13%	13.4
14%	13.6840000...
15%	13.845
16%	14.048

- Click the down arrow next to Statistics to view Percentiles for each type of data along with Six Sigma indices.

  Reverse Cumulative Frequency chart and Six Sigma indices displayed.



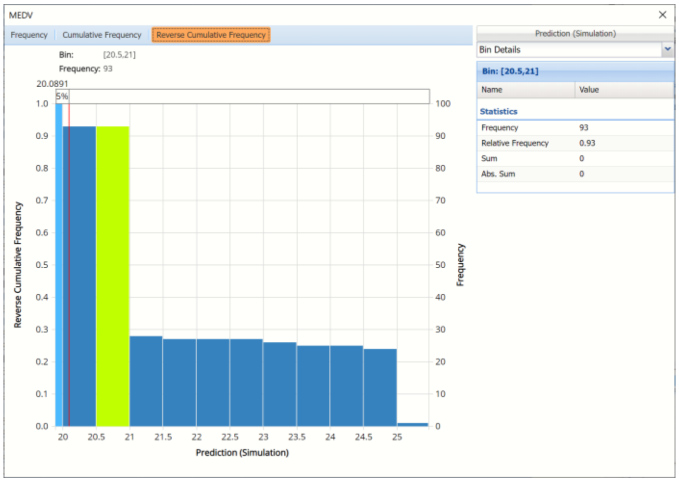
Actual

Six Sigma

Name	Actual
Six Sigma	
Cp	0.58928760...
Cpk	0.42842801...
Cpk, lower	0.42842801...
Cpk, upper	0.75014719...
Cpm	0.22671005...
PPM	111556.983...
PPM, lower	99346.5200...
PPM, upper	12210.4634...
K	0.27297297...
Lower Bound	-32.9704673...
ProbDefectShift	0.11155698...
ProbLowerShift	0.09934652...
ProbUpperShift	0.01221046...
Sigma Level	1.21828953...
Upper Bound	76.9079673...
Yield	0.88844301...

- Click the down arrow next to Statistics to view Bin Details to display information related to each bin.

  Reverse Cumulative Frequency chart and Bin Details pane displayed



Prediction (Simulation)

Bin Details

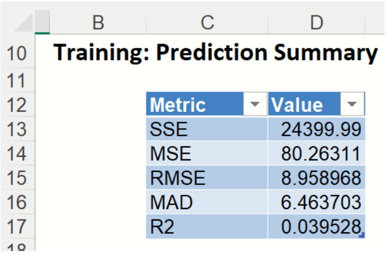
Bin: [20.5,21]

Name	Value
Statistics	
Frequency	93
Relative Frequency	0.93
Sum	0
Abs. Sum	0

- Use the Chart Options view to manually select the number of bins to use in the chart, as well as to set personalization options.

As discussed above, see the Analyze Data section of the Exploring Data chapter for an in-depth discussion of this chart as well as descriptions of all statistics, percentiles, bin metrics and six sigma indices.

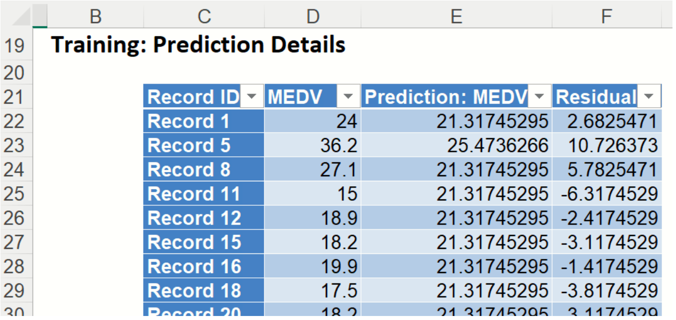
- **Training: Prediction Summary:** Click the Training: Prediction Summary link on the Output Navigator to open the Training Summary. This data table displays various statistics to measure the performance of the trained network: Sum of Squared Error (SSE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), the Median Absolute Deviation (MAD) and the Coefficient of Determination (R^2).



Training: Prediction Summary

Metric	Value
SSE	24399.99
MSE	80.26311
RMSE	8.958968
MAD	6.463703
R2	0.039528

- **Training: Prediction Details:** Scroll down to view the Prediction Details data table. This table displays the Actual versus Predicted values, along with the Residuals, for the training dataset.



Training: Prediction Details

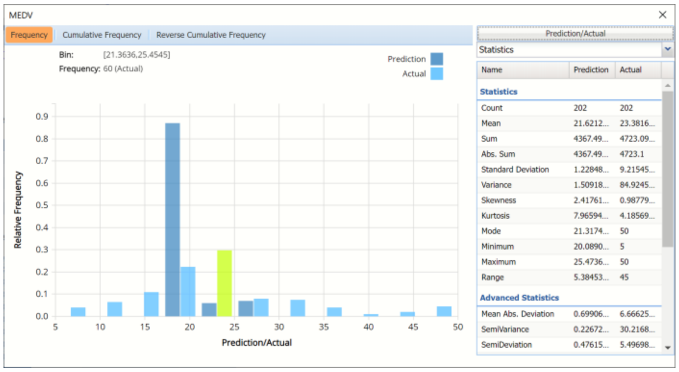
Record ID	MEDV	Prediction: MEDV	Residual
Record 1	24	21.31745295	2.6825471
Record 5	36.2	25.4736266	10.726373
Record 8	27.1	21.31745295	5.7825471
Record 11	15	21.31745295	-6.3174529
Record 12	18.9	21.31745295	-2.4174529
Record 15	18.2	21.31745295	-3.1174529
Record 16	19.9	21.31745295	-1.4174529
Record 18	17.5	21.31745295	-3.8174529
Record 20	18.2	21.31745295	3.1174529

NNP_ValidationScore

Another key interest in a data-mining context will be the predicted and actual values for the MEDV variable along with the residual (difference) for each predicted value in the *Validation* partition.

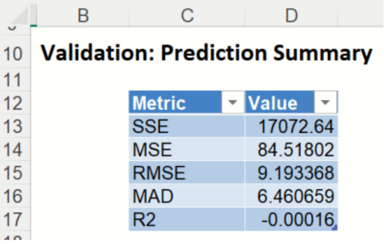
NNP_ValidationScore displays the newly added Output Variable frequency chart, the Validation: Prediction Summary and the Validation: Prediction Details report. All calculations, charts and predictions on the NNP_ValidationScore output sheet apply to the Validation partition.

- **Frequency Charts:** The output variable frequency chart for the validation partition opens automatically once the *NNP_ValidationScore* worksheet is selected. This chart displays a detailed, interactive frequency chart for the Actual variable data and the Predicted data, for the validation partition. For more information on this chart, see the NNP_TrainingScore explanation above.



Name	Prediction	Actual
Statistics		
Count	202	202
Mean	21.6212...	23.3816...
Sum	4367.49...	4723.09...
Abs. Sum	4367.49...	4723.1
Standard Deviation	1.22848...	9.21545...
Variance	1.50918...	84.9245...
Skewness	2.41761...	0.98779...
Kurtosis	7.96594...	4.18569...
Mode	21.3174...	50
Minimum	20.0890...	5
Maximum	25.4736...	50
Range	5.38453...	45
Advanced Statistics		
Mean Abs. Deviation	0.69906...	6.66625...
SemiVariance	0.22672...	30.2168...
SemiDeviation	0.47615...	5.49698...

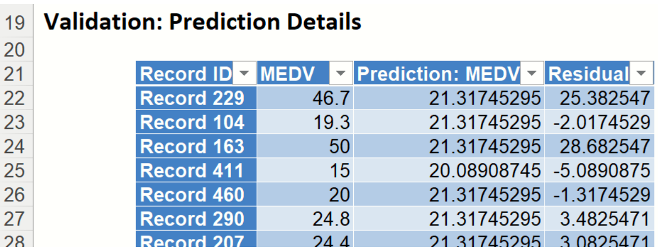
- **Prediction Summary:** In the Prediction Summary report, Analytic Solver Data Science displays the total sum of squared errors summaries for the Validation partition.



Validation: Prediction Summary

Metric	Value
SSE	17072.64
MSE	84.51802
RMSE	9.193368
MAD	6.460659
R2	-0.00016

- **Prediction Details:** Scroll down to the Validation: Prediction Details report to find the Prediction value for the MEDV variable for each record in the Validation partition, as well as the Residual value.



Validation: Prediction Details

Record ID	MEDV	Prediction: MEDV	Residual
Record 229	46.7	21.31745295	25.382547
Record 104	19.3	21.31745295	-2.0174529
Record 163	50	21.31745295	28.682547
Record 411	15	20.08908745	-5.0890875
Record 460	20	21.31745295	-1.3174529
Record 290	24.8	21.31745295	3.4825471
Record 207	24.4	21.31745295	3.0825471

RROC charts, shown below, are better indicators of fit. Read on to view how these more sophisticated tools can tell us about the fit of the neural network to our data.

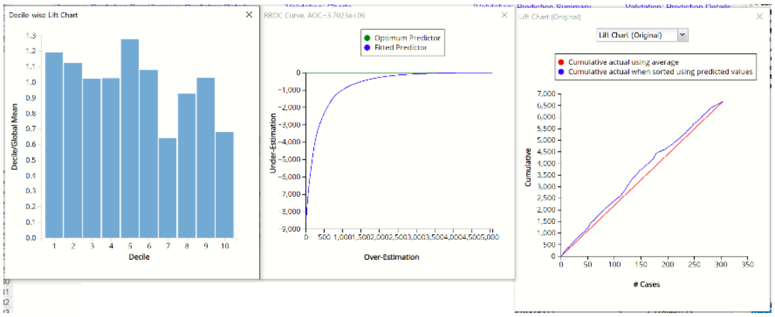
NNP_TrainingDataLiftChart & NNP_ValidationDataLiftChart

Click the **NNP_TrainingLiftChart** and **NNP_ValidationLiftChart** tabs to view the lift charts and Regression ROC charts for both the training and validation datasets.

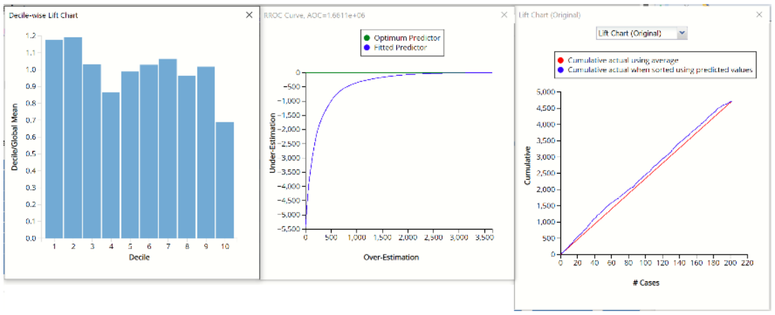
Lift charts and Regression ROC Curves are visual aids for measuring model performance. Lift Charts consist of a lift curve and a baseline. The greater the area between the lift curve and the baseline, the better the model. RROC (regression receiver operating characteristic) curves plot the performance of regressors by graphing over-estimations (or predicted values that are too high) versus underestimations (or predicted values that are too low.) The closer the curve is to the top left corner of the graph (in other words, the smaller the area above the curve), the better the performance of the model.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select NNP_TrainingLiftChart or NNP_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, RROC Curve and Lift Chart for Training Partition



Decile-wise Lift Chart, RROC Curve and Lift Chart for Valid. Partition

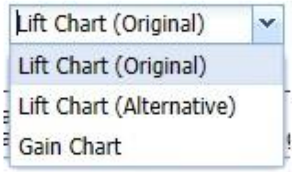


After the model is built using the training data set, the model is used to score on the training data set and the validation data set (if one exists). Then the data set(s) are sorted using the predicted output variable value. After sorting, the actual outcome values of the output variable are cumulated and the lift curve is drawn as the number of cases versus the cumulated value. The baseline (red line connecting the origin to the end point of the blue line) is drawn as the number of cases versus the average of actual output variable values multiplied by the number of cases.

The decilewise lift curve is drawn as the decile number versus the cumulative actual output variable value divided by the decile's mean output variable value. This bars in this chart indicate the factor by which the NNP model outperforms a random assignment, one decile at a time. Typically, this graph will have a "stairstep" appearance - the bars will descend in order from left to right. This means that the model is "binning" the records correctly, from highest priced to lowest. However, in this example, the left most bars are shorter than bars appearing to the right. This type of graph indicates that the model might not be a good fit to the data. Additional analysis is required.

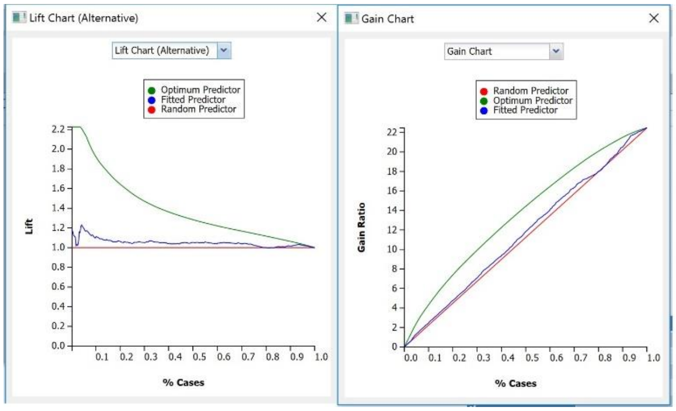
The Regression ROC curve (RROC) was updated in V2017. This new chart compares the performance of the regressor (Fitted Classifier) with an Optimum Classifier Curve. The Optimum Classifier Curve plots a hypothetical model that would provide perfect prediction results. The best possible prediction performance is denoted by a point at the top left of the graph at the intersection of the x and y axis. This point is sometimes referred to as the “perfect classification”. Area Over the Curve (AOC) is the space in the graph that appears above the ROC curve and is calculated using the formula: sigma^2 * n^2/2 where n is the number of records The smaller the AOC, the better the performance of the model.

In V2017, two new charts were introduced: a new Lift Chart and the Gain Chart. To display these new charts, click the down arrow next to Lift Chart (Original), in the Original Lift Chart, then select the desired chart.

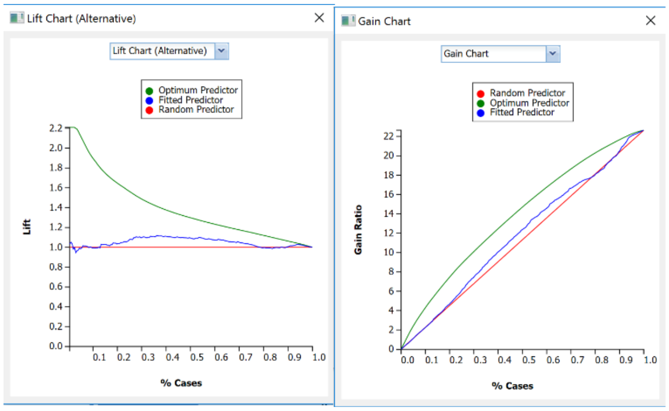


Select Lift Chart (Alternative) to display Analytic Solver Data Science's new Lift Chart. Each of these charts consists of an Optimum Classifier curve, a Fitted Classifier curve, and a Random Classifier curve. The Optimum Classifier curve plots a hypothetical model that would provide perfect classification for our data. The Fitted Classifier curve plots the fitted model and the Random Classifier curve plots the results from using no model or by using a random guess (i.e. for x% of selected observations, x% of the total number of positive observations are expected to be correctly classified).

Lift Chart (Alternative) and Gain Chart for Training Partition



Lift Chart (Alternative) and Gain Chart for Validation Partition



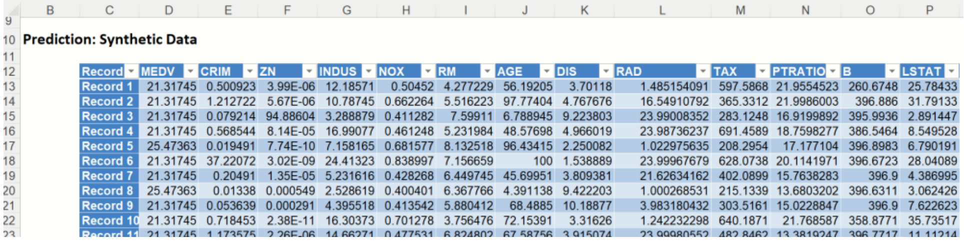
Click the down arrow and select Gain Chart from the menu. In this chart, the Gain Ratio is plotted against the % Cases.

NNP_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, NNP_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Neural Network Regression dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predicted values for the training data (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



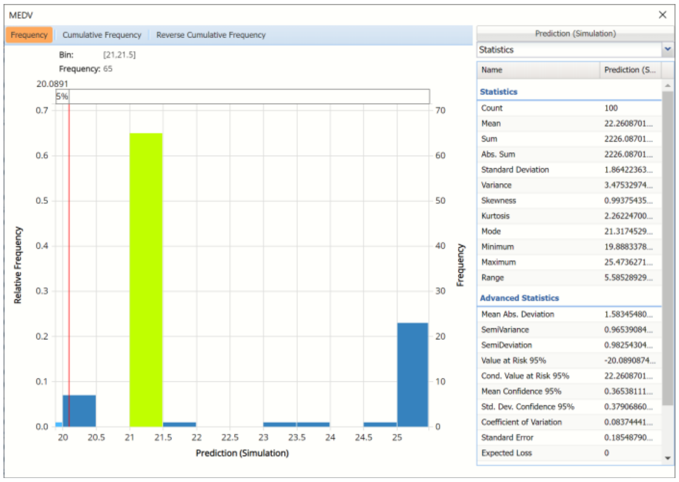
Prediction: Synthetic Data

Record	MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	21.31745	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19205	3.70118	1.485154091	597.5868	21.9554523	260.6748	25.78433
Record 2	21.31745	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404	4.767676	16.54910792	365.3312	21.9986003	396.886	31.79133
Record 3	21.31745	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945	9.223803	23.99008352	283.1248	16.9199892	395.9936	2.891447
Record 4	21.31745	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698	4.966019	23.98736237	691.4589	18.7598277	386.5464	8.549528
Record 5	25.47363	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415	2.250082	1.022975635	208.2954	17.177104	396.8983	6.790191
Record 6	21.31745	37.22072	3.02E-09	24.41323	0.838997	7.156659	100	1.538889	23.99967679	628.0738	20.1141971	396.6723	28.04089
Record 7	21.31745	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.69951	3.809381	21.62634162	402.0899	15.7638283	396.9	4.386995
Record 8	25.47363	0.01338	0.000549	2.528619	0.400401	6.367766	4.391138	9.422203	1.000268531	215.1339	13.6803202	396.6311	3.062426
Record 9	21.31745	0.053639	0.000291	4.395518	0.413542	5.880412	68.4885	10.18877	3.983180432	303.5161	15.0228847	396.9	7.622623
Record 10	21.31745	0.718453	2.38E-11	16.30373	0.701278	3.756476	72.15391	3.31626	1.242232298	640.1871	21.768587	358.8771	35.73517
Record 11	21.31745	1.173575	2.26E-06	14.66271	0.477531	6.824802	67.58756	3.915074	23.99980552	482.8462	13.3819247	396.7717	11.11214

The data contained in the Synthetic Data report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training data, the synthetic data and the expression, if it exists. (Recall that no expression was entered in this example.)

Frequency Chart for Prediction (Simulation) data

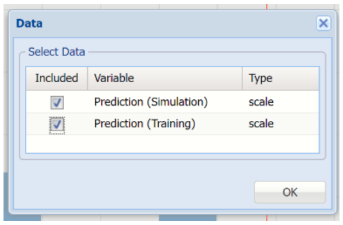


Name	Prediction (S...
Statistics	
Count	100
Mean	22.2608701...
Sum	2226.08701...
Abs. Sum	2226.08701...
Standard Deviation	1.86422363...
Variance	3.47532974...
Skewness	0.99375435...
Kurtosis	2.26224700...
Mode	21.3174529...
Minimum	19.8883378...
Maximum	25.4736271...
Range	5.58528929...
Advanced Statistics	
Mean Abs. Deviation	1.58345480...
SemiVariance	0.96539084...
SemiDeviation	0.98254304...
Value at Risk 95%	-20.0890874...
Cond. Value at Risk 95%	22.2608701...
Mean Confidence 95%	0.36538111...
Std. Dev. Confidence 95%	0.37906860...
Coefficient of Variation	0.08374441...
Standard Error	0.18548790...
Expected Loss	0

Click *Prediction (Simulation)* to add the training data to the chart.

Click *Prediction(Simulation)* and *Prediction (Training)* to change the Data view.

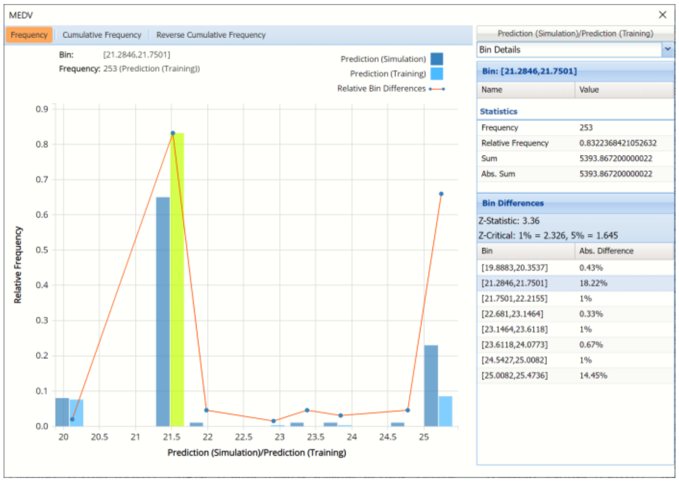
Data Dialog



Included	Variable	Type
☑	Prediction (Simulation)	scale
☑	Prediction (Training)	scale

In the chart below, the dark blue bars display the frequencies for the synthetic data and the light blue bars display the frequencies for the predicted values in the Training partition.

Prediction (Simulation) and Prediction (Training) Frequency chart for MEDV variable



Bin: [21.2846,21.7501]
Frequency: 253 (Prediction (Training))

Bin: [21.2846,21.7501]

Name	Value
Statistics	
Frequency	253
Relative Frequency	0.8322368421052632
Sum	5393.867200000022
Abs. Sum	5393.867200000022

Bin Differences

Z-Statistic: 3.36
Z-Critical: 1% = 2.326, 5% = 1.645

Bin	Abs. Difference
[19.8883,20.3537]	0.43%
[21.2846,21.7501]	18.22%
[21.7501,22.2155]	1%
[22.681,23.1464]	0.33%
[23.1464,23.6118]	1%
[23.6118,24.0773]	0.67%
[24.5427,25.0082]	1%
[25.0082,25.4736]	14.45%

The Relative Bin Differences curve charts the absolute differences between the data in each bin. Click the down arrow next to Statistics to view the Bin Details pane to display the calculations.

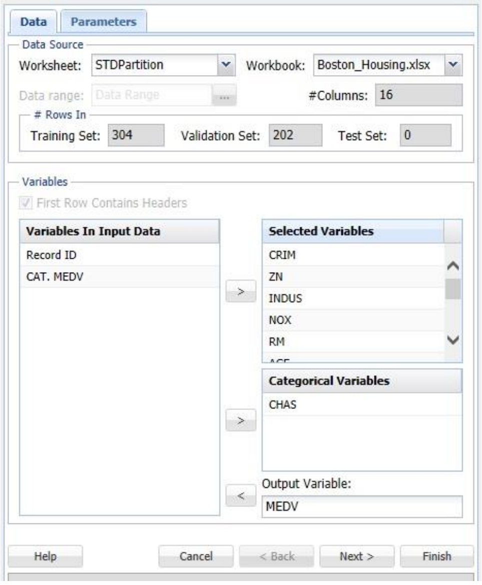
Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

For information on *NNP_Stored*, please see the “Scoring New Data” chapter within the Analytic Solver Data Science User Guide.

Neural Network Regression Method Options

The options below appear on one of the *Neural Network Regression* method dialog tabs.

Neural Network Regression dialog, Data tab



Worksheet: STDPartition
Workbook: Boston_Housing.xlsx
Data range: Data Range
#Columns: 16
Rows In
Training Set: 304
Validation Set: 202
Test Set: 0

☑ First Row Contains Headers

Variables In Input Data
Record ID
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM
AGE

Categorical Variables
CHAS

Output Variable:
MEDV

Neural Network Regression dialog, Data tab

Variables In Input Data

All variables in the dataset are listed here.

Selected Variables

Variables listed here will be utilized in the Analytic Solver Data Science output.

Categorical Variables

Place categorical variables from the Variables listbox to be included in the model by clicking the > command button. The Neural Network Regression algorithm will accept non-numeric categorical variables.

Output Variable

Select the variable whose outcome is to be predicted here.

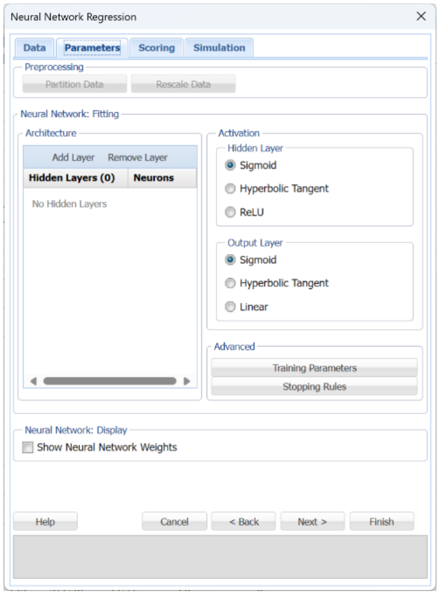
Neural Network Regression dialog, Parameters tab

See below for options appearing on the Neural Network Regression – Parameters tab. Note: The *Neural Network Automatic Regression – Parameters* tab does not include Architecture, but is otherwise the same.

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the regression method. If partitioning has already occurred on the dataset, this option will be disabled. For more information on partitioning, please see the Data Science Partitioning chapter.

Neural Network Regression dialog, Parameters tab



Neural Network Regression

Data | Parameters | Scoring | Simulation

Preprocessing: Partition Data, Rescale Data

Neural Network: Fitting

Architecture: Add Layer, Remove Layer

Hidden Layers (0)	Neurons
No Hidden Layers	

Activation

Hidden Layer: ● Sigmoid ○ Hyperbolic Tangent ○ ReLU

Output Layer: ● Sigmoid ○ Hyperbolic Tangent ○ Linear

Advanced: Training Parameters, Stopping Rules

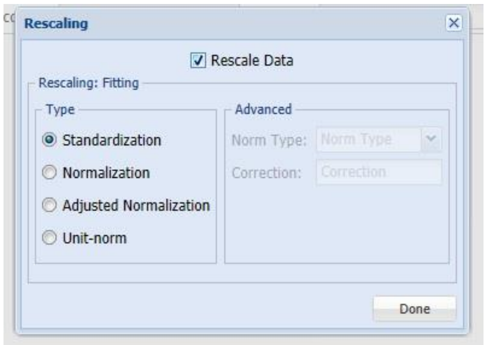
Neural Network: Display

□ Show Neural Network Weights

Help | Cancel | < Back | Next > | Finish

Rescale Data

Click **Rescale Data** to open the Rescaling dialog.



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type: ● Standardization ○ Normalization ○ Adjusted Normalization ○ Unit-norm

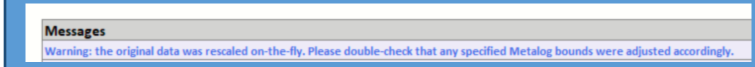
Advanced: Norm Type: Norm Type; Correction: Correction

Done

Use Rescaling to normalize one or more features in your data during the data preprocessing stage. Analytic Solver Data Science provides the following methods for feature scaling: Standardization, Normalization, Adjusted Normalization and Unit Norm. For more information on this new feature, see the Rescale Continuous Data section within the Transform Continuous Data chapter that occurs earlier in this guide.

Notes on Rescaling and the Simulation functionality

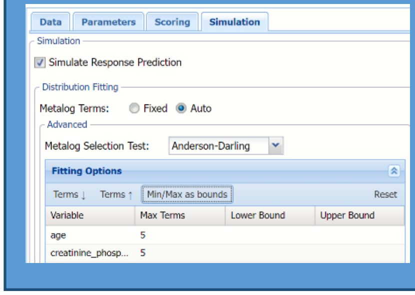
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the NNP_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Hidden Layers/Neurons

Click Add Layer to add a hidden layer. To delete a layer, click Remove Layer. Once the layer is added, enter the desired Neurons.

Hidden Layer

Nodes in the hidden layer receive input from the input layer. The output of the hidden nodes is a weighted sum of the input values. This weighted sum is computed with weights that are initially set at random values. As the network “learns”, these weights are adjusted. This weighted sum is used to compute the hidden node’s output using a *transfer function*.

Select *Sigmoid* (the default setting) to use a logistic function for the transfer function with a range of 0 and 1. This function has a “squashing effect” on very small or very large values but is almost linear in the range where the value of the function is between 0.1 and 0.9.

Select *Hyperbolic Tangent* to use the tanh function for the transfer function, the range being -1 to 1. If more than one hidden layer exists, this function is used for all layers.

ReLU (Rectified Linear Unit) is a widely used choice for hidden layers. This activation function applies max(0,x) function to the neuron values. When used instead of logistic sigmoid or hyperbolic tangent activations, some adjustments to the Neural Network settings are typically required to achieve a good performance, such as: significantly decreasing the learning rate, increasing the number of learning epochs and network parameters.

Output Layer

As in the hidden layer output calculation (explained in the above paragraph), the output layer is also computed using the same transfer function as described for *Activation: Hidden Layer*.

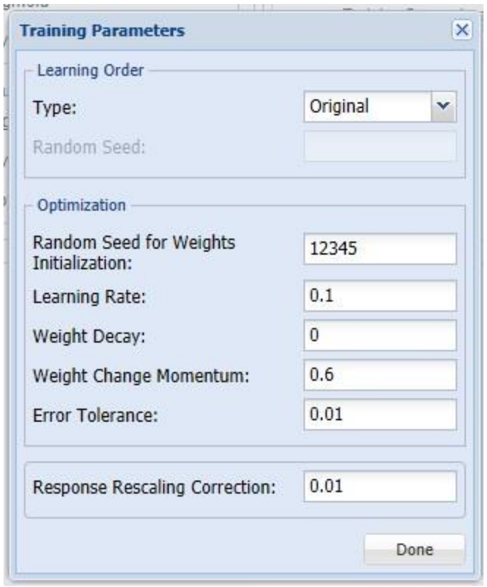
Select *Sigmoid* (the default setting) to use a logistic function for the transfer function with a range of 0 and 1.

Select *Hyperbolic Tangent* to use the tanh function for the transfer function, the range being -1 to 1.

Linear activation takes the form of const*x to create an output signal proportional to the neuron input. It is applicable and most commonly used for the output layer of regression problems to handle the continuous response that is unbounded in nature.

Training Parameters

Click Training Parameters to open the Training Parameters dialog to specify parameters related to the training of the Neural Network algorithm.



Training Parameters

Learning Order

Type: Original

Random Seed:

Optimization

Random Seed for Weights Initialization: 12345

Learning Rate: 0.1

Weight Decay: 0

Weight Change Momentum: 0.6

Error Tolerance: 0.01

Response Rescaling Correction: 0.01

Learning Order [Original or Random]

This option specifies the order in which the records in the training dataset are being processed. It is recommended to shuffle the training data to avoid the possibility of processing correlated reocrds in order. It also helps the neural network algorithm to converge faster. If Random is selected, Random Seed is enabled.

Learning Order [Random Seed]

This option specifies the seed for shuffling the training records. Note that different random shuffling may lead to different results, but as long as the training data is shuffled, different ordering typically does not result in drastic changes in performance.

Random Seed for Weights Initialization

If an integer value appears for *Random Seed for Weights Initialization*, Analytic Solver Data Science will use this value to set the seed for the initial assignment of the neuron values. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the neuron values are calculated. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box.

Learning Rate

This is the multiplying factor for the error correction during backpropagation; it is roughly equivalent to the learning rate for the neural network. A low value produces slow but steady learning, a high value produces rapid but erratic learning. Values for the step size typically range from 0.1 to 0.9.

Weight Decay

To prevent over-fitting of the network on the training data, set a weight decay to penalize the weight in each iteration. Each calculated weight will be multiplied by (1-decay).

Weight Change Momentum

In each new round of error correction, some memory of the prior correction is retained so that an outlier that crops up does not spoil accumulated learning.

Error Tolerance

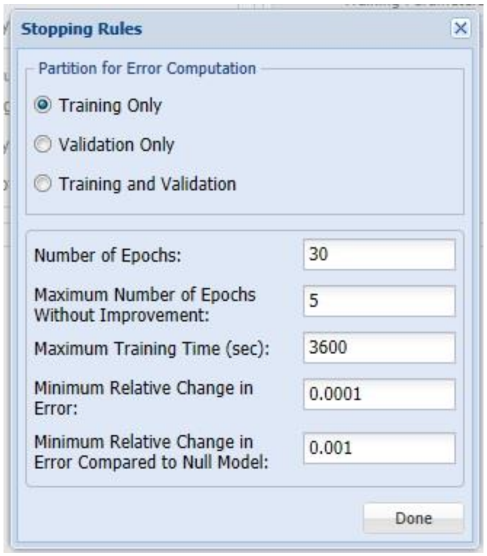
The error in a particular iteration is backpropagated only if it is greater than the error tolerance. Typically error tolerance is a small value in the range from 0 to 1.

Response Rescaling Correction

This option specifies a small number, which is applied to the Normalization rescaling formula, if the output layer activation is Sigmoid (or Softmax in Classification), and Adjusted Normalization, if the output layer activation is Hyperbolic Tangent. The rescaling correction ensures that all response values stay within the range of activation function.

Stopping Rules

Click Stopping Rules to open the Stopping Rules dialog. Here users can specify a comprehensive set of rules for stopping the algorithm early plus cross-validation on the training error.



Stopping Rules

Partition for Error Computation

● Training Only

○ Validation Only

○ Training and Validation

Number of Epochs:	30
Maximum Number of Epochs Without Improvement:	5
Maximum Training Time (sec):	3600
Minimum Relative Change in Error:	0.0001
Minimum Relative Change in Error Compared to Null Model:	0.001

Partition for Error Computation

Specifies which data partition is used to estimate the error after each training epoch.

Number of Epochs

An epoch is one sweep through all records in the training set. Use this option to set the number of epochs to be performed by the algorithm.

Maximum Number of Epochs Without Improvement

The algorithm will stop after this number of epochs has been completed, and no improvement has ben realized.

Maximum Training Time

The algorithm will stop once this time (in seconds) has been exceeded.

Keep Minimum Relative Change in Error

If the relative change in error is less than this value, the algorithm will stop.

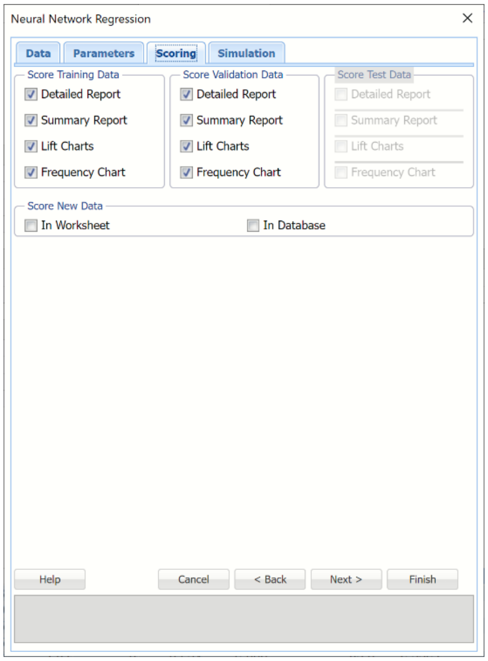
Keep Minimum Relative Change in Error Compared to Null Model

If the relative change in error compared to the Null Model is less than this value, the algorithm will stop. Null Model is the baseline model used for comparing the performance of the neural network model.

Regression Tree Dialog, Scoring tab

See below for options appearing on the Neural Network Regression – Scoring tab. Note: This tab does not exist on the *Neural Network Automatic Regression dialog.*

Neural Network Regression Dialog, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

Score New Data
□ In Worksheet
□ In Database

Score Training Data

Select these options to show an assessment of the performance of the Neural Network in predicting the value of the output variable in the training partition.

When Frequency Chart is selected, a frequency chart will be displayed when the NNP_TrainingScore worksheet are selected. This chart will display an interactive application similar to the Analyze Data feature, and explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Neural Network in predicting the value of the output variable in the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the NNP_ValidationScore worksheet is selected.

Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Neural Network in predicting the value of the output variable in the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the NNP_TestScore worksheet is selected.

Score New Data

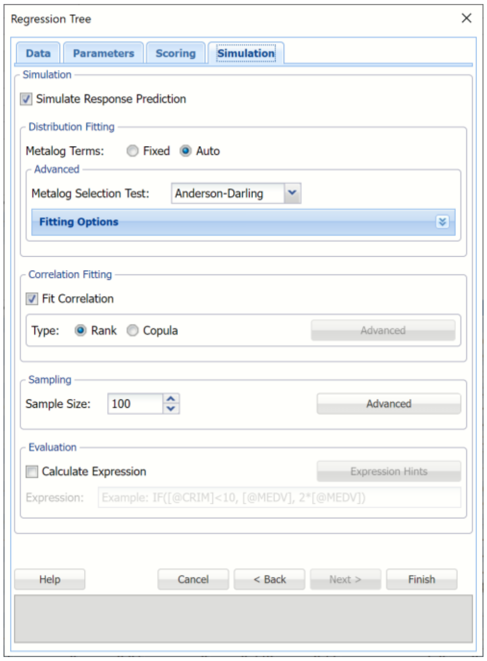
See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Regression Tree Dialog,Simulation tab

Simulation Tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, NNP_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RT_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Fitting Options

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula [Advanced]

Sampling

Sample Size: 100 [Advanced]

Evaluation

□ Calculate Expression [Expression Hints]

Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

Ensemble Methods

Analytic Solver Data Science offers three powerful ensemble methods for use with Regression: bagging (bootstrap aggregating), boosting, and random trees. Analytic Solver Data Science Regression Algorithms on their own can be used to find one model that results in good predictions for the new data. We can view the statistics and confusion matrices of the current predictor to see if our model is a good fit to the data, but how would we know if there is a better predictor just waiting to be found? The answer is that we do not know if a better predictor exists. However, ensemble methods allow us to combine multiple "weak" regression models which, when taken together form a new, more accurate "strong" regression model. These methods work by creating multiple diverse regression models, by taking different samples of the original dataset, and then combining their outputs. (Outputs may be combined by several techniques for example, majority vote for classification and averaging for regression. This combination of models effectively reduces the variance in the "strong" model. The three different types of ensemble methods offered in Analytic Solver Data Science (bagging, boosting, and random trees) differ on three items: 1.The selection of training data for each predictor or "weak" model, 2.How the "weak" models are generated and 3. How the outputs are combined. In all three methods, each "weak" model is trained on the entire training dataset to become proficient in some portion of the dataset.

Bagging, or bootstrap aggregating, was one of the first ensemble algorithms ever to be written. It is a simple algorithm, yet very effective. Bagging generates several training data sets by using random sampling with replacement (bootstrap sampling), applies the regression model to each dataset, then takes the average amongst the models to calculate the predictions for the new data. The biggest advantage of bagging is the relative ease that the algorithm can be parallelized which makes it a better selection for very large datasets.

Boosting, in comparison, builds a "strong" model by successively training models to concentrate on records receiving inaccurate predicted values in previous models. Once completed, all predictors are combined by a weighted majority vote. Analytic Solver Data Science offers three different variations of boosting as implemented by the AdaBoost algorithm (one of the most popular ensemble algorithms in use today): M1 (Freund), M1 (Breiman), and SAMME (Stagewise Additive Modeling using a Multi-class Exponential).

Adaboost.M1 first assigns a weight ($w_b(i)$) to each record or observation. This weight is originally set to 1/n and will be updated on each iteration of the algorithm. An original regression model is created using this first training set (T_b) and an error is calculated as:

$$e_b = \sum_{i-1}^{n} w_b(i) I(C_b(x_i) \neq y_i))$$

where the I() function returns 1 if true and 0 if not.

The error of the regression model in the b^th^ iteration is used to calculate the constant α_b. This constant is used to update the weight (w_b(i). In AdaBoost.M1 (Freund), the constant is calculated as:

$$\alpha_b = \ln((1-e_b)/e_b)$$

In AdaBoost.M1 (Breiman), the constant is calculated as:

$$\alpha_b = 1/2\ln((1-e_b)/e_b)$$

In SAMME, the constant is calculated as:

$$\alpha_b = 1/2\ln((1-e_b)/e_b + \ln(k-1) \text{ where k is the number of classes}$$

(When the number of categories is equal to 2, SAMME behaves the same as AdaBoost Breiman.)

In any of the three implementations (Freund, Breiman, or SAMME), the new weight for the (b + 1)th iteration will be

$$w_{b+1}(i) = w_b(i)\exp(\alpha_b I(C_b(x_i) \neq y_i))$$

Afterwards, the weights are all readjusted to sum to 1. As a result, the weights assigned to the observations that were assigned *inaccurate* predicted values are increased and the weights assigned to the observations that were assigned *accurate* predicted values are decreased. This adjustment forces the next regression model to put more emphasis on the records that were assigned inaccurate predictions. (This α constant is also used in the final calculation which will give the regression model with the lowest error more influence.) This process repeats until b = Number of weak learners (controlled by the User). The algorithm then computes the weighted average among all weak learners and assigns that value to the record. Boosting generally yields better models than bagging, however, it does have a disadvantage as it is not parallelizable. As a result, if the number of weak learners is large, boosting would not be suitable.

Random trees, also known as random forests, is a variation of bagging. This method works by training multiple “weak” regression trees using a fixed number of randomly selected features (sqrt[number of features] for classification and number of features/3 for prediction) then takes the average value for the weak learners and assigns that value to the “strong” predictor. (This ensemble method only accepts Regression Trees as a weak learner.) Typically, in this method the number of “weak” trees generated could range from several hundred to several thousand depending on the size and difficulty of the training set. Random Trees are parallelizable since they are a variant of bagging. However, since Random Trees selects a limited amount of features in each iteration, the performance of random trees is faster than bagging.

Ensemble Methods are very powerful methods and typically result in better performance than a single tree. This feature addition in Analytic Solver Data Science (introduced in V2015) provides users with more accurate prediction models and should be considered over the single tree method.

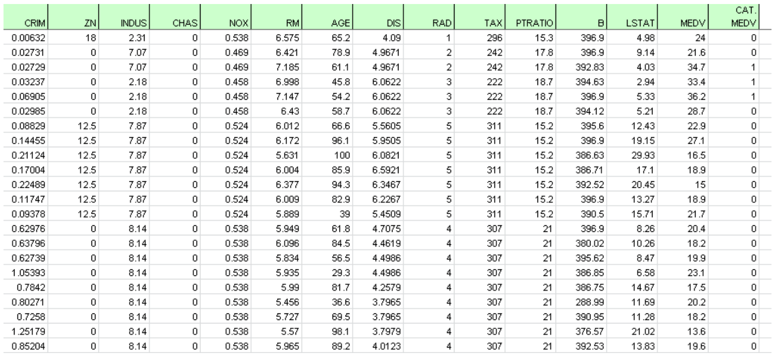
Boosting Regression Example

This example focuses on the boosting ensemble method using linear regression as the weak learner. We will use the *Boston_Housing.xlsx* example dataset. This dataset contains 14 variables, a description of each is given in the

Description tab in the example workbook. The dependent variable MEDV is the median value of a dwelling. This objective of this example is to predict the value of this variable.

Input

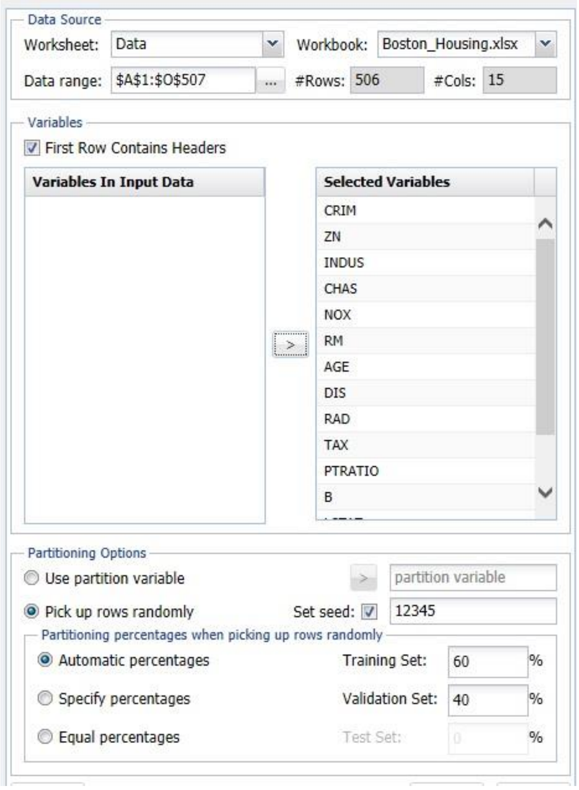
1. Open the example dataset by clicking **Help – Example Models – Forecasting / Data Science Examples – Boston Housing**. A portion of the dataset is shown below. Neither the CHAS variable nor the CAT. MEDV variable will be utilized in this example.



CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV	CAT. MEDV
0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24	0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6	0
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7	1
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4	1
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2	1
0.02985	0	2.18	0	0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7	0
0.08829	12.5	7.87	0	0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9	0
0.14455	12.5	7.87	0	0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1	0
0.21124	12.5	7.87	0	0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5	0
0.17004	12.5	7.87	0	0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9	0
0.22489	12.5	7.87	0	0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15	0
0.11747	12.5	7.87	0	0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9	0
0.09378	12.5	7.87	0	0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7	0
0.62976	0	8.14	0	0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4	0
0.63796	0	8.14	0	0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2	0
0.62739	0	8.14	0	0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9	0
1.05393	0	8.14	0	0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1	0
0.7842	0	8.14	0	0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5	0
0.80271	0	8.14	0	0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2	0
0.7258	0	8.14	0	0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2	0
1.25179	0	8.14	0	0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6	0
0.85204	0	8.14	0	0.538	5.965	89.2	4.0123	4	307	21	392.53	13.83	19.6	0

2. First, we partition the data into training and validation sets using the Standard Data Partition defaults with percentages of 60% of the data randomly allocated to the Training Set and 40% of the data randomly allocated to the Validation Set. For more information on partitioning a dataset, see the *Data Science Partitioning* chapter.

Standard Data Partitioning dialog



Data Source

Worksheet: Data Workbook: Boston_Housing.xlsx

Data range: $A$1:$O$507 #Rows: 506 #Cols: 15

Variables

☑ First Row Contains Headers

Variables In Input Data

Selected Variables: CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B

Partitioning Options

○ Use partition variable

● Pick up rows randomly Set seed: ☑ 12345

Partitioning percentages when picking up rows randomly

● Automatic percentages Training Set: 60 %

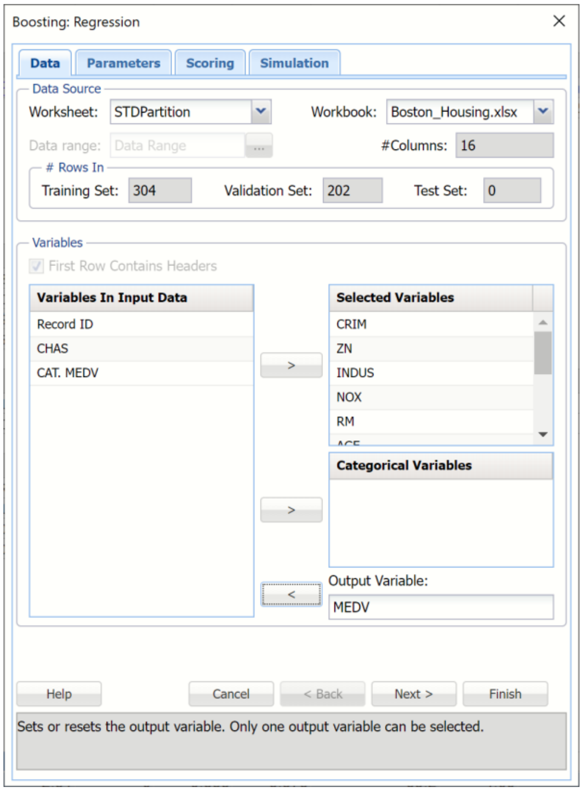
○ Specify percentages Validation Set: 40 %

○ Equal percentages Test Set: 0 %

3. Click **Predict – Ensemble** – **Boosting** on the Data Science ribbon. The *Boosting – Data* tab appears. Confirm that STDPartition is selected for Worksheet under Data Source.

4. Select **MEDV** as the *Output variable* and the **remaining variables** as *Selected Variables* (except the CAT.MEDV, CHAS and Record ID variables).

Boosting Regression dialog, Data tab



Boosting: Regression

Data Source

Worksheet: STDPartition

Workbook: Boston_Housing.xlsx

Data range: Data Range

#Columns: 16

Rows In

Training Set: 304

Validation Set: 202

Test Set: 0

Variables

☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM

Categorical Variables

Output Variable: MEDV

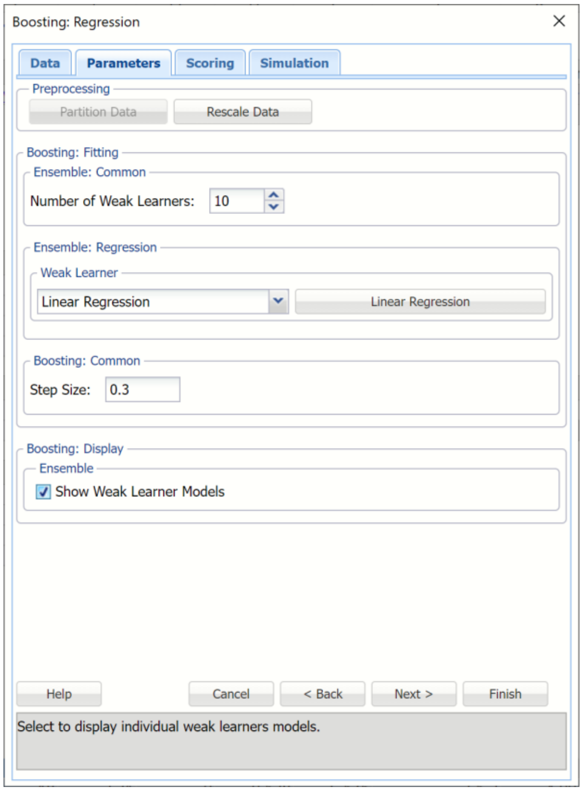
Sets or resets the output variable. Only one output variable can be selected.

5. Click **Next** to advance to the next tab.

6. Select the down arrow beneath Weak Learner and selct Linear Regression from the menu. A command button will appear to the right of the Weak Learner menu labeled Linear Regression. Click here to change any options related to this weak leaner. For more information on any of these options, see the Linear Regression chapter the appears earlier in this Guide.

7. Select Show Weak Learner Models to include this information in the output.

Boosting Regression dialog, Parameters tab



Number of Weak Learners: 10

Weak Learner: Linear Regression

Step Size: 0.3

☑ Show Weak Learner Models

Select to display individual weak learners models.

8. Click **Next** to advance to the *Boosting – Scoring* tab.
9. Select all four options for **Score Training/Validation data**.

When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the Regression Trees output.

When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the Regression Trees output.

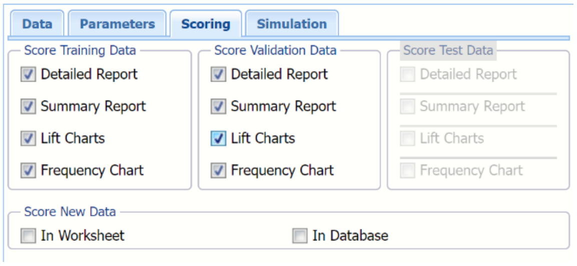
When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and ROC Curve plots in the output.

When Frequency Chart is selected, a frequency chart will be displayed when the RBoosting_TrainingScore and RBoosting_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user’s preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Since we did not create a test partition, the options for Score test data are disabled. See the chapter “Data Science Partitioning” for information on how to create a test partition.

See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.

Boosting Regression dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

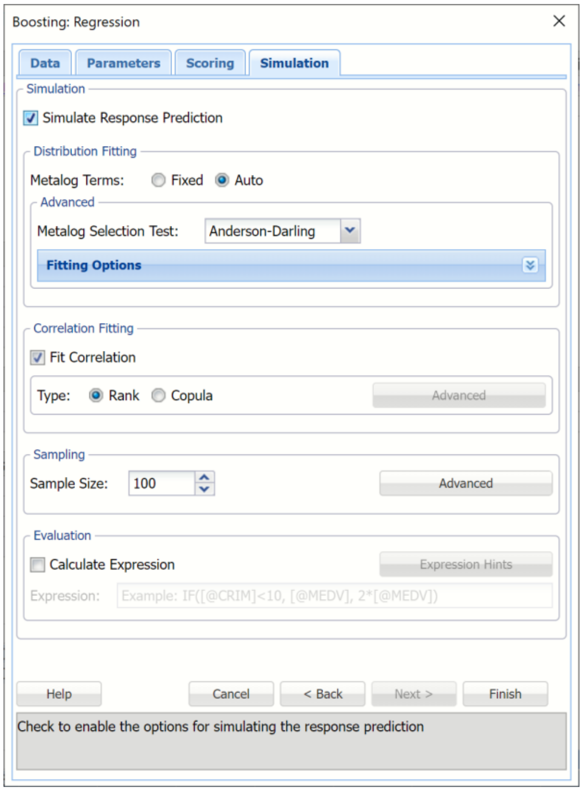
Score New Data
□ In Worksheet □ In Database

10. Click **Next** to advance to the Simulation tab.

11. Select **Simulation Response Prediction** to enable all options on the Simulation tab of the Regression Tree dialog. This tab is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

    Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, RBoosting_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

    Boosting Regression dialog, Simulation tab



Boosting: Regression

Data | Parameters | Scoring | Simulation

Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula Advanced

Sampling
Sample Size: 100 Advanced

Evaluation
□ Calculate Expression Expression Hints
Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

Help Cancel < Back Next > Finish

Check to enable the options for simulating the response prediction

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the Boosting_Simulation output tab.

Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression. Note that variable names are case sensitive. See any of the prediction methods to see the Expressison field in use.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

12. Click **Finish** to run Regression Tree on the example dataset.

Output

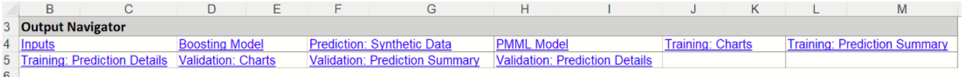
Output sheets containing the results of the Boosting Prediction method will be inserted into the active workbook, to the right of the STDPartition worksheet.

RBoosting_Output

This result worksheet includes 3 segments: Output Navigator, Inputs and Boosting Model.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

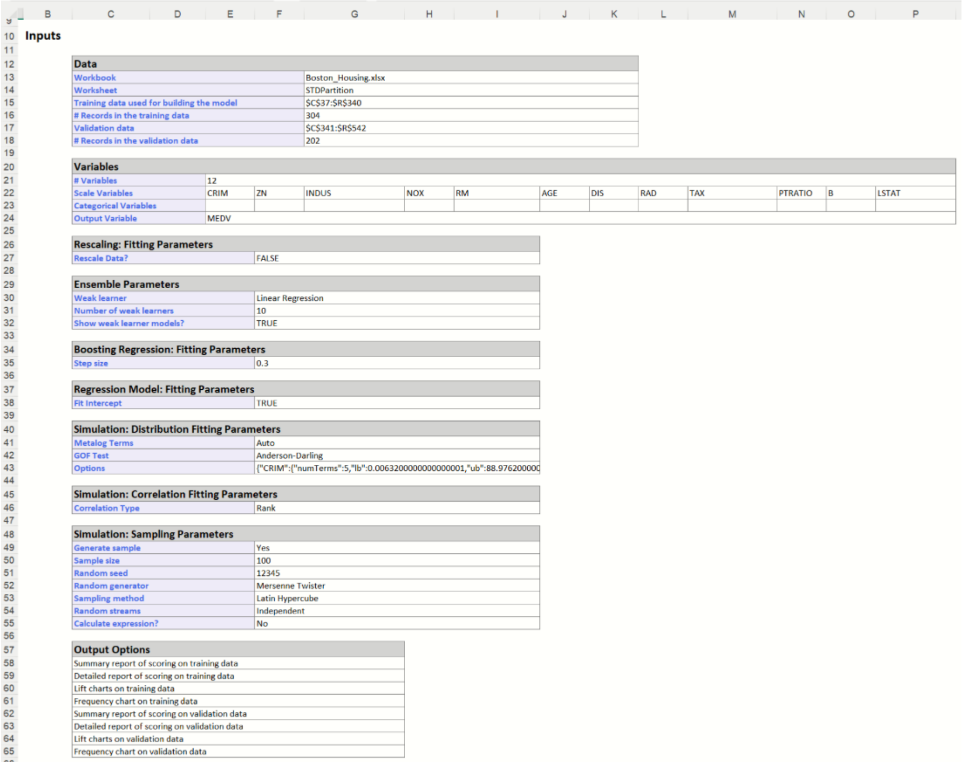
RBoosting_Output: Output Navigator



Output Navigator					
Inputs	Boosting Model	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Prediction Summary
Training: Prediction Details	Validation: Charts	Validation: Prediction Summary	Validation: Prediction Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Boosting Regression dialog.

RBoosting_Output, Inputs Report



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Ensemble Parameters	
Weak learner	Linear Regression
Number of weak learners	10
Show weak learner models?	TRUE

Boosting Regression: Fitting Parameters	
Step size	0.3

Regression Model: Fitting Parameters	
Fit Intercept	TRUE

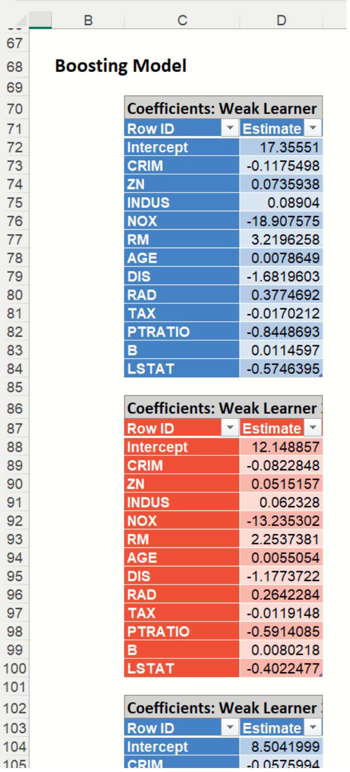
Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.0063200000000000001,"ub":88.976200000

Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Boosting Model:** Click the Boosting Model link on the Output Naviagator to view the Boosting model for each weak learner. Recall that the default is "10" on the Parameters tab.



Boosting Model

Coefficients: Weak Learner	
Row ID	Estimate
Intercept	17.35551
CRIM	-0.1175498
ZN	0.0735938
INDUS	0.08904
NOX	-18.907575
RM	3.2196258
AGE	0.0078649
DIS	-1.6819603
RAD	0.3774692
TAX	-0.0170212
PTRATIO	-0.8448693
B	0.0114597
LSTAT	-0.5746395

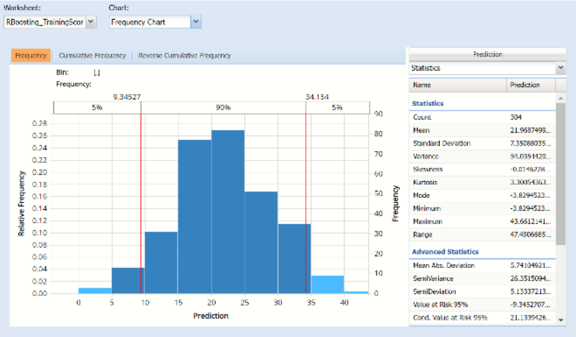
Coefficients: Weak Learner	
Row ID	Estimate
Intercept	12.148857
CRIM	-0.0822848
ZN	0.0515157
INDUS	0.062328
NOX	-13.235302
RM	2.2537381
AGE	0.0055054
DIS	-1.1773722
RAD	0.2642284
TAX	-0.0119148
PTRATIO	-0.5914085
B	0.0080218
LSTAT	-0.4022477

Coefficients: Weak Learner	
Row ID	Estimate
Intercept	8.5041999
CRIM	-0.0575994

RBoosting_TrainingScore

Click the *RBoosting_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Prediction Summary and the Training: Prediction Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

> Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



Worksheet: RBoosting_TrainingScor
Chart: Frequency Chart

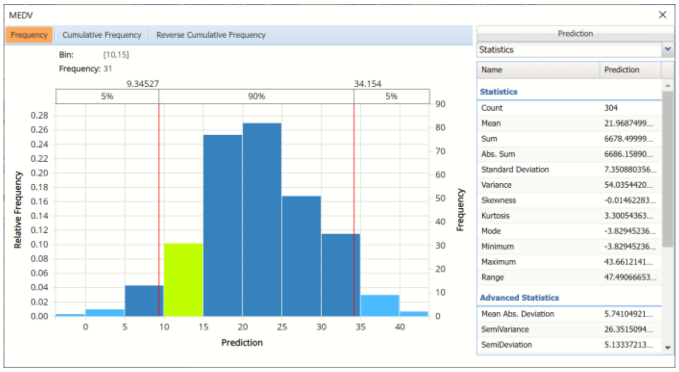
Prediction

Statistics

Name	Prediction
Statistics	
Count	304
Mean	21.9687499...
Standard Deviation	7.35088035...
Variance	54.0354420...
Skewness	-0.0146228...
Kurtosis	3.30054363...
Mode	-3.8294523...
Minimum	-3.8294523...
Maximum	43.6612141...
Range	47.4906665...
Advanced Statistics	
Mean Abs. Deviation	5.74104921...
SemiVariance	26.3515094...
SemiDeviation	5.13337213...
Value at Risk 95%	-9.3452707...
Cond. Value at Risk 95%	21.1339426...

- **Frequency Charts:** The output variable frequency chart opens automatically once the *RBoosting_TrainingScore* worksheet is selected. To close this chart, click the "x" in the upper right hand corner of the chart. To reopen, click onto another tab and then click back to the *RBoosting_TrainingScore* tab.

  Frequency chart displaying prediction data



Bin: [10,15]
Frequency: 31

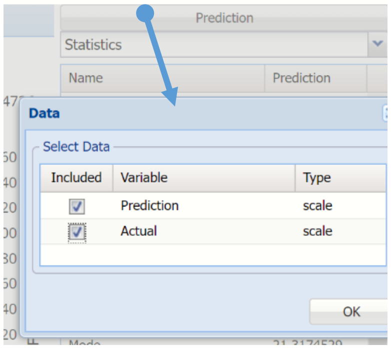
Prediction

Statistics

Name	Prediction
Statistics	
Count	304
Mean	21.9687499...
Sum	6678.49999...
Abs. Sum	6686.15890...
Standard Deviation	7.350880356...
Variance	54.0354420...
Skewness	-0.01462283...
Kurtosis	3.30054363...
Mode	-3.82945236...
Minimum	-3.82945236...
Maximum	43.6612141...
Range	47.49066653...
Advanced Statistics	
Mean Abs. Deviation	5.74104921...
SemiVariance	26.3515094...
SemiDeviation	5.13337213...

To add the Actual data to the chart, click Prediction in the upper right hand corner and select both checkboxes in the Data dialog.

Click Prediction to add Actual data to the interactive chart.



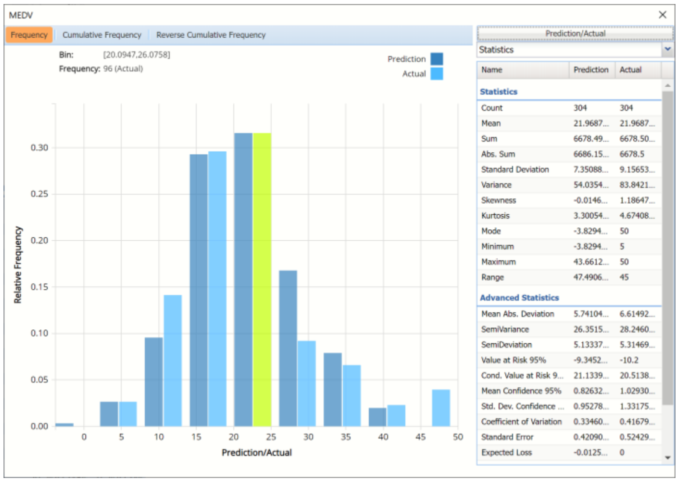
Data

Select Data

Included	Variable	Type
☑	Prediction	scale
☑	Actual	scale

Notice in the screenshot below that both the Prediction and Actual data appear in the chart together, and statistics for both appear on the right.

MEDV Frequency Chart



Bin: [20.0947,26.0758]
Frequency: 96 (Actual)

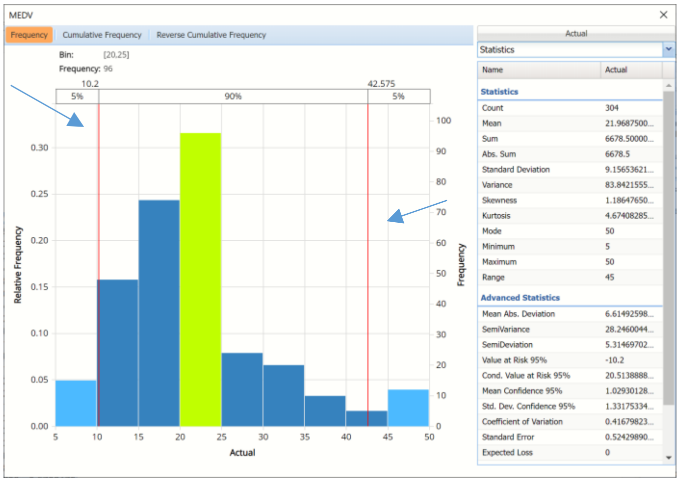
Name	Prediction	Actual
Statistics		
Count	304	304
Mean	21.9687...	21.9687...
Sum	6678.49...	6678.50...
Abs. Sum	6686.15...	6678.5
Standard Deviation	7.35088...	9.15653...
Variance	54.0354...	83.8421...
Skewness	-0.0146...	1.18647...
Kurtosis	3.30054...	4.67408...
Mode	-3.8294...	50
Minimum	-3.8294...	5
Maximum	43.6612...	50
Range	47.4906...	45
Advanced Statistics		
Mean Abs. Deviation	5.74104...	6.61492...
SemiVariance	26.3515...	28.2460...
SemiDeviation	5.13337...	5.31469...
Value at Risk 95%	-9.3452...	-10.2
Cond. Value at Risk 9...	21.1339...	20.5138...
Mean Confidence 95%	0.82632...	1.02930...
Std. Dev. Confidence ...	0.95278...	1.33175...
Coefficient of Variation	0.33460...	0.41679...
Standard Error	0.42090...	0.52429...
Expected Loss	-0.0125...	0

To remove either the Original or the Synthetic data from the chart, click Original/Synthetic in the top right and then uncheck the data type to be removed.

This chart behaves the same as the interactive chart in the Analyze Data feature found on the Explore menu.

- Use the mouse to hover over any of the bars in the graph to populate the Bin and Frequency headings at the top of the chart.
- When displaying either Prediction or Actual data (not both), red vertical lines will appear at the 5% and 95% percentile values in all three charts (Frequency, Cumulative Frequency and Reverse Cumulative Frequency) effectively displaying the 90th confidence interval. The middle percentage is the percentage of all the variable values that lie within the 'included' area, i.e. the darker shaded area. The two percentages on each end are the percentage of all variable values that lie outside of the 'included' area or the "tails". i.e. the lighter shaded area. Percentile values can be altered by moving either red vertical line to the left or right.

Frequency chart with percentage markers moved

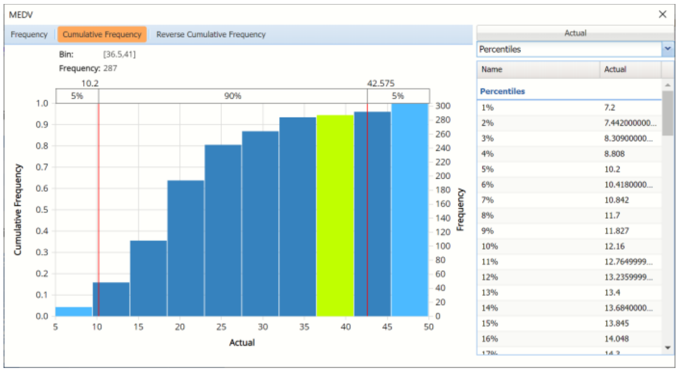


Bin: [20,25]
Frequency: 96

Name	Actual
Statistics	
Count	304
Mean	21.9687500...
Sum	6678.50000...
Abs. Sum	6678.5
Standard Deviation	9.15653621...
Variance	83.8421555...
Skewness	1.18647650...
Kurtosis	4.67408285...
Mode	50
Minimum	5
Maximum	50
Range	45
Advanced Statistics	
Mean Abs. Deviation	6.61492598...
SemiVariance	28.2460044...
SemiDeviation	5.31469702...
Value at Risk 95%	-10.2
Cond. Value at Risk 95%	20.5138888...
Mean Confidence 95%	1.02930128...
Std. Dev. Confidence 95%	1.33175334...
Coefficient of Variation	0.41679823...
Standard Error	0.52429890...
Expected Loss	0

- Click Cumulative Frequency and Reverse Cumulative Frequency tabs to see the Cumulative Frequency and Reverse Cumulative Frequency charts, respectively.

  Cumulative Frequency chart and Percentiles displayed



Bin: [36.5,41]

Frequency: 287

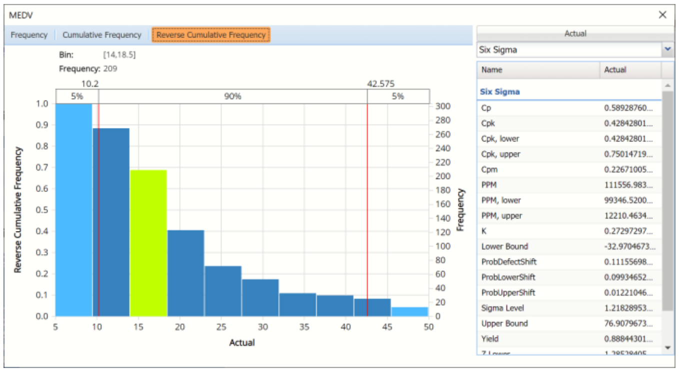
Actual

Percentiles

Name	Actual
Percentiles	
1%	7.2
2%	7.442000000...
3%	8.30900000...
4%	8.808
5%	10.2
6%	10.4180000...
7%	10.842
8%	11.7
9%	11.827
10%	12.16
11%	12.7649999...
12%	13.2359999...
13%	13.4
14%	13.6840000...
15%	13.845
16%	14.048

- Click the down arrow next to Statistics to view Percentiles for each type of data along with Six Sigma indices.

  Reverse Cumulative Frequency chart and Six Sigma indices displayed.



Bin: [14,18.5]

Frequency: 209

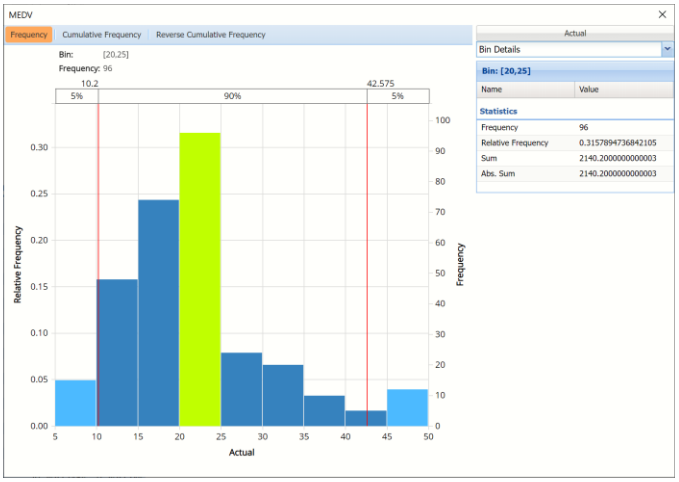
Actual

Six Sigma

Name	Actual
Six Sigma	
Cp	0.58928760...
Cpk	0.42842801...
Cpk, lower	0.42842801...
Cpk, upper	0.75014719...
Cpm	0.22671005...
PPM	111556.983...
PPM, lower	99346.5200...
PPM, upper	12210.4634...
K	0.27297297...
Lower Bound	-32.9704673...
ProbDefectShift	0.11155698...
ProbLowerShift	0.09934652...
ProbUpperShift	0.01221046...
Sigma Level	1.21828953...
Upper Bound	76.9079673...
Yield	0.88844301...

- Click the down arrow next to Statistics to view Bin Details to display information related to each bin in the chart.

  Bin Details view



Bin: [20,25]

Frequency: 96

Actual

Bin Details

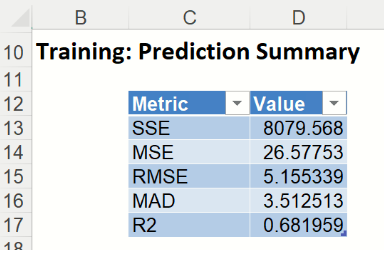
Bin: [20,25]

Name	Value
Statistics	
Frequency	96
Relative Frequency	0.3157894736842105
Sum	2140.2000000000003
Abs. Sum	2140.2000000000003

- Use the Chart Options view to manually select the number of bins to use in the chart, as well as to set personalization options.

As discussed above, see the Analyze Data section of the Exploring Data chapter for an in-depth discussion of this chart as well as descriptions of all statistics, percentiles, bin metrics and six sigma indices.

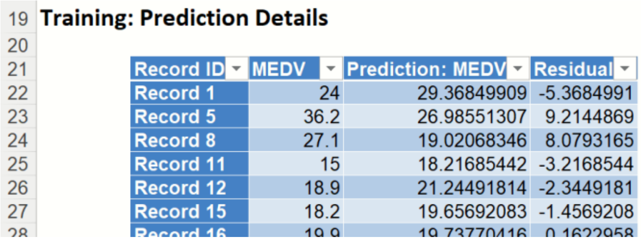
- **Training: Prediction Summary:** Click the Training: Prediction Summary link on the Output Navigator to open the Training Summary. This data table displays various statistics to measure the performance of the trained network: Sum of Squared Error (SSE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), the Median Absolute Deviation (MAD) and the Coefficient of Determination (R^2).



Training: Prediction Summary

Metric	Value
SSE	8079.568
MSE	26.57753
RMSE	5.155339
MAD	3.512513
R2	0.681959

- **Training: Prediction Details:** Scroll down to view the Prediction Details data table. This table displays the Actual versus Predicted values, along with the Residuals, for the training dataset.



Training: Prediction Details

Record ID	MEDV	Prediction: MEDV	Residual
Record 1	24	29.36849909	-5.3684991
Record 5	36.2	26.98551307	9.2144869
Record 8	27.1	19.02068346	8.0793165
Record 11	15	18.21685442	-3.2168544
Record 12	18.9	21.24491814	-2.3449181
Record 15	18.2	19.65692083	-1.4569208
Record 16	19.9	19.73770416	0.1622958

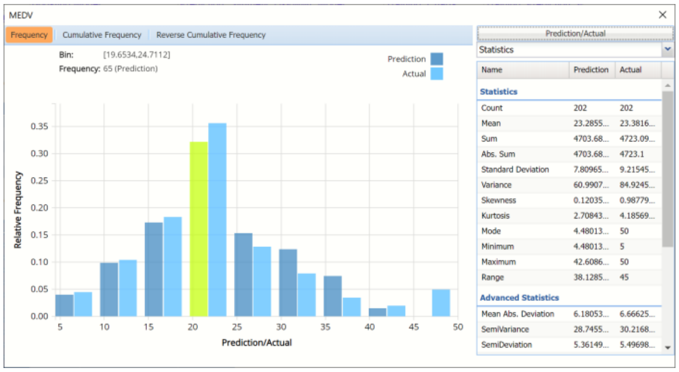
RBoosting_ValidationScore

Another key interest in a data-mining context will be the predicted and actual values for the MEDV variable along with the residual (difference) for each predicted value in the *Validation* partition.

RBoosting_ValidationScore displays the newly added Output Variable frequency chart, the Validation: Prediction Summary and the Validation: Prediction Details report. All calculations, charts and predictions on the RBoosting_ValidationScore output sheet apply to the Validation partition.

- **Frequency Charts:** The output variable frequency chart for the validation partition opens automatically once the *RBoosting_ValidationScore* worksheet is selected. This chart displays a detailed, interactive frequency chart for the Actual variable data and the Predicted data, for the validation partition. For more information on this chart, see the RBoosting_TrainingScore explanation above.

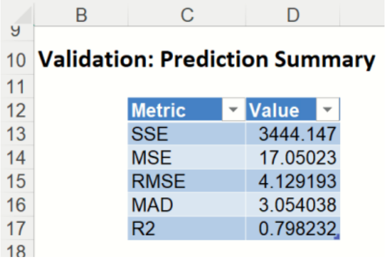
Validation Partition Frequency Chart



Name	Prediction	Actual
Statistics		
Count	202	202
Mean	23.2855...	23.3816...
Sum	4703.68...	4723.09...
Abs. Sum	4703.68...	4723.1
Standard Deviation	7.80965...	9.21545...
Variance	60.9907...	84.9245...
Skewness	0.12035...	0.98779...
Kurtosis	2.70843...	4.18569...
Mode	4.48013...	50
Minimum	4.48013...	5
Maximum	42.6086...	50
Range	38.1285...	45
Advanced Statistics		
Mean Abs. Deviation	6.18053...	6.66625...
SemiVariance	28.7455...	30.2168...
SemiDeviation	5.36149...	5.49698...

- **Prediction Summary:** In the Prediction Summary report, Analytic Solver Data Science displays the total sum of squared errors summaries for the Validation partition.

  Validation Prediction Summary

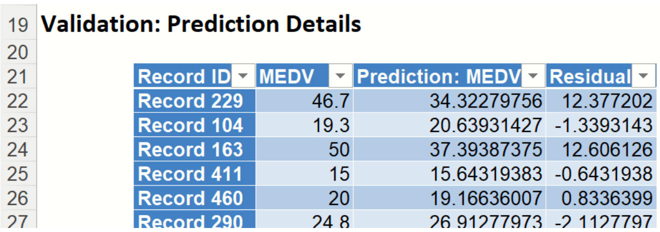

B
C
D
9
10
11
12
13
14
15
16
17
18

Validation: Prediction Summary

Metric	Value
SSE	3444.147
MSE	17.05023
RMSE	4.129193
MAD	3.054038
R2	0.798232

- **Prediction Details:** Scroll down to the Validation: Prediction Details report to find the Prediction value for the MEDV variable for each record in the Validation partition, as well as the Residual value.

  Validation Prediction Summary


19
20
21
22
23
24
25
26
27

Validation: Prediction Details

Record ID	MEDV	Prediction: MEDV	Residual
Record 229	46.7	34.32279756	12.377202
Record 104	19.3	20.63931427	-1.3393143
Record 163	50	37.39387375	12.606126
Record 411	15	15.64319383	-0.6431938
Record 460	20	19.16636007	0.8336399
Record 290	24.8	26.91277973	-2.1127797

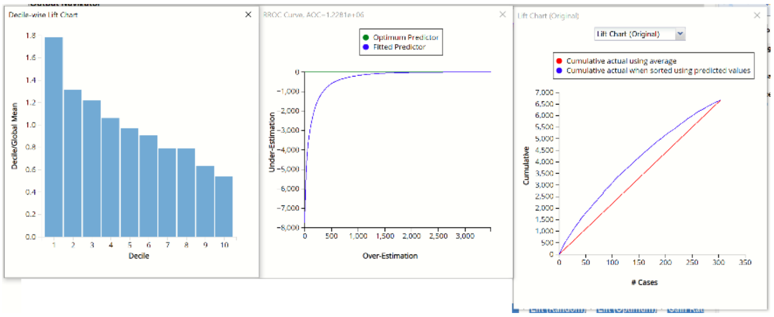
RROC charts, shown below, are better indicators of fit. Read on to view how these more sophisticated tools can tell us about the fit of the neural network to our data.

RBoosting_TrainingLiftChart & RBoosting_ValidationLiftChart

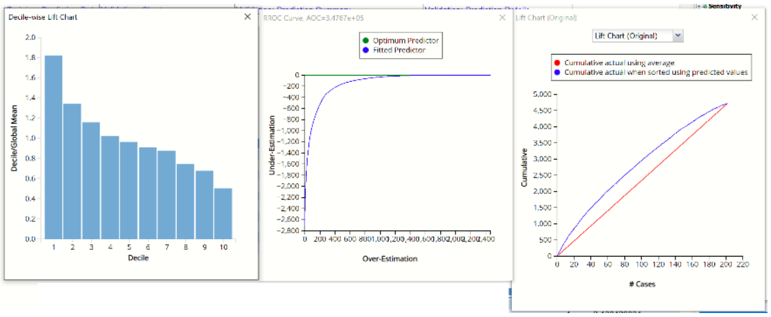
Click the **RBoosting_TrainLiftChart** and **RBoosting_ValidLiftChart** tabs to navigate to the Lift Charts and Regression RROC curves for both the training and validation datasets. For more information on how to interpret these charts, see the Mulitple Linear Regression chapter that appears in the Analytic Solver Data Science User Guide.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select RBoosting_TrainingLiftChart or RBoosting_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, RROC Curve and Lift Chart for Training Partition



Decile-wise Lift Chart, RROC Curve and Lift Chart for Valid. Partition

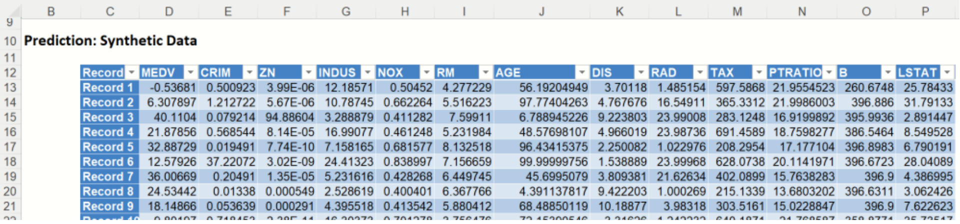


RBoosting_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, RBoosting_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Boosting Regression dialog.

This report contains the synthetic data, the predicted values for the training data (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



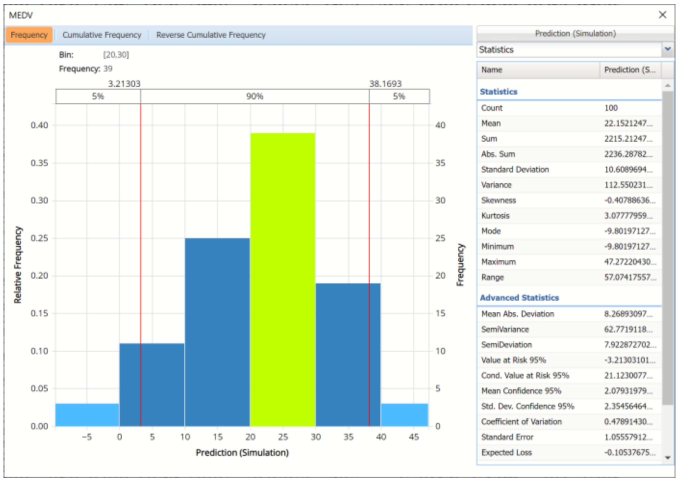
Prediction: Synthetic Data

Record	MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	-0.53681	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19204949	3.70118	1.485154	597.5868	21.9554523	260.6748	25.78433
Record 2	6.307897	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404263	4.767676	16.54911	365.3312	21.9986003	396.886	31.79133
Record 3	40.1104	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945226	9.223803	23.99008	283.1248	16.9199892	395.9936	2.891447
Record 4	21.87856	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698107	4.966019	23.98736	691.4589	18.7598277	386.5464	8.549528
Record 5	32.88729	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415375	2.250082	1.022976	208.2954	17.177104	396.8983	6.790191
Record 6	12.57926	37.22072	3.02E-09	24.41323	0.838997	7.156659	99.99999756	1.538889	23.99968	628.0738	20.1141971	396.6723	28.04089
Record 7	36.00669	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.6995079	3.809381	21.62634	402.0899	15.7638283	396.9	4.386995
Record 8	24.53442	0.01338	0.000549	2.528619	0.400401	6.367766	4.391137817	9.422203	1.000269	215.1339	13.6803202	396.6311	3.062426
Record 9	18.14866	0.053639	0.000291	4.395518	0.413542	5.880412	68.48850119	10.18877	3.98318	303.5161	15.0228847	396.9	7.622623

The data contained in the Synthetic Data report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training data, the synthetic data and the expression, if it exists. (Recall that no expression was entered in this example.)

Frequency Chart for Prediction (Simulation) data



Prediction (Simulation)

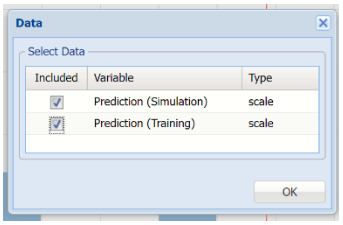
Statistics

Name	Prediction (S...
Statistics	
Count	100
Mean	22.1521247...
Sum	2215.21247...
Abs. Sum	2236.28782...
Standard Deviation	10.6089694...
Variance	112.550231...
Skewness	-0.40788636...
Kurtosis	3.07777959...
Mode	-9.80197127...
Minimum	-9.80197127...
Maximum	47.27220430...
Range	57.07417557...
Advanced Statistics	
Mean Abs. Deviation	8.26893097...
SemiVariance	62.7719118...
SemiDeviation	7.922872702...
Value at Risk 95%	-3.21303101...
Cond. Value at Risk 95%	21.1230077...
Mean Confidence 95%	2.07931979...
Std. Dev. Confidence 95%	2.35456464...
Coefficient of Variation	0.47891430...
Standard Error	1.05557912...
Expected Loss	-0.10537675...

Click *Prediction (Simulation)* to add the training data to the chart.

Click *Prediction(Simulation)* and *Prediction (Training)* to change the Data view.

Data Dialog



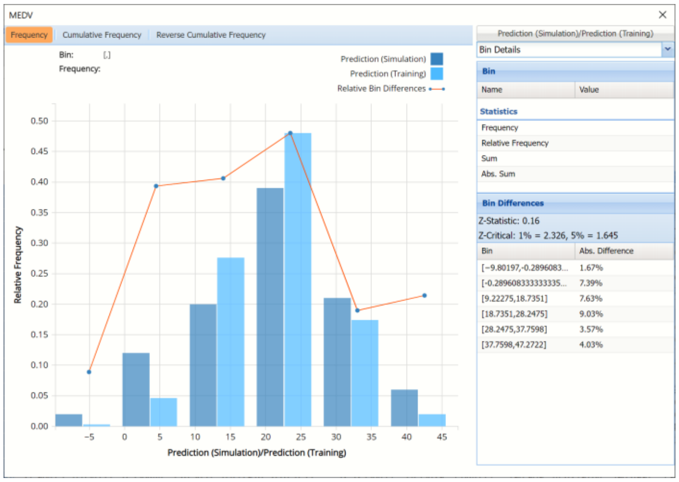
Data

Select Data

Included	Variable	Type
☑	Prediction (Simulation)	scale
☑	Prediction (Training)	scale

In the chart below, the dark blue bars display the frequencies for the synthetic data and the light blue bars display the frequencies for the predicted values in the Training partition.

Prediction (Simulation) and Prediction (Training) Frequency chart for MEDV variable



Prediction (Simulation)/Prediction (Training)

Bin Details

Bin

Name	Value
Statistics	
Frequency	
Relative Frequency	
Sum	
Abs. Sum	

Bin Differences

Z-Statistic: 0.16

Z-Critical: 1% = 2.326, 5% = 1.645

Bin	Abs. Difference
[−9.80197,-0.2896083...	1.67%
[-0.289608333333335...	7.39%
[9.22275,18.7351]	7.63%
[18.7351,28.2475]	9.03%
[28.2475,37.7598]	3.57%
[37.7598,47.2722]	4.03%

The Relative Bin Differences curve charts the absolute differences between the data in each bin. Click the down arrow next to Statistics to view the Bin Details pane to display the calculations.

Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

See the “Scoring New Data” chapter in the Analytic Solver Data Science User Guide for information on the Stored Model Sheet, *RBoosting_Stored*.

Continue on with the Bagging Neural Network Regression Example in the next section to compare the results between the two ensemble methods.

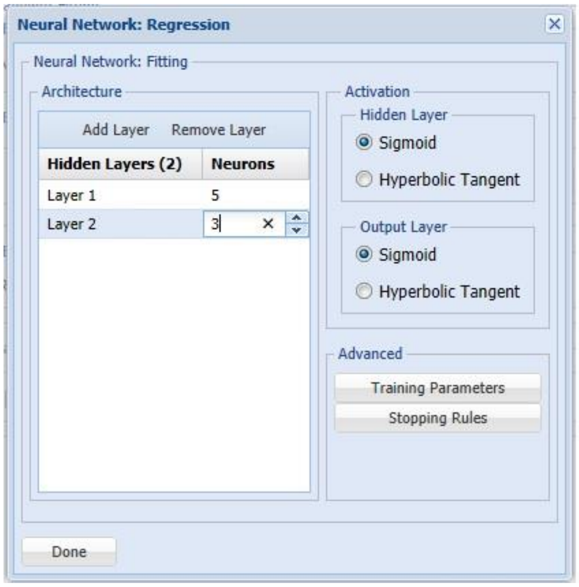
Bagging Regression Example

This example focuses on creating a Neural Network using the bagging ensemble method. See the section above to see an example on creating a Neural Network using the boosting ensemble method. This example reuses the standard data partition created in the Boosting example above.

Input

1. Click **Predict – Ensemble**– **Bagging** on the Data Science ribbon. The *Bagging – Data* tab appears.
2. As in the example above, select **MEDV** as the *Output variable* and the **remaining variables** as *Selected Variables* (except the CAT.MEDV, CHAS and Record ID variables). *(See screenshot of Boosting Regression dialog, data tab above.)*
3. Click **Next** to advance to the next tab.
4. Select the down arrow beneath Weak Learner and select **Neural Network** from the menu. A command button will appear to the right of the Weak Learner menu labeled Neural Network. Click this button and then **Add Layer** twice to add two layers with 5 and 3 neurons, respectively. For more information on any of these options, see the Neural Network chapter the appears earlier in this Guide. Click Done to return to the Parameters tab.

Bagging Weak Learner



Neural Network: Regression

Neural Network: Fitting

Architecture

Hidden Layers (2)	Neurons
Layer 1	5
Layer 2	3

Activation

Hidden Layer

● Sigmoid

○ Hyperbolic Tangent

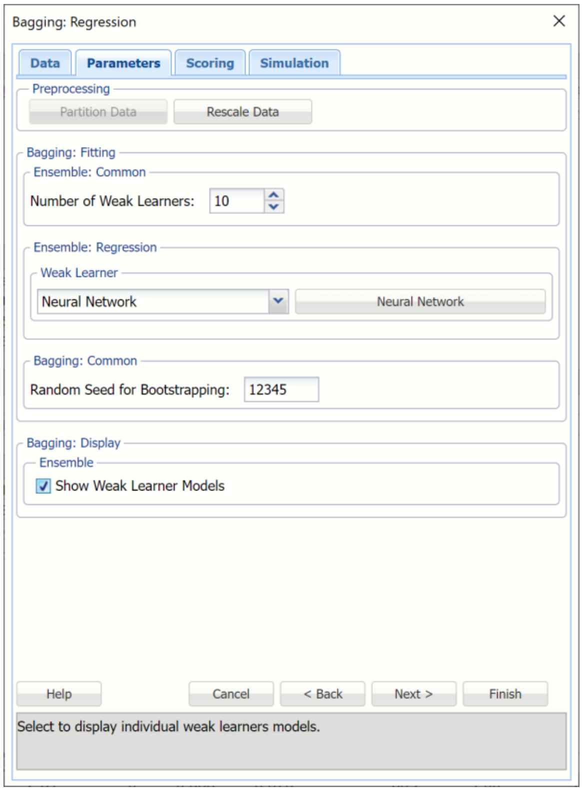
Output Layer

● Sigmoid

○ Hyperbolic Tangent

5. Select **Show Weak Learner** Models to include this information in the output.

Bagging Regression dialog, Parameters tab



Number of Weak Learners: 10

Neural Network

Random Seed for Bootstrapping: 12345

☑ Show Weak Learner Models

Select to display individual weak learners models.

6. **Next** to advance to the *Bagging – Scoring* tab.

7. Select all four options for **Score Training/Validation data**.

When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the Regression Trees output.

When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the Regression Trees output.

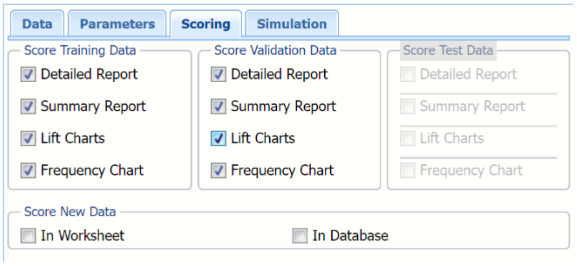
When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and ROC Curve plots in the output.

When Frequency Chart is selected, a frequency chart will be displayed when the RBagging_TrainingScore and RBagging_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Since we did not create a test partition, the options for Score test data are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.

Bagging Regression dialog, Scoring tab



Data | Parameters | Scoring | Simulation

Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
□ Detailed Report
□ Summary Report
□ Lift Charts
□ Frequency Chart

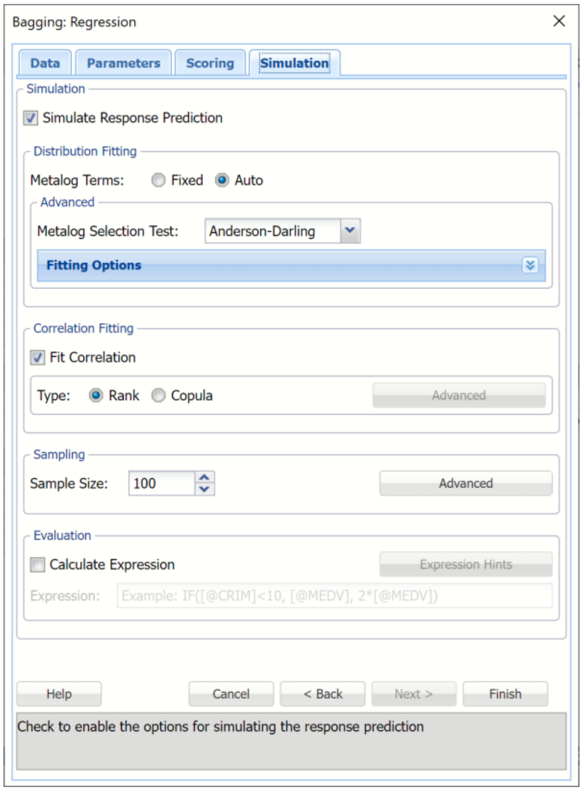
Score New Data
□ In Worksheet
□ In Database

8. Click **Next** to advance to the Simulation tab.

9. Select **Simulation Response Prediction** to enable all options on the Simulation tab of the Regression Tree dialog. This option is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

   Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, RBagging_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

   Bagging Regression dialog, Simulation tab



Bagging: Regression

Data | Parameters | Scoring | Simulation

Simulation
☑ Simulate Response Prediction

Distribution Fitting
Metalog Terms: ○ Fixed ● Auto
Advanced
Metalog Selection Test: Anderson-Darling
Fitting Options

Correlation Fitting
☑ Fit Correlation
Type: ● Rank ○ Copula
Advanced

Sampling
Sample Size: 100
Advanced

Evaluation
□ Calculate Expression
Expression Hints
Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

Help | Cancel | < Back | Next > | Finish

Check to enable the options for simulating the response prediction

   Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RBagging_Simulation output tab.

Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression. Note that variable names are case sensitive. See any of the prediction methods to see the Expressison field in use.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

10. Click **Finish** to run Bagging Ensemble Method on the example dataset.

Output

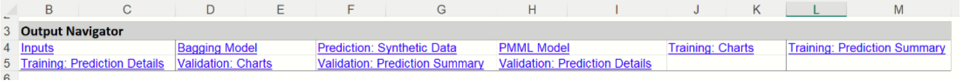
Output sheets containing the results of the Bagging Prediction method will be inserted into the active workbook, to the right of the STDPartition worksheet.

RBagging_Output

This result worksheet includes 3 segments: Output Navigator, Inputs and Bagging Model.

- **Output Navigator:** The Output Navigator appears at the top of all result worksheets. Use this feature to quickly navigate to all reports included in the output.

RBagging_Output: Output Navigator

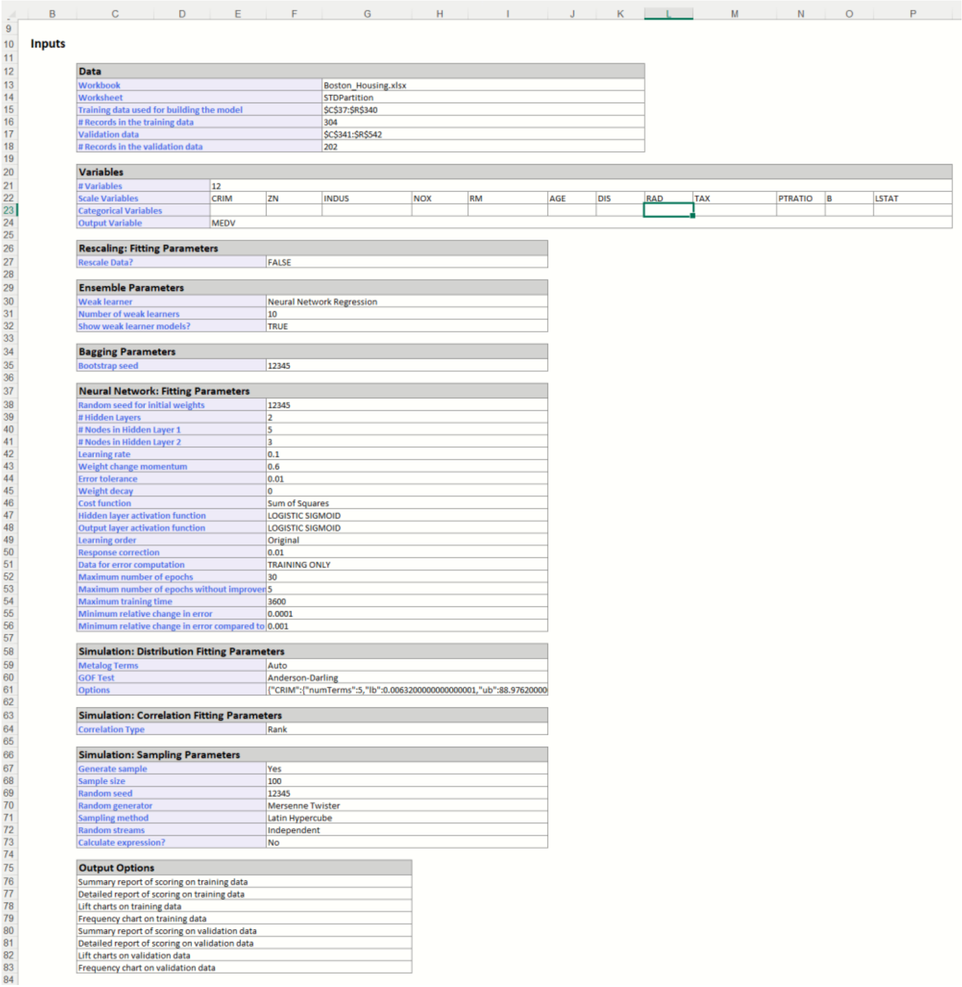


Output Navigator

Inputs	Bagging Model	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Prediction Summary
Training: Prediction Details	Validation: Charts	Validation: Prediction Summary	Validation: Prediction Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Bagging Regression dialog.

RBagging_Output, Inputs Report



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Ensemble Parameters	
Weak learner	Neural Network Regression
Number of weak learners	10
Show weak learner models?	TRUE

Bagging Parameters	
Bootstrap seed	12345

Neural Network: Fitting Parameters	
Random seed for initial weights	12345
# Hidden Layers	2
# Nodes in Hidden Layer 1	5
# Nodes in Hidden Layer 2	3
Learning rate	0.1
Weight change momentum	0.6
Error tolerance	0.01
Weight decay	0
Cost function	Sum of Squares
Hidden layer activation function	LOGISTIC SIGMOID
Output layer activation function	LOGISTIC SIGMOID
Learning order	Original
Response correction	0.01
Data for error computation	TRAINING ONLY
Maximum number of epochs	30
Maximum number of epochs without improver	5
Maximum training time	3600
Minimum relative change in error	0.0001
Minimum relative change in error compared to	0.001

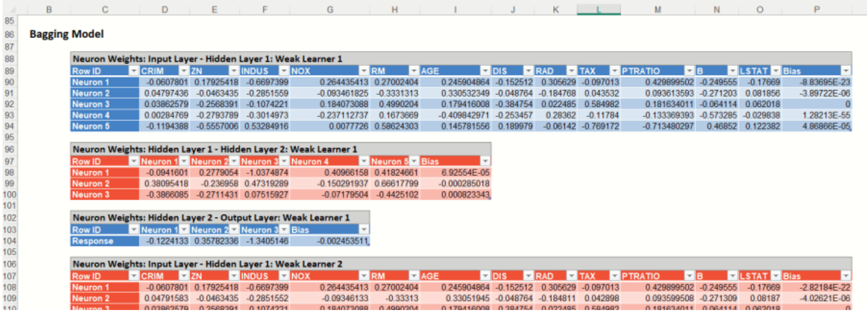
Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.0063200000000000001,"ub":88.97620000

Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

- **Boosting Model:** Click the Boosting Model link on the Output Naviagator to view the Boosting model for each weak learner. Recall that the default is "10" on the Parameters tab.



Bagging Model

Neuron Weights: Input Layer - Hidden Layer 1: Weak Learner 1

Row ID	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	Bias
Neuron 1	-0.0607801	0.17925418	-0.6697399	0.264435413	0.27002404	0.245904864	-0.152512	0.305629	-0.097013	0.429899502	-0.249555	-0.17669	-8.83695E-23
Neuron 2	0.04797436	-0.0463435	-0.2851559	-0.093461825	-0.3331313	0.330532349	-0.048764	-0.184768	0.043532	0.093613593	-0.271203	0.081856	-3.89722E-06
Neuron 3	0.03862579	-0.2568391	-0.1074221	0.184073088	0.4990204	0.179416008	-0.384754	0.022485	0.584982	0.181634011	-0.064114	0.062018	0
Neuron 4	0.00284769	-0.2793789	-0.3014973	-0.237112737	0.1673669	-0.409842971	-0.253457	0.28362	-0.11784	-0.133369393	-0.573285	-0.029838	1.28213E-55
Neuron 5	-0.1194388	-0.5557006	0.53284916	0.0077726	0.58624303	0.145781556	0.189979	-0.06142	-0.769172	-0.713480297	0.46852	0.122382	4.86866E-05

Neuron Weights: Hidden Layer 1 - Hidden Layer 2: Weak Learner 1

Row ID	Neuron 1	Neuron 2	Neuron 3	Neuron 4	Neuron 5	Bias
Neuron 1	-0.0941601	0.2779054	-1.0374874	0.40966158	0.41824661	6.92554E-05
Neuron 2	0.38095418	-0.236958	0.47319289	-0.150291937	0.66617799	-0.000285018
Neuron 3	-0.3866085	-0.2711431	0.07515927	-0.07179504	-0.4425102	0.000823343

Neuron Weights: Hidden Layer 2 - Output Layer: Weak Learner 1

Row ID	Neuron 1	Neuron 2	Neuron 3	Bias
Response	-0.1224133	0.35782336	-1.3405146	-0.002453511

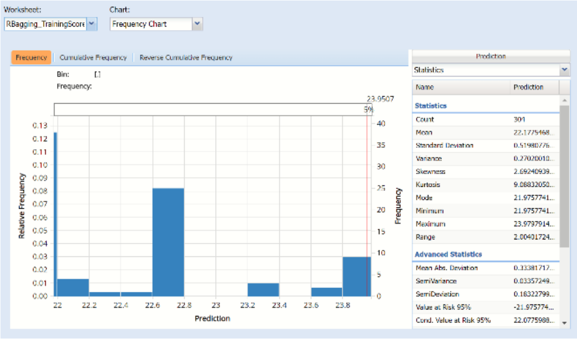
Neuron Weights: Input Layer - Hidden Layer 1: Weak Learner 2

Row ID	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	Bias
Neuron 1	-0.0607801	0.17925418	-0.6697399	0.264435413	0.27002404	0.245904864	-0.152512	0.305629	-0.097013	0.429899502	-0.249555	-0.17669	-2.82184E-22
Neuron 2	0.04791583	-0.0463435	-0.2851552	-0.09346133	-0.33313	0.33051945	-0.048764	-0.184811	0.042898	0.093599508	-0.271309	0.08187	-4.02621E-06
Neuron 3	0.03862579	-0.2568391	-0.1074221	0.184073088	0.4990204	0.179416008	-0.384754	0.022485	0.584982	0.181634011	-0.064114	0.062018	0

RBagging_TrainingScore

Click the *RBagging_TrainingScore* tab to view the newly added Output Variable frequency chart, the Training: Prediction Summary and the Training: Prediction Details report. All calculations, charts and predictions on this worksheet apply to the Training data.

Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.



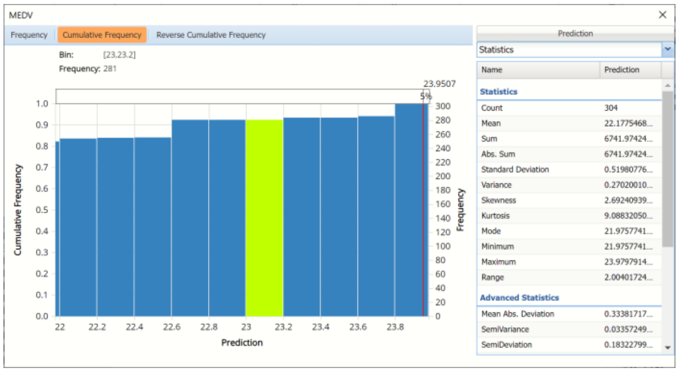
Worksheet: RBagging_TrainingScore

Chart: Frequency Chart

Name	Prediction
Statistics	
Count	301
Mean	22.1775468...
Standard Deviation	0.51980776...
Variance	0.27020010...
Skewness	2.69240939...
Kurtosis	9.08832050...
Mode	21.9757741...
Minimum	21.9757741...
Maximum	23.9797914...
Range	2.00401724...
Advanced Statistics	
Mean Abs. Deviation	0.33381717...
SemiVariance	0.03357249...
SemiDeviation	0.18322799...
Value at Risk 95%	-21.975774...
Cond. Value at Risk 95%	22.0775988...

- **Frequency Charts:** The output variable frequency chart opens automatically once the *RBoosting_TrainingScore* worksheet is selected. For more information on this dialog, see the Boosting example above.

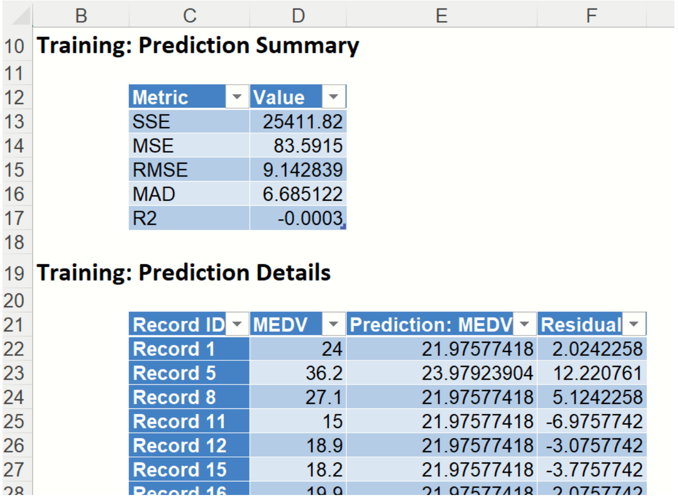
  Frequency chart displaying prediction data



Name	Prediction
Statistics	
Count	304
Mean	22.1775468...
Sum	6741.97424...
Abs. Sum	6741.97424...
Standard Deviation	0.51980776...
Variance	0.27020010...
Skewness	2.69240939...
Kurtosis	9.08832050...
Mode	21.9757741...
Minimum	21.9757741...
Maximum	23.9797914...
Range	2.00401724...
Advanced Statistics	
Mean Abs. Deviation	0.33381717...
SemiVariance	0.03357249...
SemiDeviation	0.18322799...

- **Training: Prediction Summary:** Click the Training: Prediction Summary link on the Output Navigator to open the Training Summary. This data table displays various statistics to measure the performance of the trained network: Sum of Squared Error (SSE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), the Median Absolute Deviation (MAD) and the Coefficient of Determination (R^2).

- **Training: Prediction Details:** Scroll down to view the Prediction Details data table. This table displays the Actual versus Predicted values, along with the Residuals, for the training dataset.



Training: Prediction Summary

Metric	Value
SSE	25411.82
MSE	83.5915
RMSE	9.142839
MAD	6.685122
R2	-0.0003

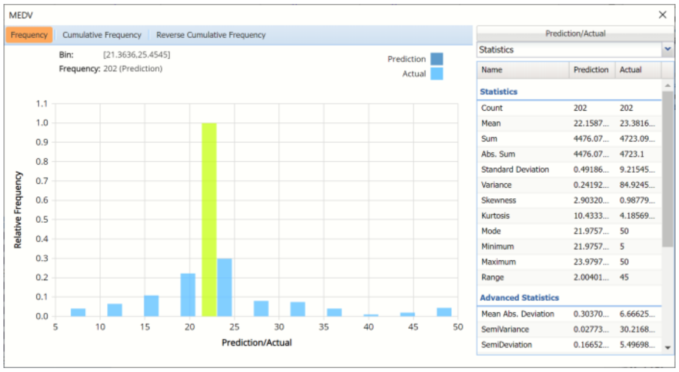
Training: Prediction Details

Record ID	MEDV	Prediction: MEDV	Residual
Record 1	24	21.97577418	2.0242258
Record 5	36.2	23.97923904	12.220761
Record 8	27.1	21.97577418	5.1242258
Record 11	15	21.97577418	-6.9757742
Record 12	18.9	21.97577418	-3.0757742
Record 15	18.2	21.97577418	-3.7757742

RBagging_ValidationScore

RBagging_ValidationScore displays the newly added Output Variable frequency chart, the Validation: Prediction Summary and the Validation: Prediction Details report. All calculations, charts and predictions on the *RBagging_ValidationScore* output sheet apply to the Validation partition.

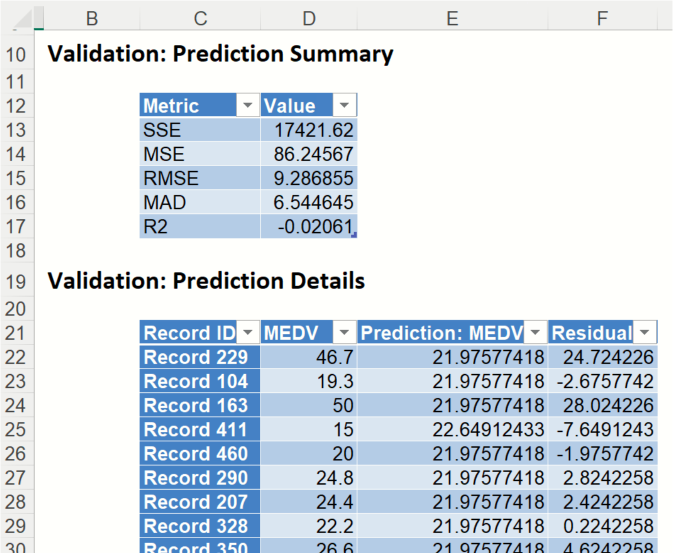
- **Frequency Charts:** The output variable frequency chart for the validation partition opens automatically once the *RBagging_ValidationScore* worksheet is selected. This chart displays a detailed, interactive frequency chart for the Actual variable data and the Predicted data, for the validation partition. For more information on this chart, see the *RBagging_TrainingScore* explanation above.



Name	Prediction	Actual
Statistics		
Count	202	202
Mean	22.1587...	23.3816...
Sum	4476.07...	4723.09...
Abs. Sum	4476.07...	4723.1
Standard Deviation	0.49186...	9.21545...
Variance	0.24192...	84.9245...
Skewness	2.90320...	0.98779...
Kurtosis	10.4333...	4.18569...
Mode	21.9757...	50
Minimum	21.9757...	5
Maximum	23.9797...	50
Range	2.00401...	45
Advanced Statistics		
Mean Abs. Deviation	0.30370...	6.66625...
SemiVariance	0.02773...	30.2168...
SemiDeviation	0.16652...	5.49698...

- **Prediction Summary:** In the Prediction Summary report, Analytic Solver Data Science displays the total sum of squared errors summaries for the Validation partition.

- **Prediction Details:** Scroll down to the Validation: Prediction Details report to find the Prediction value for the MEDV variable for each record in the Validation partition, as well as the Residual value.



Validation: Prediction Summary

Metric	Value
SSE	17421.62
MSE	86.24567
RMSE	9.286855
MAD	6.544645
R2	-0.02061

Validation: Prediction Details

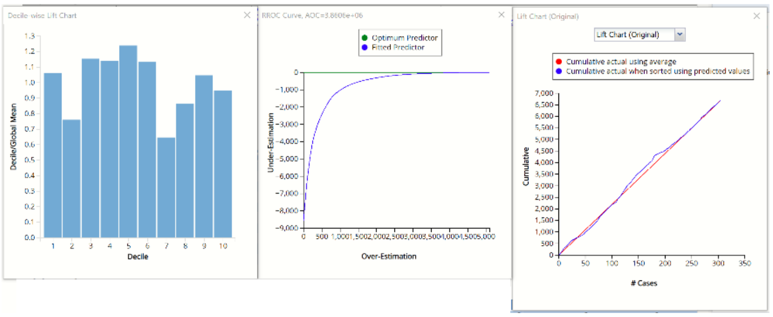
Record ID	MEDV	Prediction: MEDV	Residual
Record 229	46.7	21.97577418	24.724226
Record 104	19.3	21.97577418	-2.6757742
Record 163	50	21.97577418	28.024226
Record 411	15	22.64912433	-7.6491243
Record 460	20	21.97577418	-1.9757742
Record 290	24.8	21.97577418	2.8242258
Record 207	24.4	21.97577418	2.4242258
Record 328	22.2	21.97577418	0.2242258
Record 350	26.6	21.97577418	4.6242258

RBagging_TrainingLiftChart & RBagging_ValidationLiftChart

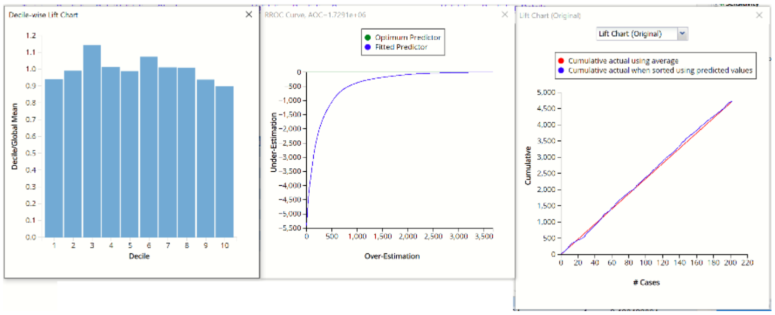
Click the **RBagging_TrainLiftChart** and **RBagging_ValidLiftChart** tabs to navigate to the Lift Charts and Regression RROC curves for both the training and validation datasets. For more information on how to interpret these charts, see the Neural Network chapter that appears above.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select RBagging_TrainingLiftChart or RBagging_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, RROC Curve and Lift Chart for Training Partition



Decile-wise Lift Chart, RROC Curve and Lift Chart for Valid. Partition

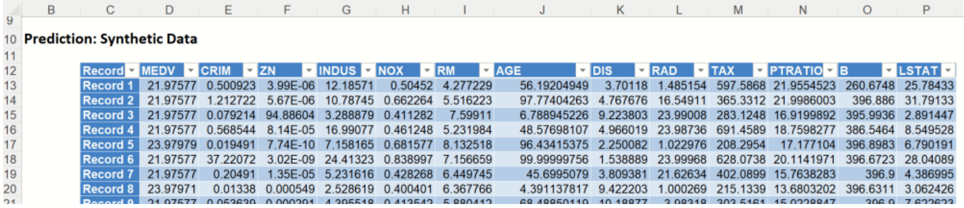


RBagging_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, RBagging_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Bagging Regression dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predicted values for the training data (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



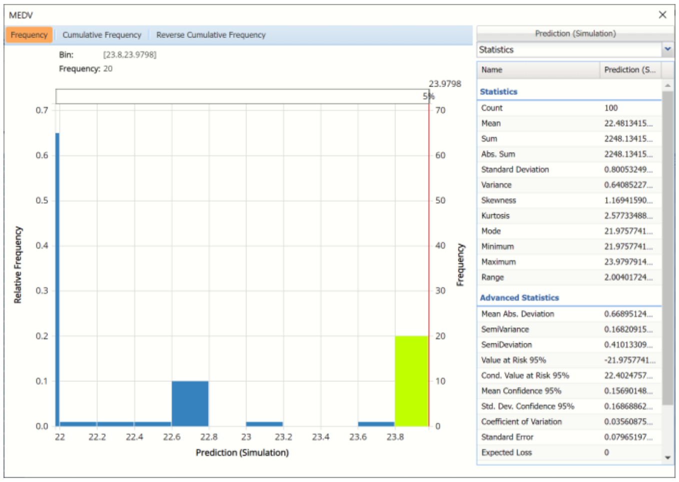
Prediction: Synthetic Data

Record	MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	21.97577	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19204949	3.70118	1.485154	597.5868	21.9554523	260.6748	25.78433
Record 2	21.97577	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404263	4.767676	16.54911	365.3312	21.9986003	396.886	31.79133
Record 3	21.97577	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945226	9.223803	23.99008	283.1248	16.9199892	395.9936	2.891447
Record 4	21.97577	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698107	4.966019	23.98736	691.4589	18.7598277	386.5464	8.549528
Record 5	23.97979	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415375	2.250082	1.022976	208.2954	17.177104	396.8983	6.790191
Record 6	21.97577	37.22072	3.02E-09	24.41323	0.838997	7.156659	99.99999756	1.538889	23.99968	628.0738	20.1141971	396.6723	28.04089
Record 7	21.97577	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.6995079	3.809381	21.62634	402.0899	15.7638283	396.9	4.386995
Record 8	23.97971	0.01338	0.000549	2.528619	0.400401	6.367766	4.391137817	9.422203	1.000269	215.1339	13.6803202	396.6311	3.062426
Record 9	21.97577	0.053639	0.000291	4.395518	0.413542	5.880412	68.48850119	10.18877	3.08318	303.5161	15.0228847	396.9	7.622623

The data contained in the Synthetic Data report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training data, the synthetic data and the expression, if it exists. (Recall that no expression was entered in this example.)

Frequency Chart for Prediction (Simulation) data

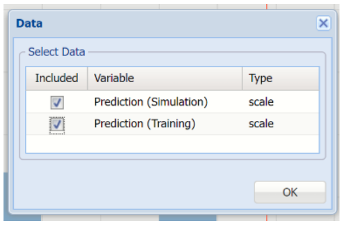


Name	Prediction (S...
Statistics	
Count	100
Mean	22.4813415...
Sum	2248.13415...
Abs. Sum	2248.13415...
Standard Deviation	0.80053249...
Variance	0.64085227...
Skewness	1.16941590...
Kurtosis	2.57733488...
Mode	21.9757741...
Minimum	21.9757741...
Maximum	23.9797914...
Range	2.00401724...
Advanced Statistics	
Mean Abs. Deviation	0.66895124...
SemiVariance	0.16820915...
SemiDeviation	0.41013309...
Value at Risk 95%	-21.9757741...
Cond. Value at Risk 95%	22.4024757...
Mean Confidence 95%	0.15690148...
Std. Dev. Confidence 95%	0.16868862...
Coefficient of Variation	0.03560875...
Standard Error	0.07965197...
Expected Loss	0

Click *Prediction (Simulation)* to add the training data to the chart.

Click *Prediction(Simulation)* and *Prediction (Training)* to change the Data view.

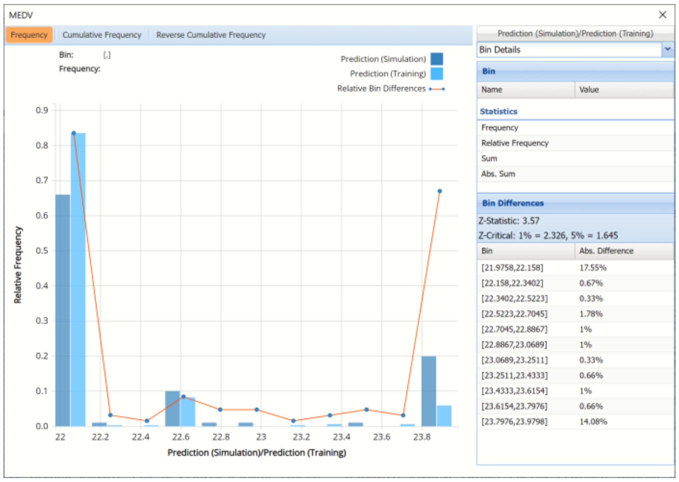
Data Dialog



Included	Variable	Type
☑	Prediction (Simulation)	scale
☑	Prediction (Training)	scale

In the chart below, the dark blue bars display the frequencies for the synthetic data and the light blue bars display the frequencies for the predicted values in the Training partition.

Prediction (Simulation) and Prediction (Training) Frequency chart for MEDV variable



Bin

Name	Value
Statistics	
Frequency	
Relative Frequency	
Sum	
Abs. Sum	

Bin Differences

Z-Statistic: 3.57

Z-Critical: 1% = 2.326, 5% = 1.645

Bin	Abs. Difference
[21.9758,22.158]	17.55%
[22.158,22.3402]	0.67%
[22.3402,22.5223]	0.33%
[22.5223,22.7045]	1.78%
[22.7045,22.8867]	1%
[22.8867,23.0689]	1%
[23.0689,23.2511]	0.33%
[23.2511,23.4333]	0.66%
[23.4333,23.6154]	1%
[23.6154,23.7976]	0.66%
[23.7976,23.9798]	14.08%

The Relative Bin Differences curve charts the absolute differences between the data in each bin. Click the down arrow next to Statistics to view the Bin Details pane to display the calculations.

Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

See the “Scoring New Data” chapter in the Analytic Solver Data Science User Guide for information on the Stored Model Sheet, *RBoosting_Stored.*

Continue on with the Random Trees Neural Network Regression Example in the next section to compare the results between the two ensemble methods.

Random Trees Ensemble Method Example

This example illustrates how to use the 3rd ensemble method, random trees, to create a regression model. We’ll again re-use the same partition of the Boston_Housing.xlsx dataset.

Input

1. Click **Predict – Ensemble – Random Trees** on the Data Science ribbon.

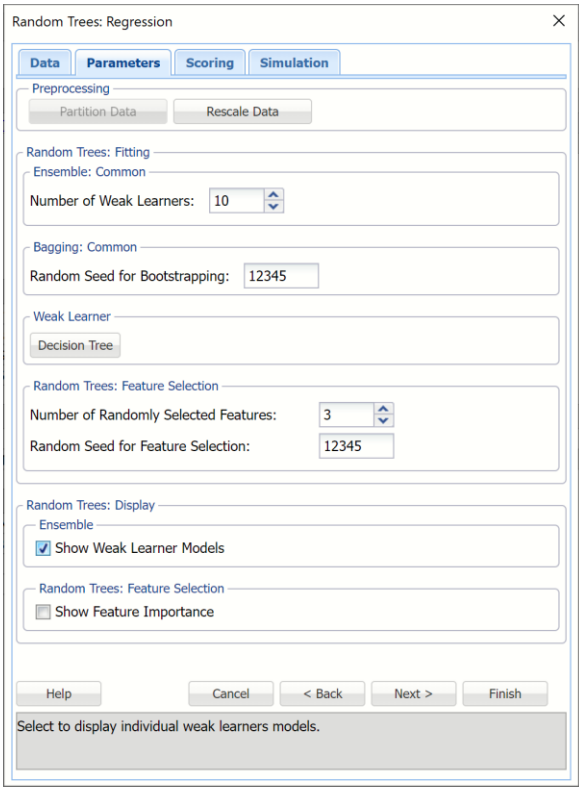
2. Select **MEDV** as the *Output variable and* **all remaining variables *except CAT.MEDV, CHAS*** and ***Record ID*** as *Selected Variables*. See the Boosting example above for a screenshot of the Data tab.

3. Click **Next** to advance to the *Random Trees***-** *Data* tab.

4. Recall that Random Trees only supports Decision Trees as a Weak Learner. Click Decision Tree to change any options associated with this algorithm. This example uses the default settings for the Decision Tree algorithm. For more information on these options, see the Regression Trees chapter that occurs earlier in this guide.

5. Select **Show Weak Learner Models** to include this information in the output. To see the output for Show Feature Importance see the Regresstion Tree chapter that appears earlier in this guide.

Random Trees Regression dialog, Parameters tab



Number of Weak Learners: 10

Random Seed for Bootstrapping: 12345

Number of Randomly Selected Features: 3

Random Seed for Feature Selection: 12345

☑ Show Weak Learner Models

☐ Show Feature Importance

Select to display individual weak learners models.

6. Click Next to advance to the *Random Trees Scoring* tab.

7. Select all four options for **Score Training/Validation data**.

When *Detailed report* is selected, Analytic Solver Data Science will create a detailed report of the Regression Trees output.

When *Summary report* is selected, Analytic Solver Data Science will create a report summarizing the Regression Trees output.

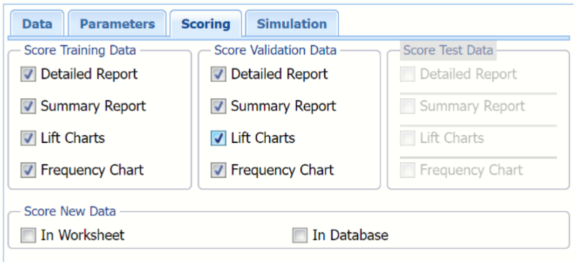
When *Lift Charts* is selected, Analytic Solver Data Science will include Lift Chart and ROC Curve plots in the output.

When Frequency Chart is selected, a frequency chart will be displayed when the RRandTrees_TrainingScore and RRandTrees_ValidationScore worksheets are selected. This chart will display an interactive application similar to the Analyze Data feature, explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Since we did not create a test partition, the options for Score test data are disabled. See the chapter "Data Science Partitioning" for information on how to create a test partition.

See the *Scoring New Data* chapter within the Analytic Solver Data Science User Guide for more information on *Score New Data in* options.

Random Trees Regression dialog, Scoring tab



Score Training Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data
☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data
☐ Detailed Report
☐ Summary Report
☐ Lift Charts
☐ Frequency Chart

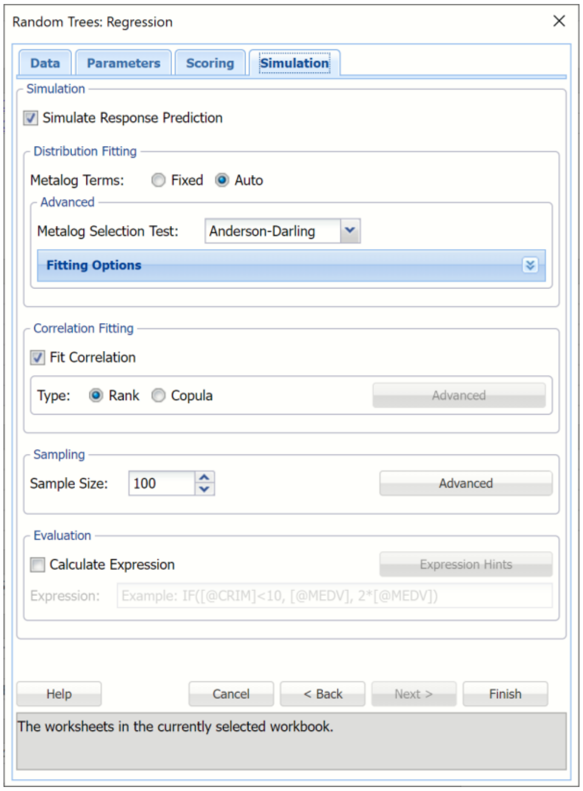
Score New Data
☐ In Worksheet
☐ In Database

8. Click **Next** to advance to the Simulation tab.

9. Select **Simulation Response Prediction** to enable all options on the Simulation tab of the Regression Tree dialog. This option is disabled in Analytic Solver Optimization, Analytic Solver Simulation and Analytic Solver Upgrade.

   Simulation tab: All supervised algorithms include a new Simulation tab. This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, RRandTrees_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Random Trees Regression dialog, Simulation tab



Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

□ Calculate Expression

Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

The worksheets in the currently selected workbook.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RRandTrees_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression. Note that variable names are case sensitive. See any of the prediction methods to see the Expressison field in use.

For more information on the remaining options shown on this dialog in the Distribution Fitting, Correlation Fitting and Sampling sections, see the Generate Data chapter that appears earlier in this guide.

10. Click **Finish** to run the Random Trees Ensemble Method on the example dataset.

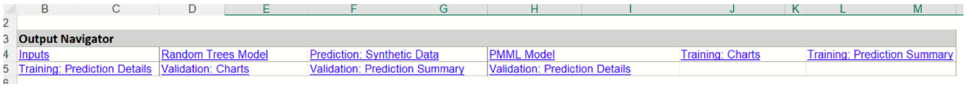
Output

The output of the Ensemble Methods algorithm are inserted at the end of the workbook.

RRandTrees_Output

This worksheet contains three sections: the Output Navigator, Inputs and Boosting Method.

- **Output Navigator:** Double click *RRandTrees_Output* to view the Output Navigator, which is inserted at the top of each output worksheet. Click any link in this table to navigate to various sections of the output.

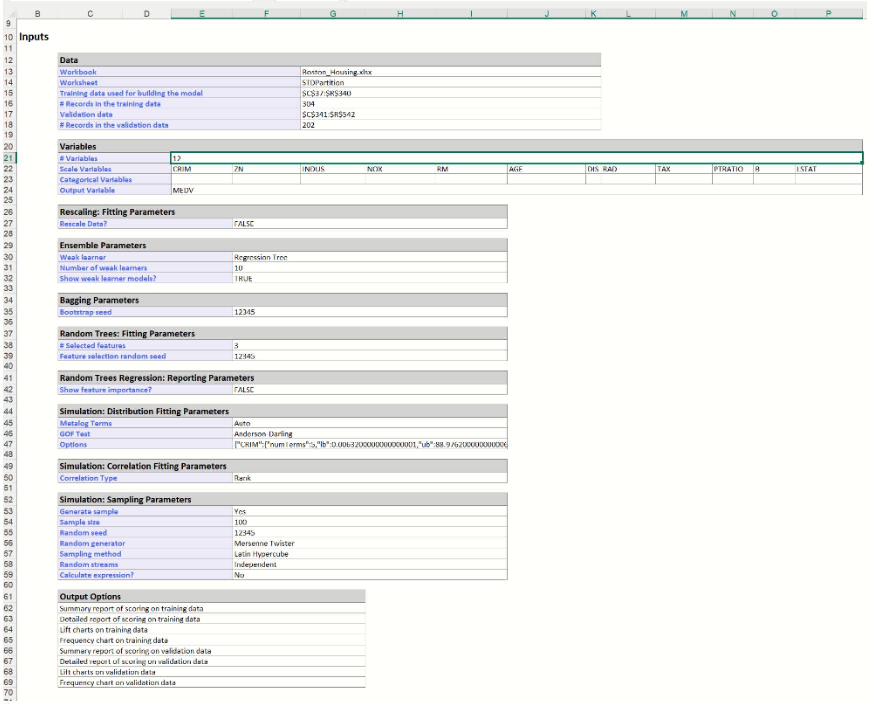


Output Navigator

Inputs	Random Trees Model	Prediction: Synthetic Data	PMML Model	Training: Charts	Training: Prediction Summary
Training: Prediction Details	Validation: Charts	Validation: Prediction Summary	Validation: Prediction Details		

- **Inputs:** Scroll down to the Inputs section to find all inputs entered or selected on all tabs of the Random Trees dialog.

RRandomTrees_Output, Inputs Report



Inputs

Data	
Workbook	Boston_Housing.xlsx
Worksheet	STDPartition
Training data used for building the model	$C$37:$R$340
# Records in the training data	304
Validation data	$C$341:$R$542
# Records in the validation data	202

Variables												
# Variables	12											
Scale Variables	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Categorical Variables												
Output Variable	MEDV											

Rescaling: Fitting Parameters	
Rescale Data?	FALSE

Ensemble Parameters	
Weak learner	Regression Tree
Number of weak learners	10
Show weak learner models?	TRUE

Bagging Parameters	
Bootstrap seed	12345

Random Trees: Fitting Parameters	
# Selected features	3
Feature selection random seed	12345

Random Trees Regression: Reporting Parameters	
Show feature importance?	FALSE

Simulation: Distribution Fitting Parameters	
Metalog Terms	Auto
GOF Test	Anderson-Darling
Options	{"CRIM":{"numTerms":5,"lb":0.006320000000000001,"ub":88.9762000000000006

Simulation: Correlation Fitting Parameters	
Correlation Type	Rank

Simulation: Sampling Parameters	
Generate sample	Yes
Sample size	100
Random seed	12345
Random generator	Mersenne Twister
Sampling method	Latin Hypercube
Random streams	Independent
Calculate expression?	No

Output Options
Summary report of scoring on training data
Detailed report of scoring on training data
Lift charts on training data
Frequency chart on training data
Summary report of scoring on validation data
Detailed report of scoring on validation data
Lift charts on validation data
Frequency chart on validation data

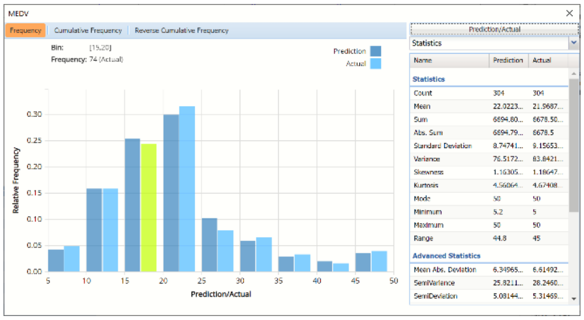
- **Boosting Model:** Click the Boosting Model link on the Output Naviagator to view the Boosting model for each weak learner. Recall that the default is "10" on the Parameters tab.

RRandTrees_TrainingScore & RRandTrees_ValidationScore

Click the *RRandTrees_TrainingScore and RRandTrees_ValidationScore* tabs to view the newly added Output Variable frequency chart, the Training\Validation: Prediction Summary and the Training\Validation: Prediction Details report.
Note: To view charts in the Cloud app, click the Charts icon on the Ribbon, select a worksheet under Worksheet and a chart under Chart.

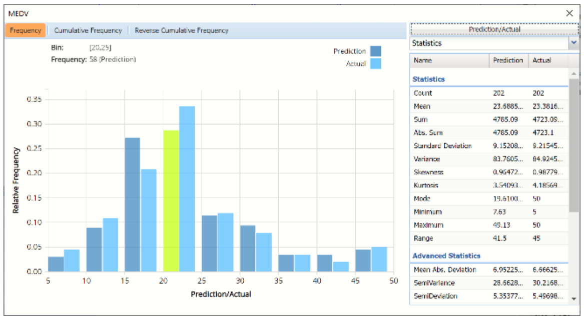
- **Frequency Charts:** The output variable frequency chart opens automatically once the *RRandTrees_TrainingScore or RRandTrees_ValidationScore* worksheet is selected. For more information on this dialog, see the Boosting example above.

RRandTrees_TrainingScore Frequency chart



Name	Prediction	Actual
Statistics		
Count	304	304
Mean	22.0223...	21.9687...
Sum	6694.80...	6678.50...
Abs. Sum	6694.79...	6678.5
Standard Deviation	8.74741...	9.15653...
Variance	76.5172...	83.8421...
Skewness	1.16305...	1.18647...
Kurtosis	4.56064...	4.67408...
Mode	50	50
Minimum	5.2	5
Maximum	50	50
Range	44.8	45
Advanced Statistics		
Mean Abs. Deviation	6.34965...	6.61492...
SemiVariance	25.8211...	28.2460...
SemiDeviation	5.08144...	5.31469...

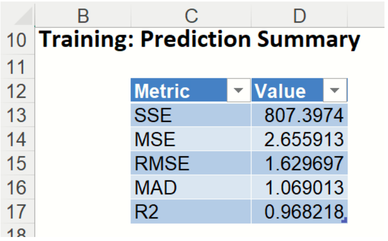
RRandTrees_ValidationScore Frequency chart



Name	Prediction	Actual
Statistics		
Count	202	202
Mean	23.6885...	23.3816...
Sum	4785.09	4723.09...
Abs. Sum	4785.09	4723.1
Standard Deviation	9.15208...	9.21545...
Variance	83.7605...	84.9245...
Skewness	0.96472...	0.98779...
Kurtosis	3.54093...	4.18569...
Mode	19.6100...	50
Minimum	7.63	5
Maximum	49.13	50
Range	41.5	45
Advanced Statistics		
Mean Abs. Deviation	6.95225...	6.66625...
SemiVariance	28.6628...	30.2168...
SemiDeviation	5.35377...	5.49698...

- **Training: Prediction Summary:** Click the Training: Prediction Summary link on the Output Navigator to open the Training Summary and the Validation: Prediction Summary link to open the Validation Summary This data table displays various statistics to measure the performance of the trained network: Sum of Squared Error (SSE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), the Median Absolute Deviation (MAD) and the Coefficient of Determination (R^2).

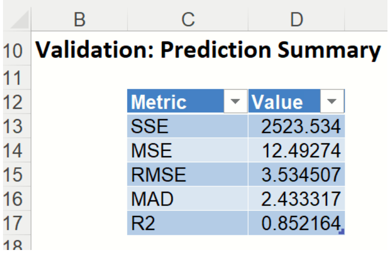
RRandTrees_TrainingScore Frequency chart



Training: Prediction Summary

Metric	Value
SSE	807.3974
MSE	2.655913
RMSE	1.629697
MAD	1.069013
R2	0.968218

RRandTrees_ValidationScore Frequency chart

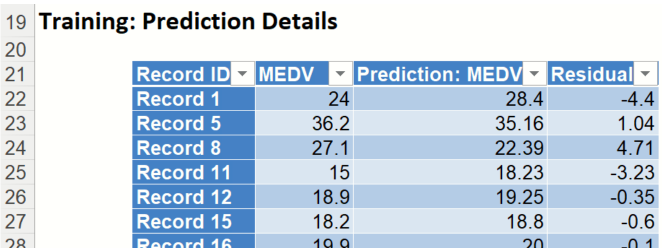


Validation: Prediction Summary

Metric	Value
SSE	2523.534
MSE	12.49274
RMSE	3.534507
MAD	2.433317
R2	0.852164

- **Training: Prediction Details:** Scroll down to view the Prediction Details data table. This table displays the Actual versus Predicted values, along with the Residuals, for the training dataset.

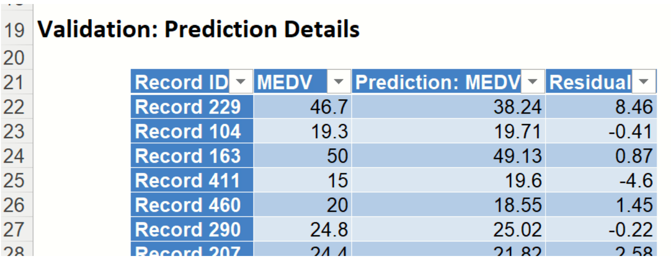
RRandTrees_TrainingScore Frequency chart



Training: Prediction Details

Record ID	MEDV	Prediction: MEDV	Residual
Record 1	24	28.4	-4.4
Record 5	36.2	35.16	1.04
Record 8	27.1	22.39	4.71
Record 11	15	18.23	-3.23
Record 12	18.9	19.25	-0.35
Record 15	18.2	18.8	-0.6
Record 16	19.9	20	-0.1

RRandTrees_ValidationScore Frequency chart



Validation: Prediction Details

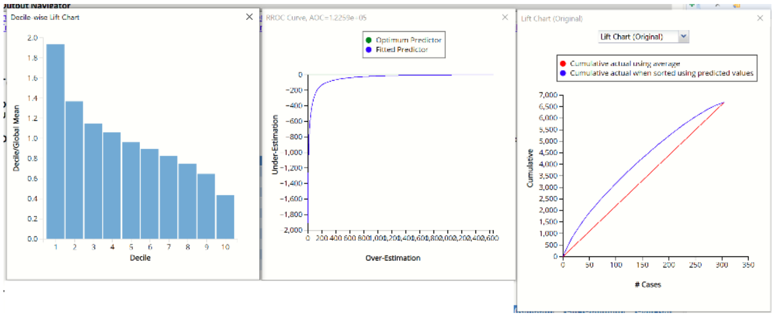
Record ID	MEDV	Prediction: MEDV	Residual
Record 229	46.7	38.24	8.46
Record 104	19.3	19.71	-0.41
Record 163	50	49.13	0.87
Record 411	15	19.6	-4.6
Record 460	20	18.55	1.45
Record 290	24.8	25.02	-0.22
Record 207	24.4	21.82	2.58

RRandTrees_TrainingLiftChart & RRandTrees_ValidationLiftChart

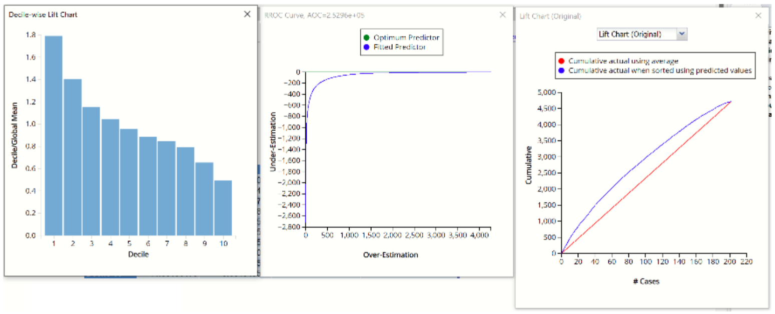
Click the **RRandTrees_TrainLiftChart** and **RRandTrees_ValidLiftChart** tabs to navigate to the Lift Charts and Regression RROC curves for both the training and validation datasets. For more information on how to interpret these charts, see the Regression Tree chapter that appears above.

Note: To view these charts in the Cloud app, click the Charts icon on the Ribbon, select RRandTrees_TrainingLiftChart or RRandTrees_ValidationLiftChart for Worksheet and Decile Chart, ROC Chart or Gain Chart for Chart.

Decile-wise Lift Chart, RROC Curve and Lift Chart for Training Partition



Decile-wise Lift Chart, RROC Curve and Lift Chart for Valid. Partition

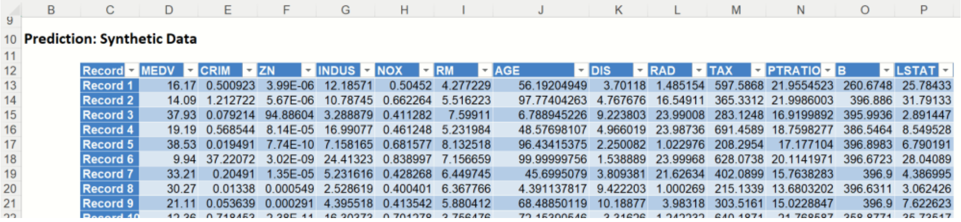


RRandTrees_Simulation

As discussed above, Analytic Solver Data Science generates a new output worksheet, RRandTrees_Simulation, when *Simulate Response Prediction* is selected on the Simulation tab of the Randon Trees Regression dialog in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.)

This report contains the synthetic data, the predicted values for the training data (using the fitted model) and the Excel – calculated Expression column, if populated in the dialog. Users can switch between the Predicted, Training, and Expression sources or a combination of two, as long as they are of the same type.

Synthetic Data



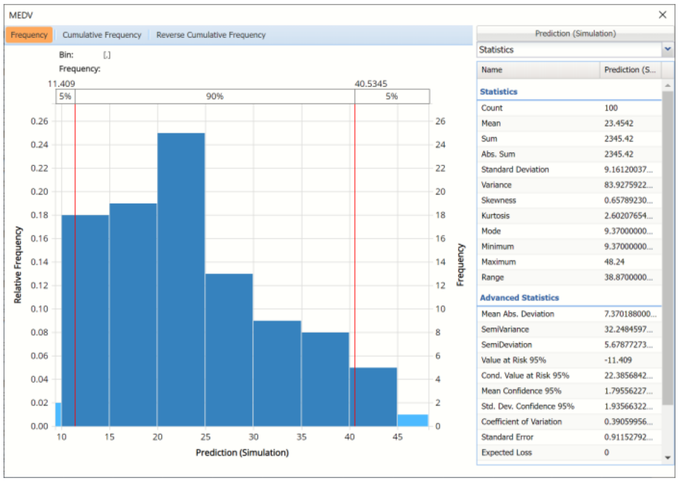
Prediction: Synthetic Data

Record	MEDV	CRIM	ZN	INDUS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT
Record 1	16.17	0.500923	3.99E-06	12.18571	0.50452	4.277229	56.19204949	3.70118	1.485154	597.5868	21.9554523	260.6748	25.78433
Record 2	14.09	1.212722	5.67E-06	10.78745	0.662264	5.516223	97.77404263	4.767676	16.54911	365.3312	21.9986003	396.886	31.79133
Record 3	37.93	0.079214	94.88604	3.288879	0.411282	7.59911	6.788945226	9.223803	23.99008	283.1248	16.9199892	395.9936	2.891447
Record 4	19.19	0.568544	8.14E-05	16.99077	0.461248	5.231984	48.57698107	4.966019	23.98736	691.4589	18.7598277	386.5464	8.549528
Record 5	38.53	0.019491	7.74E-10	7.158165	0.681577	8.132518	96.43415375	2.250082	1.022976	208.2954	17.177104	396.8983	6.790191
Record 6	9.94	37.22072	3.02E-09	24.41323	0.838997	7.156659	99.99999756	1.538889	23.99968	628.0738	20.1141971	396.6723	28.04089
Record 7	33.21	0.20491	1.35E-05	5.231616	0.428268	6.449745	45.6995079	3.809381	21.62634	402.0899	15.7638283	396.9	4.386995
Record 8	30.27	0.01338	0.000549	2.528619	0.400401	6.367766	4.391137817	9.422203	1.000269	215.1339	13.6803202	396.6311	3.062426
Record 9	21.11	0.053639	0.000291	4.395518	0.413542	5.880412	68.48850119	10.18877	3.98318	303.5161	15.0228847	396.9	7.622623

The data contained in the Synthetic Data report is syntethic data, generated using the Generate Data feature described in the chapter with the same name, that appears earlier in this guide.

The chart that is displayed once this tab is selected, contains frequency information pertaining to the output variable in the training data, the synthetic data and the expression, if it exists. (Recall that no expression was entered in this example.)

Frequency Chart for Prediction (Simulation) data

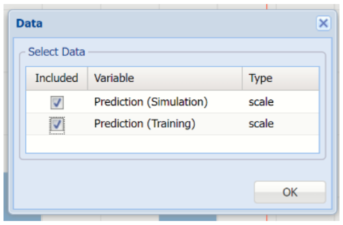


Name	Prediction (S...
Statistics	
Count	100
Mean	23.4542
Sum	2345.42
Abs. Sum	2345.42
Standard Deviation	9.16120037...
Variance	83.9275922...
Skewness	0.65789230...
Kurtosis	2.60207654...
Mode	9.37000000...
Minimum	9.37000000...
Maximum	48.24
Range	38.8700000...
Advanced Statistics	
Mean Abs. Deviation	7.370188000...
SemiVariance	32.2484597...
SemiDeviation	5.67877273...
Value at Risk 95%	-11.409
Cond. Value at Risk 95%	22.3856842...
Mean Confidence 95%	1.79556227...
Std. Dev. Confidence 95%	1.93566322...
Coefficient of Variation	0.39059956...
Standard Error	0.91152792...
Expected Loss	0

Click *Prediction (Simulation)* to add the training data to the chart.

Click *Prediction(Simulation)* and *Prediction (Training)* to change the Data view.

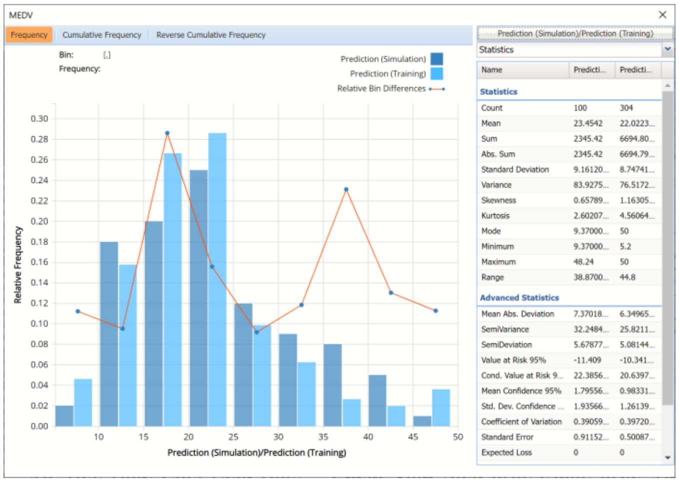
Data Dialog



Included	Variable	Type
☑	Prediction (Simulation)	scale
☑	Prediction (Training)	scale

In the chart below, the dark blue bars display the frequencies for the synthetic data and the light blue bars display the frequencies for the predicted values in the Training partition.

Prediction (Simulation) and Prediction (Training) Frequency chart for MEDV variable



Name	Predicti...	Predicti...
Statistics		
Count	100	304
Mean	23.4542	22.0223...
Sum	2345.42	6694.80...
Abs. Sum	2345.42	6694.79...
Standard Deviation	9.16120...	8.74741...
Variance	83.9275...	76.5172...
Skewness	0.65789...	1.16305...
Kurtosis	2.60207...	4.56064...
Mode	9.37000...	50
Minimum	9.37000...	5.2
Maximum	48.24	50
Range	38.8700...	44.8
Advanced Statistics		
Mean Abs. Deviation	7.37018...	6.34965...
SemiVariance	32.2484...	25.8211...
SemiDeviation	5.67877...	5.08144...
Value at Risk 95%	-11.409	-10.341...
Cond. Value at Risk 9...	22.3856...	20.6397...
Mean Confidence 95%	1.79556...	0.98331...
Std. Dev. Confidence ...	1.93566...	1.26139...
Coefficient of Variation	0.39059...	0.39720...
Standard Error	0.91152...	0.50087...
Expected Loss	0	0

The Relative Bin Differences curve charts the absolute differences between the data in each bin. Click the down arrow next to Statistics to view the Bin Details pane to display the calculations.

Click the down arrow next to Frequency to change the chart view to Relative Frequency or to change the look by clicking Chart Options. Statistics on the right of the chart dialog are discussed earlier in this section. For more information on the generated synthetic data, see the Generate Data chapter that appears earlier in this guide.

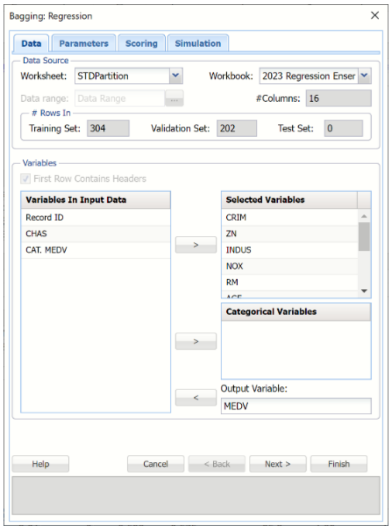
See the "Scoring New Data" chapter in the Analytic Solver Data Science User Guide for information on the Stored Model Sheet, *RRandTrees_Stored*.

Ensemble Methods for Regression Options

The following options appear on the Bagging, Boosting, and Random Trees dialogs.

Please see below for options appearing on the *Ensemble Methods- Data* tab.

Ensemble Methods Regression dialog, Data tab



Bagging: Regression

Data Source
Worksheet: STDPartition
Workbook: 2023 Regression Enser
Data range: Data Range
#Columns: 16

Rows In
Training Set: 304
Validation Set: 202
Test Set: 0

Variables
☑ First Row Contains Headers

Variables In Input Data
Record ID
CHAS
CAT. MEDV

Selected Variables
CRIM
ZN
INDUS
NOX
RM

Categorical Variables

Output Variable: MEDV

Ensemble Methods Regression dialog, Data tab

Variables In Input Data

The variables included in the dataset appear here.

Selected Variables

Variables selected to be included in the output appear here.

Categorical Variables

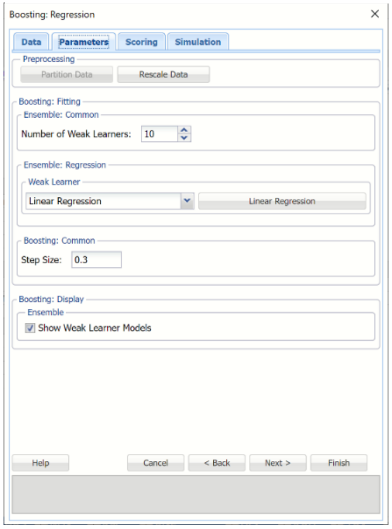
Place categorical variables from the Variables listbox to be included in the model by clicking the > command button. Ensemble Methods will accept non-numeric categorical variables.

Output Variable

The dependent variable or the variable to be classified appears here.

Please see below for options appearing on the *Boosting – Parameters* tab.

Boosting Regression dialog, Parameters tab



Boosting: Regression

Preprocessing

Boosting: Fitting
Ensemble: Common
Number of Weak Learners: 10

Ensemble: Regression
Weak Learner
Linear Regression

Boosting: Common
Step Size: 0.3

Boosting: Display
Ensemble
☑ Show Weak Learner Models

Ensemble Methods Regression dialog, Parameters tab

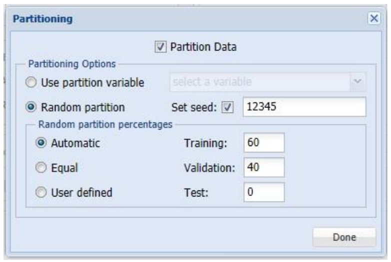
Please see below for options appearing on the Parameters tab of the Ensemble Methods Parameters tab.

Partition Data

Analytic Solver Data Science includes the ability to partition a dataset from within a classification or prediction method by clicking Partition Data on the Parameters tab. Click **Partition Data** to open the Partitioning dialog. Analytic Solver Data Science will partition your dataset (according to the partition options you set) immediately before running the regression method. If partitioning has already occurred on the dataset, this option will be disabled.

For more information on partitioning, please see the Data Science Partitioning chapter.

"On-the-fly" Partitioning dialog



Partitioning

☑ Partition Data

Partitioning Options

○ Use partition variable select a variable

● Random partition Set seed: ☑ 12345

Random partition percentages

● Automatic Training: 60

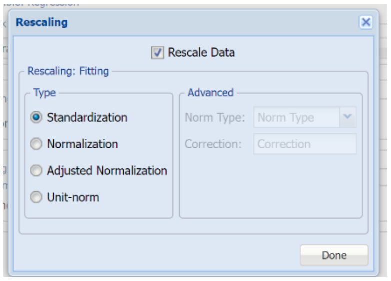
○ Equal Validation: 40

○ User defined Test: 0

Rescale Data

Recall that the Euclidean distance measurement performs best when each variable is rescaled. Here you can select how you want to standardize your variables using Standardization. Normalization, Adjusted Normalization and Unit Norm.

"On-the-fly" Rescaling dialog



Rescaling

☑ Rescale Data

Rescaling: Fitting

Type

● Standardization

○ Normalization

○ Adjusted Normalization

○ Unit-norm

Advanced

Norm Type: Norm Type

Correction: Correction

- Standardization, sometimes referred to as Z-Scores, subtracts the mean from each record's variable value and divides it by the standard deviation. (x−mean)

- Normalization subtracts the minimum value from each record's variable value and divides by the range. (x−min)/(max−min)

  The Correction option specifies a number ε that is applied as a correction to the rescaling formula. The corrected formula is [x−(min−ε)]/[(max+ε)−(min−ε)].

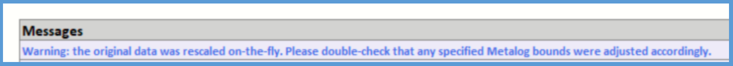
- Adjusted Normalization subtracts the minimum value from each record's variable value and divides by the range. [2(x−min)/(max−min)]−1

  The Correction option specifies a number ε that is applied as a correction to the rescaling formula. The corrected formula is {2[(x−(min−ε))/((max+ε)−(min−ε))]}−1.

- Unit Normalization is another option that is widely used in machine-learning to scale the components of a feature vector such that the complete vector has a length of one. This usually means dividing each component by the Euclidean length of the vector. In some applications it can be more practical to use the L1-norm.

Notes on Rescaling and the Simulation functionality

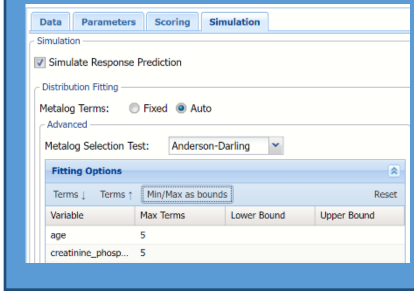
If Rescale Data is turned on, i.e. if Rescale Data is selected on the Rescaling dialog as shown in the screenshot above, then “Min/Max as bounds” on the Simulation tab will not be turned on by default. A warning will be reported in the Log on the *Ensemble Method*_Simulation output sheet, as shown below.



Messages

Warning: the original data was rescaled on-the-fly. Please double-check that any specified Metalog bounds were adjusted accordingly.

If Rescale Data has been selected on the Rescaling dialog, users can still manually use the “Min/Max as bounds” button within the Fitting Options section of the Simulation tab, to populate the parameter grid with the bounds from the *original* data, not the *rescaled* data. Note that the “Min/Max as bounds” feature is available for the user’s convenience. Users must still be aware of any possible data tranformations (i.e. Rescaling) and review the bounds to make sure that all are appropriate.



☑ Simulate Response Prediction

Metalog Terms: ○ Fixed ● Auto

Metalog Selection Test: Anderson-Darling

Variable	Max Terms	Lower Bound	Upper Bound
age	5		
creatinine_phosp...	5		

Number of Weak Learners

This option controls the number of “weak” prediction models that will be created. The ensemble method will stop when the number of prediction models created reaches the value set for this option.

Weak Learner

Under Ensemble: Regression click the down arrow beneath Weak Leaner to select one of the four featured classifiers: Linear Regression, k-NN, Neural Networks, or Decision Trees. After a weak learner is chosen, the command button to the right will be enabled. Click this command button to control various option settings for the weak leaner.

Step Size

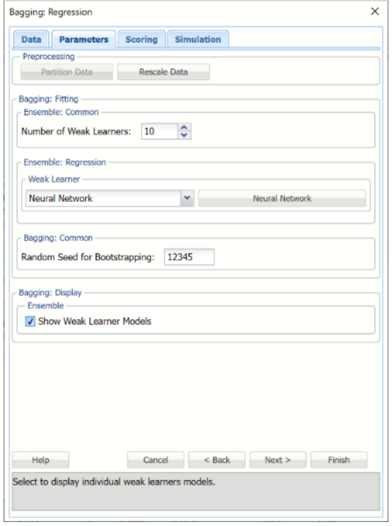
The Adaboost algorithm minimizes a loss function using the gradient descent method. The Step size option is used to ensure that the algorithm does not descend too far when moving to the next step. It is recommended to leave this option at the default of 0.3, but any number between 0 and 1 is acceptable. A Step size setting closer to 0 results in the algorithm taking smaller steps to the next point, while a setting closer to 1 results in the algorithm taking larger steps towards the next point.

Show Weak Learner

To display the weak learner models in the output, select **Show Weak Learner Models**.

Please see below for options unique to the *Bagging – Parameters* tab.

Bagging Regression dialog, Parameters tab



Preprocessing

Bagging: Fitting

Ensemble: Common

Number of Weak Learners: 10

Ensemble: Regression

Weak Learner: Neural Network

Bagging: Common

Random Seed for Bootstrapping: 12345

Bagging: Display

Ensemble

☑ Show Weak Learner Models

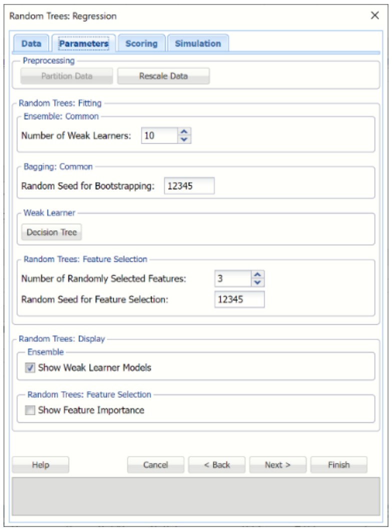
Select to display individual weak learners models.

Random Seed for Boostrapping

If an integer value appears for *Bootstrapping Random seed*, Analytic Solver Data Science will use this value to set the bootstrapping random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Please see below for options unique to the *Random Trees – Parameters* tab.

Random Trees Regression dialog, Parameters tab



Preprocessing

Random Trees: Fitting

Ensemble: Common

Number of Weak Learners: 10

Bagging: Common

Random Seed for Bootstrapping: 12345

Weak Learner

Random Trees: Feature Selection

Number of Randomly Selected Features: 3

Random Seed for Feature Selection: 12345

Random Trees: Display

Ensemble

☑ Show Weak Learner Models

Random Trees: Feature Selection

☐ Show Feature Importance

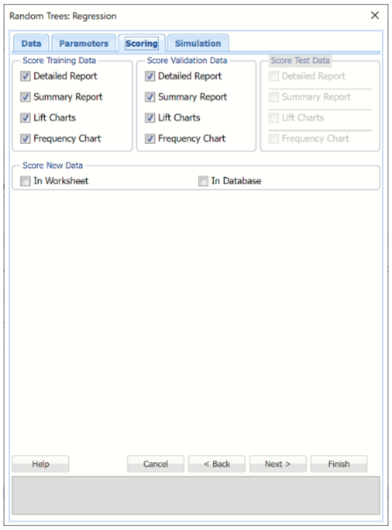
Number of Randomly Selected Features

The Random Trees ensemble method works by training multiple “weak” classification trees using a fixed number of randomly selected features then taking the mode of each class to create a “strong” classifier. The option *Number of randomly selected features* controls the fixed number of randomly selected features in the algorithm. The default setting is **3**.

Random Seed for Feature Selection

If an integer value appears for *Feature Selection Random seed*, Analytic Solver Data Science will use this value to set the feature selection random number seed. Setting the random number seed to a nonzero value (any number of your choice is OK) ensures that the same sequence of random numbers is used each time the dataset is chosen for the classifier. The default value is “12345”. If left blank, the random number generator is initialized from the system clock, so the sequence of random numbers will be different in each calculation. If you need the results from successive runs of the algorithm to another to be strictly comparable, you should set the seed. To do this, type the desired number you want into the box. This option accepts both positive and negative integers with up to 9 digits.

Ensemble Methods Regression dialog, Scoring tab



Score Training Data

☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Validation Data

☑ Detailed Report
☑ Summary Report
☑ Lift Charts
☑ Frequency Chart

Score Test Data

☐ Detailed Report
☐ Summary Report
☐ Lift Charts
☐ Frequency Chart

Score New Data

☐ In Worksheet
☐ In Database

Regression Tree Dialog, Scoring tab

Please see below for options that are unique to the *Ensemble Methods Scoring tab.*

Score Training Data

Select these options to show an assessment of the performance of the Neural Network in predicting the value of the output variable in the training partition.

When Frequency Chart is selected, a frequency chart will be displayed when the NNP_TrainingScore worksheet are selected. This chart will display an interactive application similar to the Analyze Data feature, and explained in detail in the Analyze Data chapter that appears earlier in this guide. This chart will include frequency distributions of the actual and predicted responses individually, or side-by-side, depending on the user's preference, as well as basic and advanced statistics for variables, percentiles, six sigma indices.

Score Validation Data

These options are enabled when a validation data set is present. Select these options to show an assessment of the performance of the Neural Network in predicting the value of the output variable in the validation data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the NNP_ValidationScore worksheet is selected.

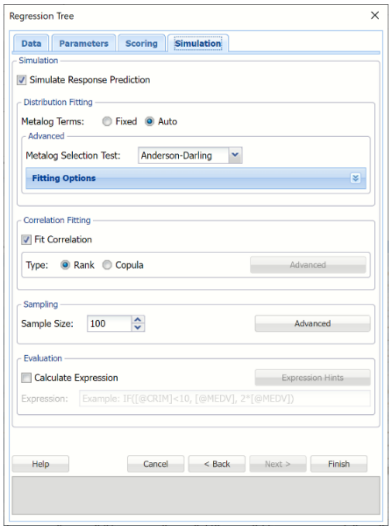
Score Test Data

These options are enabled when a test set is present. Select these options to show an assessment of the performance of the Neural Network in predicting the value of the output variable in the test data. The report is displayed according to your specifications - Detailed, Summary, and Lift charts. When Frequency Chart is selected, a frequency chart (described above) will be displayed when the NNP_TestScore worksheet is selected.

Score New Data

See the *Scoring* chapter within the Analytic Solver Data Science User Guide for more information on the options located in the *Score Test Data* and *Score New Data* groups.

Ensemble Methods Regression dialog, Simulation tab



Regression Tree

Simulation

☑ Simulate Response Prediction

Distribution Fitting

Metalog Terms: ○ Fixed ● Auto

Advanced

Metalog Selection Test: Anderson-Darling

Correlation Fitting

☑ Fit Correlation

Type: ● Rank ○ Copula

Sampling

Sample Size: 100

Evaluation

□ Calculate Expression

Expression: Example: IF([@CRIM]<10, [@MEDV], 2*[@MEDV])

Regression Tree Dialog,Simulation tab

Simulation Tab

All supervised algorithms include a new Simulation tab in Analytic Solver Comprehensive and Analytic Solver Data Science. (This feature is not supported in Analytic Solver Optimization, Analytic Solver Simulation or Analytic Solver Upgrade.) This tab uses the functionality from the Generate Data feature (described earlier in this guide) to generate synthetic data based on the training partition, and uses the fitted model to produce predictions for the synthetic data. The resulting report, NNP_Simulation, will contain the synthetic data, the predicted values and the Excel-calculated Expression column, if present. In addition, frequency charts containing the Predicted, Training, and Expression (if present) sources or a combination of any pair may be viewed, if the charts are of the same type.

Evaluation: Select Calculate Expression to amend an Expression column onto the frequency chart displayed on the RT_Simulation output tab. Expression can be any valid Excel formula that references a variable and the response as [@COLUMN_NAME]. Click the *Expression Hints* button for more information on entering an expression.

Association Rules

Introduction

The goal of association rules mining is to recognize associations and/or correlations among large sets of data items. A typical and widely-used example of association rules mining is the Market Basket Analysis. Most ‘market basket’ databases consist of a large number of transaction records where each record lists all items purchased by a customer during a trip through the check-out line. Data is easily and accurately collected through the bar-code scanners. Supermarket managers are interested in determining what foods customers purchase together, like, for instance, bread and milk, bacon and eggs, wine and cheese, etc. This information is useful in planning store layouts (placing items optimally with respect to each other), cross-selling promotions, coupon offers, etc.

Association rules provide results in the form of "if-then" statements. These rules are computed from the data and, unlike the if-then rules of logic, are probabilistic in nature. The “if” portion of the statement is referred to as the *antecedent* and the “then” portion of the statement is referred to as the *consequent*.

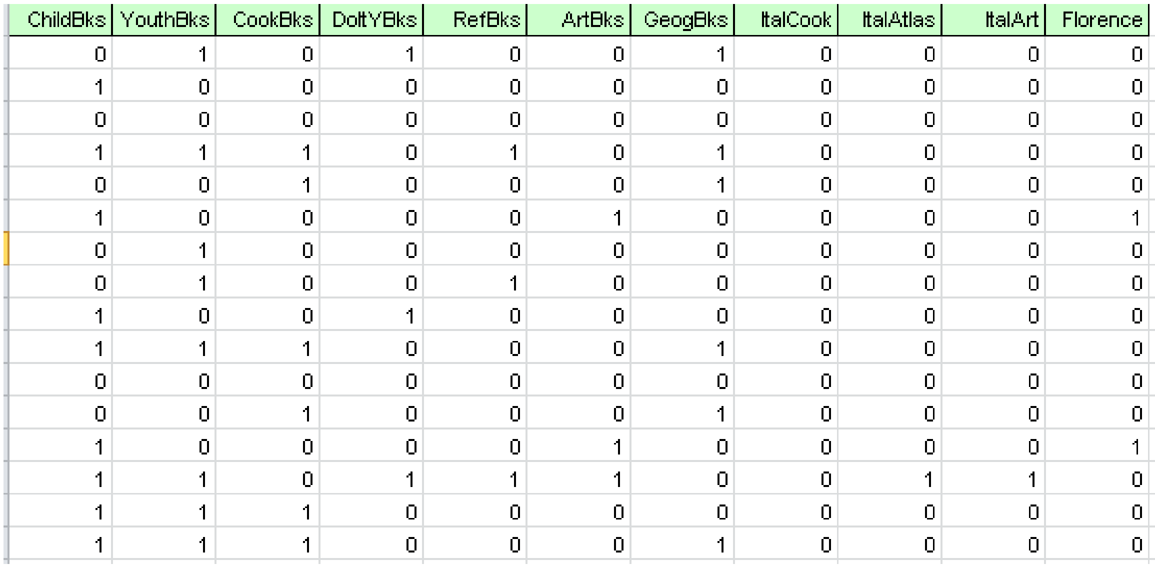
In addition to the antecedent (the "if" part) and the consequent (the "then" part), an association rule contains two numbers that express the degree of uncertainty about the rule. In association analysis the antecedent and consequent are sets of items (called itemsets) that are disjoint meaning they do not have any items in common. The first number is called the **support** which is simply the number of transactions that include all items in the antecedent and consequent. (The support is sometimes expressed as a percentage of the total number of records in the database.) The second number is known as the **confidence** which is the ratio of the number of transactions that include all items in the consequent as well as the antecedent (namely, the support) to the number of transactions that include all items in the antecedent. For example, assume a supermarket database has 100,000 point-of-sale transactions, out of which 2,000 include both items A and B and 800 of these include item C. The association rule "If A and B are purchased then C is purchased on the same trip" has a support of 800 transactions (alternatively 0.8% = 800/100,000) and a confidence of 40% (=800/2,000). In other words, support is the probability that a randomly selected transaction from the database will contain all items in the antecedent and the consequent. Confidence is the conditional probability that a randomly selected transaction will include all the items in the consequent given that the transaction includes all the items in the antecedent.

Lift is one more parameter of interest in the association analysis. Lift is the ratio of Confidence to Expected Confidence. Expected Confidence, in the example above, is the "confidence of buying A and B does not enhance the probability of buying C." or the number of transactions that include the consequent divided by the total number of transactions. Suppose the total number of transactions for C is 5,000. Expected Confidence is computed as 5% (5,000/1,000,000) while the ratio of Lift Confidence to Expected Confidence is 8 (40%/5%). Hence, Lift is a value that provides information about the increase in probability of the "then" (consequent) given the "if" (antecedent).

A lift ratio larger than 1.0 implies that the relationship between the antecedent and the consequent is more significant than would be expected if the two sets were independent. The larger the lift ratio, the more significant the association.

Association Rules Example

This example below illustrates how to use Analytic Solver Data Science's Association Rules method using the example dataset contained in the file, Associations.xlsx. Click **Help – Example Models** on the Data Science ribbon, then **Forecasting/Data Science Examples** to open this dataset.



ChildBks	YouthBks	CookBks	DoItYBks	RefBks	ArtBks	GeogBks	ItalCook	ItalAtlas	ItalArt	Florence
0	1	0	1	0	0	1	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	1	0	1	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0
1	0	0	0	0	1	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0
0	1	0	0	1	0	0	0	0	0	0
1	0	0	1	0	0	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0
1	0	0	0	0	1	0	0	0	0	1
1	1	0	1	1	1	0	0	1	1	0
1	1	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0

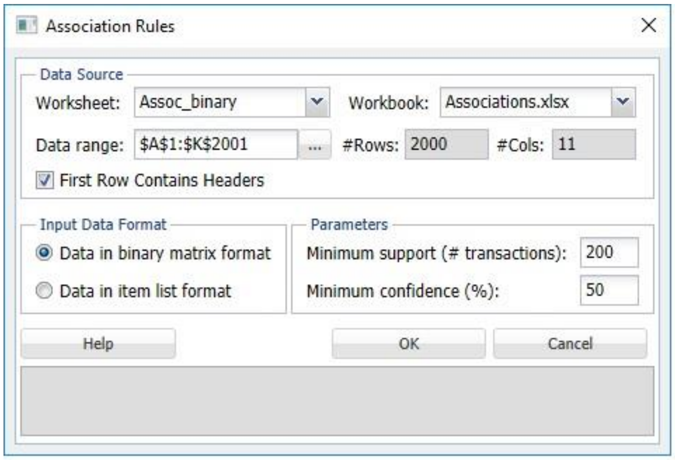
Click **Associate – Association Rules** to open the *Association Rules* dialog.

Since the data contained in the Associations.xlsx dataset are all 0's and 1's, *Data in binary matrix format* is selected by default for the option, *Input data format*. Analytic Solver Data Science will treat the data as a matrix of two entities -- zeros and non-zeros. A 0 signifies that the item is absent in that transaction and a 1 signifies the item is present.

Note: If a value other than 0 or 1 were present in the dataset, *Data in Item List Format* would have been selected by default for the option, *Input Data Format*.

Keep the default of 200 for the *Minimum Support (# transactions)*. This option specifies the minimum number of transactions in which a particular item-set must appear to qualify for inclusion in an association rule.

Keep the default of 50 for *Minimum confidence %*. This option specifies the minimum confidence threshold for rule generation. If A is the set of Antecedents and C the set of Consequents, then only those A =>C ("Antecedent implies Consequent") rules will qualify, for which the ratio (support of A U C) / (support of A) at least equals this percentage.



Association Rules

Data Source

Worksheet: Assoc_binary

Workbook: Associations.xlsx

Data range: $A$1:$K$2001

#Rows: 2000

#Cols: 11

☑ First Row Contains Headers

Input Data Format

● Data in binary matrix format

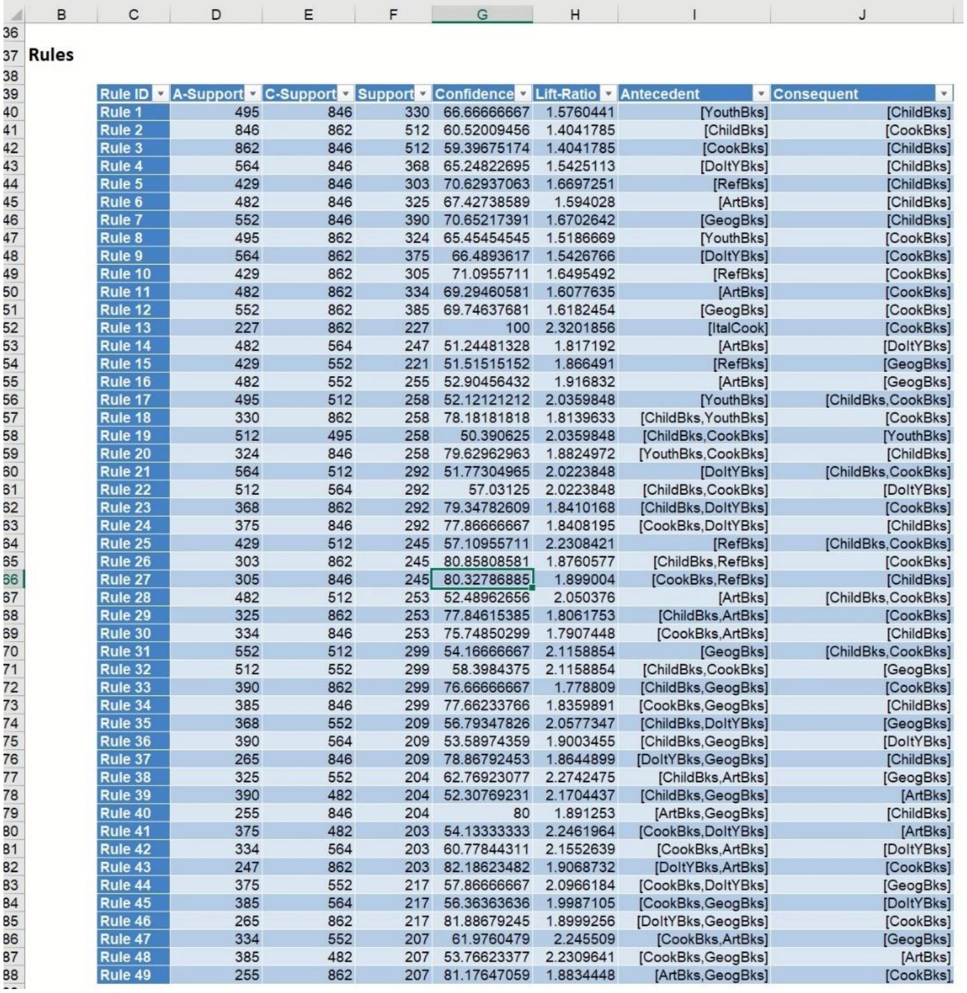
○ Data in item list format

Parameters

Minimum support (# transactions): 200

Minimum confidence (%): 50

Click **OK**. *AR_Output* is inserted to the right of the Assoc_binary worksheet.



Rules

Rule ID	A-Support	C-Support	Support	Confidence	Lift-Ratio	Antecedent	Consequent
Rule 1	495	846	330	66.66666667	1.5760441	[YouthBks]	[ChildBks]
Rule 2	846	862	512	60.52009456	1.4041785	[ChildBks]	[CookBks]
Rule 3	862	846	512	59.39675174	1.4041785	[CookBks]	[ChildBks]
Rule 4	564	846	368	65.24822695	1.5425113	[DoItYBks]	[ChildBks]
Rule 5	429	846	303	70.62937063	1.6697251	[RefBks]	[ChildBks]
Rule 6	482	846	325	67.42738589	1.594028	[ArtBks]	[ChildBks]
Rule 7	552	846	390	70.65217391	1.6702642	[GeogBks]	[ChildBks]
Rule 8	495	862	324	65.45454545	1.5186669	[YouthBks]	[CookBks]
Rule 9	564	862	375	66.4893617	1.5426766	[DoItYBks]	[CookBks]
Rule 10	429	862	305	71.0955711	1.6495492	[RefBks]	[CookBks]
Rule 11	482	862	334	69.29460581	1.6077635	[ArtBks]	[CookBks]
Rule 12	552	862	385	69.74637681	1.6182454	[GeogBks]	[CookBks]
Rule 13	227	862	227	100	2.3201856	[ItalCook]	[CookBks]
Rule 14	482	564	247	51.24481328	1.817192	[ArtBks]	[DoItYBks]
Rule 15	429	552	221	51.51515152	1.866491	[RefBks]	[GeogBks]
Rule 16	482	552	255	52.90456432	1.916832	[ArtBks]	[GeogBks]
Rule 17	495	512	258	52.12121212	2.0359848	[YouthBks]	[ChildBks,CookBks]
Rule 18	330	862	258	78.18181818	1.8139633	[ChildBks,YouthBks]	[CookBks]
Rule 19	512	495	258	50.390625	2.0359848	[ChildBks,CookBks]	[YouthBks]
Rule 20	324	846	258	79.62962963	1.8824972	[YouthBks,CookBks]	[ChildBks]
Rule 21	564	512	292	51.77304965	2.0223848	[DoItYBks]	[ChildBks,CookBks]
Rule 22	512	564	292	57.03125	2.0223848	[ChildBks,CookBks]	[DoItYBks]
Rule 23	368	862	292	79.34782609	1.8410168	[ChildBks,DoItYBks]	[CookBks]
Rule 24	375	846	292	77.86666667	1.8408195	[CookBks,DoItYBks]	[ChildBks]
Rule 25	429	512	245	57.10955711	2.2308421	[RefBks]	[ChildBks,CookBks]
Rule 26	303	862	245	80.85808581	1.8760577	[ChildBks,RefBks]	[CookBks]
Rule 27	305	846	245	80.32786885	1.899004	[CookBks,RefBks]	[ChildBks]
Rule 28	482	512	253	52.48962656	2.050376	[ArtBks]	[ChildBks,CookBks]
Rule 29	325	862	253	77.84615385	1.8061753	[ChildBks,ArtBks]	[CookBks]
Rule 30	334	846	253	75.74850299	1.7907448	[CookBks,ArtBks]	[ChildBks]
Rule 31	552	512	299	54.16666667	2.1158854	[GeogBks]	[ChildBks,CookBks]
Rule 32	512	552	299	58.3984375	2.1158854	[ChildBks,CookBks]	[GeogBks]
Rule 33	390	862	299	76.66666667	1.778809	[ChildBks,GeogBks]	[CookBks]
Rule 34	385	846	299	77.66233766	1.8359891	[CookBks,GeogBks]	[ChildBks]
Rule 35	368	552	209	56.79347826	2.0577347	[ChildBks,DoItYBks]	[GeogBks]
Rule 36	390	564	209	53.58974359	1.9003455	[ChildBks,GeogBks]	[DoItYBks]
Rule 37	265	846	209	78.86792453	1.8644899	[DoItYBks,GeogBks]	[ChildBks]
Rule 38	325	552	204	62.76923077	2.2742475	[ChildBks,ArtBks]	[GeogBks]
Rule 39	390	482	204	52.30769231	2.1704437	[ChildBks,GeogBks]	[ArtBks]
Rule 40	255	846	204	80	1.891253	[ArtBks,GeogBks]	[ChildBks]
Rule 41	375	482	203	54.13333333	2.2461964	[CookBks,DoItYBks]	[ArtBks]
Rule 42	334	564	203	60.77844311	2.1552639	[CookBks,ArtBks]	[DoItYBks]
Rule 43	247	862	203	82.18623482	1.9068732	[DoItYBks,ArtBks]	[CookBks]
Rule 44	375	552	217	57.86666667	2.0966184	[CookBks,DoItYBks]	[GeogBks]
Rule 45	385	564	217	56.36363636	1.9987105	[CookBks,GeogBks]	[DoItYBks]
Rule 46	265	862	217	81.88679245	1.8999256	[DoItYBks,GeogBks]	[CookBks]
Rule 47	334	552	207	61.9760479	2.245509	[CookBks,ArtBks]	[GeogBks]
Rule 48	385	482	207	53.76623377	2.2309641	[CookBks,GeogBks]	[ArtBks]
Rule 49	255	862	207	81.17647059	1.8834448	[ArtBks,GeogBks]	[CookBks]

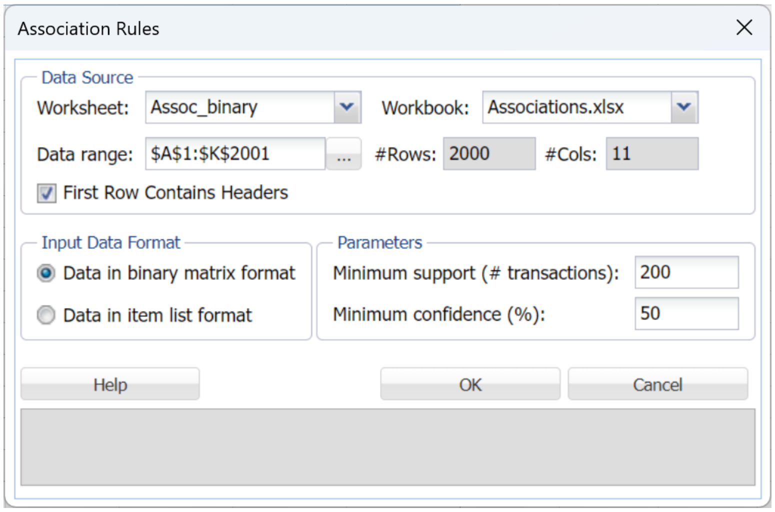
Rule 27 indicates that if a Cook book and a Reference book is purchased, then with 80% confidence a Child book will also be purchased. The *A - Support* indicates that the rule has the support of 305 transactions, meaning that 305 people bought a cook book and a Reference book. The *C - Support* column indicates the number of transactions involving the purchase of Child books. The

Support column indicates the number of transactions where all three types were purchased.

The Lift Ratio indicates how much more likely a transaction will be found where all three book types (Cook, Reference, and Child) are purchased, as compared to the entire population of transactions. In other words, the Lift Ratio is the Confidence divided by the percentage of C-Support transactions in the entire dataset. The percentage of C-Support transactions in the entire dataset for Rule 27 is .423 (846/2000). Confidence is then divided by this value to find the Lift Ratio or 0.803/.423 = 1.899. Given support at 80.3% and a lift ratio of 1.899 (lift ratio > 1), this rule can be considered “useful”.

Association Rules Options

The following options appear on the Association Rules dialog.



Association Rules

Data Source

Worksheet: Assoc_binary

Workbook: Associations.xlsx

Data range: $A$1:$K$2001

#Rows: 2000

#Cols: 11

☑ First Row Contains Headers

Input Data Format

● Data in binary matrix format

○ Data in item list format

Parameters

Minimum support (# transactions): 200

Minimum confidence (%): 50

Data Source

Worksheet: The worksheet name containing the dataset.

Workbook: The workbook name containing the dataset.

Data range: The selected data range.

#Rows: (Read only) The number of rows in the dataset.

#Cols: (Read only) The number of columns in the dataset.

First Row Contains Headers: Select this checkbox if the first row of the dataset contains column headings.

Input data format

Select **Data in binary matrix format** if each column in the data represents a distinct item. If this option is selected, Analytic Solver Data Science treats the data as a matrix of two entities -- zeros and non-zeros. All non-zeros are treated as 1's. So, effectively the data set is converted to a binary matrix which contains 0's and 1's. A 0 indicates that the item is absent in the transaction and a 1 indicates it is present.

Select **Data in item list format** if each row of data consists of item codes or names that are present in that transaction.

Minimum support (# transactions)

Specify the minimum number of transactions in which a particular item-set must appear for it to qualify for inclusion in an association rule here. The default value is 10% of the total number of rows.

Minimum confidence (%)

A value entered for this option specifies the minimum confidence threshold for rule generation. If A is the set of Antecedents and C the set of Consequents, then only those A =>C ("Antecedent implies Consequent") rules will qualify, for which the ratio (support of A U C) / (support of A) is greater than or equal to. The default setting is 50.

Appendix: Six Sigma Functions

Six Sigma Functions

These functions compute values related to the Six Sigma indices used in manufacturing and process control.

SigmaCP

A Six Sigma index, SigmaCP predicts what the process is capable of producing if the process mean is centered between the lower and upper limits. This index assumes the process output is normally distributed.

$$Cp = \frac{UpperSpecificationLimit - LowerSpecificationLimit}{6\hat{\sigma}}$$

where $\hat{\sigma}$ is the estimated standard deviation of the process.

SigmaCPK

A Six Sigma index, SigmaCPK predicts what the process is capable of producing if the process mean is not centered between the lower and upper limits. This index assumes the process output is normally distributed and will be negative if the process mean falls outside of the lower and upper specification limits.

$$Cpk = \frac{MIN(UpperSpecificationLimit - \hat{\mu}, \hat{\mu} - LowerSpecificationLimit)}{3\hat{\sigma}}$$

where $\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaCPKLower

A Six Sigma index, SigmaCPKLower calculates the one-sided Process Capability Index based on the lower specification limit.   This index assumes the process output is normally distributed.

$$Cp, lower = \frac{\hat{\mu} - LowerSpecificationLimit}{3\hat{\sigma}}$$

where $\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaCPKUpper

A Six Sigma index, SigmaCPKUpper calculates the one-sided Process Capability Index based on the upper specification limit.   This index assumes the process output is normally distributed.

$$Cp, upper = \frac{UpperSpecificationLimit - \hat{\mu}}{3\hat{\sigma}}$$

where $\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaCPM

A Six Sigma index, SigmaCPM calculates the capability of the process around a target value. This index is referred to as the Taguchi Capability Index. This index assumes the process output is normally distributed and is always positive.

$$Cpm = \frac{\hat{C}p}{\sqrt{1+(\frac{\hat{\mu}-T}{\hat{\sigma}})^2}}$$

where $\hat{C}p$ is the process capability (SigmaCP), $\hat{\mu}$ is the process mean, $\hat{\sigma}$ is the standard deviation of the process and T is the target process mean.

SigmaDefectPPM

A Six Sigma index, SigmaDefectPPM calculates the Defective Parts per Million.

$$DPMO = (\delta^{-1}(\frac{LowerSpecificationLimit - \hat{\mu}}{\hat{\sigma}}) +$$
$$1 - \delta^{-1}(\frac{UpperSpecificationLimit - \hat{\mu}}{\hat{\sigma}})) * 1000000$$

where$\hat{\mu}$ is the process mean, $\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaDefectShiftPPM

A Six Sigma index, SigmaDefectShiftPPM calculates the Defective Parts per Million with an added shift.

$$DPMOShift = (\delta^{-1}(\frac{LowerSpecificationLimit - \hat{\mu}}{\hat{\sigma}} - Shift) +$$
$$1 - \delta^{-1}(\frac{UpperSpecificationLimit - \hat{\mu}}{\hat{\sigma}} - Shift)) * 1000000$$

where$\hat{\mu}$ is the process mean, $\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaDefectShiftPPMLower

A Six Sigma index, SigmaDefectShiftPPMLower calculates the Defective Parts per Million, with a shift, below the lower specification limit.

$$DPMOshift, lower = (\delta^{-1}(\frac{LowerSpecificationLimit}{\hat{\sigma}} - Shift) * 1000000$$

where$\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaDefectShiftPPMUpper

A Six Sigma index, igmaDefectShiftPPMUpper calculates the Defective Parts per Million, with a shift, above the lower specification limit.

$$DPMOshift, upper = (\delta^{-1}(\frac{UpperSpecificationLimit}{\hat{\sigma}} - Shift) * 1000000$$

where$\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaK

A Six Sigma index, SigmaK calculates the Measure of Process Center and is defined as:

$$1 - \frac{2*MIN(UpperSpecificationLimit-\hat{\mu},\hat{\mu}-LowerSpecificationLimit)}{UpperSpecificationLimit-LowerSpecificationLimit}$$

where$\hat{\mu}$ is the process mean.

SigmaLowerBound

A Six Sigma index, SigmaLowerBound calculates the Lower Bound as a specific number of standard deviations below the mean and is defined as:

$$\hat{\mu} - \hat{\sigma} * \#StandardDeviations$$

where$\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaProbDefectShift

A Six Sigma index, SigmaProbDefectShift calculates the Probability of Defect, with a shift, outside of the upper and lower limits. This statistic is defined as:

$$\delta^{-1}(\frac{LowerSpecificationLimit - \hat{\mu}}{\hat{\sigma}} - Shift) +$$

$$1 - \delta^{-1}(\frac{UpperSpecificationLimit-\hat{\mu}}{\hat{\sigma}} - Shift)$$

where $\hat{\mu}$ is the process mean ,$\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaProbDefectShiftLower

A Six Sigma index, igmaProbDefectShiftLower calculates the Probability of Defect, with a shift, outside of the lower limit. This statistic is defined as:

$$\delta^{-1}(\frac{LowerSpecificationLimit-\hat{\mu}}{\hat{\sigma}} - Shift)$$

where$\hat{\mu}$ is the process mean ,$\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaProbDefectShiftUpper

A Six Sigma index, SigmaProbDefectShiftUpper calculates the Probability of Defect, with a shift, outside of the upper limit. This statistic is defined as:

$$1 - \delta^{-1}(\frac{UpperSpecificationLimit-\hat{\mu}}{\hat{\sigma}} - Shift)$$

where$\hat{\mu}$ is the process mean ,$\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaSigmaLevel

A Six Sigma index, SigmaSigmaLevel calculates the Process Sigma Level with a shift. This statistic is defined as:

$$-\delta(\delta^{-1}(\frac{LowerSpecificationLimit-\hat{\mu}}{\hat{\sigma}}-Shift)+$$
$$1-\delta^{-1}(\frac{UpperSpecificationLimit-\hat{\mu}}{\hat{\sigma}}-Shift))$$

where $\hat{\mu}$ is the process mean ,$\hat{\sigma}$ is the standard deviation of the process $\delta$is the standard normal cumulative distribution function, and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaUpperBound

A Six Sigma index, SigmaUpperBound calculates the Upper Bound as a specific number of standard deviations above the mean and is defined as:

$$\hat{\mu}-\hat{\sigma}*\#StandardDeviations$$

where$\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaYield

A Six Sigma index, SigmaYield calculates the Six Sigma Yield with a shift, or the fraction of the process that is free of defects. This statistic is defined as:

$$\delta^{-1}(\frac{UpperSpecificationLimit-\hat{\mu}}{\hat{\sigma}}-Shift)-$$
$$\delta^{-1}(\frac{LowerSpecificationLimit-\hat{\mu}}{\hat{\sigma}}-Shift)$$

where$\hat{\mu}$ is the process mean, $\hat{\sigma}$ is the standard deviation of the process and δ^{-1} is the standard normal inverse cumulative distribution function.

SigmaZLower

A Six Sigma index, SigmaZLower calculates the number of standard deviations of the process that the lower limit is below the mean of the process. This statistic is defined as:

$$\frac{\hat{\mu}-LowerSpecificationLimit}{\hat{\sigma}}$$

where$\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaZMin

A Six Sigma index, SigmaZMin calculates the minimum of SigmaZLower and SigmaZUpper. This statistic is defined as:

$$\frac{MIN(\hat{\mu}-LowerSpecificationLimit,UpperSpecificationLimit-\hat{\mu})}{\hat{\sigma}}$$

where$\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.

SigmaZUpper

A Six Sigma index, SigmaZUpper calculates the number of standard deviations of the process that the upper limit is above the mean of the process. This statistic is defined as:

$$\frac{UpperSpecificationLimit - \hat{\mu}}{\hat{\sigma}}$$

where$\hat{\mu}$ is the process mean and $\hat{\sigma}$ is the standard deviation of the process.