PUBLIC
SAP Predictive Analytics
2020-08-28

Automated Analytics User Guides and Scenarios

THE BEST RUN SAP

Content

1 About This Guide

This guide provides everything you need to perform your modeling workflows from the modeling concepts to data preparation and end-to-end scenarios. It is a collection of previously independent user guides that have been grouped into a single guide.

2 What's New in Automated Analytics User Guides and Scenarios

Links to information about the new features and documentation changes in the Automated Analytics user guides and scenarios for SAP Predictive Analytics 3.3.

Section	What's New	Link to More Information
Modeler - Time Series	For a trend regressive modeling, an optional weight column can be used to tweak the modeling procedure.	File Format Specifications [page 208]
Modeler - Time Series	You can change the Percentage of Variable Contributions to Keep. However, if the L2 degradation is greater than 5%, your selection is ignored.	Defining the Variables [page 218]
Modeler - Time Series	Terminology updates: Cutting strategy is now Partition strategy, Estimation is now Training, and Test is now testing.	Selecting a Partition Strategy [page 211] Selecting a Partition Strategy [page 212] Understanding the Forecast Plot [page 230] Standard Modeling [page 247] Modeling with Extra Predictable Inputs [page 249] Modeling using KxShell scripts [page 255]

3 Modeling Concepts

3.1 Operation of Automated Analytics: Overview

Automated Analytics allows you to perform supervised data mining, that is, to transform your data into knowledge, then into action, as a function of a domain-specific business issue.

The application supports various formats of source data (flat files, ODBC-compatible sources). In order to be usable by the application features, the dataset to be analyzed must be presented in the form of a single table of data, except in instances where you are using the Event Logging or Sequence Coding features of Data Manager.

To use the application's features, you must have a training dataset available that contains the target variable with all its values defined. Then, you can apply the model generated using the training dataset to one or more application datasets.

The training dataset is cut into three data sub-sets for training, validation and testing, using a partition strategy.

The different types of variables: continuous, ordinal and nominal are next encoded by the data encoding feature of Modeler, or by the Event Logging and Sequence Coding features in the case of dynamic data. Before generating the model, you must:

- Describe the data. A utility integrated with the application allows you to generate a description of the dataset to be analyzed, automatically. You need only validate that description, verifying that the type and storage format of each variable were identified correctly.
- Define the role of variables contained in the dataset to be analyzed. You may select one or more variables as target variables. These are the variables that corresponds with your business issue. The other variables of the table of data are considered to be explanatory variables: they allow calculation of the value of the target variable in a given context. They may also be used as weight variables.

For more information about the role of each feature of the application, see the description of the operational phases.

You can then generate models, capable of either explaining and predicting a phenomenon, or describing a dataset, in both cases, as a function of the previously defined target variable. This phase is called the training phase.

Once the models have been generated, you can view and interpret their relevance and robustness using:

- Performance indicators: the predictive power, which is the quality indicator, and the prediction confidence, which is the robustness indicator.
- A variety of plots, including the profit curve plot.

3.2 Data Sources Supported

Automated Analytics supports the following data sources:

- Text files (also called flat files) in which the data are separated by a delimiter, such as commas in `.csv` (Comma Separated Value) files.

  > **! Restriction**
  >
  > When accessing data in `.csv` files, Automated Analytics only supports `CR` + `LF` (common on Microsoft Windows) or `LF` (common on Linux) for line breaks.

- Database management systems that can be accessed using ODBC.

  > **i Note**
  >
  > For the list of supported ODBC-compatible sources, see the SAP Product Availability Matrix (PAM) at http://service.sap.com/sap/support/pam.
  >
  > For more information about using SAP HANA, see the related information below.
  >
  > To configure Automated Analytics modeling tools to access data in your database management system, refer to the guide *Connecting your Database Management System on Windows* or *Connecting your Database Management System on Linux*.

- SAS files

Related Information

SAP HANA as a Data Source [page 8]

3.2.1 SAP HANA as a Data Source

You can use SAP HANA databases as data sources in Data Manager and for all types of modeling analyses in Modeler: Classification/Regression, Clustering, Time Series, Association Rules, Social, and Recommendation.

SAP HANA tables or SQL views	found in the *Catalog* node of the SAP HANA database

All types of SAP HANA views	found in the *Content* node of the SAP HANA database. An SAP HANA view is a predefined virtual grouping of table columns that enables data access for a particular business requirement. Views are specific to the type of tables that are included, and to the type of calculations that are applied to columns. For example, an analytic view is built on a fact table and associated attribute views. A calculation view executes a function on columns when the view is accessed. **! Restriction** • Analytic and calculation views that use the variable mapping feature (available starting with SAP HANA SPS 09) are not supported. • You cannot edit data in SAP HANA views using Automated Analytics.
Smart Data Access virtual tables	Thanks to Smart Data Access, you can expose data from remote sources tables as virtual tables and combine them with HANA regular tables. This allows you to access data sources that are not natively supported by the application, or to combine data from multiple heterogeneous sources. **⚠ Caution** To use virtual tables as input datasets for training or applying a model or as output datasets for applying a model, you need to check that the following conditions are met: • The in-database application mode is not used. • The destination table for storing the predicted values exists in the remote source before applying the model. • The structure of the remote table, that is the column names and types, must match exactly what is expected with respect to the generation options; if this is not the case an error will occur. **⚠ Caution** In Data Manager, use virtual tables with caution as the generated queries can be complex. Smart Data Access may not be able to delegate much of the processing to the underlying source depending on the source capabilities. This can impact performance.

Prerequisites

You must know the ODBC source name and the connection information for your SAP HANA database. For more information, contact your SAP HANA administrator.

In addition to having the authorizations required for querying the SAP HANA view, you need to be granted the `SELECT` privilege on the `_SYS_BI` schema, which contains metadata on views. Please refer to SAP HANA guides for detailed information on security aspects.

3.2.2 Managing Performance when Using Databases

Before requesting data stored in a Teradata(1), Oracle(2) or SQLServer 2005 database, the application uses a feature, called the **Explain mode**, which categorizes the performances of SQL queries in several classes defined by the user. In order to be as fast and as light as possible, this categorization is done **without actually executing the full SQL query**.

> **i Note**
>
> - (1) For all versions of Teradata.
> - (2) For all versions above and including Oracle 10.

The objective is to allow estimating the workload of the SQL query before executing it and then deciding -- possibly thanks to an IT Corporate Policy-- if the SQL query can actually be used.

For example, an IT Corporate Policy may favor interactivity and then define 3 classes of SQL queries, each with its maximum time:

- *Immediate*: *duration < 1 s*. The query is accepted and executed immediately.
- *Batched*: *1s <= duration < 2 s*. The query is accepted but will be executed on next idle time.
- *Rejected*: *2s <= duration*. The query will never be executed.

The number, names and limits of classes are defined by the user in order for these values to match the current DBMS configuration and DBMS usage policy.

The Explain Mode has been Configured

If the Explain mode has been configured by your DBMS administrator, there are two possible outcomes to a query:

- **the query is accepted and executed**: this is completely transparent. The application accesses the data without further input from the user.
- **the query needs to be validated before being executed**: a pop-up window opens displaying a message configured by the DBMS administrator. A query that needs validation can be categorized in two ways:

  medium-sized  or **huge**  .

If the query is categorized as medium-sized, you will probably have to check with you administrator which action to take:

- If the administrator authorizes the query, click *Continue*. The pop-up window closes and the requested action is carried out.
- If the administrator does not authorize the query, click *Stop Query*, the pop-up window closes, but no action is executed.

If the query is categorized as huge, it means that the query will take too much time and resources. In that case, the behavior of the *Continue* button depends on the configuration set by the DBMS Administrator (for example, it can automatically refuse queries that are considered too heavy). In any case, you should check with them to know the line of action to follow.

The Explain Mode has not been Configured

If your DBMS Administrator has not configured the Explain mode, a pop-up window opens when you try to access the data.

You need to contact your Administrator who will tell you which action to take and configure the Explain mode.

If the Administrator validates the execution of the query, you may want all queries with the same duration to be executed without validation. In that case, check the box *Do not request validation anymore for similar requests*. The validation message will then only appear for larger queries. This configuration will only be used for the current session, when closing the application, it will be lost. For a permanent configuration, see your DBMS Administrator.

3.3 Training and Application Dataset

To use the application's features, you must have a training dataset available that contains the target variable with all its values defined. Then, you can apply the model generated using the training dataset to one or more application datasets.

Training Dataset

A training dataset is a dataset used for generating a model. In this set, the values of the target variable – or variable corresponding to your business issue – are known. By analyzing the training dataset, the application features will generate a model that allows explanation of the target variable, based on the explanatory variables.

To allow validation of the model generated, the training dataset is cut into three sub-sets using a partition strategy.

The training dataset may correspond to either a complete population section of your database or a sample extracted from this population. The choice depends on the type of study to be performed, the tools used and the budget allocated to the study.

Application Dataset

An application dataset is a dataset to which you apply a model. This dataset contains an unknown target variable for which you want to know the value.

The model applied to the application dataset must have been previously generated from a training dataset. The application dataset must contain exactly the same information structure as the corresponding training dataset, that is:

- The same number of variables,
- The same types of variables,
- The same order of presentation of these variables.

> ⚠ Caution
>
> The application dataset must contain a target variable that corresponds to that of the training dataset. This point is true for all instances, even if the values of this target variable are empty. When these values are defined, they may serve to detect any possible deviant observations (outliers).

3.4 Partition Strategies

A partition strategy is a technique that allows decomposition of a training dataset into three distinct sub-sets:

- A training sub-set
- A validation sub-set
- A testing sub-set

This partition allows for cross-validation of the models generated.

The following table defines the roles of the three data sub-sets obtained using partition strategies.

The dataset	Is used to...
Training	Generate different models. The models generated at this stage are hypothetical.
Validation	Select the best model among those generated using the training sub-set, which represents the best compromise between perfect quality and perfect robustness.
Testing	Verify the performance of the selected model on a new dataset.

To understand the role of partition strategies in the model generation process, see the related topic on generating a model.

To generate your models, there are two types of partition strategies that you may use:

- The customized partition strategy
- The automatic partition strategies

3.4.1 Customized Partition Strategy

The customized partition strategy allows you to define your own data sub-sets. To use this strategy, you must have prepared (before opening the application features) three sub-sets: the training, validation and testing sub-sets.

Before opening the application, cut your initial data file into three files of the size of your choice. For example:

- The first file may contain the first 1,500 observations, or lines, of your initial data file,
- The second file, observations 1,501 to 3,000,
- The third file, observations 3,001 to 5,000.

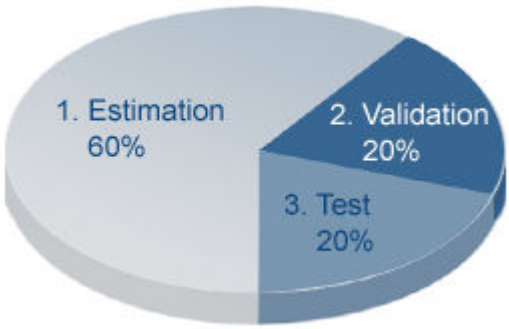
> ⚠ Caution
>
> The customized partition strategy is risky in the instance of an initial data file in which the data have been sorted. In this case, the first lines will not be representative of the overall set of data contained in the first file. To avoid this type of bias, do not forget to mix up your data prior to analysis.

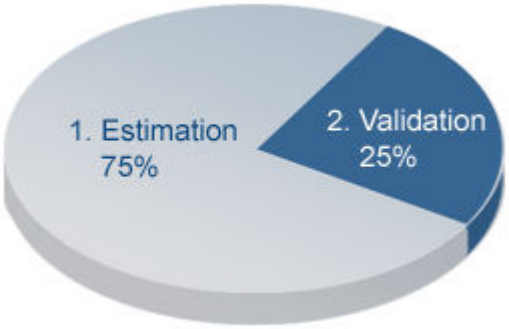
3.4.2 Automatic Partition Strategies

With the exception of the customized partition strategy, partition strategies are automatic. Automatic partition strategies operate upon a single data file, which constitutes your initial dataset.

Automatic partition strategies always cut the initial dataset into the same proportions. The following table details the proportions attributed to each dataset depending on the presence of a testing dataset.

Automatic Partition Strategies with Testing	Automatic Partition Strategies without Testing
• 3/5 of the data are used in the training sub-set, • 1/5 of the data are used in the validation sub-set, • 1/5 of the data are used in the testing sub-set.	• 3/4 of the data are used in the training sub-set, • 1/4 of the data are used in the validation sub-set.





There are seven types of automatic partition strategies available within the application.The random partition strategies distribute the data of the initial dataset in a random manner between the three sub-sets: training, validation and testing.

Random with Testing at the End

The Random with testing at the end partition strategy distributes:

- 4/5 of the initial dataset in a random manner in the two sub-sets training and validation, 3/5 being distributed in the training data sub-set and 1/5 in the validation data sub-set.
- The final 1/5 of the initial dataset is sent directly into the testing sub-set.

This is a useful strategy in cases where:

- Your database corresponds to a well-defined evolution because of the way it was built, which may mean, for example, that the data is in chronological order,
- You may wish to take this order into account when generating your model.

For example, imagine that:

- New customers are added every month to your database,
- You know that the datasets to which you apply the model will, once generated, have a better chance of resembling the most recent section of your database, that is, the section that contains the most recent customers entered.

Using the Random with testing at the end partition strategy, you decide to test the model generated on that section of your database that is most likely to resemble the state of your future application datasets.

Random Without Testing (Default Strategy)

The Random without testing strategy is the partition strategy suggested as the default setting. It distributes the whole initial dataset in a random manner to the two sub-sets of training and validation:

- 3/4 of the initial dataset are distributed to the training sub-set,
- 1/4 to the initial dataset are distributed to the validation sub-set.

As no testing sub-set is used, all the data from your training dataset can be used for sub-sets of training and validation. This can lead to a model with a better quality and robustness.

Periodic

The Periodic partition strategy is implemented by following this distribution cycle:

1. Three lines of the initial dataset are distributed to the training sub-set.
2. One line is distributed to the validation sub-set.
3. One line is distributed to the testing sub-set.
4. Distribution begins again at step 1 .

Periodic with Testing at the End

The Periodic with testing at the end strategy distributes:

- 4/5 of the initial dataset in a periodic manner to the two sub-sets of training and validation, 3/5 being distributed in the training data sub-set and 1/5 in the validation data sub-set 3/5 being distributed.
- The final 1/5 of the initial dataset is sent as a block of data to the testing sub-set.

In other words, this strategy follows this distribution cycle:

1. Three lines of the first 4/5 of the initial dataset are distributed to the training sub-set.
2. One line of the first 4/5 of the initial dataset is distributed to the validation sub-set.
3. If the entire 4/5 of the initial dataset is not yet distributed, distribution operations begin again at step 1. Otherwise, if the entire 4/5 of the initial dataset has been distributed, distribution operations go to step 4.
4. The final 1/5 of the initial dataset is sent as a block of data to the testing sub-set.

Periodic Without Testing

The Periodic without testing strategy distributes the whole initial dataset in a periodic manner to the two sub-sets of training and validation:

- 3/4 of the initial dataset are distributed to the training sub-set,
- 1/4 to the initial dataset are distributed to the validation sub-set.

In other words, this partition strategy is implemented by following this distribution cycle:

1. Three lines of the initial dataset are distributed to the training sub-set.
2. One line is distributed to the validation sub-set.
3. Distribution begins again at step 1.

As no testing sub-set is used, all the data from your training dataset can be used for sub-sets of training and validation. This can lead to a model with a better quality and robustness.

Sequential

The Sequential strategy cuts the initial dataset into three blocks, corresponding to the usual partition proportions:

- The lines corresponding to the first 3/5 of the initial dataset are distributed as a block to the training dataset,
- The lines corresponding to the next 1/5 of the initial dataset are distributed as a block to the validation dataset,
- The lines corresponding to the final 1/5 of the initial dataset are distributed as a block to the testing dataset.

Sequential without Testing

The Sequential without testing strategy cuts the initial dataset into two blocks:

- The lines corresponding to the first 3/4 of the initial dataset are distributed as a block to the training dataset,
- The lines corresponding to the next 1/4 of the initial dataset are distributed as a block to the validation dataset.

As no testing sub-set is used, all the data from your training dataset can be used for sub-sets of training and validation. This can lead to a model with a better quality and robustness.

3.5 Table of Data

A table of data is a dataset presented in the form of a two-dimensional table.

In this table:

- Each row represents an observation to be processed, such as "American individual" in the sample file Census01.csv .
- Each column represents a variable that describes observations, such as the "age" or the "gender" of individual Americans.
- Each cell, the intersection of a column and a row, represents the value of the variable in the column, for the observation in that row.

The following table is an example of a table of data.

Observations	Variable 1	Variable 2	Variable 3
Observation a	Value a1	Value a2	Value a3
Observation b	Value b1	Value b2	Value b3
...	...	...	...
Observation n	Value n1	Value n2	Value n3

Synonyms of "Observation" and "Variable"

Depending upon your profile and your area of expertise, you may be more familiar with other terms that refer to observations (in rows) and variables (in columns) when using tables of data.

The following table presents such terms, or synonyms.

Terms equivalent to the term "Observation"	Terms equivalent to the term "Variable"
Row	Column
Record	Attribute
Table	Field
Event	Property
Instance	-
Example	-

Data Formats

Whatever the data source used, the following two constraints must be accommodated:

- The data must be represented in the form of a single table, except in instances where you are using the Event Logging or Sequence Coding features.
- The target variable must be defined for each observation in the table. In the sample file Census01.csv , the variable "class" has been be defined for each individual.

> **i Note**
>
> For the list of supported ODBC-compatible sources, see the SAP Product Availability Matrix (PAM) at http://service.sap.com/sap/support/pam.

3.6 Variables

A variable corresponds to an attribute which describes the observations stored in your database. For example, in a database containing information about your customers, the "name" and "address" of those customers are examples of variables. In Automated Analytics, a variable is defined by three aspects:

- The type of variable:

Continuous	Variables whose values are numerical, continuous, and sortable. They can be used to calculate measures (such as mean or variance).
Ordinal	Variables with discrete values that are sortable.
Nominal	Variables with discrete values that are not sortable.
Textual	A type of nominal variable containing phrases, sentences or complete texts. Textual variables are used for text analyses.

- The storage format of the variable (for example, date, number, or string)
- The role of the variable: Target, Explanatory, or Weight

3.6.1 Continuous Variables

Continuous variables are variables whose values are numerical, continuous and sortable. Arithmetic operations may be performed on these values, such as determination of their sum or their mean.

During modeling, a continuous variable may be grouped into significant discrete bins.

Example

The variable "salary" is a numerical variable, but in addition, is also a continuous variable. It may, for instance, take on the following values: "$1,050", "$1,700", or "$1,750". The mean of these values may be calculated.

3.6.2 Ordinal Variables

Ordinal variables are variables with discrete values, that is, they belong to categories, and they are sortable. Ordinal variables may be:

- Numerical, meaning that its values are numbers. They are therefore ordered according to the natural number system (0, 1, 2, and so on).
- Textual, meaning that its values are character strings. They are therefore ordered according to alphabetic conventions.

Example

The variable "school grade" is an ordinal variable. Its values actually belong to definite categories and can be sorted. This variable can be :

- numerical, if its values range between "0" and "20",
- textual , if its values are A, B, C, D, E et F.

> ⚠ Caution
>
> A variable "assessment" which can have the values "good", "average", and "bad", cannot be directly treated as an ordinal variable by the application. The values would be sorted in alphabetical order ("average","bad", "good") and not according to their meaning. When a nominal variable order is important, the variable must be encoded, in letters or numbers, before it can be used by the application.

3.6.3 Nominal Variables

Nominal variables are variables whose values are discrete, that is, belong to categories, and are not sortable.

Nominal variables may be:

- Numerical, meaning that its values are numbers.
- Textual, meaning that its values are character strings.

> ⚠ Caution
>
> Binary variables are considered nominal variables.

During modeling, the values of the categorical variables are regrouped into homogeneous categories. These categories are then ordered as a function of their relative contribution with respect to the values of the target variable.

Example

The variable "zip code" is a nominal variable. The set of values that this variable may assume ("10111", "20500", "90210", for example) are clearly distinct, non-ranked categories, although they happen to be represented by numbers.

The variable "eye color" is a nominal variable. The set of values that this variable may assume ("blue", "brown", "black", for example) are clearly distinct, non-ordered categories, and are represented by character strings.

3.6.4 Variable Storage Formats

To describe the data, the application uses four types of storage formats:

- date,
- datetime,
- number,
- integer,
- string.

The following table describes these storage formats.

The storage format...	Is used to describe variables when their values correspond to...	For instance...
date	Dates expressed in the following formats: • YYYY-MM-DD • YYYY/MM/DD	• "2001-11-30" • "1999/04/28"
datetime	Dates and times expressed in the following formats: • YYYY-MM-DD HH:MM:SS • YYYY/MM/DD HH:MM:SS	• "2001-11-30 14:08:17" • "1999/04/28 07:21:58"

The storage format...	Is used to describe variables when their values correspond to...	For instance...
number	Figures, or numerical values on which operations may be performed	• The variable "salary", in US dollars: "1000.00", "1593" and "2000.54" •
integer	Figures, or numerical integer values on which operations may be performed	• The variable "age", in years: "21", "34" and "99"
string	Alphanumeric character strings	• The variable "family name": "Lake", "Martin" and "Miller" • The variable "occupation": "professor", "engineer" and "translator" • The variable "telephone": "800 555 1234" and "800 555 4321"

> **i Note**
>
> A variable that has numbers for values is not forced to be described using the number storage format. For instance, the variables "telephone" and "zip code" may instead be described using the string storage format, because no arithmetic operations that make any sense can be performed on these values. Similarly, a variable that will be used as an observation identification code in a table, and does not comply with supported number formats may be described using the string storage format.

> **⚠ Caution**
>
> For number storage formats, the decimal separator used must be a decimal point, and not a comma. So, the value "6.5" may be processed, while "6,5" will not be processed.

Date and Datetime Variables: Automatically Generated Variables

When your dataset contains date or datetime variables, the application automatically extracts date information as shown in the following tables.

For date or datetime variables:

Temporal Information	Represents...	Generated Variable Name
Day of week	the day of week according to the ISO disposition (Monday=0 and Sunday=6)	<OriginalVariableName>_DoW
Day of month	the day of month (1 to 31)	<OriginalVariableName>_DoM
Day of year	the day of the current year (1 to 366)	<OriginalVariableName>_DoY
Month of quarter	the month of the quarter (January, April, July and October = 1, February, May, August and November = 2, March, June, September and December = 3)	<OriginalVariableName>_MoQ

Temporal Information	Represents...	Generated Variable Name
Month of year	the month (1 to 12)	<OriginalVariableName>_M
Year	the year	<OriginalVariableName>_Y
Quarter	the quarter of the year (January to March = 1, April to June = 2, July to September = 3, October to December = 4)	<OriginalVariableName>_Q

For datetime variables:

Temporal Information	Represents...	Generated Variable Name
Hour	the hour	<OriginalVariableName>_H
Minute	the minute	<OriginalVariableName>_Mi
Second	the second	<OriginalVariableName>_S
µ seconds	the micro-second	<OriginalVariableName>_mu

The generated variables will appear in the model debriefing panels listing variables, such as the Contributions by Variable , the Category Significance, the Statistical Reports, as well as in the automatic variable selection feature.

3.6.5 Target Variable

A target variable is the variable that you seek to explain, or for which you want to predict the values in an application dataset. It corresponds to your domain-specific business issue.

When the target variable is a binary variable, the application considers that the target value, or target category, of this variable (that is, the value that is the object of the analysis) to be the least frequently occurring value in the training dataset. Imagine that a training dataset containing the customer information of a company contains the target variable "responded to my mailing". This target variable may take the values "Yes" or "No". If the value "Yes" is the least frequent value (for instance, if 40% of referenced customers responded to the mailing), the application considers that value to be the target category of the target variable.

Synonyms

Depending upon your profile and your area of expertise, you may be more familiar with one of the following terms to refer to target variables:

- Variables to be explained,
- Dependent variables,
- Output variables.

These terms are synonyms.

Constraints Governing Use

The following constraints govern the use of a target variable:

- Within a training dataset, all target variable values must be known.
- Only binary or continuous variables may be used as target variables.

Example

Your company is marketing two products A and B.

You have a database which contains references to:

- 1,500 of your customers. You know which product, A or B, each customer has purchased.
- 10,000 prospects. You want to know which product each customer is likely to purchase.

The variable "product purchased" is your target variable: it corresponds to your business issue. It is:

- Known for all values of the training dataset (in our example, the customers),
- Not known for the values of the application dataset (in our example, the prospects).

The application features allow you to model that target variable, and thus predict which product each of your prospects is likely to purchase.

The following table represents your database.

Name	Age	Residence	Socio-Occupational Category	Product Purchased
Charles	34	New Orleans	Manager/Administrator	Product A
John	37	Washington	Manager/Administrator	Product A
Marlène	31	Boston	Civil servant	Product B
Prospect 1	34	Oakland	Manager/Administrator	?
Prospect 2	24	Washington	Civil servant	?
...	...	...	...	...
Prospect n	35	Sacramento	Skilled tradesman	?

3.6.6 Explanatory Variable

An explanatory variable is a variable that describes your data and which serves to explain a target variable.

Synonyms

Depending upon your profile and your area of expertise, you may be more familiar with one of the following terms to refer to explanatory variables:

- Causal variables,
- Independent variables,
- Input variables.

These terms are synonyms.

Example

Your company is marketing two products A and B.

You have a database which contains references to:

- 1,500 of your customers. You know which product, A or B, each customer has purchased.
- 10,000 prospects. You want to know which product each prospect is likely to purchase.

The variables "name", "age", "address" and "socio-occupational class" are your explanatory variables: they allow you to generate a model capable of explaining and predicting the value of the target variable "product purchased".

The following table represents your database.

Name	Age	Adress	Socio-Occupational Class	Product Purchased
Charles	34	New Orleans	Manager/Administrator	Product A
John	37	Washington	Manager/Administrator	Product A
Marlene	31	Boston	Civil servant	Product B
Prospect 1	34	Oakland	Manager/Administrator	?
Prospect 2	24	Washington	Civil servant	?
...	...	...	...	...
Prospect n	35	Sacramento	Skilled tradesman	?

3.6.7 Weight Variable

A weight variable allows one to assign a relative weight to each of the observations it describes, and actively orient the training process. To declare a variable a weight variable results in creating a number of copies of each of the dataset observations, proportional to the value they possess for that variable.

Specifying a weight variable can be used:

- either to assign a higher weight to a single line
- or to do stratified sampling.

The effect of the weight can be considered as the following: a line with a weight of two in the training dataset is exactly equivalent to having two identical lines with a weight of one.

Constraints Governing Use

Only positive continuous variables may be used as weight variables.

Example

Imagine a dataset in which the observations correspond to individual Americans. These observations are described by the variable "age", among others. Defining the variable "age" as a weight variable means that for generation of the model, older individuals will be weighted more heavily than younger individuals.

3.7 Models

The term model carries many different meanings depending on its field of application. In data mining, a model describes and explains the relationships that exist between input data (explanatory variables) and output data (one or more target variables). It allows one to predict and explain phenomena, or to describe them.

To quote George E.P. Box, "All Models Are Wrong But Some Are Useful" from *"Robustness is the Strategy of Scientific Model Building" in Robustness in Statistics. Eds., R.L. Launer and G.N. Wilkinson, 1979, Academic Press*.

A model that has satisfactory performance is one that possesses both:

- High explanatory power, that is, sufficient capacity to explain the target variable. This explanatory power is indicated by the predictive power of the model.
- High robustness, that is, sufficient capacity to repeat the same performance on new datasets containing observations of a similar nature to the training dataset. This explanatory power is indicated by the prediction confidence.

In data mining, there are two types of models:

- Predictive and explanatory models, which allow one to predict and explain phenomena,
- Descriptive models, which allow one to describe datasets.

With the application, you can generate models that are both highly descriptive and highly predictive.

3.7.1 Generating a Model

The analytics are generated during a phase called the training phase, using a training dataset.

Depending on the situation, the training dataset may be cut into three sub-sets:

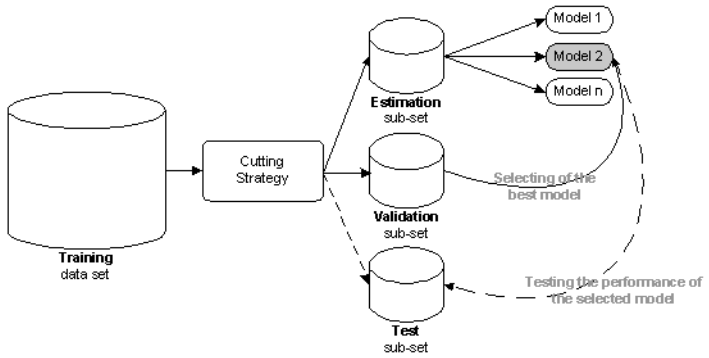
- A Training sub-set.
- A Validation sub-set.
- A Testing sub-set.

A partition strategy determines the way in which the data of the training dataset are distributed across the sub-sets.

The Training and Validation sets are used for actual training, and the Testing set (sometimes referred to as the "hold-out sample") is used to ensure that the predicted performance is correct.

> **i Note**
>
> When using the application, the data sub-sets are virtual. The file corresponding to the initial dataset remains intact at all times.

The figure below illustrates the model generation process, known as the training phase.



3.7.2 Representation of a Model

A model may be represented in many different ways, including:

- a decision tree,
- a neural network,
- a mathematical function.

In the application, models are represented in the form of mathematical functions, specifically, polynomials.

Description of the Polynomial

A polynomial may be of degree 1, 2, 3 or greater. By defining the polynomial degree, you are defining the degree of complexity of the model.

Examples of Polynomials

A polynomial of degree 1 is of the form:

```
f(X1, X2, ..., Xn) = w0 + w1.X1 + w2.X2 + ... + wn.Xn
```

A polynomial of degree 2 is of the form:

```
f(X1, X2, ..., Xn) =
 w0 + w1.X1 + w2.X2 + ... + wn.Xn +
 w11X1.X1 + w12.X1.X2 + w13.X1.X3 + ... + wij.Xi.Xj
```

Methodology

In the large majority of cases, a first degree polynomial is sufficient for generation of a relevant and robust model.

Using a higher degree of polynomial does not always guarantee better results than those obtained with a first degree polynomial. In addition, the higher the degree of polynomial you select:

- The more time needed to generate the corresponding model,
- The more time needed to apply the model to new datasets,
- The harder it is to interpret the results of modeling.

The selection of the degree of the polynomial depends on the nature of the data to be analyzed. The recommended method is to:

- First generate a model with a first order model. In the large majority of cases, this degree will be sufficient to guarantee a relevant and robust model.
- Test the results thus obtained with models of greater degree, if the performance of the first order model seems inadequate.

3.7.3 Validating the Model

Once the model has been generated, you must verify its validity by examining the performance indicators:

- The predictive power allows you to evaluate the explanatory power of the model, that is, its capacity to explain the target variable when applied to the training dataset. A perfect model possesses a predictive power of 1 and a completely random model possesses a predictive power of 0.
  No minimum threshold is required for the predictive power of a model. This depends upon the context of your work, that is, your domain of application, the nature of your data and your business issue.
  In some cases, a model with a predictive power as low as 0.1 may allow realization of a profit of several thousands dollars. In all cases, a positive predictive power indicates that the model generated will perform better than a random model.
- The prediction confidence defines the degree of robustness of the model, that is, its capacity to achieve the same explanatory power when applied to a new dataset. In other words, the degree of robustness corresponds to the predictive power of the model when applied to an application dataset.

A model with a prediction confidence inferior to 0.95 must be considered with caution. The performance of such a model is very likely to vary between the training dataset and the application datasets.

To discover how these indicators are calculated, see the related topic on Predictive Power, Prediction Confidence and Model Graphs.

> **i Note**
>
> Validation of the model is a critically important phase in the overall process of data mining. Always be sure to assign significant importance to the values obtained for the predictive power and the prediction confidence of a model.

3.7.4 How to Obtain a Better Model

Obtaining a better model is achieved by:

- Improving the prediction confidence of the model, or
- Improving the predictive power of the model, or
- Improving both the predictive power and he prediction confidence of the model.

Several techniques allow you to improve these indicators:

- You can increase the degree of complexity of the model (polynomial degree).
- The following table presents other techniques.

To improve...	You can...
The predictive power of a model	○ Add variables to the training dataset ○ Use combinations of explanatory variables that seem relevant to you
The prediction confidence of a model	Add observations to the training dataset

For more information about improving the predictive power and the prediction confidence, see the related topic.

3.8 Performance Indicators

Two indicators specific to Automated Analytics allow you to evaluate the performance of a model.

- The predictive power, which is the quality indicator of the model,
- The prediction confidence, which is the robustness indicator.

Three other indicators, commonly used in data mining, are provided to assess a model:

- the GINI index,
- the K-S
- the AUC

For clustering models, you can evaluate your model checking the performance indicators per cluster or per dataset:

- Performance indicators by cluster:
  - Residual Sum of Squares (RSS)
  - Within Cluster Variance
  - Simplified Silhouette
- Performance indicators by dataset:
  - Sum of Residual Sum of Squares (RSSSum)
  - Within Cluster Variance
  - Simplified Silhouette

3.8.1 Quality Indicator: Predictive Power

The predictive power of a model is the quality indicator of models generated using the application. This indicator corresponds to the proportion of information contained in the target variable that the explanatory variables are able to explain.

To improve the predictive power of a model, new variables may be added to the training dataset. Explanatory variables may also be combined

Example

A model with a predictive power of:

- "0.79" is capable of explaining 79% of the information contained in the target variable using the explanatory variables contained in the dataset analyzed.
- "1" is a hypothetical perfect model, capable of explaining 100% of the target variable using the explanatory variables contained in the dataset analyzed. In practice, such a predictive power would generally indicate that an explanatory variable 100% correlated with the target variable was not excluded from the dataset analyzed.
- "0" is a purely random model.

3.8.2 Robustness Indicator: Prediction Confidence

The prediction confidence is the robustness indicator of the models generated using the application. It indicates the capacity of the model to achieve the same performance when it is applied to a new dataset exhibiting the same characteristics as the training dataset.

To improve the prediction confidence of a model, additional observation rows may be added to the training dataset.

Example

A model with a prediction confidence:

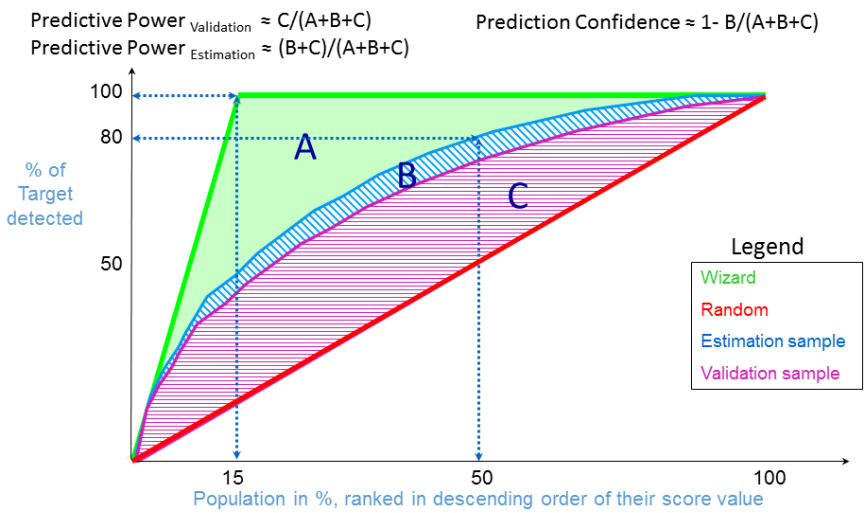
- Equal to or greater than "0.98" is very robust. It has a high capacity for generalization.
- Less than "0.95" must be considered with caution. Applying it to a new dataset will incur the risk of generating unreliable results.

3.8.3 Predictive Power, Prediction Confidence and Model Graphs

On the model graph plot:

- Of the validation dataset (default plot), the predictive power corresponds to "the area found between the curve of the model generated and that of the random model" divided by "the area found between the curve of the perfect model and that of the random model". As the curve of the generated model approaches the curve of the perfect model, the value of the predictive power approaches 1.
- Of the training and validation datasets (select the corresponding option from the list Dataset, located below the plot), the prediction confidence corresponds to one minus "the area found between the curve of the training dataset and that of the validation dataset "divided by "the area found between the curve of the *perfect model* and that of the random model".

The following graph displays the predictive power and the prediction confidence:



3.8.4 Advanced Users: Predictive Power for Continuous Targets

The regression (continuous target case) uses the predictive power (KI) for model selection. If we suppose that we want to compute both the predictive power of a score variable rr_T with respect to its target T on a

Validation dataset, then, we will refer to target categories by $T_j, j = 1 \dots B_T$. Thus, we note:

$$U_j = mean\left(T_j\right), j = 1 \dots B_T$$

$$f_j = frequency\left(T_j\right), j = 1 \dots B_T$$

The target segments are given in decreasing natural order, so that $(u_1 > u_2 > \cdots > u_{BT})$ Let μ be the global mean of the target T on the Validation dataset.

We also refer to the score categories/segments by $S_j\ for\ j = 1 \dots B_S$ and note:

$$m_j = target\ mean\left(S_j\right)\ for\ j = 1 \dots B_S$$

$$F_j = frequency\left(S_j\right), j = 1 \dots B_s$$

The wizard curve is given by the cumulative profits of the target as a function of cumulative frequencies; it is defined by the following points:

$$\left(\sum_{j=1}^{j=b} fj, \sum_{j=1}^{j=b} fj\,(uj - u)\right)\ for\ b = 1 \dots B_T|$$

This curve is normalized such that its maximum is equal to 1.

The Validation curve is given by the cumulative score profits as a function of cumulative frequencies:

$$\left(\sum_{j=1}^{j=b} Fj, \sum_{j=1}^{j=b} Fj\,(mj - u)\right)\ for\ b = 1 \dots B_S$$

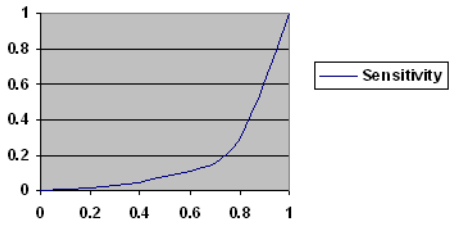
As usual, the Predictive Power value is computed from the wizard and the Validation curve areas. For example, areas can be computed using a trapezoidal rule.

> **i Note**
>
> The nominal target case can be viewed as a special case where the notion of the profit is the positive rate (which is equivalent to the binary target mean in this case).

3.8.5 Gini Index

The Gini statistic is a measure of predictive power based on the Lorenz curve. It is proportionate to the area between the random line and the model curve.

The Gini index is defined as the area under the Lorenz curve. The Gini index is the area between the 'Trade-off' curve and the obtained curve multiplied by 2. This is often pictured as the following chart:



The horizontal axis 'grows' with the score and can be associated with 1-f.;

This is simply expressed using our notations as:

$$\text{GINI} = 2\left(\frac{1}{2} - \int_{-\infty}^{+\infty}(1 - a(t))d\left((1 - a(t)) * P_G + \beta(t) * (1 - P_G)\right)\right)$$

$$\text{GINI} = 2\,(1 - P_G)\left(\frac{1}{2} - (1 - AUC)\right) = (1 - P_G)(2\,AUC - 1)$$

Using these notations, we know that the Gini index of a random model is 0 and for a perfect model is $1 - P_G$.

3.8.6 K-S

K-S is the Kolmogorov-Smirnov statistic applied here as a measure of deviation from uniform response rates across categories of a variable. Kolmogorov-Smirnov is a non-parametric, exact goodness-of-fit statistic based on the maximum deviation between the cumulative and empirical distribution functions

In the case of a binary classification task, people are interested by the difference between the Lorenz curve for the good cases $1\text{-}\alpha$, and the Lorenz curve for the bad cases β when selecting an increasing ratio of population. These curves evolve from 0 to 1 together, and the K-S statistics is the maximum deviation between these two curves. For a perfect system, the K-S statistics is 1 and that for a random system, because of the equality between the two curves, the K-S statistics is 0.

> → Tip
>
> The K-S is used to calculate the difference between two distributions in order to have an idea about the quality of a dataset.

3.8.7 AUC

The Area Under the Roc Curve (AUC) statistic is a rank-based measure of model performance or predictive power calculated as the area under the Receiver Operating Characteristic (ROC) curve. For a simple scoring model with a binary target, this represents the observed probability of a signal (responder) observation having a higher score than a non-signal (non-responder) observation. For individual variables, ordering based on score is replaced by ordering based on the response probability for the variable's categories (for example cluster ID or age range response rates).

The corresponding equation is:

$$\boldsymbol{AUC} = \int_{+\infty}^{-\infty} \boldsymbol{x(t)\, d\big(1-\beta(t)\big)} = \int_{0}^{1} \boldsymbol{x(y)dy}$$

So we have:

$$\boldsymbol{AUC_{perfect} = 1}$$

$$\boldsymbol{AUC_{random}} = \int_{0}^{1} \boldsymbol{(y)dy} = \left[\frac{x^2}{2}\right]_2^1 = \frac{\mathbf{1}}{\mathbf{2}}$$

One of the interests of the measure of AUC is its independence from the target distribution. For example, if we build another dataset and duplicate each good example twice, the AUC of the model will be the same.

> **i Note**
>
> AUC has good properties for evaluating a binary classification system. It is widely used by statisticians.

3.8.8 Error Indicators

First, some basic notations:

- Target (response value): $\boldsymbol{\gamma_i}$
- Predictor (predictor response value): $\boldsymbol{\hat{\gamma}_i}$
- Residual: $r_i = \gamma_i - \hat{\gamma}_i$
- Error: $u_i = |\gamma_i - \hat{\gamma}_i| = |r_i|$
- Weight of the tested observation: $\boldsymbol{w_i}$
- Total weight of the population: $$W = \sum_{i=1}^{n} \mathrm{w}_i$$

- Target average: $$\bar{\gamma} = \frac{1}{W}\sum_{i=1}^{N} \mathrm{w}_i\, \gamma_i$$
- Predictor average: $$\bar{\gamma} = \frac{1}{W}\sum_{i=1}^{N} \mathrm{w}_i\, \hat{\gamma}_i$$

Mean Absolute Error (L1)

Definition: mean of the absolute values of the differences between predictions and actual results, (City block distance or Manhattan distance)

Formula:

$$L1 = \frac{1}{W}\sum_{i=1}^{N} \mathrm{w}_i\, u_i$$

Root Mean Square Error (L2)

Definition: square root of the mean of the quadratic errors (Euclidian distance or root mean squared error - RMSE)

Formula:

$$RMSE = \sqrt{\frac{SSE_W}{W}} = \sqrt{\frac{1}{W}\sum_{i=1}^{N} w_i u_i^2}$$

Maximum Error (LInf)

Definition: maximum absolute difference between predicted and actual values (upper bound) (Chebyshev distance)

Formula:

$$L\infty = \max_i u_i$$

Error Mean

Definition: mean of the difference between predictions and actual values

Formula:

- Mean Percent Error (MPE):

$$MPE = \frac{1}{W}\sum_{i=1}^{N} w_i \frac{y_i - \hat{y}_i}{y_i}$$

- Mean Absolute Percent Error (MAPE):

$$MAPE = \frac{1}{W}\sum_{i=1}^{N} w_i \frac{|y_i - \hat{y}_i|}{|y_i|}$$

- Symmetric Mean Absolute Percentage Error (SMAPE):

$$SMAPE = \frac{2}{W}\sum_{i=1}^{N} w_i \frac{|y_i - \hat{y}_i|}{|y_i| + |\hat{y}_i|}$$

Error Standard Deviation

Definition: dispersion of errors around the actual result

Formula:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (u_i - \bar{u})^2}$$

where

$$\bar{u} = \frac{1}{N}\sum_{i=1}^{N} u_i$$

Classification Rate

Definition: ratio between the number of correctly classified records and the total number of records

Formula:

$$\bar{u} = \frac{1}{N}\sum_{i=1}^{N} u_i$$

Determination Coefficient (R2)

Definition: ratio between the variability (sum of squares) of the prediction and the variability (sum of squares) of the data.

Formula:

$$SSR = \sum_{i=1}^{N} W_i(\hat{\gamma}_u - \overline{\gamma})^2$$

$$SST = \sum_{i=1}^{N} w_i(\gamma_i - \overline{\gamma})^2$$

$$R2 = \frac{SSR}{SST}$$

Maximum Absolute Percentage Error (MaxAPE)

Definition: largest absolute percentage error over all records.

Formula:

$$MaxAPE = \max_{1 \le i \le N} \frac{|\gamma_i - \hat{\gamma}_i|}{|\gamma_i|}$$

Proportion of Absolute Percentage Errors under 5% (APEUnder5Pct)

Definition: proportion of the absolute percentage errors that stand below 5%.

Formula: $APEUnder5Pct = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{[0,5\%]}\left(\frac{|\gamma_i - \hat{\gamma}_i|}{|\gamma_i|}\right)$

Where $\mathbb{1}$ is the indicator function: $\mathbb{1}_A(x) = \begin{cases} 1 \text{ if } x \in A \\ 0 \text{ if } x \notin A \end{cases}$

Proportion of Absolute Percentage Errors under 10% (APEUnder10Pct)

Definition: proportion of the absolute percentage errors that stand below 10%.

Formula: $APEUnder10Pct = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{[0,10\%]}\left(\frac{|y_i - \hat{y}_i|}{|y_i|}\right)$

3.8.9 Clustering Performance Indicators

You can check the clustering performance indicators by cluster and aggregated by dataset.

Here are some preliminary information:

- C_k is the k-th cluster.
- x_i is an individual belonging to C_k.
- w_i is the weight of x_i . It's either the value from the "weight" column of the dataset or it's 1 by default.
- μ_k is the centroid of C_k.
- $W_k = \sum_{x_i \in C_k} w_i$ is the total weight of C_k.
- $W = \sum_k W_k$ is the total weight of the dataset.
- $d(x_i, x_j)$ is the distance function (Euclidian, Manhattan, etc) between x_i and x_j.

The available performance indicators per cluster are the following:

Indicator	Definition	Formula
Residual Sum of Squares (RSS)	Sum of the squared distances between the centroid and every individual.	$\sum_{x_i \in C_k} w_i\, d^2(x_i, \mu_k)$
Within Cluster Variance (also called Intraclass Inertia)	Average of the squared distances between the centroid and every individual.	$\frac{1}{W_k}\sum_{x_i \in C_k} w_i\, d^2(x_i, \mu_k)$
Simplified Silhouette	Indicator derived from the average ratio of the distances of all individuals to their two closest centroids	$1 - \frac{1}{W_k}\sum_{x_i \in C_k} w_i \frac{d(x_i, \mu_k)}{d(x_i, \pi_i)}$ where π_i is the second closest centroid to x_i.

The performance indicators aggregated by dataset are the following:

Indicator	Definition	Formula
Sum of Residual Sum of Squares (RSSSum)	Sum of the cluster Residual Sum of Squares	$\sum_k \sum_{x_i \in C_k} w_i\, d^2(x_i, \mu_k)$
Within Cluster Variance (or Intraclass Inertia)	Sum of the cluster intraclass inertias	$\sum_k \frac{1}{W_k}\sum_{x_i \in C_k} w_i\, d^2(x_i, \mu_k)$
Simplified Silhouette	Weighted average of the cluster simplified silhouette indicators	$1 - \frac{1}{W}\sum_k \sum_{x_i \in C_k} w_i \frac{d(x_i, \mu_k)}{d(x_i, \pi_i)}$ where π_i is the second closest centroid to x_i.

3.9 Profit Type

A profit type allows calculation of the profit that may be realized using the model. In general, a benefit is associated with the positive (or expected) values of the target variable and a cost is associated with the negative (or unexpected) values. For instance, in the context of a promotional mailing campaign, a person is associated with:

- A benefit for responding to the promotional mailing,
- A cost for not responding to the promotional mailing.

To visualize the profit that may be realized using a model generated by the application, you may use the following profit types:

- Detected profit,
- Lift profit,
- Standardized profit,
- Customized profit.

Detected Profit

Detected profit is the profit type shown as the default. It allows examination of the percentage of observations belonging to the target category of the target variable, that is, the least frequent category, as a function of the proportion of observations selected from the entire dataset. Using this profit:

- The value "0" is assigned to observations that do not belong to the target category of the target variable,
- The value "1" (frequency of the target category of the target variable in the dataset) is assigned to observations that do belong to the target.

Lift Profit

Lift profit allows examination of the difference between a perfect model and a random model and between the model generated by the application and a random model.

It represents the ratio between a model and the random model, that is the performance of a model compared to a model that would only allow to select observations at random from your database.

You can thus visualize how much better your model is compared with the random model.

Standardized Profit

Standardized profit allows examination of the contribution of the model generated by the application features relative to a model of random type, that is, in comparison with a model that would only allow selecting observations at random from your database.

This profit is used for the plots of variable details, which present the significance of each of the categories of a given variable with respect to the target variable.

Customized Profit

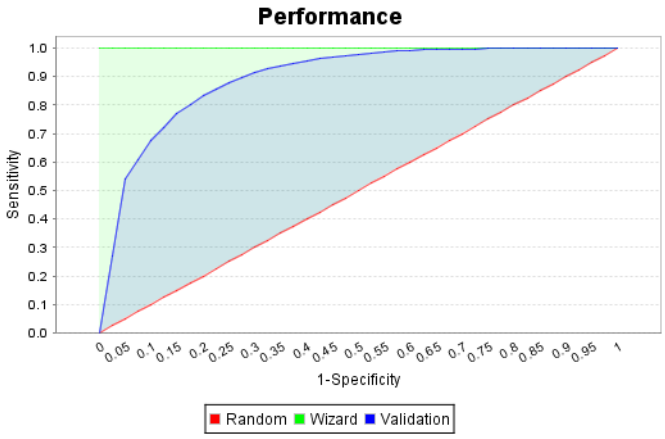
Customized profit allows you to define your own profit values, that is, to associate both a cost and a benefit to each value of the target variable. For instance, you can define:

- the cost of sending out a mailing as a negative value, for example "-5".
- the benefit brought in by the response to that mailing as a positive value, for example "20".

3.10 Advanced Model Curves

3.10.1 ROC

The ROC (Receiver Operating Characteristic) graph is derived from signal detection theory. It portrays how well a model discriminates in terms of the tradeoff between sensitivity and specificity, or, in effect, between correct and mistaken detection as the detection threshold is varied.



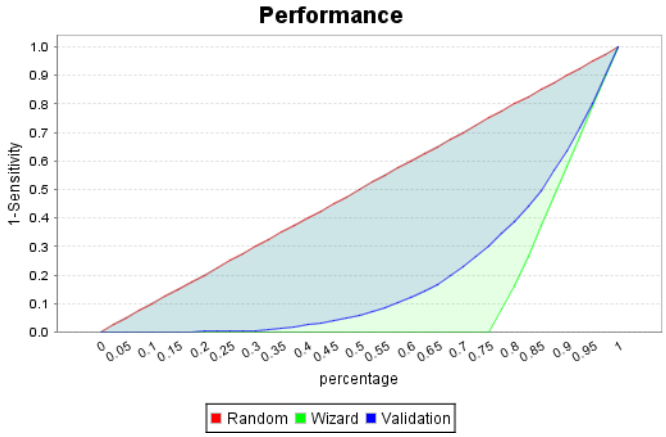
Sensitivity, which appears on the Y axis, is the proportion of CORRECTLY identified signals (true positives) found (out of all true positives on the validation dataset).

[1 – Specificity], which appears on the X axis, is the proportion of INCORRECT assignments to the signal class (false positives) incurred (out of all false positives on the validation dataset). (Specificity, as opposed to [1 – specificity], is the proportion of CORRECT assignments to the class of NON-SIGNALS – true negatives.)

3.10.2 Lorenz Curves

Lorenz "Good"

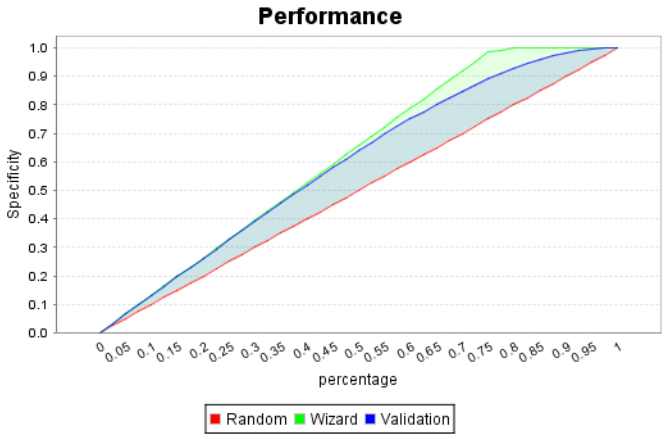
Lorenz ‘Good’ displays the cumulative proportion of missed signals (false negatives) accounted for by the records corresponding to the bottom x% of model scores.



The Y axis measures [1 – sensitivity], that is [1 – the proportion of true positives], which is equivalent to the proportion of missed signals or lost opportunity. Because the data are ordered from records predicted least likely to be signals on the left to records most likely to represent signals on the right, the slower the rise, the more sensitive the model in terms of detecting signals (or responders). The wizard line turns upward from the x-axis at the point corresponding to the proportion of non-signals in the validation dataset.

Lorenz "Bad"

Lorenz ‘Bad’ displays the cumulative proportion of true negatives (specificity) accounted for by the bottom x% of model scores. Here, the faster the rise, the lower the frequency of erroneous detection.



3.10.3 Density Curves

The density curves display the density function of the variable score in the set of events (Curve Density “Good”) and in the set of non events (Curve Density “Bad”). These curves can also be viewed as the derivate of Lorenz curves (the density function is by definition the derivate of the cumulative density function).

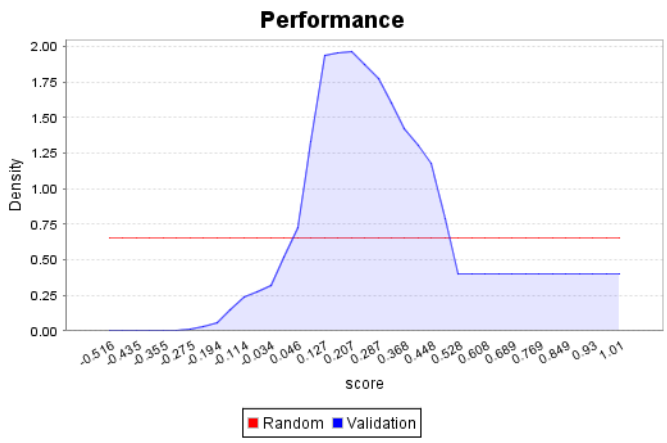
The estimated density function in a bin or interval is equal to:

$$\frac{\frac{\textit{Number of Events in the Interval}}{\textit{Total number of Events}}}{\textit{Length of the interval}}$$

The length of an interval is by definition its upper bound minus its lower bound.

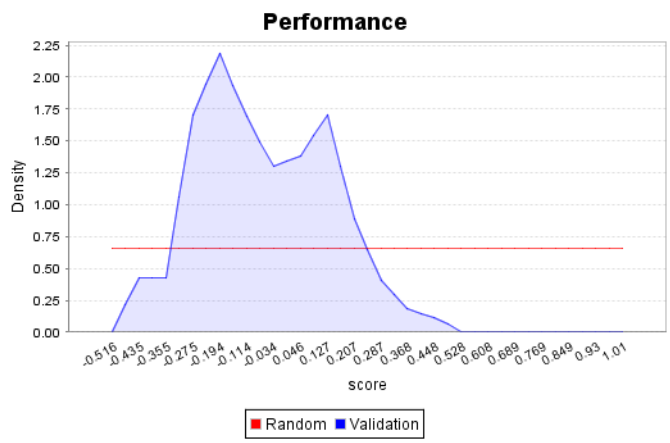
Density "Good"

This curve displays the distribution of model scores for responders/signals.



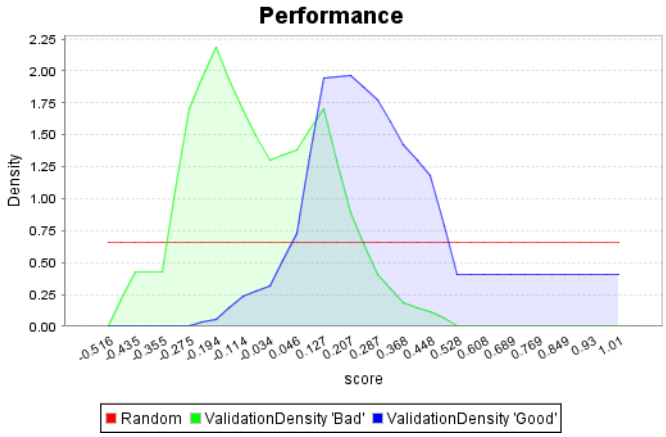
Density "Bad"

This curve displays the distribution of model scores for non-responders/non-signals.



Density "All"

This curve displays both the curves Density "Good" and Density "Bad", thus allowing you to compare both distributions.

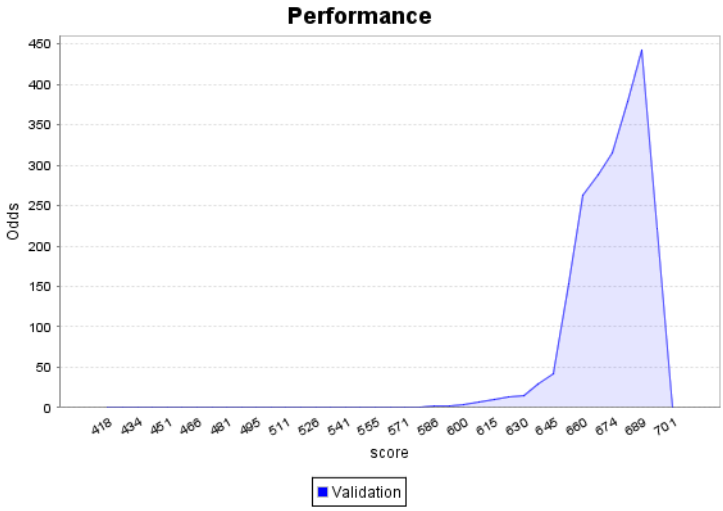


3.10.4 Risk Curves

Good/Bad Odds

The X-axis represents the risk score and the Y-axis represents the odds ratio value.

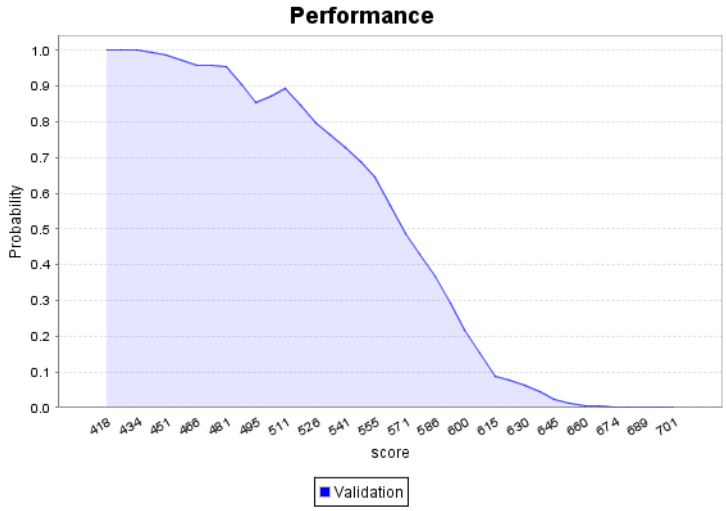
The odds ratio is equal to (1-p)/p , p is the probability of risk.



Probability of Risk

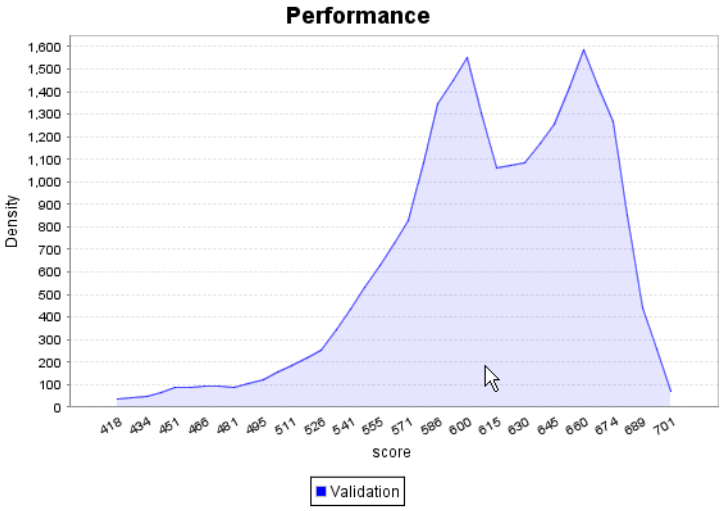
The X-axis represents the risk score and the Y-axis represents the odds ratio value.

The probability of risk p is computed for each risk score bin this way: number of "Bad" divided by the number of records in the risk score bin.



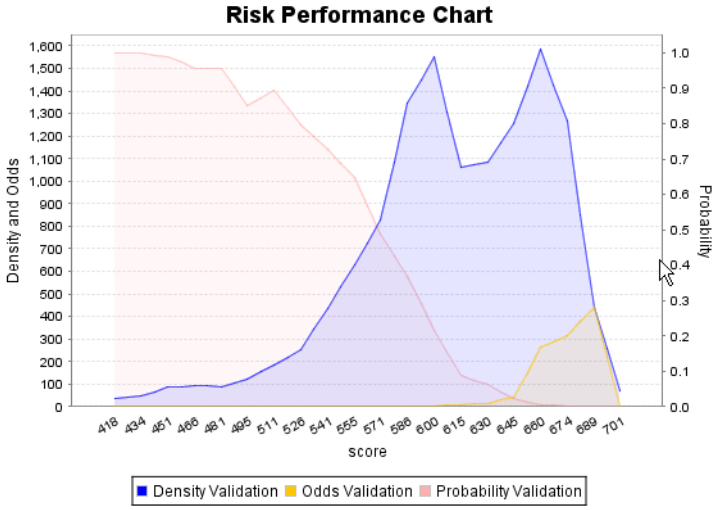
Population Density

The density is computed according to the number of records in each risk score bin (20 by default).



Risk 'All'

All three curves are displayed in the same graph. Note that the y-axis of the probability curve is on the right hand side. The y-axis of the population density and the good/bad odds on the left.


Risk Performance Chart
Density and Odds
Probability
score
Density Validation
Odds Validation
Probability Validation

4 Overview of Automated Analytics

4.1 Overview of Automated Analytics

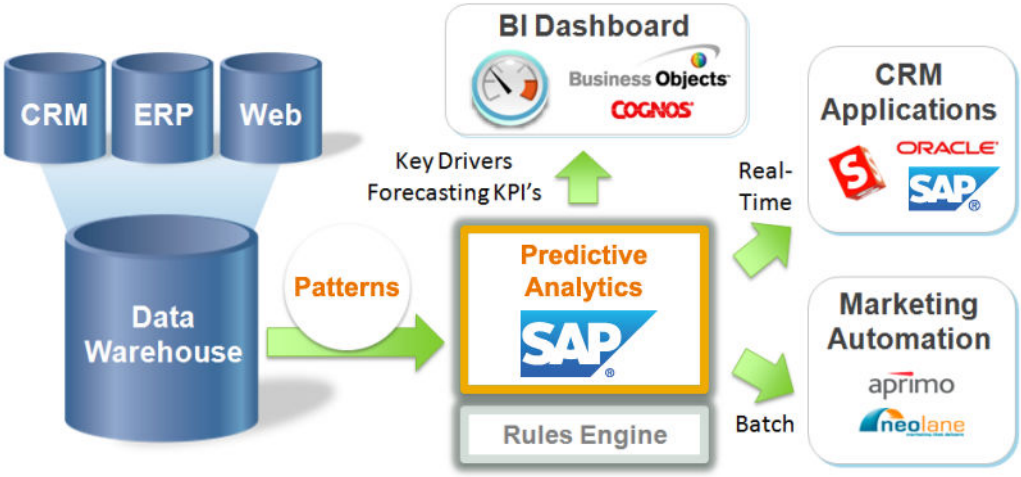
Automated Analytics provides a data mining solution for modeling your data as easily and rapidly as possible, while maintaining relevant and readily interpretable results. You will transform your data into knowledge in order to make timely strategic and operational decisions.

The application places the latest data mining techniques within reach of any non-expert user. You can access many data source formats to rapidly generate explanatory and predictive models, as well as descriptive models in a semi-automated manner.

The application allows you to concentrate on high value-added activities such as analysis of the results of data modeling, and decision-making.

4.1.1 Automated Analytics Architecture

The figure below illustrates the general architecture of Automated Analytics.



Three types of interfaces allow you to use the application:

- Graphical user interface,
- Command interpreter,
- API (Application Programming Interface) controls.

The graphical user interface is aimed primarily at non-expert users. It allows you to model your data very easily. In addition, it provides plotted output to facilitate viewing and interpretation of the results of modeling.

The KxShell command interpreter allows you to use the application by typing commands or executing scripts containing several commands.The command interpreter is an example of development based on the C++ API. Like any other API, it may be used to integrate the application with other applications or program packages.

The Control API is aimed primarily at developers, or users with programming experience. This API is used to access the complete range of functionalities and the most fine-grained parameterization of the application's features. In addition, it allows customized integration of the application's features with other applications or program packages.

Three APIs are provided with the application:

- A COM/DCOM API, usable on Microsoft platforms,
- A CORBA API, usable ov all client/server platforms,
- A C++ API, usable on all standalone platforms.

4.1.2 Automated Analytics Operational Phases

The operation of the Automated Analytics can be subdivided into four phases:

- Data access
- Data manipulation and preparation
- Data modeling
- Model presentation and deployment

Data Access Phase

The application accepts many types of data sources:

- Flat files such as .csv files, files of text tables and other files of type text.
- ODBC-compatible sources such as Oracle, SQL Server or IBM DB2 databases.

In addition, the C Data Access Application Programming Interface allows you to connect proprietary format sources, such as industrial sensor streams.

In most cases, and particularly if you are using the application features via a graphical interface, you never have to concern yourself with the data access process. Data access is accomplished in a semi-transparent manner: from the graphical user interface, you need only select the data source format to be used (flat files or ODBC-compatible data sources) and specify the location of the data file. The C Data Access Application Programming Interface is helpful to developers who want to write access code for proprietary format databases.

The application allows reading SAS data and writing the scores obtained with an application model into a SAS table.

The following formats are currently supported:

- SAS files version 6 under Windows and Linux
- SAS 7/8 under Windows and Linux
- SAS Transport Files

You can directly access a SAS table with the application interface by selecting the format of the file to analyze. Once you have built your model with the application, you can generate a SAS table containing the model application results (for example, scores, probability, cluster number, predicted value). The application interface allows you to select the output format. The generated SAS table is automatically integrated in SAS information system.

Data Manipulation and Preparation Phase

The Sequence Coding and Event Logging features of Data Manager are data manipulation and preparation features. They are used to encode data in a robust and semi-automatic manner, making them available for use by all analytical features of the application. The use of these features is transparent: all data processing is performed automatically.

Event Logging (formerly known as KEL) aggregates events into periods of time. It allows integrating transactional data with demographic customer data. It is used in cases when the raw data contains static information such as age, gender or profession of an individual, and dynamic variables, such as spending patterns or credit card transactions. Data is automatically aggregated within user defined periods without programming SQL or changing database schema. Event Logging combines and compresses this data to make it available to other features of the application.

Sequence Coding (formerly known as KSC) aggregates events into a series of transitions. For example a customer click-stream from a Web site can be transformed into a series of data for each session. Each column represents a specific transition from one page to another. Similar to Event Logging these new columns of data can be added to existing customer data and are made available to other application features for further processing.

Modeler - Data Encoding (formerly known as K2C) automatically prepares and transforms data into a format suitable for use in the application. Modeler - Data Encoding translates nominal and ordinal variables, automatically fills in missing values and detects out of range data. In addition, this feature contributes significantly to the robustness of the models generated by the application engine, by providing a robust data encoding.

Data Modeling Phase

Thanks to the statistical techniques and information technologies upon which the Regression/Classification, Segmentation/Clustering and Time Series features were built, these features require only a short modeling time to generate relevant and robust analytical models of your data.

Modeler - Regression/Classification (formerly known as K2R) generates explanatory and predictive models. The models generated by Classification/Regression explain and predict a phenomenon, or business question, by a function of the analyzed dataset, the explanatory variables. The models generated are calculated using a regression and classification algorithm. This polynomial regression is a proprietary algorithm using Vapnik's SRM (Structural Risk Minimization) principle to calculate the parameters.

Modeler - Segmentation/Clustering (formerly known as K2S) generates descriptive models, which means a function to regroup cases in a dataset into a number of clusters with similar behavior toward a business question.

Time Series (formerly known as KTS) lets you build predictive models from data representing time series. Thanks to time series models, you can:

- Identify and understand the phenomenon represented by your time series.
- Forecast the evolution of time series in the short and medium term, that is, predict their future values.

Model Presentation and Deployment Phase

Once the models have been generated, model performance indicators, plots and modeling reports in HTML format facilitate viewing and interpretation of the data modeling results.

Once the models have been validated, you can apply them to:

- One or more specific observations taken from your database (Simulation mode),
- A new, complete dataset or application dataset (Batch mode).

To facilitate deployment and integration of the models, the code corresponding to each model can also be generated in the programming language using the code generation feature.

The code generation feature (formerly known as Scorer or KMX) generates code in the following languages: C, XML, AWK, HTML, SQL, PMML2, SAS, or JAVA corresponding to a model generated by the application.

In this form, the model may be integrated into any application that supports the aforementioned languages.

The generated codes allow the application models to be integrated within any given application or software package, or to be applied directly to the data without requiring the Automated Analytics environment.

> **i Note**
>
> Code generation is only available for models using the following features: Modeler - Data Encoding, Modeler - Regression/Classification, Modeler - Segmentation/Clustering.

4.1.3 Methodological Prerequisites

Before modeling your data using Automated Analytics, you should:

- State a business issue that you want to solve,
- Possess a dataset representing this issue in the form of a set of observations.

What is your Business Issue?

The application is designed to allow supervised data analysis. The term supervised means that the data analysis does not occur completely independently, but always as a function of a particular issue: your business issue.

Consider the database that contains information about your customers. An analysis that groups your customers into homogeneous groups independently of your input is of little interest. On the other hand, an analysis that groups them as a function of a variable such as "mean business revenues earned from this customer each year" offers significant interest. You would learn the characteristic profiles of the customers that bring you the most money. Then, you can develop strategies to better influence your customers according to their characteristic profiles.

Before you begin, you need to identify and formulate your business issue.

Is your Data Usable?

Once your business issue has been identified and formulated, you need to have data on hand that will permit an answer to be found. In order for your data to be usable by the application, the following five conditions must be met:

- You must have a sufficiently large volume of data to be able to build a valid model, that is, in order for the model to be both relevant and robust. An analytical model that is generated from a dataset of 50 lines may have low generalization capacity, and contain low informative value. We can advise you on the issues of data volume.
- Your dataset must contain a target variable, that will allow you to express your business issue within the application.
- The target variable must be known for each observation of the training dataset. To express this another way, no target variable values may be missing over the range of the entire training dataset.
- The data source format must be supported by the application.
- Your data must be presented in the form of a single table of data , except in instances where you are using Event Logging or Sequence Coding features of Data Manager.

5 Modeler

5.1 Classification/Regression

5.1.1 About Modeler - Classification/Regression

This section of the guide is addressed to people who want to evaluate or use Modeler to build Regression/Classification. Use of this guide does not require any prior expertise in statistics or databases.

This part introduces you to the basic concepts and main functionalities of Modeler and its feature Regression/Classification. Using an application scenario, you can create your first model with confidence.

The purpose of Modeler - Regression/Classification is to understand and predict a phenomenon.

> Example
>
> You work for an automobile manufacturer and wish to send a promotional mailing to your prospects. Modeler - Regression/Classification allows you to:
>
> - Understand why previous prospects responded to such a mailing,
> - Predict the response rate to such a mailing sent to new prospects.

5.1.2 Files and Documentation Provided with this Guide

Sample data files are supplied with the application. These files allow you to take your first steps using various features of the application, and evaluate them. During installation, the sample files are saved in the folder: `C:\Program Files\SAP Predictive Analytics\Desktop <X.Y>\Automated\Samples\Census\`.

The following table describes those files.

File Name	Description	When is it Used?
Census01.csv	Data file	This file is used for both application scenarios used in this manual.
desc_census01.csv	Description file for the Census01.csv file	This file is used for both application scenarios used in this manual.

To obtain a detailed description of the `Census01.csv` file, see the related topic.

Complete documentation is included with the application. This documentation covers:

- The operational use of the application features
- The architecture and integration of the application API
- The Java graphical user interface

Each screen in the modeling assistant is accompanied by contextual help that describes the options presented to you, and the concepts required for their application. To display the contextual help, press F1.

To acces a searchable version of all the help for the application, select *Help* > *Open Full Searchable Help*.

5.1.3 Regression/Classification Application Scenario Description

In this scenario, you are the Marketing Director of a large retail bank. You want to enhance efficiency and master your budget using modeling. The bank wants to offer a new financial product to its customers. Your project consists of launching a direct marketing campaign aimed at promoting this product. You have a large database of prospects at your disposal and a limited and closely monitored budget, and you are also subject to significant time constraints. In order to maximize the benefits of your campaign, your business issue consists of:

- Contacting those prospects most likely to be interested in the new financial product,
- Identifying the ideal number of prospects to contact out of the entire database.

Using the Modeler - Regression/Classification (formerly known as K2R) feature, you can rapidly develop an explanatory and predictive model at the least possible cost. This model allows you to respond to your business issue and accomplish your objectives.

Your Objective

Imagine the following case.

You are the Marketing Director of a large retail bank. This bank has decided to offer its customers a new high-end savings product. It prepares to launch an extensive direct marketing campaign to promote this new product to its prospects and customers.

The bank is experiencing heavy competition and senior management, sensitive to the stakes involved in launching this new financial product, wants the marketing campaign completed as soon as possible.

Your Means

A limited and closely monitored budget: The enterprise controls of the bank are rigorous, and the budget that has been allocated to you for this marketing campaign:

- Does not allow you to contact all of the bank's prospects and customers,
- May not be exceeded.

The information at your disposal: The Marketing Department has a database for this campaign which contains the records of 1,000,000 prospects, identified by their principal characteristics, including::

- Age

- Sex
- Employer
- Nationality
- Occupation
- Education
- Number of hours worked per week

You note that the database you have at hand is not ideal. In fact, the database contains:

- Incongruous data: The database contains alphanumeric information (such as "occupation" and "nationality") as well as numerical information (such as "age" and "unreconciled accounts").
- Redundant data: Some information in the database is redundant, such as "degree" and "education", or "degree" and "area of work".
  In the field of statistics, the term "correlated variables" is used to designate such data. In classical statistical analyses, correlated variables must be processed in a particular manner. An alternate solution is to designate only one of the two correlated variables for analysis.
  Since you have neither the statistical skills not the means to handle this issue of correlation between variables, you decide to leave the database as it is.
- Missing data: Some information is missing from the database. To manage this lack of information, the Information Technology department used the following convention:
  - The symbol "?" means that an alphanumeric value (such as "occupation") is missing,
  - The value "99999" means that a numerical value (such as "age") is missing.

  Unfortunately, you have neither the time nor the necessary resources to perform a survey to fill in the missing information or to re-format the database.

The database available to you is stored in an RDBMS (relational database management system) residing on a Linux server, maintained by the Information Technology department of the bank. The technical constraints of this information environment are determining factors in selecting potential data analysis tools.

Your Approach

By virtue of the critical stakes involved in this campaign, because of your limited budget and your inability to predict customers' enthusiasm for the new product, you have chosen to minimize your risks by dividing the project into two steps:

- Test the marketing campaign on a sample of 50,000 individuals extracted from the prospects database of 1,000,000 people,
- Global launch of the marketing campaign using the entire contents of the prospects database.

The test phase of your marketing campaign allowed you to collect a sample of 50,000 individuals whose behavior with respect to this new product is known:

- 25% of the prospects showed themselves to be clearly interested. They chose to accept an invitation for a meeting with one of your sales channel agents,
- 75% of the prospects declined your invitation.

Your business issue consists of understanding the test results, by identifying the reasons which led certain individuals to respond favorably to your offer, and others to respond in the negative. Then you would be able to use the analytical model obtained to predict the behavior of each of the 1,000,000 prospects in your database. This would ensure that you optimized your marketing campaign by making the offer only to those individuals most likely to be interested.

The file containing the dataset used for the test was sent to you by the Information Technology department of the bank in the form of a flat file (`.csv`). This file corresponds to the sample file `Census01.csv` provided with the application. For more information, see the related topic about sample files.

Your Business Issue

Following the test phase of your campaign, your marketing database will contain:

- A list of 1,000,000 prospects,
- A list of 50,000 prospects, selected in a random manner during the test phase, and whose response to your test campaign is known. This sample, taken from your initial database, also contains missing values and correlated variables.

Your approach to the business issue consists of using the dataset in its present state, as a training dataset, in order to:

- Rapidly create an explanatory and predictive model,
- Next, apply this model to the entire database.

Using the model generated, you will be able to determine:

- How many individuals contained in your prospect database you should send your mailing to in order to maximize the profit/return on investment of your campaign.
- How to classify all of the individuals in your prospect database according to their interest (purchasing probability) in this new product. This interest is expressed as a score, or probability that a prospect will respond favorably to the campaign.
- What characterizes these individuals and what are their profiles? Validate the criteria (age, socio-occupational class, degree) that explain why a person expresses interest or not in the new financial product.
- How to simulate in real time, the likelihood of a single individual to respond favorably to a new offer, in particular, to allow the call center of your bank or a customer service agent to immediately know the level of interest that a prospect is likely to exhibit in this financial product.
- How to record this score in your prospect database, in order to be able to select sub-groups of the population for new campaigns at a later date.
- How to measure the quality and reliability (capacity of handling new individuals) of your model .

In order to allow you to better respond to these issues, you have access to several possible application solutions.

Your Solutions

To select the individuals to whom you will send a mailing, you have several possible solutions. You can use:

- A shotgun method ,
- An intuitive method ,
- A classical statistical method (for example: neural networks, Bayesian networks, logistic models, decision trees)

- The Automated Analytics method .

5.1.4 Scenario Solutions

5.1.4.1 Shotgun Solution Method

This method consists of performing no selection on your database, and sending out a mass mailing to every person recorded in your database. This solution guarantees that all persons likely to purchase your product are contacted.

On the other hand, the costs of this solution can be high, potentially far exceeding your budget, and it is seldom the solution applied. In addition, it runs the risk of saturating the prospects of your bank with inappropriate offers (spamming).

5.1.4.2 Intuitive Solution Method

This method consists of performing a selection that leans on your knowledge of your field, that is to say, you send your mailing to individuals selected in an intuitive manner from your database. This solution allows you to significantly reduce the cost of your marketing campaign and make it fit your budget.

This method is not optimal, because it does not allow you to:

- Control the real costs and return on investment of your marketing operation.
- Select which prospects to contact on a basis of real returns. It is true that you probably have a relatively good understanding of which individuals stand a good chance of becoming your customers some day. But optimizing your campaign means being able to identify those prospects that have every chance of becoming customers today as a result of the current marketing campaign.
- Discover new niche prospects that all your knowledge of the market had not previously allowed you to identify.
- Select a predefined number of individuals. Imagine that one of the constraints of your campaign consisted of contacting exactly 5,000 prospects. Your intuition may help you to select 2,400 of these. But how are you going to identify the remaining 2,600 prospects to be contacted? A purely random selection, thus completely non-optimized, might be your only solution.

5.1.4.3 Classical Statistical Solution Method

You may decide to use a classical statistical method to better manage the effectiveness of your campaign, and thus, your budget.

On the basis of the information that you have, a Data Mining expert could create predictive models. In other words, you could ask a statistical expert to create a mathematical model that would allow you to predict the probability of a given individual to respond to your marketing campaign, as a function of his profile.

To implement this method, the statistician must:

- Perform a detailed analysis of your test campaign,
- Prepare your database down to the smallest detail, specifically, encoding the different types of data in such a way that they can be used by the analytical tools he will apply,
- Test different types of algorithms (for example: neural networks, Bayesian networks, logistic models, decision trees) and select the one best suited to your business issue.

Typically, after a few weeks, the statistician will be able to associate a value with each individual in your database, indicating the probability of being interested or not interested in your marketing campaign.

This method presents significant constraints. You must:

- Ensure that your statistical expert, perhaps from a department external to the Marketing Department, is available for the scheduled period,
- Ensure that the cost for using this scarce resource will fit into your budget,
- Spend time explaining your domain-specific business issue to him,
- Spend time understanding the results that are provided.

5.1.4.4 Automated Analytics Solution Method

The simplicity and automatic nature of the application will allow you to perform the statistical analysis of your database yourself in a short amount of time.

Using the application, you will be able to create a model that allows you to:

- Determine which individuals have the highest probability (score) of being interested in your marketing campaign (predictive modeling). You may then apply the model to your entire database.
- Break out the determining factors that describe the phenomenon that you hope to model, that is, the fact of being "interested" or "not interested" in the new financial product of the bank (descriptive modeling).

The profit curve, an important validation and control tool, allows you to compare the performance of models generated using the application features with that of a hypothetical random model and that of a hypothetical perfect model. At the same time, it also allows you to determine the optimal number of persons to contact to maximize the profit generated by your campaign. The application also provides you with indicators of the quality of the model you generate (predictive power), and its capacity to generalize and remain relevant to new datasets (prediction confidence).

The application provides you with the means to customize your direct marketing campaign with respect to your different customer profiles, increasing your powers of persuasion.

5.1.5 Introduction to Sample Files

This guide is accompanied by the following sample data files:

- A data file `Census01.csv`.
- The corresponding description file `desc_census.csv`.

These files allow you to evaluate Automated Analytics features and take your first steps in using it.

`Census01.csv` is the sample data file that you will use to follow the scenarios for Regression/Classification and Segmentation/Clustering. This file is an excerpt from the American Census Bureau database, completed in 1994 by Barry Becker.

> **i Note**
>
> For more information about the American Census Bureau, see http://www.census.gov .

This file presents the data on 48,842 individual Americans, of at least 17 years of age. Each individual is characterized by 15 data items. These data, or variables, are described in the following table.

Variable	Description	Example of Values
age	Age of individuals	Any numerical value greater than 17
workclass	Employer category of individuals	• Private • Self-employed-not-inc
fnlwgt	Weight variable, allowing each individual to represent a certain percentage of the population	Any numerical value, such as "0", "2341" or 205019".
education	Level of study, represented by a schooling level, or by the title of the degree earned	• 11th • Bachelors
education-num	Number of years of study, represented by a numerical value	A numerical value between 1 and 16
marital-status	Marital status	• Divorced Never-married
occupation	Job classification	• Sales • Handlers-cleaners
relationship	Position in family	• Husband • Wife
race	Ethnicity	• White • Black
sex	Gender	• Male • Female
capital-gain	Annual capital gains	Any numerical value
capital-loss	Annual capital losses	Any numerical value
native country	Country of origin	• United States • France
class	Variable indicating whether or not the salary of the individual is greater or less than $50,000	• "1" if the individual has a salary of greater than $50,000 • "0" if the individual has a salary of less than $50,000

> **i Note**
>
> In order to avoid complicating the application scenarios, the variable fnlwgt is used as a regular explanatory variable in these scenarios, and not as a weight variable.

5.1.6 SAP Predictive Analytics

To accomplish the scenario, you will use the Automated Analytics toolset of SAP Predictive Analytics.

5.1.6.1 To Start SAP Predictive Analytics

1. Select *Start* > *Programs* > *SAP Business Intelligence* > *SAP Predictive Analytics Desktop* > *SAP Predictive Analytics*.

   The SAP Predictive Analytics start panel appears.
2. Click the feature you want to use.

5.1.6.2 Editing the Options

- To edit the appliation options:

1. In the *File* menu, click *Preferences...*.

   The window *Edit Options...* appears.
2. The following options can be modified for Classification/Regression:

Option	Description
Category	Options
General	○ Country ○ Language ○ Message Level ○ Log Maximum Size ○ Message Level for Strange Values ○ Display the Parameter Tree ○ Number of Store in the History ○ Always Exit without Prompt ○ Include Test in Default Partition Strategy ○ SQL Statement Separator
Stores	○ Default Store for Apply-in Dataset ○ Default Store for Apply-out Dataset ○ Default Store to Save Models
Metadata Repository	○ Store the Metadata in the Same Place as the Data

Option	Description
	○ Store the Metadata in a Single Place ○ Edit Variable Pool Content
Graphic	○ Graphic theme ○ Font Size ○ Profit Curve Points ○ Bar Count Displayed ○ Display 3D Chart ○ Disable Double Buffering ○ Optimize for Remote Display
Report	○ Report Custom Banner ○ Style Sheet Repository ○ Active Style Sheet
Geolocation	○ Geographic Information System Protocol

5.1.6.2.1 Customizing Style Sheets

SAP Predictive Analytics offers the possibility to customize the generated reports. The default style sheet, called *SAP Predictive Analytics (default)*, cannot be modified. You have to create your own style sheets to modify the settings.

> **i Note**
>
> Before you create, load or save a style sheet, you must first select a *Style Sheet Repository* folder.

5.1.6.2.1.1 To Create a New Style Sheet

1. In the field *Folder*, click the button (*Browse*).
2. Select a folder.

   This folder is your style sheets repository.
3. Click the button (*Add*).

   A new style sheet has been created.
4. Click the button .

   The panel *Report Style Sheet Editor* opens.
5. In the field *Style Sheet Name*, enter a name for the new style sheet.

   The extension `.krs` is automatically added.

   > **i Note**
   >
   > You can duplicate a style sheet by changing the name of your style sheet. The previous one is not deleted.

5.1.6.2.1.2 To Delete a Style Sheet

1. Select one of the displayed style sheets.
2. Click the button (Remove).

> **i Note**
>
> The style sheet is not only deleted from the list but also from the data source.

5.1.6.2.1.3 To Edit the General Settings

Select settings:

Option	Description
Settings...	Options...
Reports Background Color	○ choose a color ○ make transparent **i Note** Only the PDF and HTML formats can display a background color.
Edit Configuration	○ font size ○ font style ○ font color ○ text background color ○ table configuration **i Note** Check the option *Dynamically render option changes* or click *Apply* when editing the settings, so that you can visualize the result.

The selected settings will be applied to both the wizard and the generated reports.

5.1.6.2.1.4 To Edit the Charts Settings

Select settings:

Option	Description
Settings...	Options...
Chart Colors	modify the chart colors

Option	Description
Default Chart Bars Orientation	○ horizontal ○ vertical **i Note** It is possible to set another default orientation for specific report items.

5.1.6.2.1.5 To Edit Report Items

1. Set the properties of your choice.

Properties...	Functions...
Displayed as	name of the label
View Type	choose between Tabular, HTML and Graphical. The last one is only available if the report item can be displayed as a graph.
Chart Type	select one of the proposed chart types Note that this option is only available for report items of the view type *Graphical*.
Switch Bar Orientation	this option allows having another bar orientation as the default one for a specific report item
Sort by/Sort Order	you can select a column to sort by and choose between an ascending or a descending order
Visibility	you can hide columns of a report item or even menu items Note that at least one column of a report item must remain visible.

2. Click Save to validate.

   A window opens, indicating that your style sheet has been successfully saved.
3. Click *OK*.

5.1.6.2.1.6 To Apply the New Style Sheet to the Generated Reports

1. In the panel *Report*, select the new style sheet.
2. Click *OK*.

   A window opens, indicating that you have to restart the modeling assistant to take the edited options into account.

3. Click *OK*.

   When training a model, all the generated reports (the learn/excel/statistical reports) are now customized.

5.1.6.2.2 Defining a Metadata Repository

The metadata repository allows you to specify the location where the metadata should be stored.

1. Choose between storing the metadata in the same place as the data or in a single place by checking the option of your choice.
2. In the list *Data Type*, select the type of data you want to access. For some types of data, you will need a specific license.
3. Use the *Browse* button corresponding to the *Folder* field to select the folder or database containing the data. In case of a protected database, you will need to enter the user name and the password in the fields *User* and *Password*.
4. Click the button *Edit Variable Pool Content* to edit the parameters of the variables stored in the variable pool.
5. Click *OK* to validate.

5.1.7 Creating a Model Using Modeler

Data modeling with the application is subdivided into four broadly defined stages:

- Defining the Modeling Parameters
- Generation and Validation of the Model
- Analysis and Understanding of the Analytical Results
- Using a Generated Model

5.1.7.1 Step 1 - Defining the Modeling Parameters

To respond to your business issue you want to:

- Identify and understand the factors that determine whether a prospect reacts positively or negatively to your marketing campaign.
- Thereby be able to predict the behavior of new prospects with respect to your campaign.

The Regression/Classification feature (formerly known as K2R) allows you to create explanatory and predictive models.

The first step in the modeling process consists of defining the modeling parameters:

- Select a data source to be used as training dataset.
- Describe the dataset selected.
- Select the variables: the target variables, the explanatory variables and possibly a weight variable.
- Check the modeling parameters.

- Setting the Advanced Parameters (degree, target key, variable auto-selection and correlations). This step is optional.

5.1.7.1.1 Selecting a Data Source

Use the file `Census01.csv` as a training dataset.

This file represents the sample that you had extracted from your database and used for the test phase of your direct marketing campaign. As specified in your test plan, this file contains data concerning 50,000 prospects, for whom you now know the behavior with respect to the new financial product:

- 25% of the prospects showed themselves to be clearly interested. They chose to accept a an invitation for a meeting with one of your sales channel agents,
- 75% of the prospects declined your invitation.

In this file, you created a new variable Class, which corresponds to the reaction of prospects contacted during the test. You assigned:

- The value "1" to those prospects who responded positively to your invitation,
- The value "0" to those prospects who responded negatively to your invitation.

To select a data source:

1. On the screen *Select a Data Source*, select the data source format to be used (*Text files*, *Data Base*, ...).
2. Click the *Browse* button.

   In the *Data Selection* dialog, browse to the `<Installation Path>/Samples` folder.

   > **i Note**
   >
   > Depending on your environment, the `Samples` folder may or may not appear directly at the root of the list of folders. If you selected the default settings during the installation process, you will find the Samples folder located in `C:Program Files/SAP Analytics/Desktop <version number>/Automated/`.

3. Double-click the *Samples* folder, then the *Census* folder.
4. Select the file `Census01.csv`, then click *OK*.

   The name of the file appears in the *Dataset* field.
5. Click *Next*.

   The screen *Data Description* appears. Now you are ready to describe the data selected.

5.1.7.1.2 Describing the Data Selected

For this scenario:

- Select *Text Files* as the *Data Type*.
- Use the file `Desc_Census01.csv` as the description file for the `Census01.cvs` data file.

To Select a Description File:

1. On the screen *Data Description*, click the button *Open Description*.

   The Load a Description window opens.
2. Select the type of your description file.
3. In the *Folder* field, select the folder where the description file is located with the *Browse* button.

   The folder selected by default is the same as the one you selected on the screen *Select a Data Source*.
4. In the *File Name* field, select the file containing the dataset description with the *Browse* button.

   > ⚠ Caution
   >
   > When the space used for model training contains a physical variable named `KxIndex`, it is not possible to use a description file without any key for the described space.
   >
   > When the space used for model training does not contain a physical variable named `KxIndex`, it is not possible to use a description file including a description about a `KxIndex` variable since it does not exist in current space.
5. Click *OK*.

   The window *Load a Description* closes and the description is displayed on the screen *Data Description*.
6. Click *Next*.

5.1.7.1.2.1 Why Describe the Data Selected?

In order for the application to interpret and analyze your data, the data must be described. To put it another way, the description file must specify the nature of each variable, determining the storage format and type:

- Storage format: number (*number*), character string (*string*), date and time (*datetime*) or date (*date*).

  > ℹ Note
  >
  > When a variable is declared as *date* or *datetime*, the date coder feature automatically extracts date information from this variable such as the day of the month, the year, the quarter and so on. Additionnal variables containing this information are created during the model generation and are used as input variables for the model.
  >
  > The date coder feature is disabled for Time Series.
- Type: *continuous*, *nominal*, *ordinal* or *textual*.

For more information about data description, see the related topics.

5.1.7.1.2.2 How to Describe Selected Variables

To describe your data, you can:

- Either use an existing description file , that is, taken from your information system or saved from a previous use of the application,

- Or create a description file using the *Analyze* option.. In this case, it is important that you validate the description file obtained . You can save this file for later re-use. If you name the description file `KxDoc_<SourceFileName>`, it will be automatically loaded when clicking the *Analyze* button.

> ⚠ Caution
>
> The description file obtained using the *Analyze* option results from the analysis of the first 100 lines of the initial data file. In order to avoid all bias, we encourage you to mix up your dataset before performing this analysis.

Each variable is described by the fields detailed in the following table:

The Field...	Gives information on...
Name	the variable name (which cannot be modified)
Storage	the type of values stored in this variable: • *Number*: the variable contains only "computable" numbers (be careful a telephone number, or an account number should not be considered numbers) • *String*: the variable contains character strings • *Datetime*: the variable contains date and time stamps • *Date*: the variable contains dates
Value	the value type of the variable: • *Continuous*: a numeric variable from which mean, variance, etc. can be computed • *Nominal*: categorical variable which is the only possible value for a string • *Ordinal*: discrete numeric variable where the relative order is important • *Textual*: textual variable containing phrases, sentences or complete texts ⚠ Caution When creating a text coding model, if there is not at least one textual variable, you will not be able to go to the next panel.
Key	whether this variable is the key variable or identifier for the record: • 0 the variable is not an identifier; • 1 primary identifier; • 2 secondary identifier...
Order	whether this variable represents a natural order. (0: the variable does not represent a natural order; 1:the variable represents a natural order). If the value is set at 1, the variable is used in SQL expressions in an "order by" condition. There must be at least one variable set as Order in the Event data source. ⚠ Caution If the data source is a file and the variable stated as a natural order is not actually ordered, an error message will be displayed before model checking or model generation.

The Field...	Gives information on...
Missing	the string used in the data description file to represent missing values (e.g. "999" or "#Empty" - without the quotes)
Group	Name of the group to which the variable belongs. You can assign a variable to one group only. All variables of a group convey the same information. Thus different groups cannot be crossed when the model has an order of complexity over 1.
Description	an additional description label for the variable
Structure	this option allows you to define your own variable structure, which means to define the variables categories grouping.

5.1.7.1.2.3 Viewing the Data

To help you validate the description when using the *Analyze* option, you can display the first hundred lines of your dataset.

1. Click the button *View Data*.

   A new window opens displaying the dataset top lines:
2. In the field *First Row Index*, enter the number of the first row you want to display.
3. In the field *Last Row Index*, enter the number of the last row you want to display.
4. Click the *Refresh* button to see the selected rows.

5.1.7.1.2.4 A Comment about Database Keys

For data and performance management purposes, the dataset to be analyzed must contain a variable that serves as a key variable. Two cases should be considered:

- If the initial dataset does not contain a key variable, a variable index `KxIndex` is automatically generated by the application. This will correspond to the row number of the processed data.
- If the file contains one or more key variables, they are not recognized automatically. You must specify them manually in the data description.

To Specify that a Variable is a Key

1. In the *Key* column, click the box corresponding to the row of the key variable.
2. Type in the value "1" to define this as a key variable.

5.1.7.1.2.5 Defining a Variable Structure

There are three ways to define a variable structure:

- by first extracting the categories from the variable statistics, then editing or validating the suggested structure.
- by importing the structure from an existing model.
- by building a new structure from scratch.

The option Optimal Grouping allows you to let Data Encoding group together the categories groups defined in the variable structure if they bring the same information.

The last column of the description table indicates the state of the structure of each variable. The following table lists the possible states of a variable structure.

Icon	State	Description
	undefined	Data Encoding will automatically determine the categories grouping depending on their interaction with the target variable.
	non-editable	The structure for an ordinal string variable cannot be modified.
	defined by extraction from the variable statistics	The user must open and validate the variable structure.
	defined by the user or imported from an existing model	

> **i Note**
>
> A translation of the variable categories has no influence on the variable structure, which has to be set according to the original values of the variable.

5.1.7.1.2.5.1 To Extract a Variable Structure

1. In the *Data Description* pane, select the variables for which you want to extract the structure.
2. Go to the *Structures* tab of the ribbon, the available options are separated in two parts: *Edit* and *Extract*.
3. Click the option *From Statistics* in the *Extract* part.

   A progression bar is displayed while the structure is being extracted. Once the extraction is done, the icons corresponding to the selected variables change, indicating that the operation was a success and allowing you to easily identify them. You can then modify the variable structure as you need.

5.1.7.1.2.5.2 To Import the Variable Structure from a Model

1. Select the variables for which you want to extract the structure.
2. Go to the *Structures* tab of the ribbon, the available options are separated in two parts: *Edit* and *Extract*.
3. Select the option *From Model* and choose the desired option.

   The window *Loading Model* opens.

4. In the list *Data Type*, select the type of store the model is saved in.
5. Use the *Browse* button located next to the *Folder* field to select the folder or database containing the model.
6. In the displayed models list, select the model from which you want to extract the variable structure.
7. Click *OK*.
8. In the list *Target from Loaded Model*, select the target of the model.

   The variables you have selected are displayed in a list with the corresponding variables from the loaded model. You can add or remove variables from this list and view the model variables structure (see the related topics).
9. Once all the variables for which you want to import the structure from the model are displayed in the list, Click *OK*.

   The selection window closes and the structure state changes.

5.1.7.1.2.5.3 To Add a Variable to the List of Variables

1. In the list *Variable from Loaded Model*, select the variable you want to add to the list of variables for which the structure will be imported.
2. Click the *Add* button.
   - The variable appears in the list.

5.1.7.1.2.5.4 To Remove a Variable from the List of Variables

1. In the list located in the lower part of the panel, select the variable for which you do not want to import the structure.
2. Click the *Remove* button.

   The variable is removed from this list and added to the list *Variable from Loaded Model*.

5.1.7.1.2.5.5 To View a Variable Structure Defined in the Loaded Model

1. If the variable has not been added yet to the list of variables located on the lower part of the panel:
   a. In the list *Variable from Loaded Model*, select the variable for which you want to see the structure defined in the model.
   b. Click the *View* button.

      The variable structure opens in a new window.
2. If the variable has already been added to the list of variables located on the lower part of the panel, double-click the variable for which you want to see the structure defined in the model.

5.1.7.1.2.5.6 To Create or Modify a Variable Structure

1. Double-click the Structure icon corresponding to the variable for which you want to edit or create the structure.

   The edit window opens. If the structure had been extracted from the variable statistics or from a model, the fields are already filled.
2. For details on how to use the structure editor, see the topics for the type of variable structure.

5.1.7.1.2.5.6.1 Structure for a Continuous Variable

The structure for a continuous variable is defined by several intervals each made of :

- a lower bound ([ ]) that can be either open or closed,
- a minimum value (Minimum),
- a maximum value (Maximum),
- a higher bound ([ ]) that can be either open or closed.

All intervals must be adjoining: there can be no gap or overlap between two intervals.

The option *Add Missing* allows you to indicate with which interval the missing values should be grouped.

The option *Include Smaller Data* allows you to include to the first interval any value smaller than its lower bound. In the same way, the option *Include Higher Data* allows you to include to the last interval any value higher than its higher bound.

To Create an Interval

1. Click the *Add* button.
   The edit window opens.
2. Select the lower bound type by clicking the] button.
3. Enter the minimum value for the interval in the left text field.
4. Enter the maximum value for the interval in the right text field.
5. Select the higher bound type by clicking the [button.
6. Check the option *Add Missing* if the missing values must be grouped with this interval.
7. Click the *Yes* button to validate your interval.

To Split an Interval

1. Select the interval to split.
2. Click the *Split* button.
   The selected interval is automatically split into two equal intervals.

To Merge an Interval

1. Select the intervals to merge. You can only select adjoining intervals.
2. Click the *Merge* button.

To Delete an Interval

1. Select one or more intervals. You can only select adjoining intervals.
2. Click the *Remove* button. The previous and next intervals are extended to include the values previously contained in the deleted intervals, so that no gap is left between intervals.

5.1.7.1.2.5.6.2 Structure for an Ordinal Variable

The structure for an ordinal variable is similar to that of a continuous variable with the exception of the bounds which are always closed and cannot be modified.

> ⚠ Caution
>
> The structure for an ordinal string variable cannot be edited.

5.1.7.1.2.5.6.3 Structure for a Textual Variable

The structure for a textual variable cannot be edited.

5.1.7.1.2.5.6.4 Structure for a Nominal Variable

The structure for a nominal variable is made of groups containing the variable categories.

To Create a New Category Group

1. In the list *Category Edition*, select the categories you want to add in a new group. Use the `Ctrl` key to select several categories.
2. Click the button *Add New Group*. A group containing the selected categories is created in the list *Group Structure*.

To Include Missing Values in a Group

1. In the list *Group Structure*, select the group in which you want to add the missing values.
2. Click the button *Add Missing* located under the list *Category Edition*. The `KxMissing` category, which represents the missing values, is added to the selected group and the button *Add Missing* is deactivated. As any category, the `KxMissing` category can only belong to one group at a time.

To Create a New Category

- In the field right of the button *New Category*, enter the name of the category to add.
- Click the button *New Category*. The category is created in the list *Category Edition*.

To Add Categories to a Group

- In the list *Category Edition*, select the category (or categories) to add to a group.
- In the list *Group Structure*, select the group in which you want to add the selected categories.
- Click the button *Add Category*.

To Delete a Group

- In the list *Group Structure*, select the group to delete.
- Click the button *Remove Group*. All the categories belonging to this group are re-added to the list *Category Edition*.

To Remove a Category from a Group

- In the list *Group Structure*, select the category or categories you want to remove from the group.
- Click the button *Remove Category*. The selected categories are removed from the group and re-added to the list *Category Edition*.

5.1.7.1.2.6 Working Without any Defined Structure

If you let the structure as undefined, the application uses consistent coder to automatically determine the categories grouping depending on their interaction with the target variable. You can configure two parameters in this case:

- The band count for continuous variables
- Modeler - Data Encoding optimal grouping for all variables.

5.1.7.1.2.6.1 Band Count for Continuous Variables

When you work with no defined structure, you can set the band count for continuous variables. The allowed values for this parameter are between 1 and 20.

The population is thus divided into as many segments of similar size. These segments are used to build descriptive statistics, particularly the distribution of target variables for each segment, which affects the coding of the variable with respect to target variables.

The band count has an influence on the calculation of the predictive power: the more there are segments, the more accurate is the calculation of the predictive power for the explanatory variable.. However, this influence is very small.

To Edit the Band Count for Continuous Variables

- Click the row corresponding to the continuous variable to be edited.
- Go to the *Structures* tab of the ribbon.
- Click *Edit User Band Count*. The *Set Band Count* screen displays.

If you want to...	Then...
modify the band count for all the continuous variables of the model	○ Type in the desired band count in the field at the bottom of the panel. ○ Click *Set the Same Band Count for All Variables*. ○ Click *OK*.
modify the band count for the variable being edited	○ Type in the desired band count in the column *Band Count* at the top of the panel. ○ Click *OK*.

5.1.7.1.2.6.2 Optimal Grouping for All Variables

When working with a defined structure, if want to keep your categories as they are defined for the model building, you must disable this option.

If not or if you work with no defined structure, *Optimal Grouping* allows in a large number of cases to increase the prediction confidence of the model with a minimal loss of predictive power. Where possible, similar adjacent segments are gathered to reduce artifacts between the training and validation datasets.

To Enable Modeler - Data Encoding Optimal Grouping for All Variables

- Right click the row corresponding to the variable to be edited.
- Select *Define Structure*.
- Select *Optimal Grouping*, in such a way that the option is checked.

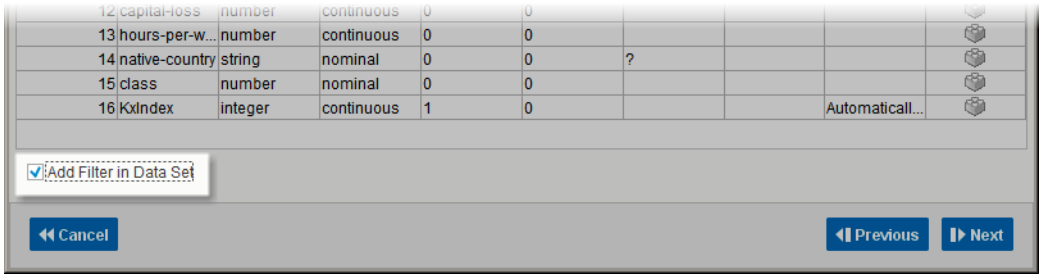
5.1.7.1.3 Filtering the Dataset

In order to accelerate the learn process and to optimize the resulting model, you can apply a filter to your dataset.

> **i Note**
>
> For this scenario, do not use the filtering option.

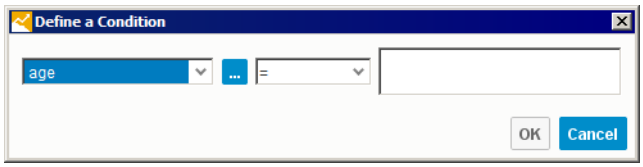
5.1.7.1.3.1 To Filter a Dataset

1. Check the option *Add a Filter in Dataset*.
2. Click *Next*.



5.1.7.1.3.2 To Add a Condition

1. Click the button *Add a Condition*. The window *Define a Condition* opens.



2. Choose a variable in the first list.
3. Choose an operator in the second list.
4. Indicate a value in the third list:
   - For a variable with number storage, type a value.
   - For a variable with string storage, choose a variable in the list. If the list is empty click the button  to extract the variable categories.
5. Click *OK*.

> **i Note**
>
> You can edit a condition by double-clicking it.

5.1.7.1.3.3 To Add a Logical Conjunction

Click the button *Add Logic 'And'* or the button *Add Logic 'Or'*.

You can change a conjunction by double-clicking it.

5.1.7.1.3.4 To Change the Order

You can change the order of the nodes to accelerate the filtering process by setting the conditions with the highest probability to be false at the top of the list.

1. Select the node you want to move up or down.
2. Use the buttons ^ and ˅ to change its position in the filter.

5.1.7.1.3.5 To Delete a Node

1. Select the node you want to delete.
2. Click the button *Remove Selected Node*.

5.1.7.1.3.6 To Display the Filtered Dataset

You can visualize the dataset that you will obtain after the application of the filter.

Click the button *View Data*.

A pop-up window opens.

5.1.7.1.3.7 To Save a Filter

You can save the filter you created to be able to reuse it at a later moment without being obliged to recreate the same conditions.

1. Click the button *Save Filter*.

   A pop-up window is displayed.
2. In the list *Data Type*, select the format in which you want to save the filter.
3. Use the *Browse* button located on the right of the *Folder* field to select the folder or database where you want to save the filter.
4. In the *File Name* field, enter the name of the file or table in which you want to save the filter.
5. Click the *OK* button.

5.1.7.1.3.8 To Load an Existing Filter

To apply a filter to the dataset, you can use a file created during a previous use of the dataset in Automated Analytics.

1. Click the button *Load Existing Filter*.

   A pop-up window is displayed.
2. Use the list *Data Type* to select the format of the filter.
3. Use the *Browse* button located on the right of the *Folder* field to select the folder or the database in which the filter is stored.
4. Use the *Browse* button located on the right of the *File* field to select the file or the table containing the filter.
5. Click the *OK* button.

5.1.7.1.4 Translating the Variable Categories

You can translate the categories of a nominal variable, save the translation or load an existing translation. This translation has no influence on the variable structure, which has to be set according to the original values of the variable.

> **i Note**
>
> The variable "Target Key", which is used in the advanced settings, does not take into account the translation when displaying the possible values of this variable.

5.1.7.1.4.1 To Translate the Variable Categories

1. Click a nominal variable to translate its categories.

2. Go to the *Edition* tab of the ribbon and click the option *Translate Categories*.

   A new window appears.
3. Choose into which languages you want to translate.

   By default, the language of the user interface is displayed as a column.
4. Click the button to extract the variable categories from the dataset.
5. Translate the categories.

   > **i Note**
   >
   > You do not need to fill all fields.
6. Click the *OK* button.

5.1.7.1.4.2 To Save the Categories Translation

1. After translating the variable categories, click the *Save* button.
2. Choose a *Data Type*.
3. Select a *Folder*.
4. Enter a *Name* for the file or table.
5. Click the *OK* button.

5.1.7.1.4.3 To Load an Existing Translation File

1. Click a nominal variable to translate its categories.
2. Go to the *Edition* tab of the ribbon and click the option *Translate Categories*.

   A new window appears.
3. Click the *Load* button.
4. Select the format of the translation in the list *Data Type*.
5. Use the *Browse* button located on the right of the *Folder* field to select the folder or the database in which the description is stored.
6. Use the *Browse* button located on the right of the field *Table* or *File* to select the file or the table containing the description.
7. Click the *OK* button.
8. Click the button *Update* to refresh the display of the categories.
9. If the list of columns is not named correctly, use the *Advanced Settings* to set a header line and update again.
10. Map the language names with those from the loaded translation, by clicking the categories and choosing the corresponding language in the contextual menu.
11. Click the *OK* button.

5.1.7.1.5 Selecting Variables

Once the training dataset and its description have been entered, you must select the following variables:

- One or more Target Variables
- Possibly a Weight Variable
- The Explanatory Variables

5.1.7.1.5.1 Target Variables

For this scenario, select the variable Class as your target variable, that is, the variable that indicates the probability of an individual responding in a positive or negative manner to your campaign.

1. On the screen *Selecting Variables*, in the section *Explanatory variables selected* (left hand side), select the variables you want to use as Target Variables.

   > **i Note**
   >
   > On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2. Click the button > located on the left of the screen section *Target Variables* (upper right hand side).

   The variable moves to the screen section *Target Variables*.

   You can also select a variable in the screen section *Target Variables* and click the button < to move the variables back to the screen section *Explanatory variables selected*.

5.1.7.1.5.2 Weight Variable

Selecting a Weight Variable enables to set the Weight Quantum option available in the Advanced Model Parameters.

For this scenario, do not select a weight variable.

1. On the screen *Selecting Variables*, in the section *Explanatory variables selected* (left hand side), select the variables you want to use as Weight Variables.

   > **i Note**
   >
   > On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2. Click the button > located on the left of the screen section *Weight Variables* (middle right hand side).

   The variable moves to the screen section *Weight Variables*.

   You can also select a variable in the screen section *Weight Variables* and click the button < to move the variables back to the screen section *Explanatory variables selected*.

5.1.7.1.5.3 Explanatory Variables

By default, and with the exception of key variables (such as `KxIndex`), all variables contained in your dataset are taken into consideration for generation of the model. You may exclude some of these variables.

For the first analysis of your dataset, we recommend that you retain all variables. It is particularly important to retain even the variables that seem to have no impact on the target variable. If indeed these variables have no impact on the target variable, the model will confirm this. In the contrary case, the model will allow you to recognize previously unidentified correlations between these variables and the target variable.

Depending on the results obtained from the first analysis, which included all of the variables of the dataset, you can generate a second model by excluding the variables too closely correlated with the target variable.

For this Scenario:

- Exclude the variable `KxIndex`, as this is a key variable. Since the initial dataset does not contain a key variable, the application generated `KxIndex` automatically.
- Retain all the other variables.

1. To exclude some variables from data analysis, on the screen *Selecting Variables*, in the section *Explanatory Variables Selected* (left hand side), select the variable to be exluded.

   > **i Note**
   >
   > On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2. Click the button > located on the left of the screen section *Excluded Variables* (lower right hand side). The variable moves to the screen section *Excluded Variables*. Also, select a variable in the screen section Excluded Variables and click the button < to move the variables back to the screen section *Explanatory Variables Selected*.

   By default, any variable defined as a key is put in the *Excluded Variables* . However, the user has the possibility to move a key variable in the *Explanatory Variables Selected* if he wants this variable to have this role.

3. Click *Next*.

   The screen *Parameters of the Model* appears.

5.1.7.1.6 Checking Modeling Parameters

The screen *Summary of Modeling Parameters* allows you to check the modeling parameters just before generating the model.

> **i Note**
>
> The screen *Summary of Modeling Parameters* contains an *Advanced* button. By clicking this button, you access the screen *Advanced Model Parameters*. For more information about these parameters, see the related topic.

- The *Model Name* is filled automatically. It corresponds to the name of the target variable (class for this scenario), followed by the underscore sign (_) and the name of the data source, minus its file extension (`Census01` for this scenario).
- You have the possibility to display the results as a decision tree based on the five most contributive variables. To activate this option, check the box *Compute Decision Tree*.
- The *Autosave* button allows you to activate the feature that will automatically save the model once it has been generated. When the autosave option is activated, a green check mark is displayed on the *Autosave* button.

5.1.7.1.6.1 Activating the Autosave Option

The *Model Autosave* panel allows you to activate the option that will automatically save the model at the end of the generation process and to set the parameters needed when saving the model.

To activate the option, proceed as follows:

1. In the *Summary of Modeling Parameters* panel, click the *Autosave* button.
   The *Model Autosave* panel is displayed.
2. Check *Enable Model Autosave*.
3. Set the parameters listed in the following table.

Parameter	Description
Model Name	This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
Description	This field allows you to enter the information you want, such as the name of the training dataset used, the polynomial degree or the performance indicators obtained. This information could be useful to you later for identifying your model. Note that this description will be used instead of the one entered in the *Summary of Modeling Parameters* panel.
Data Type	This list allows you to select the type of storage in which you want to save your model. The following options are available: ○ *Text Files*, to save the model in a text file. ○ *Data Base*, to save the model in a database. ○ *Flat Memory*, to save the model in the active memory.
Folder	Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
File/Table	This field allows you to enter the name of the file or table that is to contain the model. When saving the model as a text file, you must enter one of the following format extensions `.txt` (text file in which the data is separated by tabs) or `.csv` (text file in which the data is separated by commas).

4. Click *OK*.

5.1.7.1.7 Setting the Advanced Parameters

On the screen *Summary of Modeling Parameters*, click the *Advanced* button.

The screen *Advanced Model Parameters* appears.

5.1.7.1.7.1 General Tab

The *General* tab allows you to define the general settings of the model, that is, the degree of the model, the score bin count, the number of correlations to display and the target key value.

5.1.7.1.7.1.1 Defining the Degree of the Model (optional)

The model generated by Modeler - Regression/Classification is represented by a polynomial. This polynomial may be of degree 1, 2, 3 or greater. By defining the polynomial degree, you will define the degree of complexity of the model.

It is greatly recommended that you always use a degree of "1" (default value) for the first analysis of a dataset. Using a higher degree of polynomial does not guarantee that you will in all cases obtain a more powerful model.

For more information about the polynomial degree, see the related topic.

For this scenario, keep the polynomial degree set to the default value – that is "1".

To define the degree of the model, in the *Polynomial degree* field, enter the value corresponding to the degree of complexity of the model that you want to obtain.

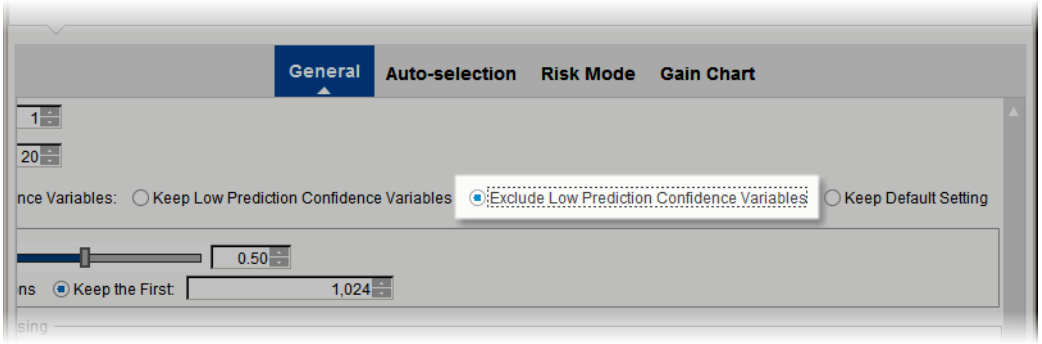
5.1.7.1.7.1.2 Setting the Score Bin Count

This option allows you to define the number of bins to create for the score. This value must be set between 20 and 100 since a lower or higher number of bins would lead to poor model quality.

5.1.7.1.7.1.3 Exclusion of Low KR Variables

This option allows you to enable the exclusion of variables based on the value of their prediction confidence (KR). The application uses an internally computed threshold to decide whether a variable has a low prediction confidence. This threshold depends mainly on the dataset size and target distribution.

To automaticallyeExclude variables with low prediction confidence, check the option *Exclusion of Low Prediction Confidence Variables*.



5.1.7.1.7.1.4 Defining the Number of Correlations to Display

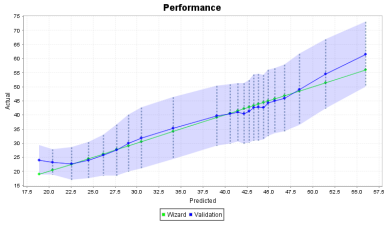
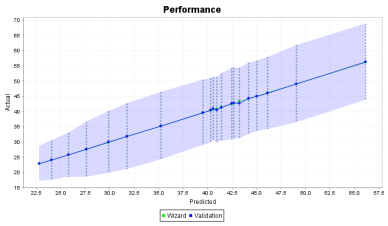
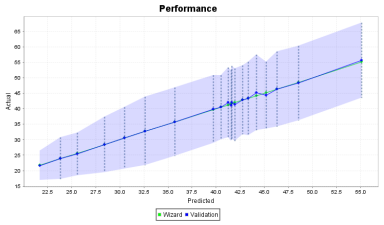
The section *Correlations Settings* allows you to set the parameters for the *Correlation* debriefing panel. That is, to select how many correlations should be displayed in that panel.

To say that variables are correlated implies that they each contribute some of the same information with respect to the target variable. A correlation contains two variables and a correlation rate. When you modify the number of correlations to display, the engine excluded the ones with the lowest correlation rate, thus keeping only the more significant ones.

5.1.7.1.7.1.5 Enabling the Post-Processing

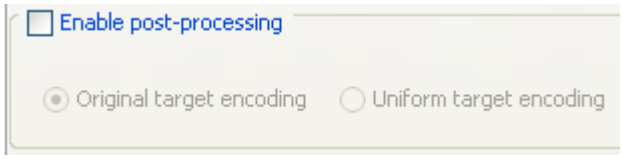
This section allows setting some regression parameters according to three strategies. This option can only be activated when the model contains at least one continuous target variable.

The description of these strategies and an example of performance curve for each strategy are provided in the table below.

Regression Strategy	Description	Example of Performance Curve
Without post-processing	The first strategy consists in disabling the regression post-processing during the learning model phase in order to create a regression similar to the one used in previous versionsof the application. In this case, a standard regression is performed. No special improvement is made to the final scores. Original target values are used and raw score values are produced as outputs.	
With original target encoding	The second strategy, which applies to regressions using a post-processing, consists in using the original target value during the learning model phase to compute regression coefficients. The result of the regression is then transformed to align target segment means and score segment means in the post-processing phase. **i Note** This is the default strategy.	
With uniform target encoding	The last strategy, which applies to regressions using a post-processing, consists in using first an encoded target value instead of the original target value during the learning model phase in order to have a uniform distribution: it is the pre-processing phase. Then, regression coefficients are computed and scores are transformed in the original target space during the post-processing phase. **i Note** This strategy is to be preferred when the default strategy does not produce models with enough quality, which is often the case with very skewed target distributions.	

5.1.7.1.7.1.5.1 Regression Without Post-processing

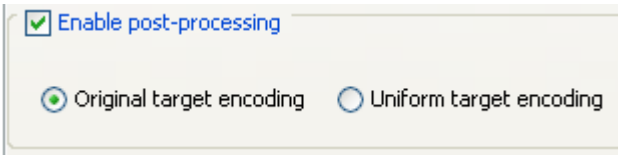
Uncheck the option *Enable Post-processing*.



> **i Note**
>
> It is not possible to change the target encoding strategy when the post-processing is disabled.

5.1.7.1.7.1.5.2 Regression with Original Target Values

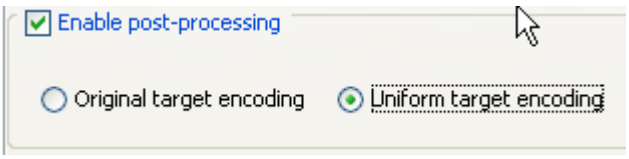
1. Check the option *Enable Post-processing*.
2. Select the radio button *Original target encoding*.



> **i Note**
>
> This strategy is set by default.

5.1.7.1.7.1.5.3 Regression with Uniform Target Encoding

1. Check the option *Enable Post-processing*.
2. Select the radio button *Uniform target encoding*.



5.1.7.1.7.1.6 Defining the Target Key Values

For the binary targets you have the option to select which value is the key category for each target. By default, the category selected by the application is the least represented in the dataset.

The *Advanced Model Parameters* screen lists all the binary targets of the current model allowing you to define the key category for each target, that is the expected value of the target.

In this scenario, do not define a value for the target variable. The application will automatically select "1" as the key category for the Class variable.

To define the key category value for a target variable, in the *Target Key* field corresponding to the chosen target, enter the key value.

5.1.7.1.7.2 Auto-selection Tab

The *Auto-selection* tab allows you to define the parameters of the automatic variable selection.

Setting the Variables Auto-Selection

The section *Auto-selection* allows you to automatically reduce the number of variables in the model in relation to quality criteria. This selection is done by successive iterations. There are two selection modes, one based on the number of variables to keep, and the other on the amount of information that should be kept. In this instance, the information is the sum of the variables contributions.

To use the auto-selection, check the box *Enable Auto-Selection*. The corresponding options are activated. By default the parameters are set to: *Select the best model keeping between 1and all variables*. Any parameter that can be changed is marked as a hyperlink ((blue, underlined).

Choosing the Selection Mode

To Select the Selection Mode

1. Click the link indicating the type of model to keep. For example, the best model in the sentence: *Select the best model keeping between 1and all variables*.
   A drop-down menu is displayed offering the following options:
   - *the best model*
   - *the last model*
2. Select the desired option.
3. Click *OK*.

Selecting the number of variables is mandatory and allows fixing the minimum and the maximum number of variables in the final model. To select the number of variables

1. In the sentence defining the number of variables *Select the best model keeping between 1and all variables*, select the minimum number of variables (for example, 1 variable) and the maximum number of variables (for example, all variables) .
2. For the minimum number of variables, a slide is displayed ranging from 1 to the total number of variables in the model.
   - Move the cursor on the slide to select the quantity of your choice.

- For the maximum number of variables you can either confirm the minimum number selected previously by choosing *Keep all variables*, or choose a maximum number of variables.

3. Click *OK*.

Choosing the Stopping Criteria

Choose between two variable selection parameters:

- *Each step removes 1 variable*
  This option allows you to set the number of variables that should be excluded at each iteration.
- *Each step keeps 95.0% of information*
  This option allows you to set the amount of information that should be kept at each iteration (thus limiting the loss of information).

To select the number of variables:

1. Click the link indicating the number of variables in the sentence *Each step removes 1 variable*. A slide is displayed ranging from 1 to the total number of variables in the model.
2. Move the cursor on the slide to select the number of your choice.
3. Click *OK*.

To select the information amount:

- Click the link indicating the amount of information to keep in the sentence *Each step keeps 95.0% of information*. A slide is displayed.
- Move the cursor on the slide to select the quantity of your choice.
- Click *OK*.

The quality loss can be set in the sentence *Search process stops with a drop of 1.0% of KI and KR*. To set the authorized quality loss:

1. Click the link indicating the percentage of loss (for example, 5.0%). A slide is displayed.
2. Select the maximum percentage of authorized quality loss with the cursor.
3. Click *OK*.
4. Click the quality criterion. A drop-down list is displayed offering the following options:
   - Based on *KI + 2 KR*, the quality loss is based on both the predictive power (KI) and twice the prediction confidence (KR).
   - *KI and the KR*, the quality loss is limited for both the predictive power (KI) and the prediction confidence (KR). It is the default value.
   - *KI*, the quality loss is limited for the predictive power (KI) only.
   - *KR*, the quality loss is limited for the prediction confidence (KR) only.
5. Select the option of your choice.
6. Click *OK*.

5.1.7.1.7.3 Risk Mode Tab

This tab allows you to select a specific learning mode for your model.

To Enable the Risk Mode

1. Select the tab *Risk Mode*.
2. Check the box *Enable*. The tab activates and the *Risk Mode* settings are displayed.

5.1.7.1.7.3.1 Setting Risk Mode

Risk Mode allows advanced users to ask a classification model to translate its internal equation obtained with no constraints into a specified range of scores associated with good/bad odds ratio.

When this mode is activated, the different encodings that are used internally for continuous and ordinal variables are merged in a single representation, allowing a simpler view of the model internal equations. This is particularly useful when the usage of predictive model is subject to legal restrictions: the model equations are now simple enough to be understood by legal departments, and can be exposed, not only in programming language, as it was already the case before, but even in simple words.

The underlying technology is also used to display so called 'score cards'.

To use this mode, you need to choose:

- a *Risk Score* associated with a *Good/Bad Odds* ratio

  > **i Note**
  >
  > The odds ratio is the ratio between 'good' and 'bad', i.e. (1-p)/p where p is the probability of risk.

- the number of *Points to double the odds*

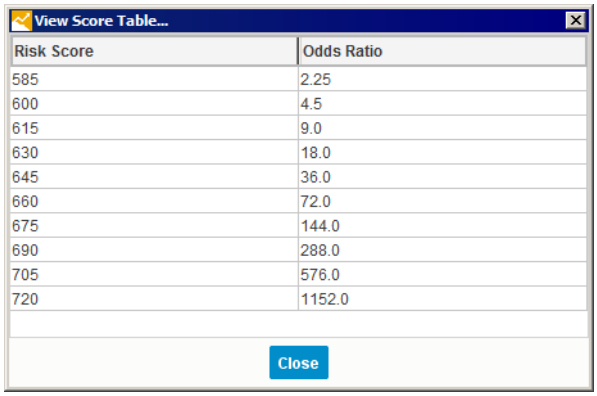
  > **i Note**
  >
  > The points to double the odd (PDO) are the number of risk points needed to double the odds ratio.

For example, considering a *Risk Score* equal to 615, an odds ratio of 9 :1 and 15 points to double the odds. In this case, the application automatically re-scales the internal scores to scores in *Risk Mode* space and associates an odds ratio to each score in *Risk Mode* space.

In this scenario, do not activate the Risk mode.

To Define the Risk Mode Parameters

1. In the field *Risk Score*, enter the score you want to associate with a good/bad odds ratio.
2. In the field for *good/bad odds ratio of*, enter the ratio.
3. Indicate the increase of score points needed to double the odds in the field *Points to double odds*.
4. Click the button *View Score Table* to display the table of scores associated with the corresponding good/bad odds ratio.



View Score Table...

Risk Score	Odds Ratio
585	2.25
600	4.5
615	9.0
630	18.0
645	36.0
660	72.0
675	144.0
690	288.0
705	576.0
720	1152.0

Close

5.1.7.1.7.3.2 Risk Fitting Domain

This option allows the user to control the way risk score fitting is performed, that is, how InfiniteInsight® fits its own scores to the risk scores.

The risk fitting has two modes:

- *PDO based*: the area equals [Median Score - N*PDO ; Median Score + N*PDO] . N (number of PDOs around the median score) must be specified by the user. By default, N is set to 2 .

  > **i Note**
  >
  > PDO stands for Points to double the odds.

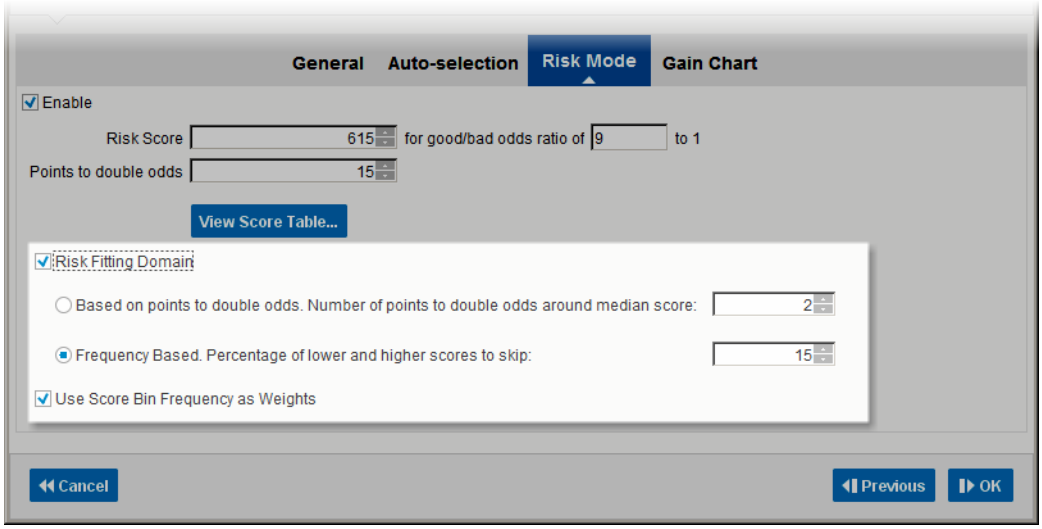
- *Frequency based*: the area equals[Quantile(Freq) ; Quantile(1.0 - Freq)] . The frequency of higher and lower scores to be skipped must be specified by the user. By default, the frequency is set to 15% .

If you do not check the box *Risk Fitting Domain*, the mode Frequency based will be used by default.

The fitting can be weighted or not.

5.1.7.1.7.3.2.1 To Set the Risk Fitting Parameters

1. Check the box Risk Fitting Domain.



2. Select the mode you want to use.
3. Depending on the selected mode, set the appropriate value in the corresponding field.
4. If you want to use weights for the fitting, check the box *Use Score Bin Frequency as Weights*.

5.1.7.1.7.4 Gain Chart Tab

This tab allows you to compute the gain chart on the training dataset, that is to rank your data in order of descending scores and split it into exact quantiles (decile, vingtile, percentile). This option can be useful to check the model performance on the validation dataset.

To Compute the Gain Chart

1. Select the tab *Gain Chart*.
2. Check the box *Compute Gain Chart on Training Data*.
3. In the list, select the *Number of Quantiles* you want your data to be segmented in.
4. You can add additional variables in order to estimate profits per segments of the population:
   1. In the *Variables* list, select the variables you want to add to the gain chart. Use the `CTRL` key to select multiple variables.
   2. Click the > button to add the selected variables to the list *Values for Gain Chart*.
5. The sum of each selected variable will be calculated for each segment of the population.
6. Click *OK* to save the advanced parameters and go back to the panel *Summary of Modeling Parameters*.

The result of the gain chart computation is available after the model has been generated in the *Statistical Reports* panel, section *Model Performance*.

5.1.7.2 Step 2 - Generating and Validating the Model

Once the modeling parameters are defined, you can generate the model. Then you must validate its performance using the predictive power (KI) and the prediction confidence (KR):

- If the model is sufficiently powerful, you can analyze the responses that it provides in relation to your business issue (see Step 3 - Analyzing and Understanding the Model Generated), and then apply it to new datasets (see Step 4 - Using the Model).
- Otherwise, you can modify the modeling parameters in such a way that they are better suited to your dataset and your business issue, and then generate new, more powerful models.

5.1.7.2.1 Generating the Model

To generate the model:

1. On the *Advanced Model Parameters* screen, click *Generate*.

   The screen *Training the Model* appears. The model is being generated. A progress bar allows you to follow the process.
2. If the *Autosave* option has been activated in the panel *Summary of Modeling Parameters*, a message is displayed at the end of the learning process confirming that the model has been saved.
3. Click *Close*.
4. Once the model has been generated, click *Next* to go to panel *Using the Model*.

5.1.7.2.2 Following the Progress of the Generation Process

There are two ways for you to follow the progress of the generation process:

- The Progress Bar displays the progression for each step of the process. It is the screen displayed by default.
- The Detailed Log displays the details of each step of the process.

To display the Progression Bar

Click *View Type* and select (Progress).

The progression bar screen appears.

To Display the Detailed Log

Click *View Type* and select the (Log) button.

The detailed log displays the details of each step of the process.

To Stop the Learning Process

1. Click the (Stop Current Task) button.
2. Click the *Previous* button.
   The screen *Summary of Modeling Parameters* appears.
3. Go back to the section on checking modeling parameters.

5.1.7.2.3 Validating the Model

Once the model has been generated, you must verify its validity by examining the performance indicators:

- The predictive power allows you to evaluate the explanatory power of the model, that is, its capacity to explain the target variable when applied to the training dataset. A perfect model would possess a predictive power equal to 1 and a completely random model would possess a predictive power equal to 0.
- The prediction confidence defines the degree of robustness of the model, that is, its capacity to achieve the same explanatory power when applied to a new dataset. In other words, the degree of robustness corresponds to the predictive power of the model applied to an application dataset.

For this scenario, the model generated has the following performance indicators:

- A quality indicator KI equal to 0.808,
- A robustness indicator KR equal to 0.992.

The model performs sufficiently well. You do not need to generate another.

To Validate the Model Generated

1. Verify the Predictive Power (KI) and Prediction Confidence (KR) of the model.
   - If the performance of the model meets your requirements, go to Step 3 - Analyzing and Understanding the Model Generated.
   - Otherwise, go to the procedure To Generate a New Model.
2. You can also check other indicators provided in addition to KI and KR during the model generation. For example, you could view the total elapsed time required to generate the model and information on the standard error rate.

To Generate a New Model

You have two options. On the screen *Training the Model*, you can:

- Either click the *Previous* button to return to the modeling parameters defined initially. Then you can modify the parameters one by one.
- Or click the *Cancel* button to return to the main screen of the modeling assistant. Then you must redefine all the modeling parameters.

5.1.7.3 Step 3 - Analyzing and Understanding the Model Generated

The suite of plotting tools within the application allows you to analyze and understand the model generated:

- The performance of the model with respect to a hypothetical perfect model and a random type of model,
- The contribution of each of the explanatory variables with respect to the target variable,
- The significance of the various categories of each variable with respect to the target variable.

5.1.7.3.1 Presentation of the User Menu

Once the model has been generated, Click *Next*. The screen *Using the Model* appears.

The screen *Using the Model* presents the various options for using a model, that allow you to:

- Display the information relating to the model just generated or opened (*Display* section), referring to the model curve plots, contributions by variables, the various variables themselves, HTML statistical reports, table debriefing. Some information is only displayed upon request from the user: the display of Modeler - Regression/Classification results as a decision tree, which can be specified in the modeling parameters before the model generation, or the display of model parameters, which can be requested in the general user options.
- Apply the model just generated or opened to new data, to run simulations, and to refine the model by performing automatic selection of the explanatory variables to be taken into consideration (*Run* section).
- Save the model, or generate the source code (*Save/Export* section).

5.1.7.3.2 Model Overview

The *Model Overview* screen displays the same information as the training summary.

Overview

Name	Name of the model, created by default from the target variable name and the dataset name
Dataset	Name of the dataset
Initial Number Variables	Number of explanatory variables used
Number of Selected Variables	Number of explanatory variables actually used by the resulting model.
Number of Records	Number of records in the dataset
Building Date	Date and time when the model was built
Learning time	Total learning time
Engine name	Depending on the feature used: • `Kxen.RobustRegression` • `Kxen.SmartSegmenter` • `Kxen.TimeSeries` • `Kxen.AssociationRules` • `Kxen.EventLog` • `Kxen.SequenceCoder` • `Kxen.SocialNetwork`

Modeling Warnings

Monotonic Variables Detected	Indicates if monotonic variables have been found in the dataset, that is, variables which direction of variation is constant, in the reading order of the data in the training dataset.
Suspicious Variables Detected	This report presents a list of variables that are considered to be suspicious. These suspicious variables have a predictive power over 0.9, they are very correlated to the target variable. This means these variables probably bring a biased information and should not be used for the modeling. A special attention should be taken towards those variables. A more detailed report lists which variables exactly are suspicious and at which extent (see *Statistical Reports* > *Expert Debriefing* > *Suspicious Variables*).

Targets

For each nominal variable

<Name>	Name of the target variable

Target key	Wanted target value
Target categories Frequency	Percentage of all the target value in the Training dataset, when dealing with a nominal target

For each continuous target variable:

<TargetName>	Name of the target variable
Min	Minimum value found for the target variable in the Training dataset
Max	Maximum value found for the target variable in the Training dataset
Mean	Mean of the target variable values on the Training dataset
Standard deviation	Mean of the distance between the target values and the Mean

Performance Indicators

For each target

*rr_*<TargetName> *kc_*<TargetName>	Target name Note that *rr_* indicates a regression/classification and *kc_* indicates a segmentation/clustering.
Predictive Power (KI)	Quality indicator that corresponds to the proportion of information contained in the target variable that the explanatory variables are able to explain.
Prediction Confidence (KR)	Robustness indicator that signifies the capacity of the model to achieve the same performance when it is applied to a new dataset exhibiting the same characteristics as the training dataset.

5.1.7.3.2.1 Model Overview Options

To Copy the Model Overview

Click the  (*Copy*) button.

The application copies the HTML code of the screen. You can paste into a word processing or spreadsheet program, a text editor, ...

To Save the Model Overview

Click the  (*Save*) button situated under the title.

The file is saved in HTML format.

To Print the Model Overview

1. Click the  (*Print*) button situated under the title.
   A dialog box appears, allowing you to select the printer to use.
2. Select the printer to use and set other print properties if need be.
3. Click OK.
   The report is printed.

To Export to PowerPoint

Click the 

(*Export to PowerPoint*) button.

5.1.7.3.3 Model Graphs

Depending on the type of the target, the model graph plot allows you to:

- View the realizable profit that pertains to your business issue using the model generated when the target is nominal.
- Compare the performance of the model generated with that of a random type model and that of a hypothetical perfect model when the target is nominal.
- Compare the predicted value to the actual value when the target is continuous.

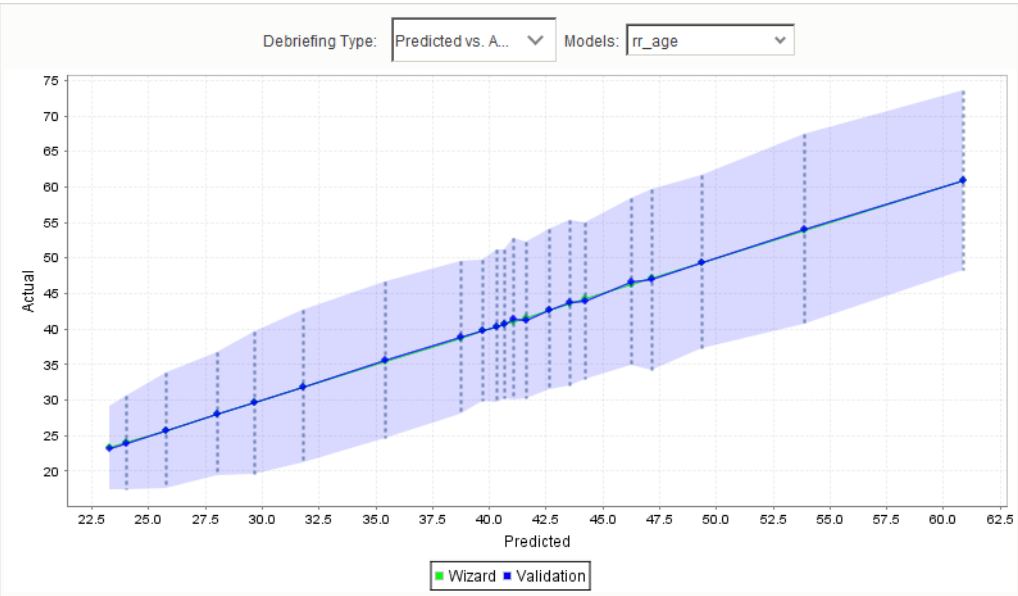
On the plot, for each type of model, the curves represent:

- When the target is nominal, the realizable profit (on the Y axis) as a function of the ratio of the observations correctly selected as targets relative to the entire initial dataset (on the X axis) .
- When the target is continuous, the predicted value or score (on the X axis) in respect with the actual value or target (on the Y axis) .

To Display the Model Graph

1. On the screen *Using the Model*, click the *Model Graphs* option.
   The model graphs appears. When the target is nominal, the following curve is displayed:

   The default parameters display the profit curves corresponding to the *Validation* sub-set (blue line), the hypothetical perfect model (*Wizard*, green line) and a random model (*Random*, red line). The default setting for the type of profit parameter is *Detected* profit, and the values of the abscissa are provided in the form of a percentage of the entire dataset.
   When the target is continuous, the following curve is displayed:



   The default parameters display the curves corresponding to the *Validation* sub-set (blue line) and the hypothetical perfect model (*Wizard*, green line). The blue area represents the standard deviation of the current model. For more information on the meaning of model curves, see the topic on understanding model graphs.

2. When there is more than one target, select the target for which you want to see the curves in the *Models* list.

> **i Note**
>
> To each variable corresponds a model. The name of each model is built from the *rr_* (*Robust Regression*) prefix and the model target name.

3. Select the viewing options that interest you.

5.1.7.3.3.1 Plot Options

To Display the Graphs for the Training, Validation, and Testing Sub-sets

Click *Dataset* and select one of the following options that allow you to switch between:

- the graph for the *Validation* sub-set: 
- the graphs for all the sub-sets: 

To Change the View Type

Click *View Type* and select the desired option.

To Copy the Model Graph

Click the  (*Copy*) button and select the desired option.

The application copies the parameters of the plot. You can paste it into a spreadsheet program (such as Excel) and use it to generate a graph.

To Save the Model Graph

1. Click the  (*Save*) button.
   A dialog box appears, allowing you to select the file properties.
2. Type a name for your file.
3. Select the destination folder.
4. Click OK.

The plot is saved as a PNG formatted image.

To Print the Model Graph

1. Click the  (*Print*) button situated under the title.
   A dialog box appears, allowing you to select the printer to use.
2. Select the printer to use and set other print properties if need be.
3. Click OK.
   The report is printed.

To Export the Model Graph to Microsoft Excel

Click the  (*Export to Excel*) button situated under the title. An Excel sheet opens containing the model graph you are currently viewing along with its data.

To Open the Current Graph in a New Window

Click the  (*Pin View*) button. The current graph is displayed in a new window.

5.1.7.3.3.1.1 Plot Options for a Nominal Target

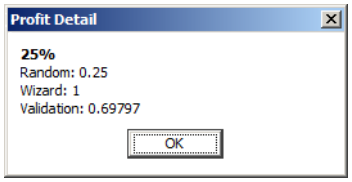
On the model curve plot, different options allow you to visualize:

- Exact profit values for a point for all the displayed curves.
- The curves for the different profit types: Detected, Lift, Normalized, and Customized.

For more information on profit types, see the related topic.

To Display the Exact Profit Values for a Given Point

On the screen *Model Curves*, on the plot, click a point on one of the curves presented. For instance, by clicking a point on any one of the curves whose value on the abscissa is 25%, the exact profit values appear.



To Select a Profit Type

1. On the screen Model Curves, beneath the plot, click the drop-down list associated with the Profit field.
   The list of profit types appears.
2. Select a profit type.
   The corresponding profit curves appear.

5.1.7.3.3.1.2 Plot Options for a Continuous Target

To Display the Exact Profit Values for a Given Point

On the screen *Model Graphs*, on the plot, click a point on one of the curves presented.

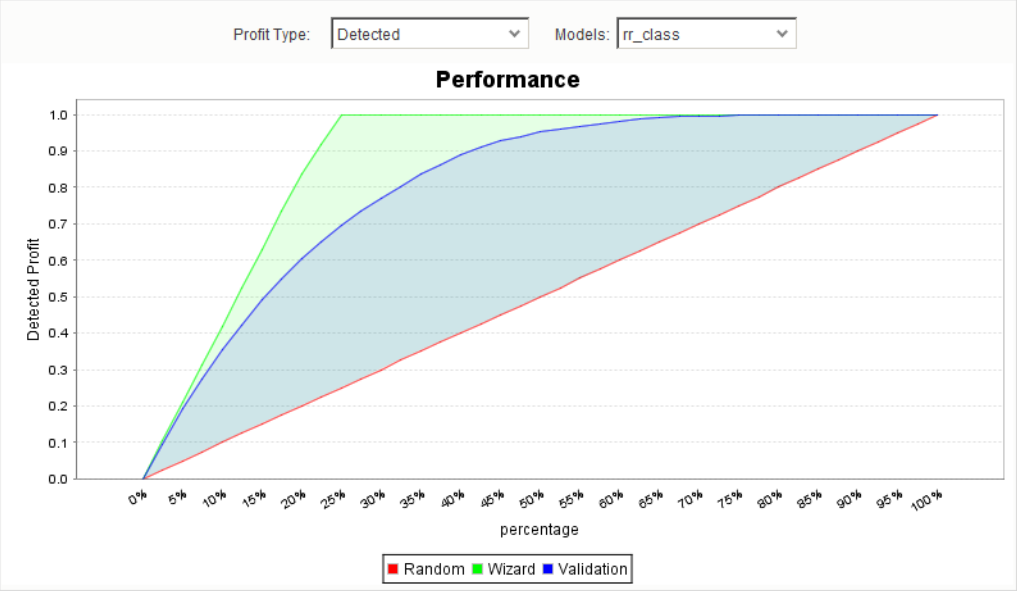
To Select the Debriefing Type

1. On the screen Model Graphs, above the plot, click the drop-down list associated with the Debriefing Type field.
   The list of debriefing types appears.
2. Select a debriefing type.
   The corresponding plot appears.

5.1.7.3.3.2 Understanding Model Graphs

5.1.7.3.3.2.1 Understanding Graphs for a Model with a Nominal Target

The following figure represents the model graph produced using the default parameters.



On the plot, the curves for each type of model represent the profit that may be realized (Y axis), that is, the percentage of observations that belong to the target variable, in relation to the number of observations selected from the entire initial dataset (X axis). On the X axis, the observations are sorted in terms of decreasing "score", that is, the decreasing probability that they belong to the target category of the target variable.

In the application scenario, the model curves represent the ratio of prospects likely to respond in a positive manner to your marketing campaign, relative to the entire set of prospects contained in your database.

Detected profit is the default setting for type of profit. Using this type of profit:

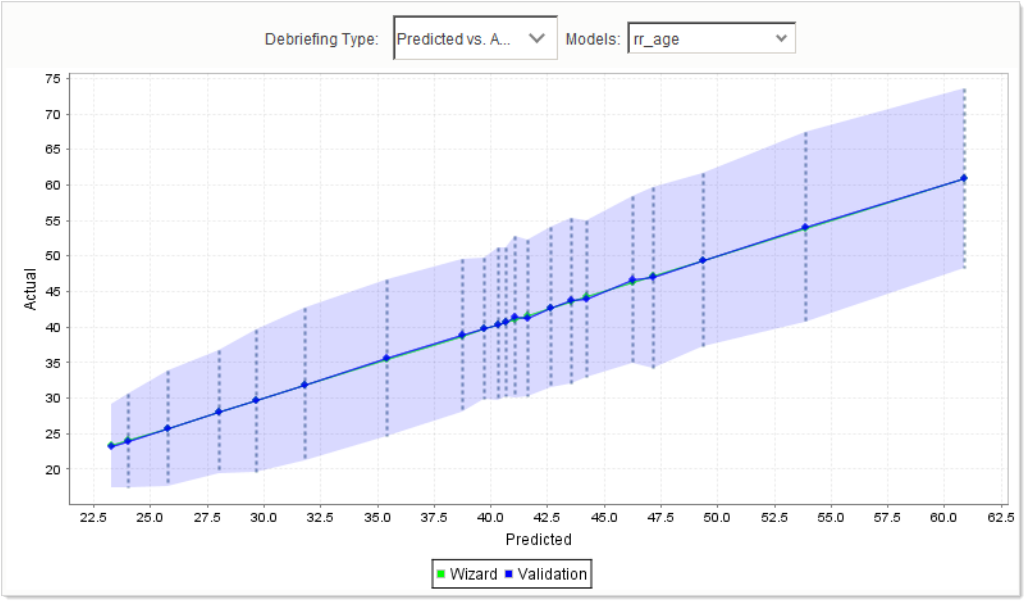
- The value "0" is assigned to observations that do not belong to the target category of the target variable,
- The value "1/(frequency of the target variable in the dataset)" is assigned to observations that do belong to the target category of the target variable.

The following table describes the three curves represented on the plot created using the default parameters.

The curve...	Represents...	For instance, by selecting...
Wizard (green curve, at the top)	The profit that may be achieved using the hypothetical perfect model that allows one to know with absolute confidence the value of the target variable for each observation of the dataset	25% of the observations from your entire dataset with the help of a perfect model, 100% of observations belonging to the target category of the target variable are selected. Thus maximum profit is achieved. Note that these 25% correspond to the proportion of prospects who responded in a positive manner to your marketing campaign, during your test phase. For these prospects, the value of the target variable, or profit, is equal to 1.
Validation (blue curve, in the middle)	The profit that may be achieved using the model generated by Modeler - Regression/Classification that allows one to perform the best possible prediction of the value of the target variable for each observation of the dataset	25% of the observations from your initial dataset with the help of the model generated, 66.9% of the observations belonging to the target category of the target variable are selected.
Random (red curve, at the bottom)	The profit that may be achieved using a random model that does not allow one to know even a single value of the target variable for each observation of the dataset	25% of the initial dataset using a random model, 25% belonging to the target category of the target variable are selected.

5.1.7.3.3.2.2 Understanding Graphs for a Model with a Continuous Target

The following graph represents the model curve plot produced using a continuous target.



The default graphic displays the actual target values as a function of predicted target values. Two curves are displayed: one for the *Validation* sub-set (blue line) and another for the hypothetical perfect model (*Wizard*, green line). The *Validation* curve gives *Actual Target* value as a function of *Predicted Target* value. For example, when the model predicts 35, the average actual value is 37. The *Wizard* curve is just X=Y, meaning that all the predicted values are equal to the actual values. The graph is an easy way to quickly see model error. When the curve is going far from *Wizard*, it means that the predicted value is suspicious.

The graph is computed as follow:

- about 20 segments or bins of predicted values are built. Each of these segments represents roughly 5% of the population.
- for each of these segments, some basic statistics are computed on actual value, such as the mean of the segment (SegmentMean), the mean of the associated target (TargetMean) and the variance of this target within that segment (TargetVariance). For example for predicted value in [17; 19], the mean would be 18.5, the actual target mean would be 20.5 and the actual target variance would be 9. In this case we could say that, if the predicted value is between 17 and 19, the model is underestimating a bit the actual value.

For each curve, a dot on the graph corresponds to the segment mean on the X-axis, and the target mean on the Y-axis.

The blue area represents the expected deviation of the current model. The blue area shows where about 70% of the actual values are expected to be. In other words, it means that, in case of a Gaussian distribution, about 70% of the actual points should be in the blue area (keep in mind that this is a theoretical percentage that may not be observed every time). The default setting for the type of curve parameter is Predicted versus Actual. The extreme values for prediction ranges are {TargetMean - (sqrt(TargetVariance)); TargetMean + (sqrt(TargetVariance))}.

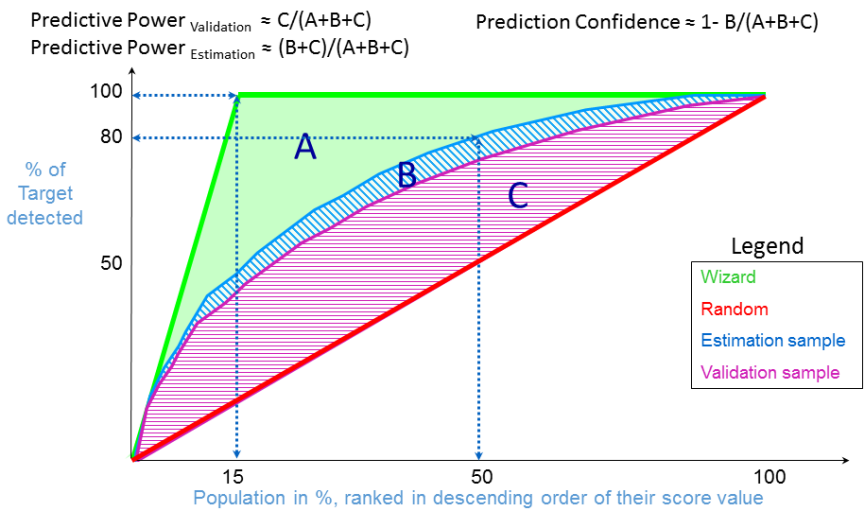
> **i Note**
>
> sqrt(TargetVariance) is equal to the Standard Deviation.

5.1.7.3.3.3 Predictive Power, Prediction Confidence and Model Graphs

On the model graph plot:

- Of the validation dataset (default plot), the predictive power corresponds to "the area found between the curve of the model generated and that of the random model" divided by "the area found between the curve of the perfect model and that of the random model". As the curve of the generated model approaches the curve of the perfect model, the value of the predictive power approaches 1.
- Of the training and validation datasets (select the corresponding option from the list Dataset, located below the plot), the prediction confidence corresponds to one minus "the area found between the curve of the training dataset and that of the validation dataset "divided by "the area found between the curve of the *perfect model* and that of the random model".

The following graph displays the predictive power and the prediction confidence:



5.1.7.3.4 Contributions by Variables

The *Contributions by Variables* plot allows you to examine the relative significance of each of the variables within the model. On this plot, each bar represents the contribution of an explanatory variable with respect to the target variable.

The following four types of plots allow you to visualize contributions by variables:

- *Variable Contributions*, that is, relative importance of each variable in the built model.
- *Variable Weights*, that is, weights (in the final polynomial) of the normalized variables.
- *Smart Variable Contributions*, that is, the variables internal contributions.

- *Maximum Smart Variable Contributions*, that is, the maximum smart variable contributions including only the maximum of similar variables. For example, only binned encoding of the continuous variable age will be displayed. This is the chart displayed by default.

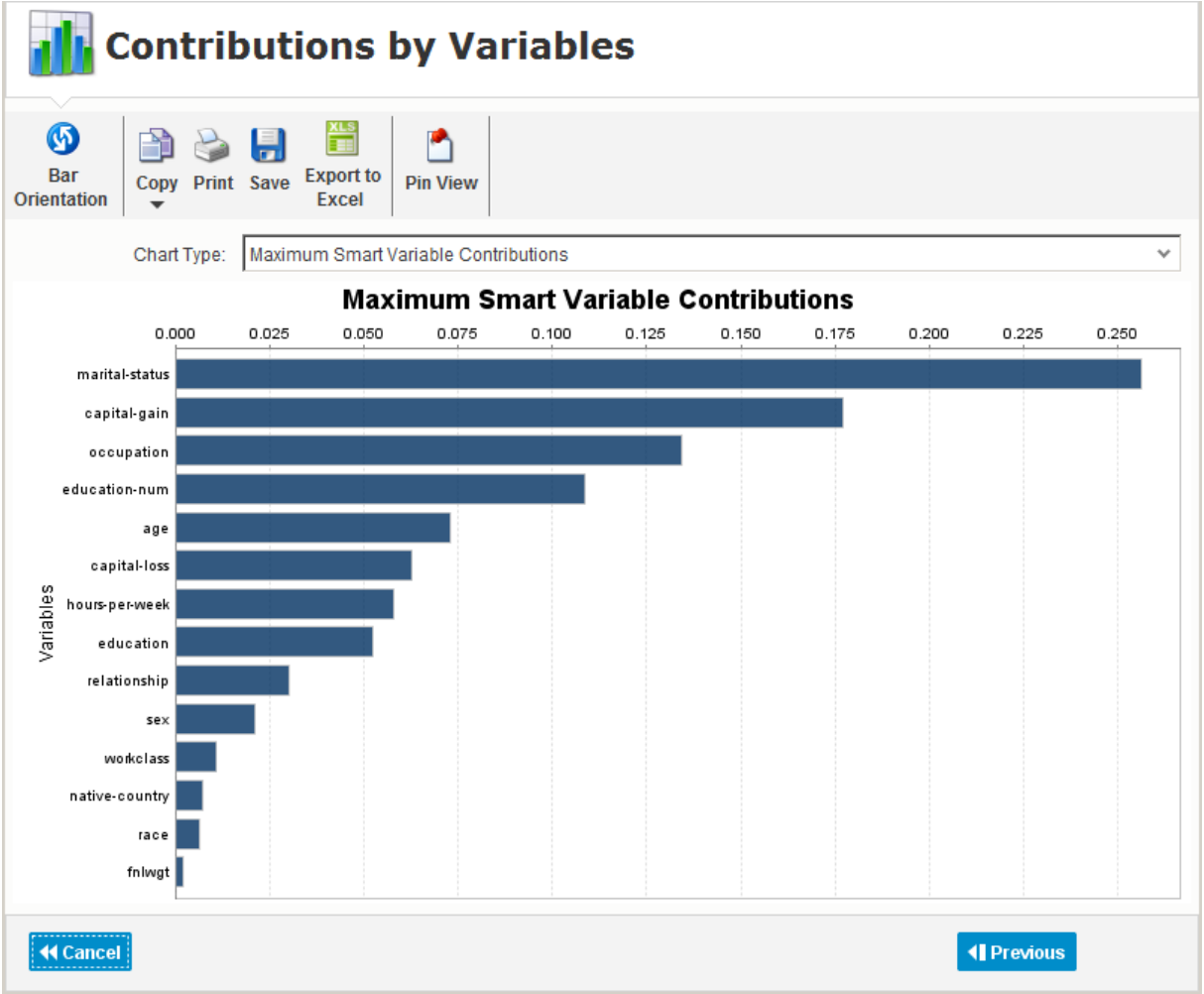
Displaying Contributions by Variables

1. On the screen *Using the Model*, click the option *Contributions by Variables*.
   The plot *Contributions by Variables* appears.The default plot type is *Maximum Smart Variable Contributions.*
   If your dataset contains date or datetime variables, automatically generated variables can appear in this panel. For more information, refer to the section on date and date-time variables in the topic on variable storage formats.
2. You can drill down on a variable, that is, display the plot of details of this variable, where the categories of the variable can be seen. To zoom in on a variable, double-click the corresponding bar. Go to section *Significance of Categories*.

Understanding Contributions by Variables

Only the plot, *Maximum Smart Contributions by Variables*, the default selection, is presented in this guide.

The *Contributions by Variables* option allows the user to examine the relative significance of each of the explanatory variables in relation to the target variable. This significance is relative, as the weight of each variable is pro-rated as a function of the significance of the other explanatory variables.



The plot above corresponds to the model generated, and illustrates the two variables that contribute the most to the target variable, which, in this scenario, are:

- marital-status,
- capital-gain.

In other words, the marital-status and capital-gain variables are those which have the greatest effect on whether a prospect will respond positively or negatively to your marketing campaign. Among all the variables included in the sample dataset, these two are the most discriminatory variables with respect to the target variable *Class*.

Correlated Variables

To say that variables are correlated implies a certain level of redundancy, that they each contribute some of the same information with respect to the target variable. Two variables said to be highly correlated would describe the same information, or the same concept, to an even greater degree.

The plot *Smart Variable Contributions* reflects the correlation that may exist between various explanatory variables. When two variables A and B are strongly correlated:

- Variable A, with a greater contribution than B with respect to the target variable, becomes the "primary variable": the plot displays all its information, including what it has in common with variable B.
- Variable B, with a smaller contribution than A with respect to the target variable, becomes the "secondary variable": only its marginal contribution is displayed on the plot, meaning that only the supplementary contribution to target variable information, or the values that B does not share with A, are displayed. This difference of information is noted [variable_B]-[variable_A] .

Encoded Variables

Creating a model uses not only the original variables but also, in case of continuous or ordinal variables, their value as encoded by the appliation. This is called dual-encoding and allows the application to find all the information contained in each variable.

The encoded variables appear on the variable contributions plots with the prefix c_. For example, the encoded version of a continuous variable named age is noted c_age.

> **i Note**
>
> In Modeler, on the *Data Description* panel, if you enable the *Natural Encoding* for a given variable, its value encoded by the data encoding engine (`c_variableName`) will not be generated.

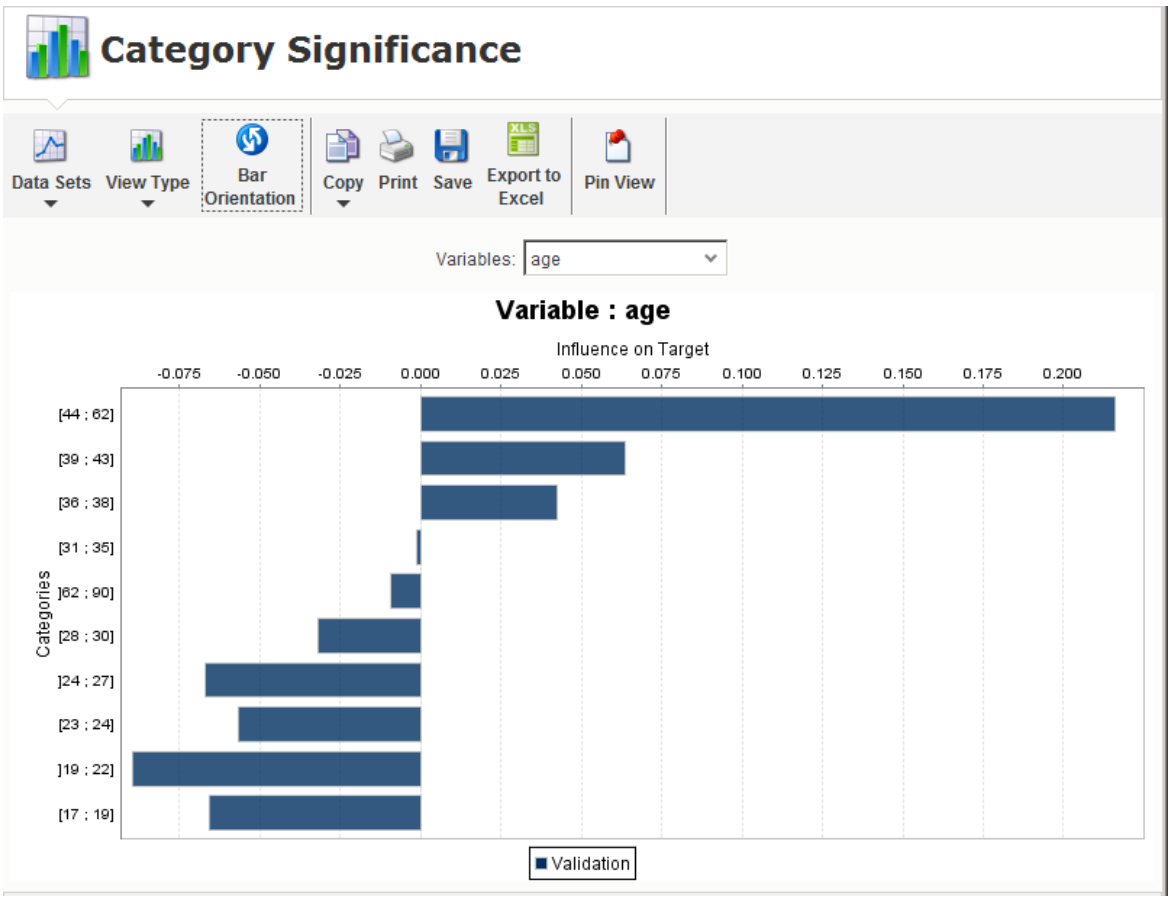
5.1.7.3.5 Category Significance

Definition

The Significance of Categories plot illustrates the relative significance of the different categories of a given variable with respect to the target variable.

Displaying the Significance of Categories Plot

1. On the screen *Using the Model*, click *Category Significance*. The plot *Category Significance* appears.



2. In the *Variables* list located above the plot, select the variable for which you want to display the categories. If your dataset contains date or datetime variables, automatically generated variables can appear in the Variables list. For more information, refer to section on date and date-time variables in the topic on variable storage formats.

> **i Note**
>
> - You can display the relative significance of the categories of a variable directly from the plot Contributions by Variables. On the plot Contributions by Variables, double-click the bar of the variable which interests you.
> - In case no user structure has been defined for a continuous variable, the plot category significance displays the categories created automatically using the band count parameter. The number of categories displayed corresponds to the value of the band count parameter. For more information about configuring this parameter, please refer to the section Band Count for Continuous Variables.

Plot Options

To Switch Between "Validation Dataset" and "All Datasets" Plots

1. Click *Datasets* and select the  (*All Datasets*) button to display all datasets.

The plot displaying all datasets appears.

2. Click *Datasets* and select the  (*Validation Only*) button to go back to the *Validation Dataset* plot.

To Switch between Curve and Bar Charts

1. Click *View Type* and select the  button to display the curve chart. The curve plot appears.
2. Click *View Type* and select the  button to go back to the bar chart.

> **i Note**
>
> You can combine the different types of plot. For example, you can display *All Datasets* in a curve chart or the *Validation Dataset* in a bar chart.

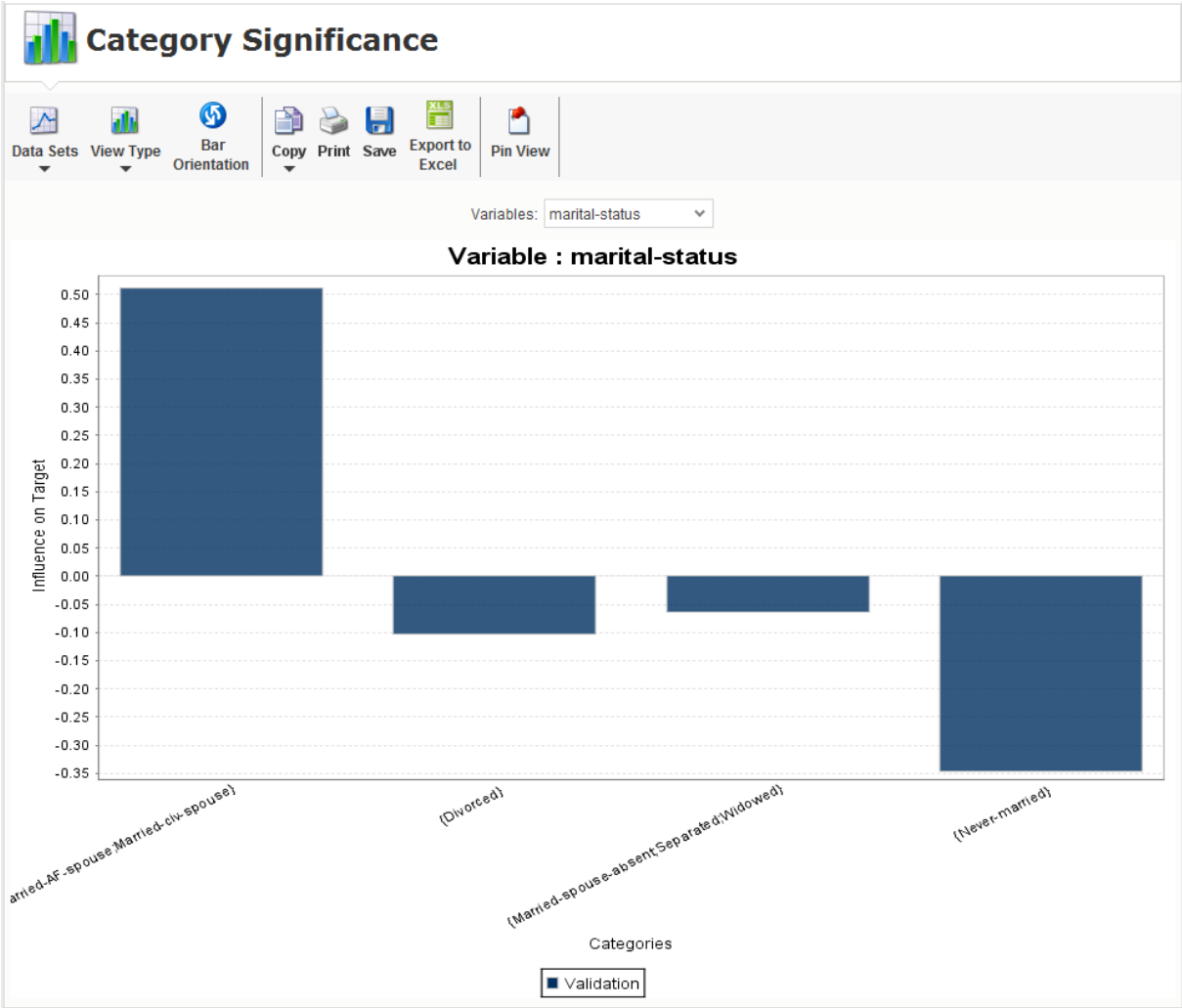
Understanding the Plots of Variables

For this scenario, select the variable `marital-status`, which is the explanatory variable that contributes the most to the target variable `Class`. This plot presents the effect of the categories of the `marital-status` variable on the target variable. For an explanation of the plot, see the topic on variable categories and profit.

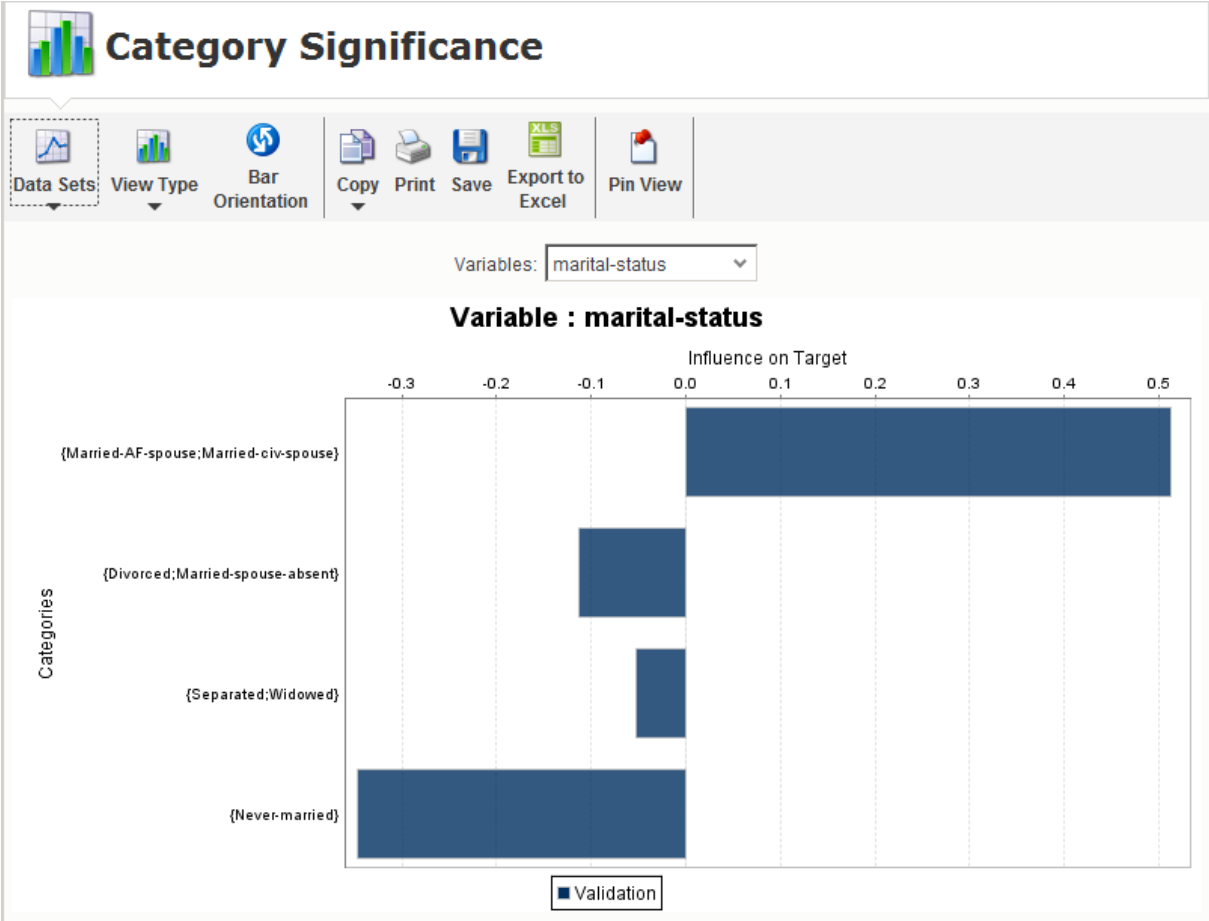
5.1.7.3.5.1 Variable Categories and Profit

The *Category Significance* plot shows the relative significance of the different categories of a given variable with respect to the target variable.

Each bar on the plot shows the amount of influence that category has on the target category (or hoped-for value) of the target variable. The bars are ordered by their amount of influence. In this example, the *Bar Orientation* option shows the bars vertically, so the bar the furthest to the left represents the category with the greatest positive effect, and the bar the furthest to the right, the category with the least effect.



If the *Bar Orientation* option is set to show the bars horizontally, the highest bar on the plot represents the category with the greatest positive effect. In other words, the higher a category appears on the plot, the more representative that category is of the target category of the target variable.



The length and direction of a bar correspond to the profit contributed by that category, in other words, whether the category has more or fewer observations belonging to the target category. A positive bar (*Influence on Target* greater than 0.0) indicates that the category contains more observations belonging to the target category than the mean (calculated on the entire dataset). A negative bar (*Influence on Target* less than 0.0) indicates that the category contains fewer observations belonging to the target category than the mean.

> **i Note**
>
> You can display the profit curve for the selected variable by clicking the  (*Display Profit Curve*) button located in the tool bar under the title.

The importance of a category depends on both its difference to the target category mean and the number of represented cases. High importance can result from any of the following:

- A high discrepancy between the category and the mean of the target category of the target variable
- A minor discrepancy combined with a large number of records in the category
- A combination of both

Use the *Variables* pull down menu to select and graph any of the variables in the model. Use the tool bar located under the title to copy the coordinates to the clipboard, print the plot, or save it in PNG format. The values are normalized and their sum always equals to "0". Depending on the chosen profit strategy, or on the continuous target variables value type, you can obtain all positive importance or negative and positive importance.

5.1.7.3.5.1.1 Axes

The X-axis shows the influence of the variable categories on the target. The significance of the different numbers on the X-axis are detailed in the following table:

Number on the X-axis	Indicates that the category has
positive number	a positive influence on the target
0	no influence on the target (the behavior is the same as the average behavior of the whole population)
negative number	a negative influence on the target

The Y-axis displays the variable categories. Categories sharing the same effect on the target variable are grouped. They appear as follow [Category_a;Category_b;Category_c]. Categories not containing sufficient numbers to provide robust information are grouped in the KxOther category. When a variable is associated with too many missing values, the missing values are grouped in the KxMissing category. Both categories are created automatically by the application.

5.1.7.3.5.1.2 Category Importance Definition

The following definition applies to continuous targets; some wording may be simplified for binary targets. The formulas presented below can also be applied to the binary target case (use categories instead of segments in this case).

We consider the case where a Regression/Classification regression model is trained on a continuous target/signal S with the help of an input variable X.

Regression/Classification starts by binning the continuous target S into B segments: S1,...,SB and by computing the basic statistics of the inputs and the cross statistics of the inputs with respect to the target.

We will suppose that the input X is a nominal (categorical) variable, though the whole process can be extended easily to the case of ordinal and continuous inputs.

We will suppose that X has N categories: X1, ..., XN. We are interested in assessing the importance of a category Xi with respect to the target S. The importance of a category depends on two factors:

- The fact that the distribution of the target for this category is significantly skewed towards high values or low values when compared with the distribution of the target on the entire population.
- The frequency of this category.

High importance can result from either of the following:

- A high discrepancy between the target distribution for cases associated to this category and the distribution of the target variable for the entire population.
- A minor discrepancy combined with a large number of records in the category
- a combination of both

The application uses a non-parametric setting in which the category importance is defined as:

$$CategoryImportance(X_i) = NormalProfit(X_i) * \frac{Freq(X_i)}{2}$$

where :

- normalProfit(Xi) is the normal profit of category Xi, see below for a definition
- Freq (Xi) is the global frequency of the category Xi
- Z is a normalization constant.

We give below the details of the computation of these quantities.

Normal Profit

Each category of the target Sj is associated with a profit profit(Sj) defined such that:

$$\sum_{j=1}^{j=B} profit(S_j)Freq(S_j) = 0$$

The profit of a target category is a value in the range [-1; +1]. It is defined the following way from the (cumulated) target category frequencies:

$$profit(S_j) = 2\sum_{k=1}^{j-1} Freq(S_k) + Freq(S_j) - 1$$

The normal profit of a category Xi is then defined as:

$$NormalProfit(X_i) = \sum_{j=1}^{j=B} Profit(S_j)Prob[S_j|X_i]$$

Where Prob[Sj |Xi] is the conditional probability of observing the target category Sj in the variable category Xi (cross statistics):

$$Prob[S_j|X_i] = \frac{Freq(S_j : X_i)}{Freq(X_i)}$$

The fact that these formulas rely only on frequencies makes them resistant to any monotonic transformation of the target S.

Normalization Constant

The normalization can be approximated for non pathological continuous targets (that is continuous targets without distribution peak (Dirac)) as:

$$Z = P[S > median(S)] * (1 - P[S > median(S)])$$

In most cases, a good approximation is = 0.25.

Normal Profit Properties

There are several interesting things to note about normal profit:

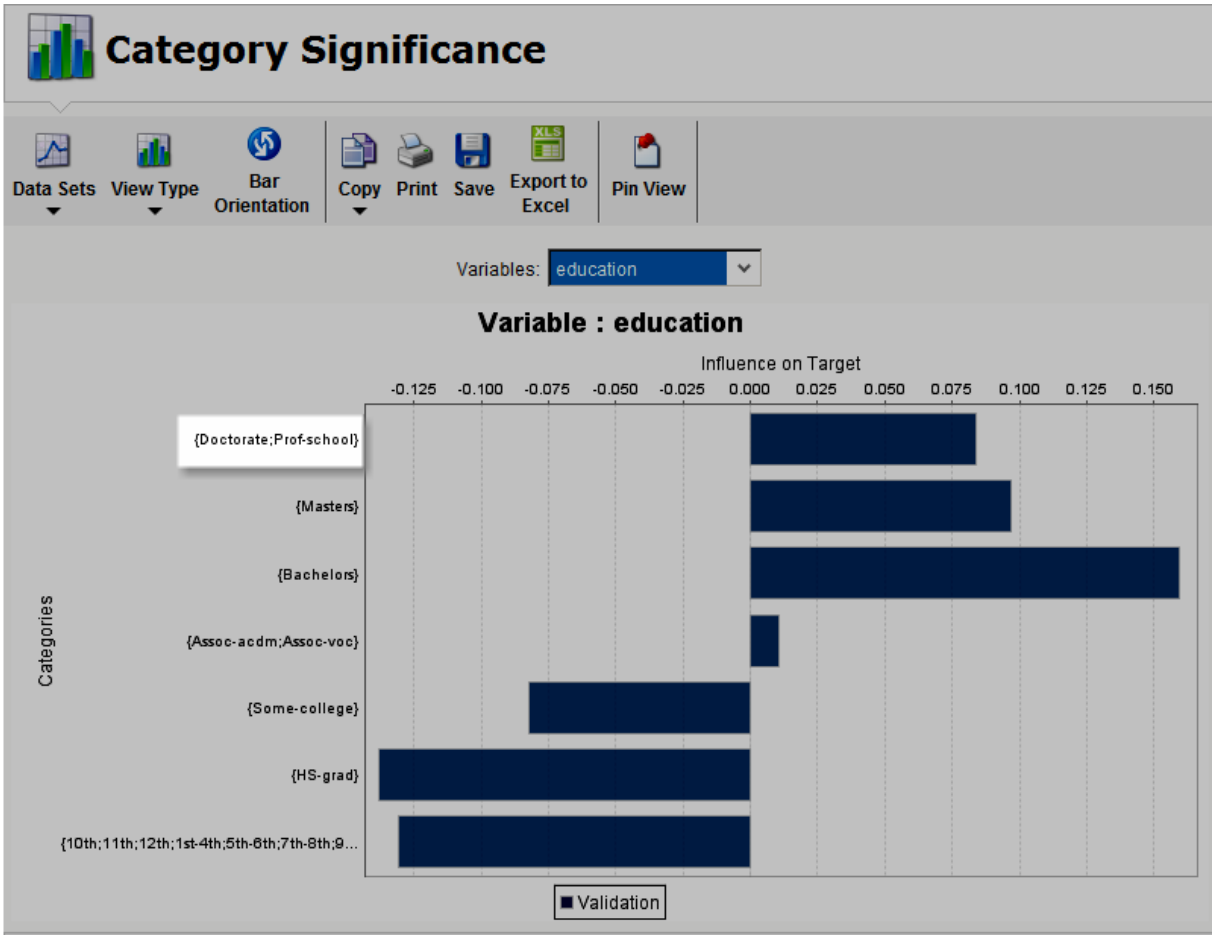
- The normal profit of category is independent of the target values themselves (user can change the target value through monotonic transformations; the normal profit of the categories with respect to this target will not change). This belongs to non-parametric metrics.
- A consequence of 1 is that this metric is resistant to outliers: when there are a few occurrences of the target with very high values with respect to the rest of the target value distributions, the notion of normal profit is not impacted.
- The weighted sum of the normal profit for all categories of a given variables will always be 0.

5.1.7.3.5.2 Grouping Categories

On the plot of details of a variable, categories may appear grouped.

When the option *Optimal Grouping* is enabled, the application groups those categories sharing the same effect on the target variable. In the plot below (example in English), for the variable `education`, the categories `Doctorate` and `Prof-School` are grouped. If the explanatory variable is continuous, the application identifies the points where behavioral changes occur with respect to the target variable and automatically crops the variable into intervals exhibiting homogeneous behavior with respect to the target.

For more information, please see the related topic.



When categories do not contain sufficient numbers to provide robust information, they are grouped in the KxOther category, that is created automatically.

When a variable is associated with too many missing values, the missing values are grouped in the KxMissing category, that is also created automatically.

To understand the value of the categories KxOther and KxMissing, consider the following example. The database of corporate customers of a business contains the variable "web address". This variable contains the Web site address of the corporate customers contained in the database. Some companies have a Web site; others do not. In addition, each Web site address is unique. In this case, the application automatically transforms the "web address" variable into a binary variable with two possible values: KxOther (the firm has a Web site) and KxMissing (the firm does not have a Web site).

5.1.7.3.6 Statistical Reports

Statistical Reports provide you with a set of tables that allow you a more detailed debriefing of your model. These reports are grouped in different levels of debriefing:

- The Descriptive Statistics, which provides the statistics on the variables, their categories and the datasets, as well as the variables cross-statistics with the target.

> **i Note**
>
> - If your dataset contains date or datetime variables, automatically generated variables will appear in the statistical reports.
> - In the section *Cross Statistics with the Target(s)*, the number of displayed categories corresponds to:
> - The number of categories as defined in the user structure
> - The band count if no user structure has been defined
>
> For more information, see the related topic on bad count.
> - In the section *Grouped Cross Statistics with the Target(s)*, if the option *Optimal Grouping* is enabled, the number of displayed categories is lower than that defined:
> - In the user structure
> - By the parameter band count if no user structure has been defined.
>
> .

- The *Model Performance*, in which you will find the model performance indicators, the variables contributions and the score detailed statistics.
- The *Control for Deviations*, which allows you to check the deviations for each variable and each variable category between the validation and test datasets.
- The *Expert Debriefing*, in which you will find more specialized performance indicators, as well as the variables encoding, the excluded variables during model generation and the reason for exclusion, and so on.

5.1.7.3.6.1 Variable Exclusion Cause

Statistical reports include the section *Variable Exclusion Causes*. For regression and classification models, this section presents the reason why any variable was excluded from the model.

- *Overall Exclusions* shows the variables excluded from the whole model.
- *Target Specific Exclusions* shows the variables excluded towards a particular target.

The table below shows the possible variable exclusion causes:

Possible Variable Exclusion Clauses

Overall Exclusions / Target Specific Exclusions	Name	Explanation
Overall Exclusions	Constant	The variable has only one value (continuous variables) or one category (nominal or ordinal variables) in the dataset. The variable is discarded with respect to all targets.
Overall Exclusions	Small variance	For continuous variables, the variance is small. The variable variation is noise. The variable is discarded with respect to all targets.

Overall Exclusions / Target Specific Exclusions	Name	Explanation
Target Specific Exclusions	Fully Compressed	The variable has been fully compressed with respect to the target. It will be excluded from the model with respect to this target.
Target Specific Exclusions	Small KI on training	The variable has a small KI on Training dataset with respect to the target. It will be excluded from the model with respect to this target.
Target Specific Exclusions	Small KI on validation	The variable has a small KI on Validation dataset with respect to the target. It will be excluded from the model with respect to this target.
Target Specific Exclusions	Large KI difference	A large KI difference has been observed for this variable between Training and Validation datasets with respect to the target. It will be excluded from the model with respect to this target.
Target Specific Exclusions	Small KR	The variable has a small KR with respect to the target. It will be excluded from the model with respect to this target.

5.1.7.3.6.2 Statistical Report Options

A toolbar is provided allowing you to modify how the current report is displayed, to copy the report, to print it, to save it or to export it to Excel.

Display Options

Menu	Option
View	Display the current report view in the graphical table that can be sorted by column.
	Display the current report view as an HTML table.

Menu	Option
	Some reports can be displayed as a bar chart. This bar chart can be sorted by ascending or descending values, or by ascending or descending alphabetical order. You can also select which data should be displayed.
	Display certain reports as a pie chart.
	Display certain reports as a line chart.
Sort	When the current report is displayed as a bar chart, this option allows you to change the orientation of the bars (from horizontal to vertical and vice versa).
	Display the current report with no sorting.
	Sort the current report by ascending values.
	Sort the current report by descending values.
	Sort the current report by ascending names.
	Sort the current report by descending names.
Series	Select which columns to display for current report.

Usage Options

Menu	Option
	Copy the data from the current view of the displayed report. The data can then be pasted in a text editor, a spreadsheet, a word processing software.
	Print the current view of the selected report depending on the chosen display mode (HTML table, graph, ...).
	Save under different formats (text, html, pdf, rtf) the data from the current view of the selected report.
	Save under different formats (text, html, pdf, rtf) the data from all the views of the selected report.

Menu	Option
	Export to Excel.
	Save all reports.
	Save the customized style sheet.

5.1.7.3.7 Scorecard

This screen provides you with the coefficients associated to each category for all variables in a regression model.

class

Category	Score (age)
Empty	1.968e-2
< = 17	-7.913e-2
]17 ; 18]	-2.331e-4*[age]+-7.517e-2
]19 ; 21]	3.614e-4*[age]+-8.527e-2
]22 ; 23]	1.312e-3*[age]+-1.043e-1
]24 ; 25]	5.224e-3*[age]+-1.892e-1
[26 ; 27]	1.084e-2*[age]+-3.339e-1
]27 ; 28]	9.982e-3*[age]+-3.107e-1
[29 ; 30]	6.447e-3*[age]+-2.105e-1
]30 ; 32]	4.278e-3*[age]+-1.455e-1
[33 ; 35]	3.973e-3*[age]+-1.356e-1
]35 ; 36]	9.495e-3*[age]+-3.289e-1
[37 ; 38]	3.025e-3*[age]+-9.528e-2
]38 ; 39]	3.966e-3*[age]+-1.31e-1
[40 ; 43]	5.014e-4*[age]+5.736e-3
]43 ; 47]	3.872e-4*[age]+1.065e-2
[48 ; 51]	3.872e-4*[age]+1.065e-2

To Obtain a Score

Add all the coefficients corresponding to the selected value of each variable.

> **i Note**
>
> In the case of a continuous variable, the scorecard always includes a number of categories that is higher than in the user defined structure or as given by the parameter *band count* if no user structure has been set. Indeed, the encoding of variables for the scorecard adds target curve points to increase the accuracy of

coding according to the training dataset. These points split some existing categories and thus increase the number of categories in the scorecard.

5.1.7.3.7.1 Risk Mode

The representation of a model equation is easier to read and to interpret in the Risk Mode due to stepwise encoding for ordinal and continuous variables.

In the Risk Mode, it is easy to define which category has a negative or positive effect on the risk score and consequently on the odds or on the probability of risk.

In order to illustrate the advantages of a scorecard in interpreting results, the variable age will be used for this example.

The segment]24;27] has a risk score of about 30 and the segment [37;43] has a risk score of about 15. According to the parameter PDO (set in this example to 15), it is easy to conclude that the segment [37;43] is two times more risky or that the odds of the segment [37;43] are two times inferior to the segment]24;27].

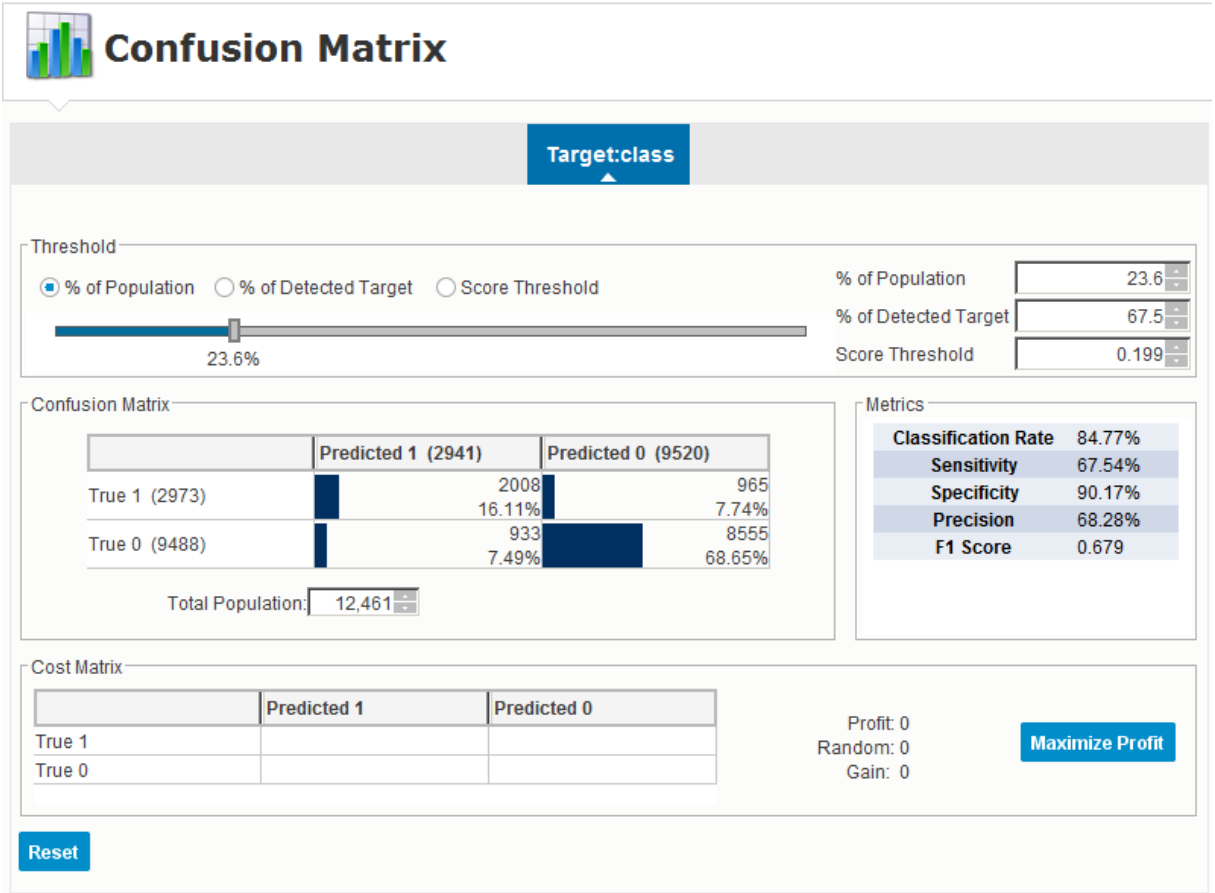
class

age	
Category	**Score**
Empty	34.57
< = 17	33.03
[17 ; 24]	33.03
]24 ; 27]	29.46
[28 ; 30]	24.74
[31 ; 33]	21.92
[34 ; 36]	18.79
[37 ; 43]	15.15
[44 ; 53]	13.04
]53 ; 62]	17.31
]62 ; 90]	24.95
> 90	24.95
Default	15.15

5.1.7.3.8 Confusion Matrix

You can use the Confusion Matrix to compare the predicted value of the target variable with its actual value.

When generating the model (see step Generating the Model [page 87]), you have set up the decision threshold and decided on the score above which the observations are considered as positive. A positive observation is an observation that belongs to the population you want to target. A negative observation is an observation that does not belong to this target population.



Understanding the Confusion Matrix

The threshold

There are three ways to set the threshold using the displayed slide bar:

- By selecting the *percentage of population to target* if the population is sorted by descending order of score (% of Population),
- By selecting the *percentage of positive observations* you want to detect (% of Detected Target)),
- By selecting the *score* used to differentiate positive observations from negative ones (Score Threshold). Any observation with a score above the threshold is considered positive, on the contrary any observation with a score below the threshold is considered negative.

The slide is graduated from the lowest score (on the left) to the highest score (on the right). The values corresponding to each option are displayed under the slide.

When you move the cursor, the confusion matrix is updated accordingly.

The Confusion Matrix

The following table details how to read the confusion matrix.

	Predicted [Target Category] Positive Observations Predicted	Predicted [Non-target Category] Negative Observations Predicted
True [Target Category] Actual Positive Observations	Number of correctly predicted positive observations (True Positive =TP)	Number of actual positive observations that have been predicted negative (False Negative = FN)
False [Non-target Category] Actual Negative Observations	Number of actual negative observations that have been predicted positive (False Positive = FP)	Number of correctly predicted negative observations (True Negative = TN)

By default, the *Total Population* is the number of records in the Validation dataset. You can modify this number to see the confusion matrix for the population on which you want to apply your model.

Example

Consider a classification system that organizes the e-mails into two categories: relevant e-mails and junk e-mails. You want to know how many relevant e-mails are wrongly identified as junk e-mails and how many junk e-mails are not identified as junk e-mails.

For the exercise, suppose that you are testing the classification system with 100 relevant e-mails and 100 junk e-mails.

The confusion matrix could be:

	Predicted [Normal e-mails] Positive Observations Predicted	Predicted [Junk e-mails] Negative Observations Predicted
True [Normal e-mails] Actual Positive Observations	95	5
False [Junk e-mails] Actual Negative Observations	3	97

And is read as follows:

- Among the 100 relevant e-mails, 95 e-mails are predicted as relevant e-mails and 5 are predicted as junk e-mails.
- Among the 100 junk e-mails, 3 are predicted as relevant e-mails and 97 are predicted as junk e-mails.
- Among the e-mails predicted as relevant e-mails (TP + FP), 3 are actually junk e-mails.
- Among the e-mails predicted as junk e-mails (FN + TN), 5 are actually relevant e-mails.

The Metrics:

The following table details how to read the confusion matrix.

Metrics	Definitions	Formula
Classification Rate	Percentage of observations accurately classified by the model when applied on the training dataset.	
Sensitivity	Percentage of actual positive observations that have been correctly predicted.	TP/(TP+FN)
Specificity	Percentage of negative observations that have been correctly predicted.	TN/(FP+TN)
Precision	Percentage of detected positive observations that are actually positive observations.	TP/(TP+FP)
Recall	Percentage of actual positive observations that have been detected as positive.	TP/(TP+FN)
F1 score	Harmonic mean of Precision and Recall (Recall and Precision are evenly weighted).	F1 = 2 (Precision X Recall)/ (Precision + Recall).

The Cost Matrix

This section allows you to visualize your profit depending on the selected score, or to automatically select the score depending on your profit parameters.

For each observation category, enter a profit or a cost per observation. The total profit is automatically displayed on the right of the table.

To know the threshold that will give you a maximum profit for the profit parameters you have set, click the button *Maximize Profit*.

Example

In the following profit/cost table, each positive observation correctly identified will yield $15, but each negative observation identified as positive will cost you $8.

Category	Predicted[1]	Predicted[0]
True[1]	15	0
True[0]	-8	0

5.1.7.3.9 Decision Tree

The panel Decision Tree allows you to display the results of a regression or classification model generated by Modeler as a decision tree based on the five most contributive variables. You can only access the Decision Tree panel if you selected the option Compute Decision Tree before generating the model.

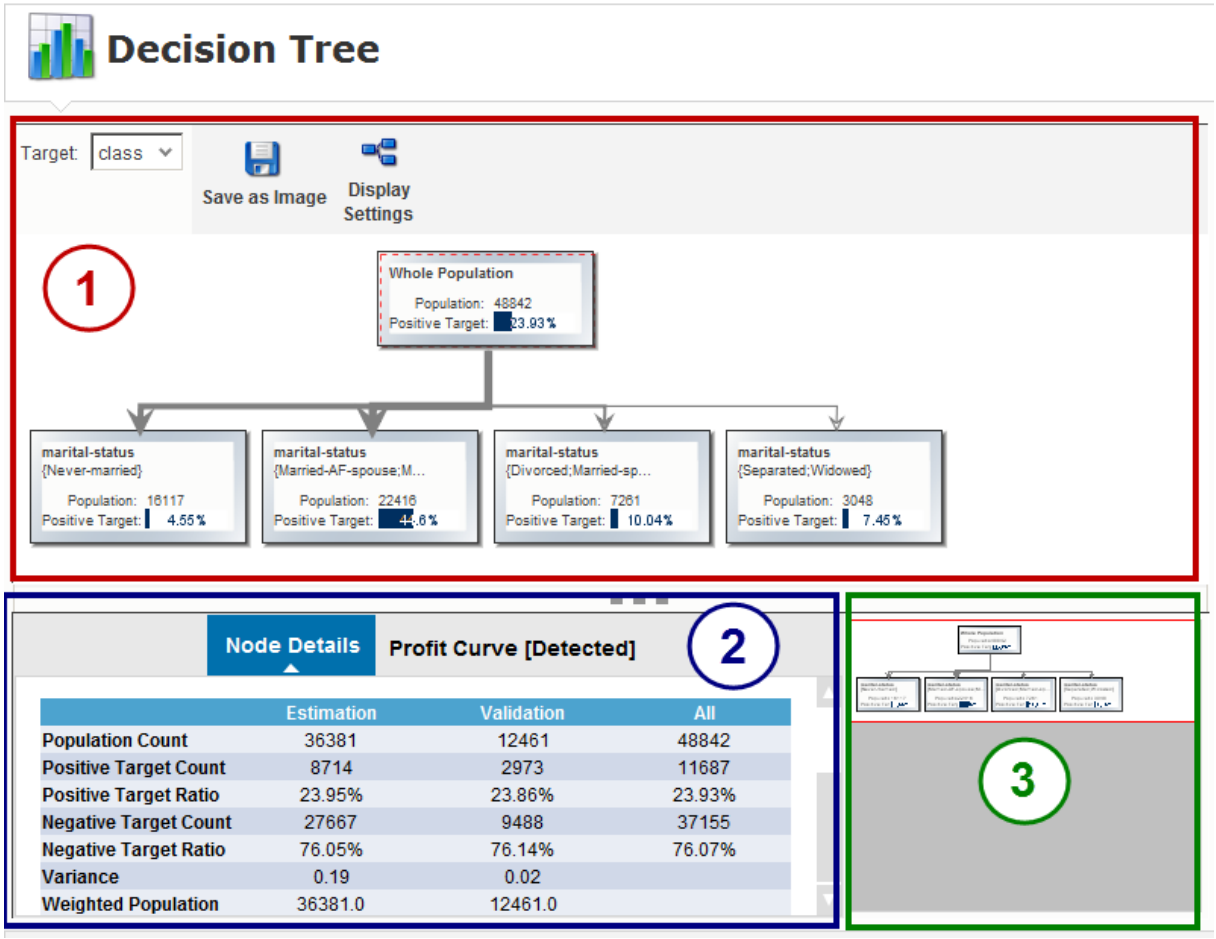
To Display the Decision Tree for a Target

In the Target list, select the target for which you want to display the decision tree.

Understanding the Decision Tree Panel

The panel Decision Tree is split into three parts:

1. The decision tree itself, which is displayed in the upper section of the panel
2. Two tabs located in the left bottom part of the panel provide you with information on the nodes and with the profit curve corresponding to the current decision tree
3. A navigator allowing you to visualize what part of the tree you are studying is displayed in the right bottom part of the panel.



	Estimation	Validation	All
Population Count	36381	12461	48842
Positive Target Count	8714	2973	11687
Positive Target Ratio	23.95%	23.86%	23.93%
Negative Target Count	27667	9488	37155
Negative Target Ratio	76.05%	76.14%	76.07%
Variance	0.19	0.02	
Weighted Population	36381.0	12461.0	

The Decision Tree

Each node in the tree displays:

- The name of the expanded variable, for example `Marital-status`.
- The categories on which the node population has been filtered, for example `{Married-AF-spouse;Never-married}`
- The *Population* of the node
- The ratio of *Positive Target* (for nominal targets) or the *Target Mean* (for continuous targets)

When you go over a node, several options are offered:

Option	Description
	Select the variable to be used to expand the next level of the decision tree.
	Automatically expand the next level using the most contributive variable not yet used in the current decision tree.
	Fold the section of the tree displayed below the current node.

The thickness of the arrows depends on the amount of population in the node. In the following example, the arrow leading to the node corresponding to the category `[0;4386[` of `capital-gain` is thicker since the node population is significantly higher than the one from the node `capital-gain ]4386;41310]`.

Node Details

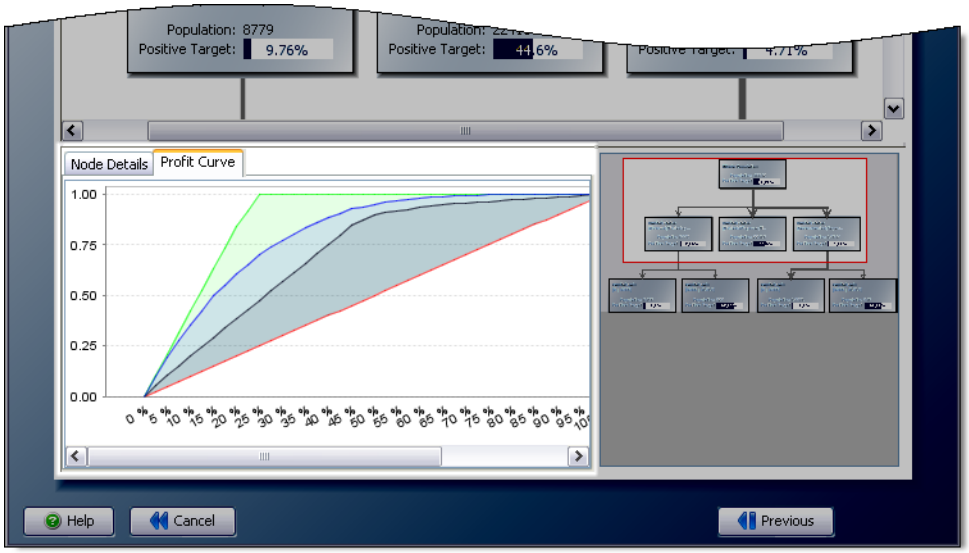
When you select a node, the node information is displayed in the tab *Node Details* located in the lower part of the panel.

This tab indicates the target for which the current decision tree is displayed and provides you with the following information for each dataset in the model:

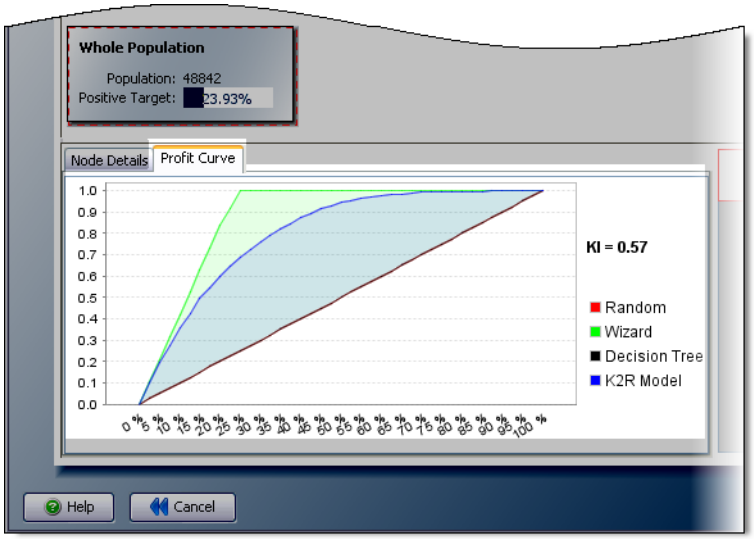
- *Population Count*, that is, the number of records found in the current node
- For continuous targets:
  - *Target Mean*, that is, the mean of the target for the current node.
- For nominal targets:
  - *Positive Target Count*, that is, the number of records for which the target is positive,
  - *Positive Target Ratio*, that is, the percentage of the node population for which the target is positive,
  - *Negative Target Count*, that is, the number of records for which the target is negative,
  - *Negative Target Ratio*, that is, the percentage of the node population for which the target is negative,
  - *Variance*, that is, the variance for the current node,
  - *Weighted Population*, that is, the number of records when using a weight variable.

Profit Curve

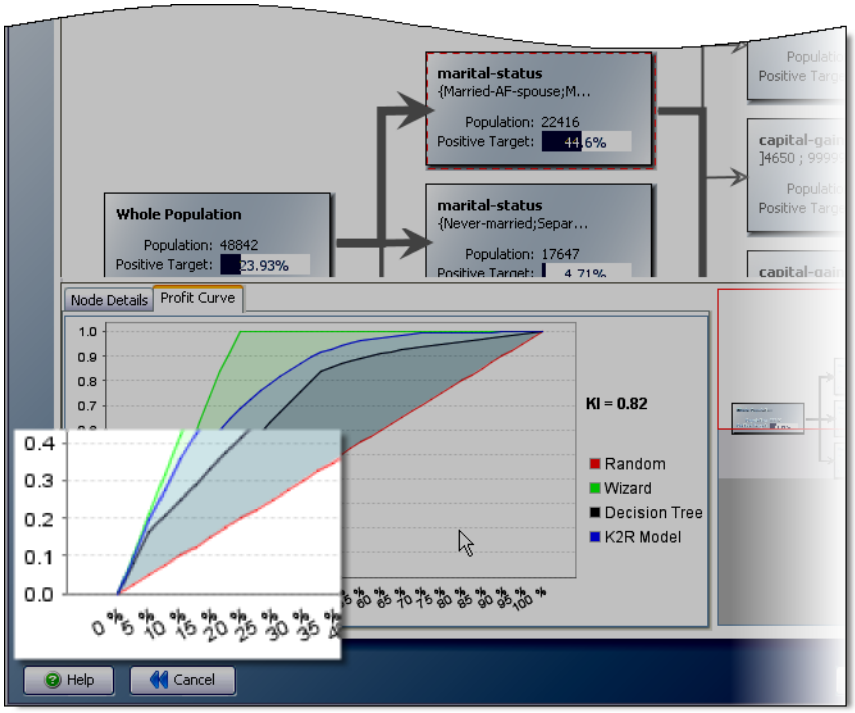
The profit curve for the current decision tree is displayed in the tab Profit Curve located in the lower part of the panel. This profit curve changes with every modification made on the decision tree.



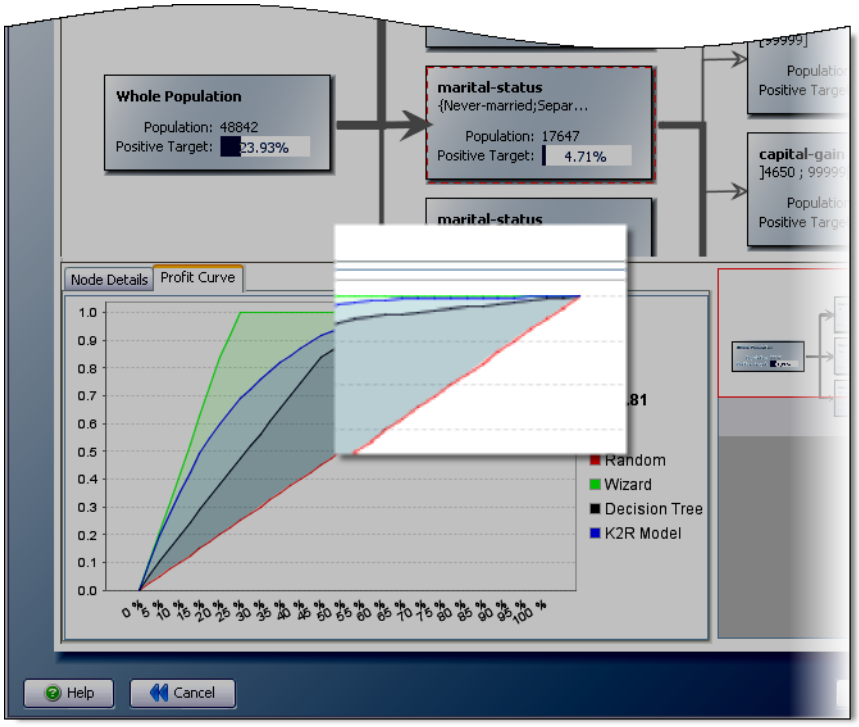
The profit curve corresponding to the node containing the whole population is equal to the random curve.



When you expand the node with the highest percentage of positive target, the profit curve improves over the first percentiles, which means that the model will detect the population with the highest scores.



If you expand the node with the lowest percentage of positive target, the profit curve will improve over the last percentiles.



However, if the node you expand contains a very small population, the profit curve will not be impacted. So you need to find the best compromise between the size of the population and the percentage of positive target.

Customizing the Display

The button *Display Settings* allows you to customize some of the display settings for the decision tree.

- *Orientation*: this setting allows you to select if you want to display the tree horizontally or vertically.
- *Display Type*: this setting allows you to display the decision tree as a standard decision tree (*Compact*) or with a specific look provided by SAP Predictive Analytics (*Full*). The *Compact* option uses less screen space, but the *Full* option is easier to read.

When you have set the display parameters, click the *Close* button.

5.1.7.4 Step 4 - Using the Model

Once generated, a classification model may be saved for later use.

A classification model may be applied to additional datasets. The model thus allows you to perform predictions on these application datasets, by predicting the values of a target variable. The model can also be used to carry out simulations on specific observations, on a case-by-case basis.

Moreover, you can refine a classification model, by re-generating it with an optimized list of explanatory variables. The application allows you to select the variables most pertinent to your business issue automatically, with pertinence defined as: producing the minimum area between the predictive curve and the hypothetical perfect curve, and thus maximizing the volume of information explained by the model.

So that you to apply the model to any other database, the application allows you to generate different source code of the model (for example, C, XML, AWK, HTML, SQL, PMML2, SAS, and JAVA).

5.1.7.4.1 Analyzing Deviations

The option Analyze Deviations is a tool that provides you with a diagnostic of the data statistical variation.

This option can be used for several purposes:

- to compare the distribution of a new dataset with the distribution of the dataset used to train the model,
- to check the quality of new data after loading them,
- to check if your data have evolved over time and thus if the model need to be adapted to the new data.

5.1.7.4.1.1 Selecting the Dataset to Analyze

First, you need to select the dataset for which you want to analyze the deviations.

For the results to make sense, the new dataset should contain the same columns as the dataset that was originally used to train the model including the target variable, which must be filled.

To Select a Dataset

1. On the screen *Analyze Deviations*, select the data source format to be used (*Text file*, *ODBC*, ...).
2. Click the *Browse* button .
3. In the *Data Selection* dialog, browse to the `<Installation Path>/Samples` folder.
4. Select the file you want to use, then click *OK*. The name of the file appears in the *Dataset* field.
5. Click *Next*. The screen *Deviation Analysis Debriefing* is displayed.

5.1.7.4.1.2 Following the Deviation Analysis Progress

The panel *Deviation Analysis Debriefing* allows you to follow the analysis process thanks to a progression bar.

At the end of the process, a debriefing panel is displayed. For details on the debriefing panel, see the topic on Understanding the Deviation Analysis.

You can use the toolbar provided on the upper part of the panel to:

- stop the analysis process, by clicking the button,
- display the text log detailing the process, by clicking the button,
- copy, print or save the debriefing panel .

To Copy the Report

Click the (*Copy*) button.

The application copies the HTML code of the screen that you can the paste into another document (for example, a spreadsheet or text editor).

To Print the Report

1. Click the (*Print*) button situated under the title. A dialog box appears, allowing you to select the printer to use.
2. Select the printer to use and set other print properties if need be.
3. Click *OK*. The report is printed.

To Save the Report

Click the (*Save*) button situated under the title.

The file is saved in HTML format.

5.1.7.4.1.3 Understanding the Deviation Analysis

The first step to take to know if there are any deviations in your data is to look at the debriefing report and compare the performances (KI and KR) obtained on the original data with those obtained on the control dataset.

Then to visualize which variables have changed, you should look into the Control for Deviation Reports.

Debriefing Report

The section Control for Deviation Overview provides you with basic statistics on the Dataset used for Deviation Control (also called control dataset) such as:

- the name of the dataset (Dataset),
- the source file (Source),
- the number of records contained in the dataset (Number of Records)
- and the number of variables for which the application has found deviations in comparison to the dataset originally used to train the model (Number of variables showing deviation).

The second and third section of the debriefing report allows you to compare the performance of your model on the original dataset with the its performance on the control dataset:

- the section Performance Indicators displays for each target the KI and KR indicators obtained by the model on the original dataset.
- the section Performance on Control Dataset displays for each target the KI and KR indicators obtained by the model on the control dataset.

If the KI and/or KR of the model on the control dataset are significantly lower, it means that the relation between the variables and the target variable has changed. As a consequence the model should be rebuilt on the new data.

If the KI and KR are not much different, it means that the relation between the input variables and the target behavior has not changed, but it does not mean that differences of distributions are not possible.

Control for Deviations Reports

The panel *Control for Deviations* provides you with six options that can be separated in three groups:

- The first one, made of the options *Probability of Deviation*, *Probability of Category Deviation* and *Probability of Grouped Category Deviation*, enumerates the probabilities of deviation of each variable distribution, be it by variable, variable category or group of categories. A probability over 0.95 indicates that the variable or category global distribution is significantly different in the control dataset than in the reference dataset.

> **ℹ Note**
>
> The probability of deviation is actually a standardized chi-square test. It is significant above 0.95.

- The second group, comprised of the options *Probability of Target Deviation* and *Probability of Target Deviation for Grouped Categories*, lists for each variable the probabilities of deviation of the categories and the grouped categories with respect to the target variable. A probability over 0.95 indicates that there is a change of behavior with respect to the target variable in the category or group of categories.
- The last group contains only the option *Category with Problem*. For each dataset (reference datasets and control dataset), all variable categories with a probability over 0.95 are listed. This allows you a quick visualization of possible problems without having to analyze all the reports.

> **⚠ Caution**
>
> In all the report panels the control dataset is referred as the *ApplyIn* dataset.

Options

You can select which report sections to save:

1. Click the button  *Save the reports* (located in the bottom left corner). A selector window opens.
2. In the displayed list, check the sections you want to save.
3. In the list *Report Style*, select the type of output you want. Three styles of output are available:
   - *Automatic*: saves the default view displayed in the interface
   - *Graphical*: saves the reports as graphs if such a view exists
   - *Textual*: saves the reports as tables

> **⚠ Caution**
>
> When selecting the options Automatic or Graphical , be careful to choose an appropriate file type such as pdf, rtf or HTML.

4. Click *OK*.
5. Select the folder in which you want to save the report.
6. Enter the name of the file.

5.1.7.4.2 Applying the Model to a New Dataset

The currently open model may be applied to additional datasets. The model allows you to perform predictions using the application datasets, and specifically, to predict the values of the target variable.

Constraints of Model Use

In order to apply a model to a dataset, the format of the application dataset must be identical to that of the training dataset used to generate the model. The same target variable, in particular, must be included in both datasets, even if the values are not contained in the application dataset.

> **i Note**
>
> If the *KxIndex* variable of the model is virtual, the application dataset must not contain a physical *KxIndex* variable.

5.1.7.4.2.1 To Apply the Model to a New Dataset

For this scenario, due to technical constraints, a dataset corresponding to the database of 1,000,000 customers that will be used in this scenario can not be provided to you. You will apply the model to the file `Census01.csv`, which you used to generate the model. In this manner, you will be able to compare the predictions provided by the model to the real values of the target variable *Class* for each of the observations.

In the procedure:

- Select the format *Text files*,
- In the *Generate* field, select the option *Individual Contributions*,
- Select the folder of your choice in which to save the results file (*Model-Generated Output*).
- Do not select the option *Keep only outliers*.

1. On the screen *Using the Model*, click the option *Applying the model to a new dataset*.

   The screen *Applying the Model* appears.
2. In the section *Application Dataset*, select the format of the data source in the list *Data Type*.
3. Click the *Browse* button to select:
   - In the *Folder* field, the folder which contains your dataset.
   - In the *Data* field, the name of the file corresponding to your dataset.
4. In the section *Results generated by the model*, select the file format for the output file in the list *Data Type*.
5. You may also opt to select *Keep only outliers*.

   If you select this option, only the outlier observations will be presented in the results file obtained, after applying a model.
6. Click the *Apply* button.

   The screen *Applying the Model* appears.

Once application of the model has been completed, the results files of the application is automatically saved in the location that you had defined from the screen *Applying the Model*.

5.1.7.4.2.2 Classification Decision

The screen *Classification Decision* allows you to select how many observations you want the model to detect after application on the new dataset.

To apply a Classification Decision

1. On the screen *Applying the Model*, follow all the steps of the procedure To Apply a Model to a New Dataset.
2. In the Generate drop-down list, select the option *Decision* .
3. Click the *Apply* button.
   The screen *Classification Decision* appears.
4. Use the slide to set the percentage of population to detect.
5. Click *Next*.
   The model is applied to the new dataset.

Understanding the Classification Decision Screen

The screen *Classification Decision* allows you to either select a percentage of the population who will respond positively to your campaign (*% of Detected Target*) or a percentage of the entire population (*% of Population*).

When moving the cursor on the scale, the different values are updated accordingly.

For example, if you select the option *% of Detected Target* and set the cursor to 80%, the value of the field *% of Population* will be 32.0, which means that if you want that 80% of the people who will respond positively to your campaign receive your mailing, you will have to send it to 32% of the entire population.

On the other hand, if you select the option *% of Population* and set the cursor to 20% on the scale, the value of the field *% of Detected Target* will be 60.4, which means that if your budget only allows you to send your mailing to 20% of the entire population, you will touch 60% of the population who will respond positively.

For more details on how to use the *Confusion Matrix*, see the related topic.

5.1.7.4.2.3 Using the Option Direct Apply in the Database

This optimized scoring mode can be used if all the following conditions are met:

- the apply-in dataset (table, view, select statement, data manipulation) and the results dataset are tables coming from the same database,
- the model has been computed while at least one physical key variable was defined in the application,
- no error has occurred,
- the in-database apply mode is not deactivated,
- granted access to read and write (create table).

To use the in-database apply mode, check the option *Use the Direct Apply in the Database* and automatically the option *Add Score Deviation* is selected as well.

5.1.7.4.2.4 Advanced Apply Settings

5.1.7.4.2.4.1 General Outputs

Copy the Weight Variable

This option allows you to add to the output file the weight variable if it had been set during the variable selection of the model.

Copy Dataset Id

This option allows you to add to the output file the name of the sub-dataset the record comes from (*Training*, *Validation* or *Testing*).

> **i Note**
>
> This option cannot be used with the in-database apply feature.

Copy the Variables

This option allows you to add to the output file one or more variables from the dataset.

- To Add All the Variables
  - Check the *All* option.
- To Select only Specific Variables
  1. Check the *Individual* option.
  2. Click the >> button to display the variable selection table.
  3. In the *Available* list select the variables you want to add (use the `Ctrl` key to select more than one variable).
  4. Click the > button to add the selected variables to the *Selected* list.

User Defined Constant Outputs

This option allows you to add to the output file constants such as the apply date, the dataset name, or any other information useful for using the output file.

A user defined constant is made of the following information:

Parameter	Description	Value / Warnings
Visibility	indicates if the constant will appear in the output or not	*checked*: the constant appears in the output *unchecked*: the constant does not appear in the output
Name	the name of the user defined constant	• The name cannot be the same as the name of an existing variable of the reference dataset. • If the name is the same as an already existing user defined constant, the new constant will *replace* the previous one
Storage	the constant type (number, string, integer, date, ...)	*number* *string* *integer* *date* *datetime*
Value	the value of the constant	date format: YYYY-MM-DD datetime format: YYYY-MM-DD HH:MM:SS
Key	indicates if the constant is a key variable or identifier for the record. You can declare multiple keys. They will be built according to the indicated order (1-2-3-...).	*0*: the variable is not an identifier *1*: primary identifier *2*: secondary identifier ...

- To Define a Constant
  1. Click the *Add* button. A pop-up window opens allowing you to set the constant parameters.
  2. In the field *Output Name*, enter the constant name.
  3. In the list *Output Storage*, select the constant type.
  4. In the field *Output Value,* enter the constant value.
  5. Click the *OK* button to create the constant. The new constant appears in the list. You can choose whether to generate the defined constants or not by checking the *Visibility* box.

5.1.7.4.2.4.2 Gain Chart

This tab allows you to compute the gain chart on the apply dataset, that is to rank your data in order of descending scores and split it into exact quantiles (decile, vingtile, percentile).

If you have computed the gain chart while creating your model, two gain charts will be computed during the application process:

- a transversal gain chart allowing you to check the frequency deviation between the validation and the application gain chart,
- an apply gain chart allowing you to have the exact number of targets inside each tiles.

1. Check the box *Compute Gain Chart on Apply-in Data*.
2. In the list, select the *Number of Quantiles* you want your data to be segmented in.
3. You can add additional variables in order to estimate profits per segments of the population:
   a. In the *Variables* list, select the variables you want to add to the gain chart. Use the CTRL key to select multiple variables.
   b. Click the > button to add the selected variables to the list *Values for Gain Chart*.
4. The sum of each selected variable will be calculated for each segment of the population.
5. Click *Validate* to save the advanced parameters and go back to the panel *Applying the Model*.

The result of the gain chart computation is available at the end of the model application. It can also be found in the *Statistical Reports*, in the section *Model Performance*.

If several gain charts have been computed, select the dataset in the proposed list to display the gain chart you want to visualize.

5.1.7.4.2.4.3 Outputs by Targets

5.1.7.4.2.4.3.1 Reason Codes

Reason codes are variables whose values have the most influence in a score-based decision (typically a risk score). An example of the use of reason codes is to provide a customer with the reasons why the automatic scoring system did not approve his loan.

To Generate Reason Codes

1. In the tree *Advanced Apply Settings* located on the left of the panel, open the node *Outputs for Target* `<Target Name>`.
2. Select *Reason Codes*.
3. Click the + button located on the right of the displayed table.
4. Click in the cell corresponding to the parameter you want to set. The following table sums up the available parameters:

Parameter	Values	Description
Number of Reason Codes	Integer Default: 3	Number of reason codes you want to generate.

Parameter	Values	Description
Threshold	○ Mean (default) ○ Maximum ○ Minimum	Threshold used for computing the most important reason codes. For each variable the contribution corresponding to the customer score is compared to its contribution for the whole population. The variables for which the contribution is the most differential are selected as the most important reason codes. For example, if you select Mean , the customer variable contribution will be compared to the mean of the whole population contribution to determine which variables are the most differential.
Criterion	○ Below (default) ○ Above	Indicates whether you want to generate the reason codes when the customer variable contribution is above or below the threshold Warning - Using Below with the Minimum threshold or Above with the Maximum threshold will generate an error.

5. If you want to generate several types of reason codes, repeat steps 3 and 4 for each type.

Output

The output table contains two columns for each reason code requested:

- `reason_name_<criterion>_<threshold>_<rank>_rr_<target name>`: contains the name of the variable selected as a reason code.
  For example, the output column named reason_name_Below_Mean_1_rr_class contains the name of the variable being the most important (1) reason code with respect to the target variable class. Among the variables whose contribution is below (Below) the mean (Mean) of the population contribution, the selected variable will be the one having the highest deviation with it.
- `reason_value_<criterion>_<threshold>_<rank>_rr_<target name>`: contains the value of the reason code.

5.1.7.4.2.4.3.2 Continuous Target

Option	Output Column Name	This option allows you to...
Predicted Value	`rr_`<target variable>	generate in the output file the value predicted by the model for the target variable. This option is checked by default.
Confidence	`bar_rr_`<target variable>	add to the output file the confidence level for the value that has been predicted, this is also known as the error bar. It is computed with "3" standard deviations on the validation dataset and bin per bin. The percentage of population corresponding to the "3" standard deviations is of about 99%. Calculation formula: {TargetMean - 3 (sqrt(TargetVariance)); TargetMean + 3 (sqrt(TargetVariance))} When (sqrt(TargetVariance)) is equal to the Standard Deviation TargetMean +/- Standard Deviation is equal to the Confidence Interval
Outlier Indicator	`outlier_rr_`<target variable>	to show in the output file which observations are outliers. An observation is qualified as outlier once its corresponding forecasting error is considered to be abnormal relative to the forecasting error mean observed on the estimation dataset. The forecasting error indicator is the absolute difference between the actual and predicted values. This is also called the residue. The residue abnormal threshold is set to 3 times the standard deviation of the residue values on an estimation (or validation) dataset.. Possible values are *1* if the observation is an outlier with respect to the current target, else *0*.

Option	Output Column Name	This option allows you to...
Contributions	`contrib_`<variable>`_rr_`<target variable> For example, if "marital-status" is an explanatory variable for the target variable "class", the column contrib_marital-status_rr_class will be generated in the output file.	add the variables contributions for the current variable to the output file. You can add the contributions of all variables or select only the contributions of specific variables (see procedure below).

Predicted Value

This option is checked by default. It allows you to generate in the output file the value predicted by the model for the target variable. It appears in the output file as `rr_`<target variable>.

Outlier Indicator

This option allows you to show in the output file which observations are outliers. An observation is qualified as outlier once its corresponding forecasting error is considered to be abnormal relative to the forecasting error mean observed on the estimation dataset. The forecasting error indicator is the absolute difference between the actual and predicted values. This is also called the residue. The residue abnormal threshold is set to 3 times the standard deviation of the residue values on an estimation (or validation) dataset. It appears in the output file as `outlier_rr_`<target variable>. Possible values are 1 if the observation is an outlier with respect to the current target, else 0.

Predicted Value Quantile

This option allows you to cut the output file in quantiles and to assign to each observation the number of the quantile containing it.

Approximate quantiles are constructed based on the sorted distribution and the boundaries of predicted scores from the validation sample. The score boundaries are used to determine *approximate* quantiles on the apply dataset.

> **i Note**
>
> - Exact quantile computation would require a full sort of the scores obtained on the apply dataset which can be consuming.
> - A *Gain Chart* option is available for this purpose.

It appears in the output file as `quantile_rr_`<target variable>_<number of quantiles>, for example for a target variable named "class" and a number of quantiles equal to 10, the generated column will be named `quantile_rr_`class_10.

1. Check the option *Predicted Value Quantiles*.
2. In the field *Number of Quantiles*, enter the number of quantiles you want to create.Check the option *Predicted Value Quantiles*.

Contributions

This option allows you to add the variables contributions for the current variable to the output file. You can add the contributions of all variables or select only the contributions of specific variables.

It appears in the output file as `contrib_`<variable>`_rr_`<target variable>.

For example, if "marital-status" is an explanatory variable for the target variable "class", the column `contrib_`marital-status`_rr_`class will be generated in the output file.

- To Add All Variables Contributions
  - Check the *All* option.
- To Add Specific Variable Contributions
  1. Check the *Individual* option.
  2. Click the >> button to display the variable selection table.
  3. In the *Available* list, select the variables you want to add (use the `Ctrl` key to select more than one variable).
  4. Click the > button to add the selected variables to the *Selected* list.

5.1.7.4.2.4.3.3 Nominal Target

Outputs by Rank

Scores

This option allows you to generate in the output file the best score(s) for each observation. For each line in the application dataset, Automated Analytics compares the scores obtained by the current observation for each category of the target variable and displays the best score in the column `best_rr_`<Target Variable>`_1`. You can generate up to 2 best scores since a target variable has 2 categories. The second best score is displayed in the column `best_rr_`<Target Variable>`_2`. When using this option with the *Decision* option described below, you can link the best score with the category that has obtained it.

Decision

This option allows you to generate in the output file the best decision(s) for each observation. Like for the previous option, the scores obtained for each category of the target variable are compared and the category with the best score for the current record is displayed in the column `decision_rr_`<Target Variable>. You can

generate up to 2 best decisions since a target variable has 2 categories. The category with the second best score is displayed in the column `decision_rr_`<Target Variable>.`_2`

Probabilities

This option allows you to generate in the output file the probability of the best decisions for each observation. Like for the previous options, the scores obtained for each category of the target variable are compared and the probability of the category with the best score for the current record is displayed in the column `proba_rr_`<Target Variable>. You can generate up to 2 probabilities of the best decisions since a target variable has 2 categories. The probability of the category with the second best score is displayed in the column `proba_rr_`<Target Variable>`_2`.

Outputs by Reference Category

Score

This option allows you to generate in the output file the score corresponding to each dataset line for the different categories of the target variable. You can generate the scores for all the target variable categories or select specific categories.

It appears in the output file as `rr_`<Target Variable> for the target variable key category and `rr_`<Target Variable>_<Category> for its other categories.

- To Add the Score of All Target Variable Categories
  - Check the *All* option.
- To Add Only the Scores of Selected Categories
  1. Check the *Individual* option.
  2. In the *Selection* column, check the boxes corresponding to the categories for which you want to add the score in the output file.

Prediction Probability

This option allows you to generate in the output file the probability for one or more target variable categories, that is for each observation the probability of the target variable value to be the selected category.

It appears in the output file as `proba_rr_`<Target Variable> for the target variable key category and as `proba_rr_`<Traget Variable>_<Category> for the other categories of the target variable.

- To Add the Probabilities of All Target Variable Categories
  - Check the *All* option.
- To Add Only the Probabilities of Selected Categories
  1. Check the *Individual* option.

2. In the *Selection* column, check the boxes corresponding to the categories for which you want to add the probabilities in the output file.

Miscellaneous Outputs

Outlier Indicator

This option allows you to show in the output file which observations are outliers. An observation is considered an outlier when the prediction error is greater than 3 times the average prediction error found on similar observations. It appears in the output file as `outlier_rr_`<target variable>. Possible values are 1 if the observation is an outlier with respect to the current target, else 0.

Predicted Value Quantile

This option allows you to cut the output file in quantiles and to assign to each observation the number of the quantile containing it.

Approximate quantiles are constructed based on the sorted distribution and the boundaries of predicted scores from the validation sample. The score boundaries are used to determine *approximate* quantiles on the apply dataset.

> **i Note**
>
> - Exact quantile computation would require a full sort of the scores obtained on the apply dataset which can be consuming.
> - A *Gain Chart* option is available for this purpose.

It appears in the output file as `quantile_rr_`<target variable>_<number of quantiles>, for example for a target variable named "class" and a number of quantiles equal to 10, the generated column will be named `quantile_rr_`class_10.

1. Check the option *Predicted Value Quantiles*.
2. In the field *Number of Quantiles*, enter the number of quantiles you want to create.

Contributions

This option allows you to add the variables contributions for the current variable to the output file. You can add the contributions of all variables or select only the contributions of specific variables.

It appears in the output file as `contrib_`<variable>`_rr_`<target variable>.

For example, if "marital-status" is an explanatory variable for the target variable "class", the column `contrib_`marital-status`_rr_`class will be generated in the output file.

- To Add All Variables Contributions
  - Check the *All* option.
- To Add Specific Variable Contributions
  1. Check the *Individual* option.
  2. Click the >> button to display the variable selection table.
  3. In the *Available* list select the variables you want to add (use the `Ctrl` key to select more than one variable).
  4. Click the **>** button to add the selected variables to the *Selected* list.

5.1.7.4.2.5 Types of Results Available

The application of a model to a dataset allows you to obtain four types of results, which are described in the following table.

Type of Results	Description
score, or predicted value	For a continuous variable, the predicted value corresponds to the value predicted by the model for the target variable of each observation. The "predicted values" correspond to the values read off the X axis of the profit curve plot. The "predicted value" of each observation is calculated by replacing the parameters of the polynomial representing the model, by the values of each of the variables of that observation. For a binary variable, the model outputs a score.
probability	Corresponds to the probability of each observation belonging or not to the target category of the target variable.
prediction range, or maximum error	The prediction range allows you to identify outlier observations. An observation is considered an outlier if the difference between its "predicted value" and its "real value" exceeds the value of the prediction range. In other words, the prediction range is a deviation measure of the values aroud the predicted score.
individual contributions	The individual contributions by variables contained in the dataset with respect to the target variable. The sum of all those individual contributions corresponds with the predicted value (score) to the nearest whole number.

Type of Results	Description
decision	The "decision" option can only be used for classification models, that is, when the target variable is nominal. It allows to generate a classification decision based on the "scores" (or "predicted values") generated by the model. The result file obtained contains a column in which a category of the target variable is assigned to every observation. The decision is taken on the basis of a threshold that is applied on the scores generated by the model. The target category of the target variable is assigned to observations whose scores are superior to the threshold. The default threshold (computed during the generation " or training " of the model) is chosen so that the way the categories of the target variable are assigned to observations is representative from their distribution in the training dataset.

Upon the level of information desired, you can choose to generate among several results' files, described in the table below.

Selecting the option...	Will generate a results' file containing the following information...
Predicted value only	Only the predicted value of observations (`rr_`<target variable name>)
Probability	• the predicted value • the probability (`proba_rr_`<target variable name>) • the prediction range (`bar_rr_`<target variable name>)
Individual Contributions	• the predicted value • the probability • the prediction range • the individual contributions of variables (`contrib_VariableName_rr_`<target variable name>)
Decision	• the predicted value • the decision (`decision_rr_`<target variable name>) • the decision probability (`proba_decision_rr_`<target variable name>) • the probability

5.1.7.4.2.5.1 Analyzing the Results of the Application

For this scenario, open the results file in Microsoft Excel, in the text format that you obtained when you applied the model to the `Census01.csv` file.

To Open the Model Application Results File

1. Depending upon the format of the results file generated, use Microsoft Excel or another application to open the file. The figure below presents the headings and columns of the results file obtained for this scenario.

	A	B	C	D	E	F	G
1	KxIndex	class	rr_class	proba_rr_class	bar_rr_class	contrib_age	contrib_workclass
2	1	0	0,02628554	0,177313641	1,17675817	-0,00014918	0,004575992
3	2	0	0,268730283	0,567965508	1,496288657	-0,00604592	0,004575992
4	3	0	-0,192986876	0,023777327	0,457313478	0,00038689	-0,002218456
5	4	0	-0,027590154	0,094321787	0,946476877	-0,00765412	-0,002218456
6	5	0	0,230136439	0,548656046	1,521923184	0,00574755	-0,002218456
7	6	0	0,35434714	0,737949133	1,401365995	0,00092295	-0,002218456
8	7	0	-0,358695984	0,00095057	0,467965275	-0,00550985	-0,002218456
9	8	1	0,188632727	0,435952127	1,483268023	-0,00711805	0,004575992
10	9	1	0,634126425	0,962523282	0,575068474	0,00413935	-0,002218456
11	10	1	0,540619075	0,916772068	0,933868706	-0,00175738	-0,002218456
12	11	1	0,235637605	0,551528811	1,519493461	0,00092295	-0,002218456
13	12	1	0,317775369	0,676707566	1,426785707	0,00467542	0,004575992
14	13	0	-0,192047745	0,024259994	0,463258654	0,00842789	-0,002218456
15	14	0	-0,11635045	0,058997665	0,731234848	0,00360329	-0,002218456
16	15	1	0,164729878	0,406741321	1,466872215	-0,00068525	-0,002218456

2. You can now analyze the results obtained and use these results of your analysis to make the right decisions.

Description of the Results File

Depending upon which options you selected, the results file will contain some or all of the following information, in the same order as seen below:

- The key variable defined during data description at the setting model parameters step.
- Possibly, the target variable given as known values if the latter appeared in the application dataset, as is the case in this scenario.
- The predicted value (score) provided by the model for the target variable of each observation. The name of this column corresponds to the name of the target variable prefixed by `rr_` , or in this case, `rr_Class` .
- The decision is based on the score. For example, its value can be of 1 if the observation is considered as interesting or 0 if it is considered as uninteresting for the model. The name of this column corresponds to the name of the target variable prefixed by `decision_rr_` , or in this case, `decision_rr_class`.
- The probability decision is also based on the score and provides the probability of the decision. The higher it is, the more it will confirm the decision value. The name of this column corresponds to the name of the target variable prefixed by `proba_decision_rr_`, or in this case, `proba_decision_rr_class`.
- The probability for each observation that it does or does not belong to the target category of the target variable. The name of this column corresponds to the name of the target variable prefixed by `proba_rr_` , or in this case, `proba_rr_class`.
- The prediction range , or "maximum error". The name of this column corresponds to the name of the target variable prefixed by `bar_rr_` , or in this case, `bar_rr_Class`.
- The individual contributions by variables contained in the dataset with respect to the target variable. The names of the columns of individual contributions correspond to the names of each of the variables, prefixed by `contrib_` , or in this case, `contrib_age` , `contrib_workclass`, and so on.

5.1.7.4.3 Performing a Simulation

The open model may be used to carry out simulations on specific observations, one at a time. To define the observation to be analyzed, the variables of your choice must be associated with values. For instance, if you have selected the occupation (profession category) and workclass (socio-professional category) variables, they must contain values. During execution of the simulation, the application will automatically assign values to certain variables when values are missing, but essential to proper completion of the simulation.

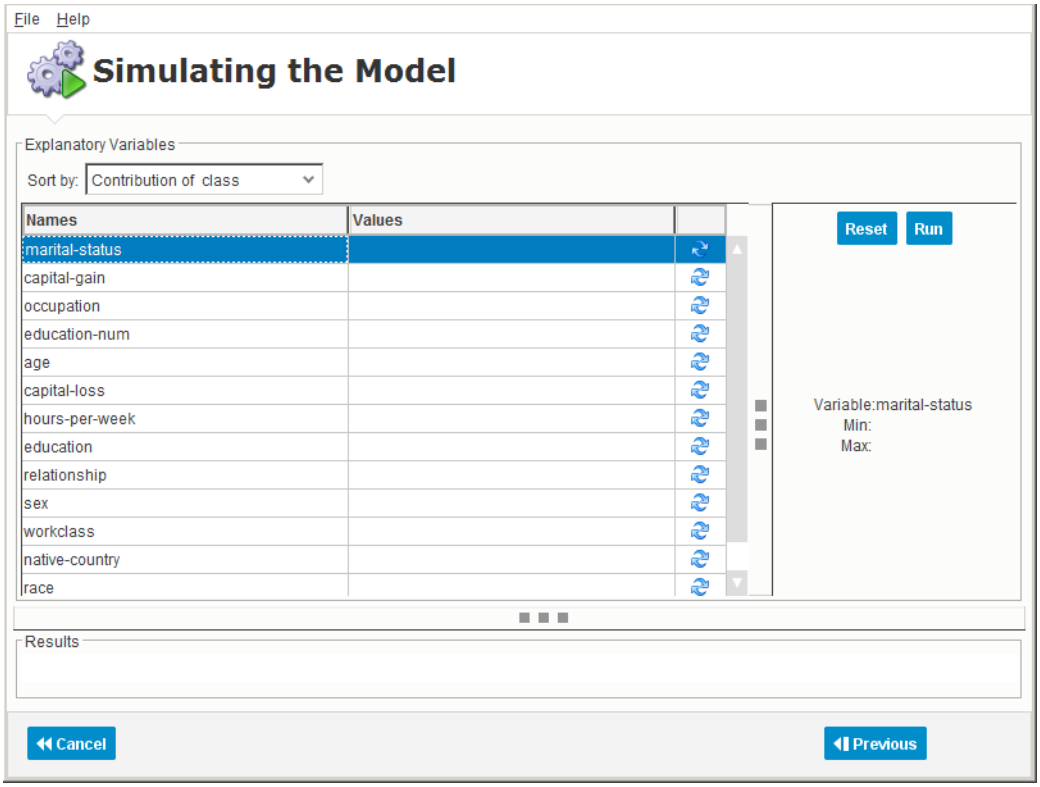
Once the simulation is complete, you will obtain the following results:

- The predicted value (score),
- The probability that this observation belongs to the target category of the target variable.

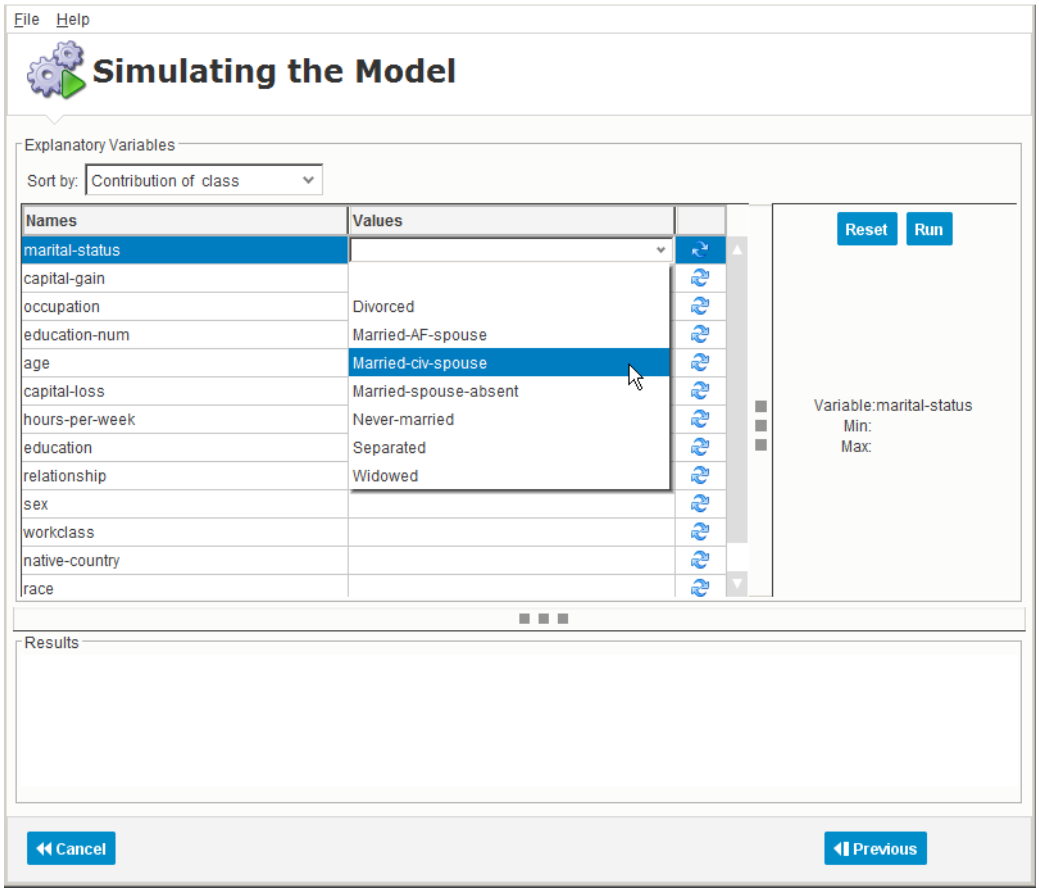
To Simulate a Model

1. On the screen *Using the Model*, click the option *Simulation*.

   The screen *Simulating the Model* appears.
2. On the left side of the screen (*Explanatory variables*), select a variable, such as `marital-status`. Its values appear in the section *Modifying values*, on the right side of the screen.



3. In the section *Modifying values*, in the *Value* field, select or enter a value, such as `Married-civ-spouse`. The value appears in the table of *Explanatory variables*, across from the selected variable.



4. If you would like to select other explanatory variables, go back to step 2. Otherwise, go to step 5.
5. Click the *Run* button to perform a model simulation.

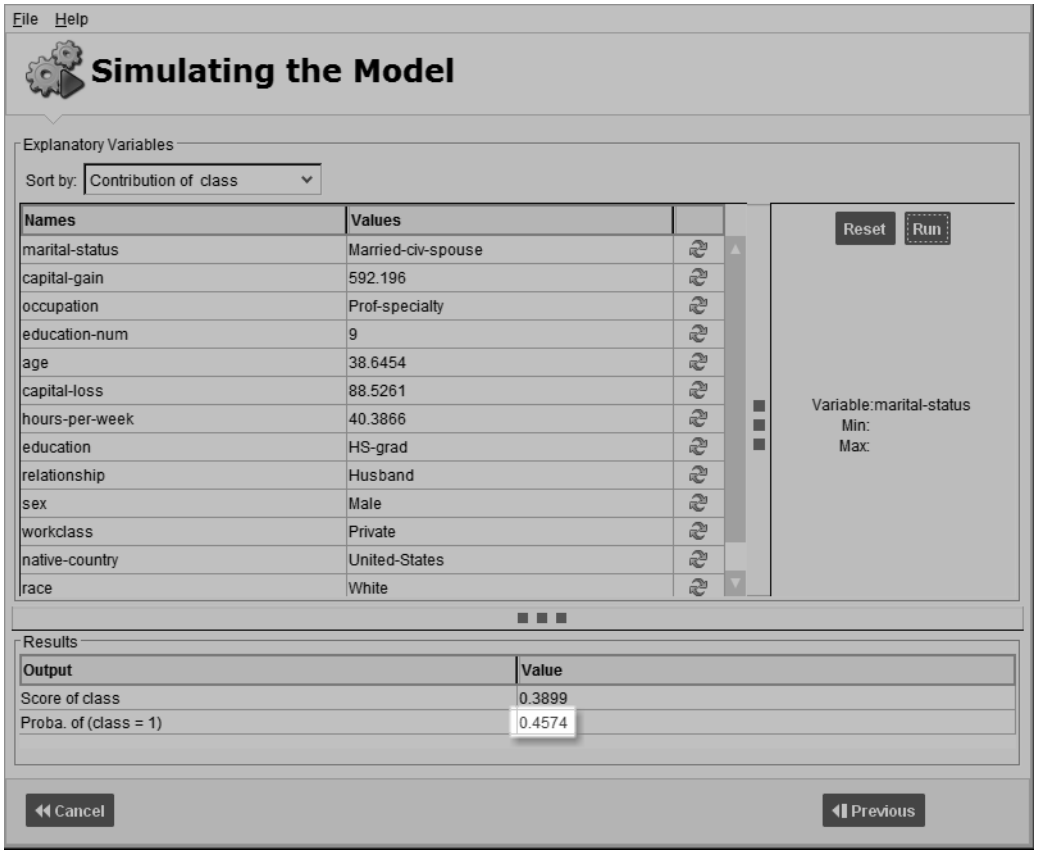
   The results of the simulation appears in the Results section. You will obtain the predicted value (score) of the observation described in the table of Explanatory variables, as well as the probability that this observation belongs to the target category of the target variable. In our example, only one variable has been defined. The probability that this observation belongs to the target category of the target variable is 0.1120.

   Note that certain variables of the table of Explanatory variables were automatically completed upon execution of the simulation. In fact, the model automatically completed certain missing values that were essential to the simulation.

   These values are listing in the following table.

Type of variable	Default value
continuous variable	the mean value
nominal variable	the most frequent category
ordinal variable	the most frequent category

   These changes are reflected in the left part of the screen after clicking the *Run* button



6. You can modify the value of an explanatory variable and run the simulation again to measure the effect of that change with respect to the target variable. For instance:
   a. Assign the value Widowed to the variable marital-status in place of the value `Married-civ-spouse`.
   b. Run the simulation. The probability now obtained is 0.0040.
7. Click the *Reset* button to run the simulation again.

5.1.7.4.4 Refining a Model

The application allows you to refine a currently open model. For instance, you can:

- Reduce the number of explanatory variables used by the model while maintaining the initial quality (KI) and robustness (KR),
- Generate a model of degree 2 using the most significant variables of the degree 1 model.

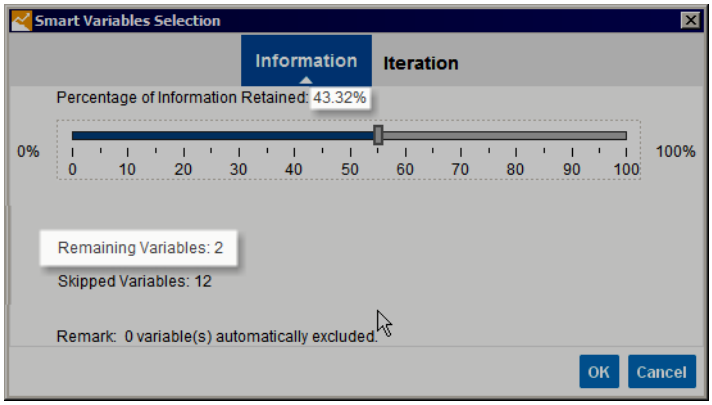
> **i Note**
>
> If your dataset contains date or datetime variables, automatically generated variables will appear in this panel. For more information, refer to the information on date and datetime variables in the topic on variable storage formats.

1. On the screen *Using the Model*, click the option *Select Variables*.

   The screen *Selecting Contributory Variables* appears.

2. In the *Targets* list, select the target variable for which you want to select the contributory variables.
3. Click the button *Smart Selection*. The window *Smart Variables Selection* opens.
4. On the bar *Percentage of Information Retained*, move the cursor to change the amount of information to keep; the number of variables selected changes accordingly.

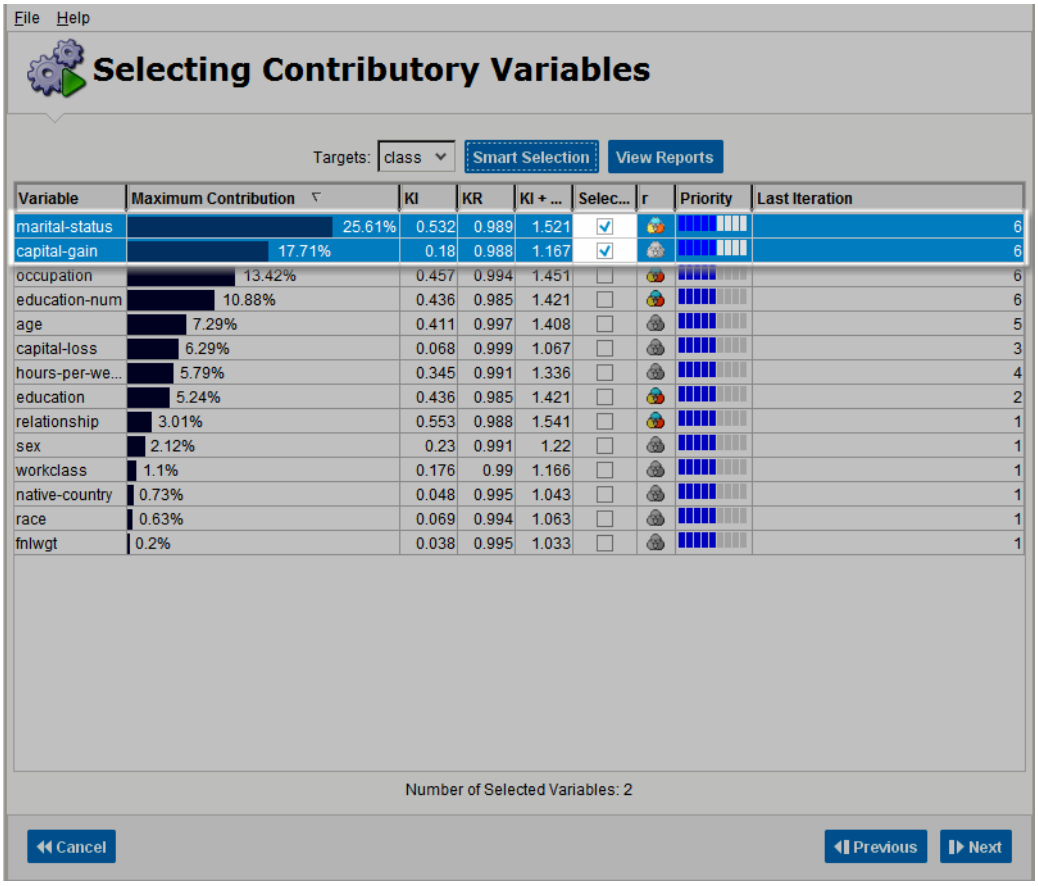
   The further this cursor is moved to the left, the more variables are excluded. The variables excluded are selected automatically as a function of their significance with respect to the model. For instance, the figure below shows that to retain only two variables out of the original fourteen, you should keep 43.07% of the information contributed by the model.



> **i Note**
>
> Certain variables in the training dataset may contribute no information, such as constant value variables. These can therefore be automatically excluded from the model during the training phase. The number of variables excluded is displayed as a *Remark*. In the figure above, this number is equal to "0".

5. Click *OK*.

   The window *Smart Variables Selection* closes and the panel *Selecting Contributory Variables* is updated with the selected variables, allowing you to view the kept variables and the excluded ones. In our example, the application automatically determined that the two explanatory variables that contributed the most information to explain the target variable were the variables `marital-status`, and `capital-gain`.



6. Click *Next*.

   A message "*This will reset the current. Do you really want to do this?*" appears.
7. Click *Yes* to move to the screen *Selecting Variables*.
8. Resume the model configuration from the step selecting variables.

5.1.7.4.5 Generating the Source Code of a Model

1. In the list *Target to be used*, choose the target of model.
2. Use the list *Information to be generated* to select the type of results:

Selected Option	Results of the Generated Model
Score/Estimates	score value (classification) or estimates (regression)
Probability	score value and probability value, except for HTML and all SQL codes for which only the probability value is provided.
Bar	score value and error bar value, except for HTML and all SQL codes for which only the error bar value is provided.

> ⚠ Caution
>
> Both options Probability and Bar are only available for regression/classification models with nominal targets.

> ℹ Note
>
> In the case of a continuous variable, the generated code (SQL for example) always includes a number of categories that is higher than in the user defined structure or as given by the parameter band count if no user has structure has been set. Indeed, the encoding of variables adds target curve points to increase the accuracy of coding according to the training dataset. These points split some existing categories and thus increase the number of categories in the generated code.

3. In the section *Code Settings*, select the code type to be generated.
4. Click the *Browse* button associated with the *Folder* field and select a folder to save the generated file.
5. In the field *Generated File*, enter the name of the exported file. If you want to replace an existing file, use the *Browse* button to select it.
6. If you have selected the option *View Generated Code*, it is displayed at the end of the generation process.
7. Click the *Generate* button.

5.1.7.4.5.1 List of Generated Codes

The following table lists the available codes with their particularities:

Generated Code	Comment
AWK	
C	see C Code Generator documentation
CCL Code	for scoring in SAP HANA Smart Data Streaming
CPP	
DB2UDF	
HANA	
HANAUDF	
Hive	
HTML (Javascript)	contains a form to fill which reproduces the model
JAVA	needs the KxJRT.jar package to run
JSON	
ORACLE	
OracleUDF	
PMML3.2	
PostgreSQL	

Generated Code	Comment
SAS	
ScoreCard	only available for regression/classification models
SPARK	
SQLDB2	
SQLNetezza	
SQLServer	wraps variable names with [ ]
SQLServerUDF	
SQLTeradata	Teradata databases
SQLVertica	
SybaseIQ	
SybaseIQUDF	
TERAUDF	
VORA	

> **i Note**
>
> When generating SQL and SAS codes, you will be asked to provide the names of the key column and of the dataset used.

> **⚠ Caution**
>
> Only SQLServer key code handles trimmed data during its execution. For other codes, if data are not trimmed it may generate some differences.

5.1.7.4.5.2 Advanced Settings

UNICODE Mode

The option Activate UNICODE Mode allows you to generate the code selected in Unicode so that it supports non-latin languages such as Japanese, Russian, and so on. This option is particularly useful for SQL codes.

SQL/UDF Options

- The option *Do not generate code for non-contributive variables* allows you to exclude from the code all variables with a contribution of 0 since they do not influence the result. In some cases, this can significantly reduce the size of the generated code.
- You can either *Use the default separator* ("GO") or *Use a custom separator*.

5.1.7.4.5.3 Exporting the Model as a KxShell Script

The KxShell script export allows you to generate a KxShell script reproducing the current model. This script can be used to run models in batches.

One easy way to get special settings in exported KxShell scripts is to first do the corresponding operation in the graphical user interface. For example, if you run an auto-selection of variables before exporting the shell script, then the exported script will include the code needed to do the auto-reduction.

To save the KxShell script:

1. In the section *Save/Export* of the menu *Using the Model*, select the option *Export KxShell Script*.

   The panel *KxShell Script Generation* is displayed.
2. Use the *Browse* button located to the right of the *Folder* field to select where the script will be saved.
3. In the field *KxShell Script*, enter the name of the file in which the script will be saved.
4. In the frame *Model Dataset Description Saving*, select where you want to save the data description. The four available options are:

Option	Description
Save the Description in the Script	The data description is added in the KxShell script. Only one file is generated.
Save the Description with the Script	The data description is saved in an additional file in the same folder as the KxShell script.
Save the Description with the Data	The data description is saved in an additional file in the same folder as data used for the model.
Save the Description Separately	The data description is saved in an additional file. Indicate the type of the description (text file, data base, flat memory, ...) and the location where the data description should be saved.

> **i Note**
>
> When saving the description in an additional file, the file is named following this syntax: `KxDesc_<Dataset Role>_<Dataset Name>`. For example, for a training dataset named `Census.csv`, the description file name will be `KxDesc_Training_Census.csv`.

5. Additionally you can export the variable structure with relation to a target variable by checking the option *Generate Variable Structure From Statistics* and selecting the target variable in the list *Select a Target*.

   This option allows you to force the grouping of categories when training the model on new datasets.
6. Before exporting the script you can view the script by clicking the button *Script Preview*.
7. Validate to start the generation process. Once the script has been generated, the menu *Using the Model* is displayed.

5.1.7.4.6 Saving the Model

Once a model has been generated, you can save it. Saving it preserves all the information that pertains to that model, that is, the modeling parameters, its profit curves, and so on.

1. On the screen *Using the Model*, click the option *Save Model*.

   The screen *Saving the Model* appears.
2. Complete the following fields:

Option	Description
Model Name	This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
Description	This field allows you to enter the information of your choosing, such as the name of the training dataset used, the polynomial degree or the predictive power and prediction confidence obtained for the model. This information could be useful to you later for identifying your model

Data Type	This list allows you to select the type of storage in which you want to save your model. The following options are available:
	○ *Text files*, to save the model in a text file.
	○ *Database*, to save the model in a database.
	○ *Flat Memory*, to save the model in the active memory.
	○ *SAS Files*, to save the model in a SAS compatible file for a specified version of SAS and a specified platform (SAS v6 or 7/8 for Windows or Linux).
	○ *SAS Transport*, to save the model in a generic SAS compatible file:
	  ○ *Folder*: Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
	  ○ *File/Table*: This field allows you to enter the name of the file or table that is to contain the model. When saving as a text file, you must enter one of the following format extensions `.txt` (text file in which the data is separated by tabs) or `.csv` (text file in which the data is separated by commas).

5.1.7.4.6.1 Files Created When Saving a Model

When saving a model, the application creates a set of files/tables in the specified store. Some of these files are specific to the type of model. The following table lists the files or tables created when saving a model and in which case.

File	Description	Used By
`KxAdmin`	Lists all the models contained in the folder/database with additional information (date, version, name of the model, comments).	All models created with Automated Analytics
`<Model_name>`	File named after the model and containing all the model data, except graphs information. Graphs are stored in additional tables (see below).	All models created with Automated Analytics

File	Description	Used By
`KxInfos`	Indicates which additional tables are needed by the model.	All models created with Automated Analytics
`KxOlapCube`	Stores the OLAP Cube used by the decision tree when the option Regression/Classification as Decision is activated.	Regression/Classification models with decision tree
`KxLinks`	Contains the links from the graphs of the model.	Social model only
`KxNodes`	Lists all the nodes from all the graphs and their attributes.	Social model only
`KxCommunities`	Matches the nodes to their communities, if the community detection was enabled.	Social model only

> ⚠ Caution
>
> When sharing or sending a model, all these files must be joined to the model or the recipient will not be able to open the model.

5.1.7.4.7 Opening a Model

Once saved, models may be opened and reused in the application.

1. On the main application screen, select *Load a Model*. The screen Opening a Model appears.
2. In the *Data Type* list, select one of the following options depending upon the format of the model that you want to open:
   - *Text files*
   - *Database*
   - *SAS files*
   - *SAS Transport*
3. 3 Click the *Browse* button.

   The *Data Selection* dialog appears.
4. Select the folder that holds the model that you want to open. The list of models contained in that folder appears providing the following information for each model.

Column	Description	Values
Name	Name under which the model has been saved	Character string

Column	Description	Values
Class	Class of the model, that is the type of the model	○ `Kxen.Classification`: Classification/Regression with nominal target ○ `Kxen.Regression`: Classification/Regression with continuous target ○ `Kxen.Segmentation`: Clustering with SQL Mode ○ `Kxen.Clustering`: Clustering without SQL Mode ○ `Kxen.TimeSeries`: Time Series ○ `Kxen.AssociationRules`: Association Rules ○ `Kxen.SimpleModel`: Classification/Regression and Clustering multi-target models, any other model
Version	Number of the model version when the model has been saved several times	Integer starting at 1
Date	Date when the model has been saved	Date and time in the format yyyy-mm-dd hh:mm:ss
Comment	Optional user defined comment that can be used to identify the model	Character string

5. Select a model from the list.
6. Click the *Open* button.

   The screen *Using the Model* appears.

5.2 Segmentation/Clustering

5.2.1 About Modeler - Segmentation/Clustering

This section of the guide is addressed to people who want to evaluate or use Modeler to build Segmentation/Clustering models. Use of this guide does not require any prior expertise in statistics or databases.

This part introduces you to the basic concepts and main functionalities of Modeler and its feature Segmentation/Clustering. Using an application scenario, you can create your first model with confidence.

The purpose of Modeler - Segmentation/Clustering is to describe a dataset, by breaking it down into homogeneous data groups, or clusters.

> **Example**
>
> Your firm is in the process of bringing products A and B to market. Modeler - Segmentation/Clustering allows you to:
>
> - Regroup your customers into several homogeneous groups,
> - Understand the behavior of each of these groups with respect to products A and B.

5.2.2 Segmentation/Clustering Application Scenario Description

In this scenario, you are the Marketing Director of a large retail bank. You want to customize your communications using data modeling. The bank wants to offer a new financial product to its customers. Your project consists of launching a direct marketing campaign aimed at promoting this product.

In order to customize the marketing messages from the bank and improve communication with the various customers and prospects for this new product, the senior management of the bank asks you to build a segmentation model of the customers of this product.

Using Modeler - Segmentation/Clustering, you can rapidly develop a descriptive model with the least possible cost. This model shows the characteristic profiles of the customers interested in your new product, and thus responds to your business issue and fulfills your objectives.

Your Objective

Consider the following case.

Using Regression/Classification, you have contacted the prospects most likely to be interested in your new financial product, and identified the ideal number of prospects to contact out of the entire database meeting the deadlines and within the budget you were allowed.

To improve the rate of return of your campaign, senior management asks you to:

- Build a segmentation model of your customers,
- Analyze the characteristics of the identified clusters,
- Define customized communications for each cluster.

The segmentation model in particular should allow you to distinguish customer clusters by virtue of their propensity to purchase the new high-end savings product proposed by your firm. You will optimize your understanding of your customers.

Your Approach

For organizational reasons, you want to define five groups of customers, or clusters, and describe the customer profiles for each of these groups.

To accomplish this project, you will use the sample of 50,000 people who responded to your first test, during the previous campaign. This file corresponds to the sample file `Census01.csv`, provided with the application and described in the section Introduction to Sample Files.

Your Business Issue

In your marketing database, you have:

- A list of 1,000,000 prospects,
- A list of 50,000 prospects (people selected during the test phase of your campaign), whose responses to the campaign are known. This sample thus constitutes a training dataset. This sample, taken from the complete database, also exhibits some missing values.

Your business issue thus consists of:

- Rapidly building a segmentation model using the training dataset (or sample). The clusters obtained will allow you to better understand the profiles of the individuals in your database as a function of their propensity to purchase.
- Then applying the segmentation model obtained from the training data to the entire list of prospects to determine which cluster each individual should belong.

5.2.3 Scenario Solutions

5.2.3.1 Intuitive Method

This method consists of using your knowledge of the various profiles exhibited by your customers. Thanks to the domain-specific knowledge that you have of your customers, you determine the criteria of the segmentation model intuitively, and build the clusters yourself.

The main disadvantage of this method is that the number of information items available for each customer will invariably grow with time. The more data your database accumulates, the harder it is for you to manually create clusters that take all data into consideration and to develop a response to your business issue. Furthermore, as the increasing volume of information requires you to build segmentation models with increasing frequency, the time required to build these segmentation models becomes increasingly more significant.

Finally, management may want you to rationalize your methods, and to perform your segmentation using a method not based purely on your intuition. Defending your segmentation method based on intuition may be difficult.

5.2.3.2 Classical Statistical Method

On the basis of the information that you have, a data mining expert could build a segmentation model. In other words, you could ask a statistical expert to create a mathematical model that would allow you to build clusters based on the profiles of your customers.

To implement this method, the statistician must:

- Perform a detailed analysis of your database.
- Prepare your database down to the smallest detail, specifically, encoding the variables as a function of their type (nominal, ordinal or continuous) in preparation for segmentation. The encoding strategy used will determine the type of segmentation model obtained. At this step, the statistician may unconsciously bias the results.

- Test different types of algorithms (K-means, both ascending and descending hierarchical segmentation models) and select the one best suited to your business issue.
- Evaluate the relevance of the clusters obtained, in particular, the response to your domain-specific business issue.

After a few weeks, the statistical expert will be able to provide a certain number of clusters, or homogeneous groups, to which each of the individuals of your database are assigned.

This method presents significant constraints. You must:

- Ensure that your statistical expert, who is usually from an external department, is available for the scheduled period,
- Ensure that the modeling costs will fit into your budget,
- Spend time explaining your domain-specific business issue to the statistician,
- Spend time understanding the results that are provided,
- Ask a programmer to write a program to determine the cluster associated with any new individual added to your database.

In addition, this method is not systematic. Two statisticians performing this segmentation on the same dataset could obtain different results.

5.2.3.3 Automated Analytics Method

Segmentation/Clustering allows you to build a segmentation model of your customers in a few minutes, taking into consideration the interest expressed by your customers in your new product.

Segmentation/Clustering automatically detects interactions between the variables to build homogeneous subsets, or clusters. Each cluster is homogeneous with respect to the entire set of variables, and in particular with respect to the target variable, that is, for example, "responded positively to my test".

You will discover the characteristics of different clusters, such as those clusters with an excellent response rate and those with a poor response rate. In addition, if your customer database contains customer expenditures on your other products, you will also obtain information on product sale synergies, by cluster.

Using Segmentation/Clustering, you have access to all the analytical features needed to define the type of message to be sent to the cluster for each customer. You have homogeneous clusters that will allow you to respond to your business issue. Of particular importance, this segmentation is systematic: the results obtained do not represent a particular point of view of your data, and is robust or consistent. Two people performing this segmentation using the method would obtain the same results.

5.2.4 Introduction to Sample Files

This guide is accompanied by the following sample data files:

- A data file `Census01.csv`.
- The corresponding description file `desc_census.csv`.

These files allow you to evaluate Automated Analytics features and take your first steps in using it.

`Census01.csv` is the sample data file that you will use to follow the scenarios for Regression/Classification and Segmentation/Clustering. This file is an excerpt from the American Census Bureau database, completed in 1994 by Barry Becker.

> **i Note**
>
> For more information about the American Census Bureau, see http://www.census.gov .

This file presents the data on 48,842 individual Americans, of at least 17 years of age. Each individual is characterized by 15 data items. These data, or variables, are described in the following table.

Variable	Description	Example of Values
age	Age of individuals	Any numerical value greater than 17
workclass	Employer category of individuals	• Private • Self-employed-not-inc
fnlwgt	Weight variable, allowing each individual to represent a certain percentage of the population	Any numerical value, such as "0", "2341" or 205019".
education	Level of study, represented by a schooling level, or by the title of the degree earned	• 11th • Bachelors
education-num	Number of years of study, represented by a numerical value	A numerical value between 1 and 16
marital-status	Marital status	• Divorced Never-married
occupation	Job classification	• Sales • Handlers-cleaners
relationship	Position in family	• Husband • Wife
race	Ethnicity	• White • Black
sex	Gender	• Male • Female
capital-gain	Annual capital gains	Any numerical value
capital-loss	Annual capital losses	Any numerical value
native country	Country of origin	• United States • France
class	Variable indicating whether or not the salary of the individual is greater or less than $50,000	• "1" if the individual has a salary of greater than $50,000 • "0" if the individual has a salary of less than $50,000

> **i Note**
>
> In order to avoid complicating the application scenarios, the variable fnlwgt is used as a regular explanatory variable in these scenarios, and not as a weight variable.

5.2.5 SAP Predictive Analytics

To accomplish the scenario, you will use the Automated Analytics toolset of SAP Predictive Analytics.

5.2.5.1 To Start SAP Predictive Analytics

1. Select *Start* > *Programs* > *SAP Business Intelligence* > *SAP Predictive Analytics Desktop* > *SAP Predictive Analytics*.

   The SAP Predictive Analytics start panel appears.
2. Click the feature you want to use.

5.2.5.2 Editing the Options

- To edit the application options:

1. In the *File* menu, click *Preferences...*.

   The window *Edit Options...* appears.
2. The following options can be modified for Segmentation/Clustering:

Option	Description
Category	Options
General	○ Country ○ Language ○ Message Level ○ Log Maximum Size ○ Message Level for Strange Values ○ Display the Parameter Tree ○ Number of Store in the History ○ Always Exit without Prompt ○ Include Testing in Default Partition Strategy
Stores	○ Default Store for Apply-in Dataset ○ Default Store for Apply-out Dataset ○ Default Store to Save Models
Metadata Repository	○ Enable Single Metadata Repository ○ Edit Variable Pool Content

Option	Description
Graphic	○ Profit Curve Points ○ Bar Count Displayed ○ No InfiniteInsight® Look and Feel ○ Display 3D Chart ○ Disable Double Buffering ○ Optimize for Remote Display ○ Remember Size and Position when Leaving
Report	○ Number of Variables of Interest ○ Active Style Sheet ○ Customize Style Sheets

5.2.6 Creating a Model Using Modeler

Data modeling with the application is subdivided into four broadly defined stages:

- Defining the Modeling Parameters
- Generation and Validation of the Model
- Analysis and Understanding of the Analytical Results
- Using a Generated Model

5.2.6.1 Step 1 - Defining the Modeling Parameters

To respond to your business issue, you want to:

- Break down the sample of 50,000 prospects who responded to the test phase of your marketing campaign into homogeneous groups.
- Describe each of these groups and provide customized communication for each of these different groups.

The Segmentation/Clustering feature allows you to create descriptive models.

The first step in the modeling process consists of defining the modeling parameters:

- Select a data source to be used as a training dataset.
- Describe the selected dataset.
- Select the variables.
- Select the explanatory variables.
- Check the Modeling Parameters.
- Define the number of clusters. This step is optional.

5.2.6.1.1 Selecting a Data Source

Use the file `Census01.csv` as a training dataset.

This file represents the sample that you had extracted from your database and used for the test phase of your direct marketing campaign. As specified in your test plan, this file contains data concerning 50,000 prospects, for whom you now know the behavior with respect to the new financial product:

- 25% of the prospects showed themselves to be clearly interested. They chose to accept a an invitation for a meeting with one of your sales channel agents,
- 75% of the prospects declined your invitation.

In this file, you created a new variable Class, which corresponds to the reaction of prospects contacted during the test. You assigned:

- The value "1" to those prospects who responded positively to your invitation,
- The value "0" to those prospects who responded negatively to your invitation.

To select a data source:

1. On the screen *Select a Data Source*, select the data source format to be used (*Text files*, *Data Base*, ...).
2. Click the *Browse* button.

   In the *Data Selection* dialog, browse to the `<Installation Path>/Samples` folder.

   > **i Note**
   >
   > Depending on your environment, the `Samples` folder may or may not appear directly at the root of the list of folders. If you selected the default settings during the installation process, you will find the Samples folder located in `C:Program Files/SAP Analytics/Desktop <version number>/Automated/`.

3. Double-click the *Samples* folder, then the *Census* folder.
4. Select the file `Census01.csv`, then click *OK*.

   The name of the file appears in the *Dataset* field.
5. Click *Next*.

   The screen *Data Description* appears. Now you are ready to describe the data selected.

5.2.6.1.2 Describing the Data Selected

For this scenario:

- Select *Text Files* as the *Data Type*.
- Use the file `Desc_Census01.csv` as the description file for the `Census01.cvs` data file.

To Select a Description File:

1. On the screen *Data Description*, click the button *Open Description*.

   The Load a Description window opens.
2. Select the type of your description file.
3. In the *Folder* field, select the folder where the description file is located with the *Browse* button.

   The folder selected by default is the same as the one you selected on the screen *Select a Data Source*.
4. In the *File Name* field, select the file containing the dataset description with the *Browse* button.

> ⚠ Caution
>
> When the space used for model training contains a physical variable named `KxIndex`, it is not possible to use a description file without any key for the described space.
>
> When the space used for model training does not contain a physical variable named `KxIndex`, it is not possible to use a description file including a description about a `KxIndex` variable since it does not exist in current space.

5. Click *OK*.

   The window *Load a Description* closes and the description is displayed on the screen *Data Description*.
6. Click *Next*.

5.2.6.1.2.1 Why Describe the Data Selected?

In order for the application to interpret and analyze your data, the data must be described. To put it another way, the description file must specify the nature of each variable, determining the storage format and type:

- Storage format: number (*number*), character string (*string*), date and time (*datetime*) or date (*date*).

  > ℹ Note
  >
  > When a variable is declared as *date* or *datetime*, the date coder feature automatically extracts date information from this variable such as the day of the month, the year, the quarter and so on. Additionnal variables containing this information are created during the model generation and are used as input variables for the model.
  >
  > The date coder feature is disabled for Time Series.
- Type: *continuous*, *nominal*, *ordinal* or *textual*.

For more information about data description, see the related topics.

5.2.6.1.2.2 How to Describe Selected Data

To describe your data, you can:

- Either use an existing description file , that is, taken from your information system or a previously created description file from the application features,
- Or create a description file using the Analyze option from the modeling assistant. In this case, you must validate the description file obtained. You can save this file for later use.

> ⚠ Caution
>
> The description file obtained using the *Analyze* option results from the analysis of the first 100 lines of the initial data file. In order to avoid all bias, we encourage you to randomly sort your dataset outside the application before performing this analysis.

5.2.6.1.2.3 Viewing the Data

To help you validate the description when using the *Analyze* option, you can display the first hundred lines of your dataset.

1. Click the button *View Data*.

   A new window opens displaying the dataset top lines:
2. In the field *First Row Index*, enter the number of the first row you want to display.
3. In the field *Last Row Index*, enter the number of the last row you want to display.
4. Click the *Refresh* button to see the selected rows.

5.2.6.1.2.4 A Comment about Database Keys

For data and performance management purposes, the dataset to be analyzed must contain a variable that serves as a key variable. Two cases should be considered:

- If the initial dataset does not contain a key variable, a variable index `KxIndex` is automatically generated by the application. This will correspond to the row number of the processed data.
- If the file contains one or more key variables, they are not recognized automatically. You must specify them manually in the data description.

To Specify that a Variable is a Key

1. In the *Key* column, click the box corresponding to the row of the key variable.
2. Type in the value "1" to define this as a key variable.

5.2.6.1.2.5 Defining a Variable Structure

There are three ways to define a variable structure:

- by first extracting the categories from the variable statistics, then editing or validating the suggested structure.
- by importing the structure from an existing model.
- by building a new structure from scratch.

The option Optimal Grouping allows you to let Data Encoding group together the categories groups defined in the variable structure if they bring the same information.

The last column of the description table indicates the state of the structure of each variable. The following table lists the possible states of a variable structure.

Icon	State	Description

	undefined	Data Encoding will automatically determine the categories grouping depending on their interaction with the target variable.
	non-editable	The structure for an ordinal string variable cannot be modified.
	defined by extraction from the variable statistics	The user must open and validate the variable structure.
	defined by the user or imported from an existing model	

> **i Note**
>
> A translation of the variable categories has no influence on the variable structure, which has to be set according to the original values of the variable.

5.2.6.1.3 Filtering the Dataset

In order to accelerate the learn process and to optimize the resulting model, you can apply a filter to your dataset. For more information, see the related topic.

5.2.6.1.4 Translating the Variable Categories

You can translate the categories of a nominal variable, save the translation or load an existing translation. This translation has no influence on the variable structure, which has to be set according to the original values of the variable. For more information, see the related topic.

5.2.6.1.5 Selecting Variables

Once the training dataset and its description have been entered, you must select different variables:

- One or more targets variables.
  The Segmentation/Clustering feature is capable of segmenting a dataset independently, that is, it does not require that a target variable be selected. However, even though this is not required, we strongly recommend selecting a target variable. For the process of segmenting, a dataset gains maximum meaning only when it is accomplished in relation to a domain-specific business issue, expressed in the form of a target variable.
- Possibly a weight variable.
- The explanatory variables.

5.2.6.1.5.1 Target Variables

For this scenario, select the variable Class as your target variable, that is, the variable that indicates the probability of an individual responding in a positive or negative manner to your campaign.

1. On the screen *Selecting Variables*, in the section *Explanatory variables selected* (left hand side), select the variables you want to use as Target Variables.

   > **i Note**
   >
   > On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2. Click the button > located on the left of the screen section *Target Variables* (upper right hand side).

   The variable moves to the screen section *Target Variables*.

   You can also select a variable in the screen section *Target Variables* and click the button < to move the variables back to the screen section *Explanatory variables selected*.

5.2.6.1.5.2 Weight Variable

Selecting a Weight Variable enables to set the Weight Quantum option available in the Advanced Model Parameters.

For this scenario, do not select a weight variable.

1. On the screen *Selecting Variables*, in the section *Explanatory variables selected* (left hand side), select the variables you want to use as Weight Variables.

   > **i Note**
   >
   > On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2. Click the button > located on the left of the screen section *Weight Variables* (middle right hand side).

   The variable moves to the screen section *Weight Variables*.

   You can also select a variable in the screen section *Weight Variables* and click the button < to move the variables back to the screen section *Explanatory variables selected*.

5.2.6.1.5.3 Explanatory Variables

By default, and with the exception of key variables, all variables contained in your dataset are taken into consideration for generation of the model. You may exclude some of these variables.

The decision whether to include or exclude a variable for generation of your segmentation model depends upon domain-specific considerations. Your domain-specific knowledge allows you to determine which variables

are the most useful for description of the clusters or homogeneous groups. A regression model generated using Regression/Classification would also be used as a tool to determine the variables with the greatest explanatory power for a given phenomenon.

For this Scenario:

- Exclude the variable `KxIndex`, as this is a key variable. Since the initial dataset does not contain a key variable, the application generated `KxIndex` automatically.
- Retain all the other variables.

1. To exclude some variables from data analysis, on the screen *Selecting Variables*, in the section *Explanatory Variables Selected* (left hand side), select the variable to be exluded.

   > **i Note**
   >
   > On the screen *Selecting Variables*, variables are presented in the same order as that in which they appear in the table of data. To sort them alphabetically, select the option *Alphabetic sort*, presented beneath each of the variables list.

2. Click the button > located on the left of the screen section *Excluded Variables* (lower right hand side). The variable moves to the screen section *Excluded Variables*. Also, select a variable in the screen section Excluded Variables and click the button < to move the variables back to the screen section *Explanatory Variables Selected*.
3. Click *Next*.

   The screen *Parameters of the Model* appears.

5.2.6.1.6 Checking Modeling Parameters

The screen *Summary of Modeling Parameters* allows you to check the modeling parameters just before generating the model.

> **i Note**
>
> The screen *Summary of Modeling Parameters* contains an *Advanced* button. By clicking this button, you access the screen *Advanced Model Parameters*. For more information about these parameters, Setting Up the Advanced Options.

The *Model Name* is filled automatically. It corresponds to the name of the target variable (class for this scenario), followed by the underscore sign ("_") and the name of the data source, minus its file extension (`Census01` for this scenario).

Before generating the model, you can define the number of clusters that you want to obtain. These fields allow you to specify how many clusters will be generated by the model. By default the number of clusters is set to 10. The higher the number of segments, the lower the robustness (KR). The lower the number of segments, the lower the information (KI). One should generally start with the default number and then go further with more or fewer clusters based on the results.

- For supervised segmentation (with a target), choose the best number of segments, for example, [5;10] means that you are requesting 5 to 10 clusters. The application computes the "best number of clusters" using the metric KI+KR. For instance, you may have 7 clusters.

- For unsupervised segmentation (without target), the application chooses the minimum number of clusters, for instance [10;10] which means that you requested 10 clusters.

Choosing to *Calculate SQL Expressions* allows you to see, in the model debriefing, the SQL Expressions used to generate each cluster.

> **i Note**
>
> When you activate the option *Calculate SQL Expressions*, the application generates an additional cluster that contains the unassigned records. For more details on SQL expressions and unassigned records, see Difference Between Standard Cross Statistics and SQL Expressions.

For this scenario, keep the default settings.

5.2.6.1.6.1 Setting Up the Advanced Options

The *Advanced Model Parameters* panel provides you with several options.

You may calculate the cross-statistics for the model to be generated, define the target key value, choose the distance computing option or choose the encoding strategy.

Calculating the Cross Statistics

The *Set Target Keys* value option lists the target variables selected in the *Selecting Variables* screen and allows you to choose their key value.

To Define the target key value In the *Target Key* field, enter the key value of the target variable.

Choosing the Distance Computing Method

The *Distance* list allows you to specify the distance used to compare input data encoded by the data encoder.

To choose the distance computing method, in the *Distance* drop-down list, select among these options:

- *Chessboard*: maximum of absolute differences between coordinates (LInf).
- *Euclidean*: square root of sum of square differences between coordinates (L2).
- *City Block*: sum of absolute differences between coordinates (L1).
- *System Determined* (default value): Lets the system determine the best distance to be used according to the model build settings.

> **i Note**
>
> The current policy is to use LInf either in unsupervised mode or when the clusters SQL expressions have been asked for, and L2 otherwise.

5.2.6.1.6.1.1 Encoding Strategy

The *Encoding Strategy* option refers to the kind of encoding the segmentation engine is expecting from the data encoder of Automated Analytics.

- To Choose an Encoding Strategy:
  - Choose among the following options from the drop-down list:

Option	Description
System Determined	Lets the system select the best encoding according to the model parameters. The *Target Mean* encoding is used for supervised models.Otherwise, variables are encoded using the *Unsupervised* scheme.
Target Mean	**Default value for supervised clustering** Each value of a continuous input variable is replaced by the mean of the target for the segment the value belongs to. Each category of a nominal input variable is replaced by the mean of the target for this category. In case of a nominal target variable, the mean of the target corresponds to the percentage of positive cases of the target variable for the input variable category.
Uniform	Each variable segment is encoded in the range `[-1;+1]` so that the distribution of the variables is uniform.
Unsupervised	**Default value for unsupervised clustering** A target free strategy. Only segment frequency is used to encode variables.

The following options will only be displayed when all variables are continuous:

Option	Description
Natural	This option does not transform the input data.
Min Max	This option encodes the categories of the variable in the range [0,1], where 0 corresponds to the minimum value of the variable and 1 corresponds to the maximum value.
Standard Deviation Normalization	This option performs a normalization based on the variable mean and standard deviation. $\frac{x - Mean}{StdDev}$

5.2.6.1.6.2 Activating the Autosave Option

The *Model Autosave* panel allows you to activate the option that will automatically save the model at the end of the generation process and to set the parameters needed when saving the model.

To activate the option, proceed as follows:

1. In the *Summary of Modeling Parameters* panel, click the *Autosave* button.
   The *Model Autosave* panel is displayed.
2. Check *Enable Model Autosave*.
3. Set the parameters listed in the following table.

Parameter	Description
Model Name	This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
Description	This field allows you to enter the information you want, such as the name of the training dataset used, the polynomial degree or the performance indicators obtained. This information could be useful to you later for identifying your model. Note that this description will be used instead of the one entered in the *Summary of Modeling Parameters* panel.
Data Type	This list allows you to select the type of storage in which you want to save your model. The following options are available: ○ *Text Files*, to save the model in a text file. ○ *Data Base*, to save the model in a database. ○ *Flat Memory*, to save the model in the active memory.
Folder	Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
File/Table	This field allows you to enter the name of the file or table that is to contain the model. When saving the model as a text file, you must enter one of the following format extensions `.txt` (text file in which the data is separated by tabs) or `.csv` (text file in which the data is separated by commas).

4. Click *OK*.

5.2.6.2 Step 2 - Generating and Validating the Model

Once the modeling parameters are defined, you can generate the model. Then you must validate its performance using the predictive power (KI) and the prediction confidence (KR):

- If the model is sufficiently powerful, you can analyze the responses that it provides in relation to your business issue (see Step 3 - Analyzing and Understanding the Model Generated), and then apply it to new datasets (see Step 4 - Using the Model).
- Otherwise, you can modify the modeling parameters in such a way that they are better suited to your dataset and your business issue, and then generate new, more powerful models.

5.2.6.2.1 Generating the Model

To generate the model:

1. On the *Advanced Model Parameters* screen, click *Generate*.

   The screen *Training the Model* appears. The model is being generated. A progress bar allows you to follow the process.
2. If the *Autosave* option has been activated in the panel *Summary of Modeling Parameters*, a message is displayed at the end of the learning process confirming that the model has been saved.
3. Click *Close*.
4. Once the model has been generated, click *Next* to go to panel *Using the Model*.

5.2.6.2.2 Following the Progress of the Generation Process

There are two ways for you to follow the progress of the generation process:

- The Progress Bar displays the progression for each step of the process. It is the screen displayed by default.
- The Detailed Log displays the details of each step of the process.

To display the Progression Bar

Click *View Type* and select (Progress).

The progression bar screen appears.

To Display the Detailed Log

Click *View Type* and select the (Log) button.

The detailed log displays the details of each step of the process.

To Stop the Learning Process

1. Click the (Stop Current Task) button.
2. Click the *Previous* button.
   The screen *Summary of Modeling Parameters* appears.
3. Go back to the section on checking modeling parameters.

5.2.6.2.3 Validating the Model

Once the model has been generated, you must verify its validity by examining the performance indicators:

- The predictive power allows you to evaluate the explanatory power of the model, that is, its capacity to explain the target variable when applied to the training dataset. A perfect model would possess a predictive power equal to 1 and a completely random model would possess a predictive power equal to 0.
- The prediction confidence defines the degree of robustness of the model, that is, its capacity to achieve the same explanatory power when applied to a new dataset. In other words, the degree of robustness corresponds to the predictive power of the model applied to an application dataset.

To see how the predictive power and the prediction confidence are calculated, see Predictive Power, Prediction Confidence and Model Graphs.

Besides the Predictive Power (KI) and the Prediction Confidence (KR) , the application also provides you with two commonly known indicators:

- the classification rate , in case of a classification model,
- the Pearson Square Correlation coefficient (named R2 in the application), in case of a regression model.

Both indicators can be used to compare the application results with results obtained through other data mining tools.

> **i Note**
>
> Validation of the model is a critically important phase in the overall process of Data Mining. Always be sure to assign significant importance to the values obtained for the predictive power and the prediction confidence of a model.

To validate a segmentation model, you can also observe the value of the indicators "frequency" and "target mean" for each of the identified clusters. Specifically, the most interesting clusters of the segmentation model will possess an elevated "frequency" and a "target mean" that deviates from the "target mean" of the entire dataset. Note that a segmentation model with a low predictive power may conceal precisely this type of cluster.

To find out how the frequency and target mean for a cluster are calculated, see Understanding the Detailed Description of Clusters.

For this scenario, the model generated possesses:

- A predictive power equal to 0.703,
- A prediction confidence equal to 0.987.

The model performs sufficiently well. You do not need to generate another.

To Validate the Model Generated

1. Verify the Predictive Power (KI) and the Prediction Confidence (KR) of the model.

   > **i Note**
   >
   > As a general note, other indicators are provided in addition to the predictive power and the prediction confidence during the generation of the model. For example, you could view the Learning Time required to generate the model and information on the targets

2. To verify the indicators in the detailed log, click (*Log*).
3. You can then display the screen *Using the Model*.
   1. If the performance of the model satisfies you, go to Step 3 - Analyzing and Understanding the Model Generated.
   2. Otherwise, go to the procedure To Generate a New Model.

To Generate a New Model

You have two options. On the screen *Training the Model*, you can:

- Either click the *Previous* button to return to the modeling parameters defined initially. Then you can modify the parameters one by one.
- Or click the *Cancel* button to return to the main screen of the modeling assistant. Then you must redefine all the modeling parameters.

5.2.6.3 Step 3 - Analyzing and Understanding the Model Generated

The suite of plotting tools within the application allows you to analyze and understand the model generated:

- The performance of the model with respect to a hypothetical perfect model and a random type of model,
- The characteristics of each of the clusters,
- The significance of the various categories of each variable with respect to the target variable (cross statistics).

5.2.6.3.1 Presentation of the User Menu

Once the model has been generated, Click *Next*. The screen *Using the Model* appears.

The screen *Using the Model* presents the various options for using a model, that allow you to:

- Display the information relating to the model just generated or opened (*Display* section), referring to the model curve plots, plotting of clusters, contributions by variables and the profiles of variables of each cluster.
- Apply the model just generated or opened to new data (*Run* section).
- Save the model, or generate the source code (*Save/Export* section).

5.2.6.3.2 Model Overview

The screen *Model Overview* displays the same information as the training summary.

Overview

Model	name of the model, created from the target variable name and the dataset name
Dataset	name of the dataset
Initial Number of Variables	number of variables in the dataset
Number of Selected Variables	number of explanatory variables used to build the model
Number of Records:	number of records in the dataset
Building Date	date and time when the model was built
Learning Time	total learning time
Engine name	name of the feature used to build the model; Kxen.KMeans for a segmentation
Minimum / Maximum Requested Number of Clusters	number of clusters that have been asked for by the user
SQL Expressions	indicates if the SQL expressions for the clusters definitions have been calculated (Enabled) or not (Disabled)

Nominal Target Variables

For each nominal target:

<TargetVariableName>	name of the target variable for which the statistics are displayed
Target Key	wanted target value
<NonTargetCategory> - Frequency	frequency in percentage of the non-target valuein the entire dataset
<TargetCategory> - Frequency	frequency in percentage of the wanted target valuein the entire dataset

Continuous Targets (Number)

For each continuous target:

<TargetVariableName>	name of the target variable for which the statistics are displayed
Min	minimum value for the target
Max	maximum value for the target
Mean	mean of the target

Standard Deviation	mean of the distance between the target values and the Mean

Performance Indicators

For each target variable:

Predictive Power (KI)	For more information on the predictive power, see section Performance indicators (on page 39).
Prediction Confidence (KR)	For more information on the prediction confidence, see section Performance indicators (on page 39)

Cluster Counts

For each target variable:

Initial Number of Clusters	number of clusters that have been asked for by the user
Final Number of Clusters	number of clusters found by the model
Overlap	number of observations described by two different SQL expressions, thus appearing in two clusters. For more information, see Difference Between Standard Cross Statistics and SQL Expressions [page 192]
Unassigned Observations	number of observations that cannot be described by the SQL expressions and that are left outside of clusters. For more information, see Difference Between Standard Cross Statistics and SQL Expressions [page 192]

5.2.6.3.2.1 Model Overview Options

To Copy the Model Overview

Click the  (*Copy*) button.

The application copies the HTML code of the screen. You can paste into a word processing or spreadsheet program, a text editor, ...

To Save the Model Overview

Click the  (*Save*) button situated under the title.

The file is saved in HTML format.

To Print the Model Overview

1. Click the  (*Print*) button situated under the title.
   A dialog box appears, allowing you to select the printer to use.
2. Select the printer to use and set other print properties if need be.
3. Click OK.
   The report is printed.

To Export to PowerPoint

Click the 

(*Export to PowerPoint*) button.

5.2.6.3.3 Model Graphs

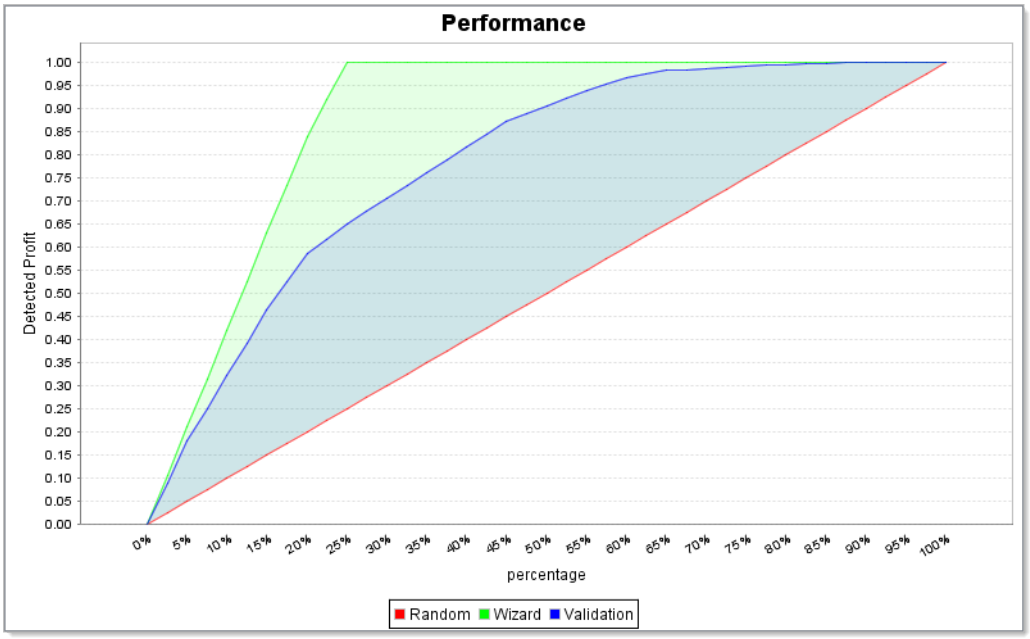
The model graphs allow you to:

- View the realizable profit that pertains to your business issue using the model generated.
- Compare the performance of the model generated with that of a random type model and that of a hypothetical perfect model.

On the plot, for each type of model, the curves represent the realizable profit (Y axis, or ordinate) as a function of the ratio of the observations correctly selected as targets relative to the entire initial dataset (X axis, or abscissa).

Displaying the Model Graphs

1. On the screen *Using the Model*, click the *Model Graphs* option. The model graph appears.



The default parameters display the profit curves corresponding to the *Validation* sub-set (blue line), the hypothetical perfect model (*Wizard*: green line) and a random model (*Random*: red line). The default setting for the type of profit parameter is Detected profit, and the values of the abscissa are provided in the form of a percentage of the entire dataset.

2. When there is more than one target, select the target for which you want to see the curves in the *Models* list.

> **i Note**
>
> One model corresponds to each variable. The name of each model is built from the `kc_` prefix and the model target name.

3. Select the viewing options that interest you. For more information about viewing options,

5.2.6.3.3.1 Plot Options

To Display the Graphs for the Training, Validation, and Testing Sub-sets

Click *Dataset* and select one of the following options that allow you to switch between:

- the graph for the *Validation* sub-set: 
- the graphs for all the sub-sets: 

To Change the View Type

Click *View Type* and select the desired option.

To Copy the Model Graph

Click the (*Copy*) button and select the desired option.

The application copies the parameters of the plot. You can paste it into a spreadsheet program (such as Excel) and use it to generate a graph.

To Save the Model Graph

1. Click the (*Save*) button.
   A dialog box appears, allowing you to select the file properties.
2. Type a name for your file.
3. Select the destination folder.
4. Click OK.
   The plot is saved as a PNG formatted image.

To Print the Model Graph

1. Click the (*Print*) button situated under the title.
   A dialog box appears, allowing you to select the printer to use.
2. Select the printer to use and set other print properties if need be.
3. Click OK.
   The report is printed.

To Export the Model Graph to Microsoft Excel

Click the (*Export to Excel*) button situated under the title. An Excel sheet opens containing the model graph you are currently viewing along with its data.

To Open the Current Graph in a New Window

Click the (*Pin View*) button. The current graph is displayed in a new window.

5.2.6.3.3.2 Understanding the Model Graphs

5.2.6.3.3.2.1 Understanding Graphs for a Model with a Nominal Target

The following figure represents the model graph produced using the default parameters.

On the plot, the curves for each type of model represent the profit that may be realized (Y axis), that is, the percentage of observations that belong to the target variable, in relation to the number of observations selected from the entire initial dataset (X axis). On the X axis, the observations are sorted in terms of decreasing "score", that is, the decreasing probability that they belong to the target category of the target variable.

In the application scenario, the model curves represent the ratio of prospects likely to respond in a positive manner to your marketing campaign, relative to the entire set of prospects contained in your database.

Detected profit is the default setting for type of profit. Using this type of profit:

- The value "0" is assigned to observations that do not belong to the target category of the target variable,
- The value "1/(frequency of the target variable in the dataset)" is assigned to observations that do belong to the target category of the target variable.

The following table describes the three curves represented on the plot created using the default parameters.

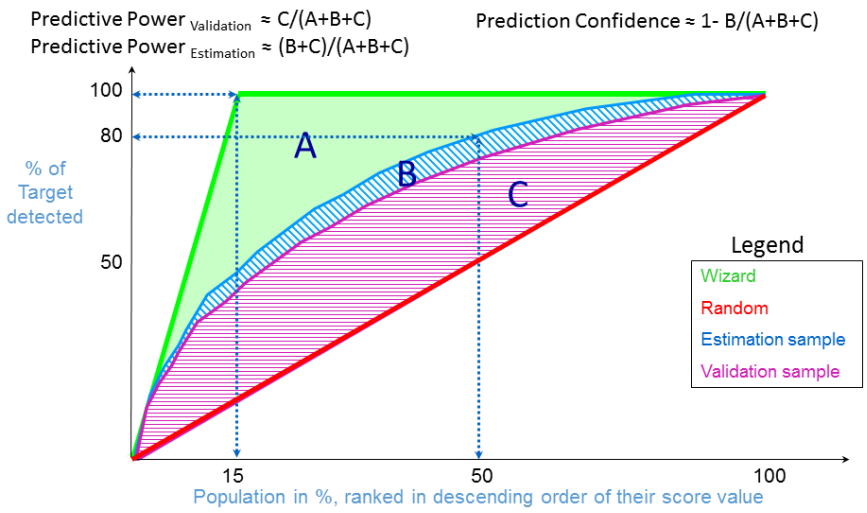
The curve...	Represents...	For instance, by selecting...
Wizard (green curve, at the top)	The profit that may be achieved using the hypothetical perfect model that allows one to know with absolute confidence the value of the target variable for each observation of the dataset	25% of the observations from your entire dataset with the help of a perfect model, 100% of observations belonging to the target category of the target variable are selected. Thus maximum profit is achieved. Note that these 25% correspond to the proportion of prospects who responded in a positive manner to your marketing campaign, during your test phase. For these prospects, the value of the target variable, or profit, is equal to 1.
Validation (blue curve, in the middle)	The profit that may be achieved using the model generated by Modeler - Regression/Classification that allows one to perform the best possible prediction of the value of the target variable for each observation of the dataset	25% of the observations from your initial dataset with the help of the model generated, 66.9% of the observations belonging to the target category of the target variable are selected.
Random (red curve, at the bottom)	The profit that may be achieved using a random model that does not allow one to know even a single value of the target variable for each observation of the dataset	25% of the initial dataset using a random model, 25% belonging to the target category of the target variable are selected.

5.2.6.3.3.3 Predictive Power, Prediction Confidence and Model Graphs

On the model graph plot:

- Of the validation dataset (default plot), the predictive power corresponds to "the area found between the curve of the model generated and that of the random model" divided by "the area found between the curve of the perfect model and that of the random model". As the curve of the generated model approaches the curve of the perfect model, the value of the predictive power approaches 1.
- Of the training and validation datasets (select the corresponding option from the list Dataset, located below the plot), the prediction confidence corresponds to one minus "the area found between the curve of the training dataset and that of the validation dataset "divided by "the area found between the curve of the *perfect model* and that of the random model".

The following graph displays the predictive power and the prediction confidence:



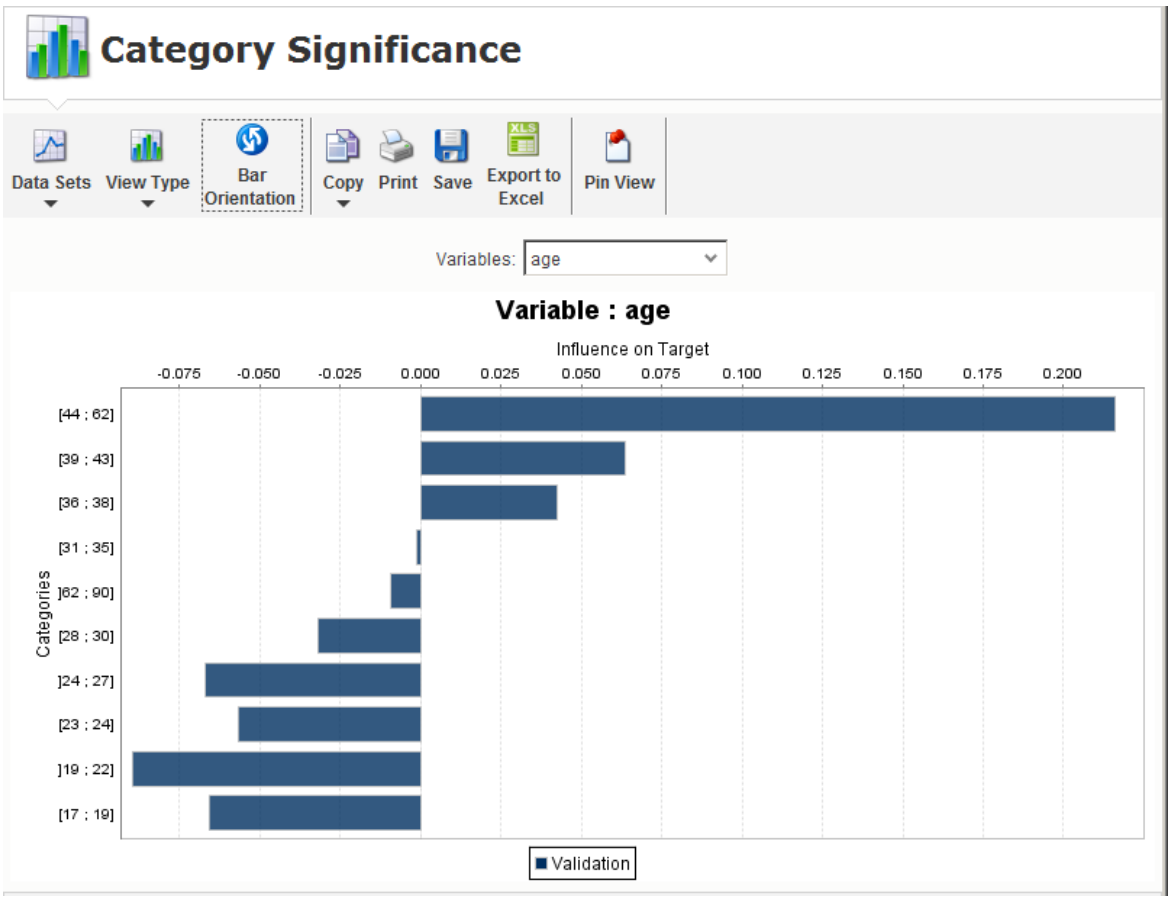
5.2.6.3.4 Category Significance

Definition

The Significance of Categories plot illustrates the relative significance of the different categories of a given variable with respect to the target variable.

Displaying the Significance of Categories Plot

1. On the screen *Using the Model*, click *Category Significance*. The plot *Category Significance* appears.



2. In the *Variables* list located above the plot, select the variable for which you want to display the categories. If your dataset contains date or datetime variables, automatically generated variables can appear in the Variables list. For more information, refer to section on date and date-time variables in the topic on variable storage formats.

> **i Note**
>
> - You can display the relative significance of the categories of a variable directly from the plot Contributions by Variables. On the plot Contributions by Variables, double-click the bar of the variable which interests you.
> - In case no user structure has been defined for a continuous variable, the plot category significance displays the categories created automatically using the band count parameter. The number of categories displayed corresponds to the value of the band count parameter. For more information about configuring this parameter, please refer to the section Band Count for Continuous Variables.

Plot Options

To Switch Between "Validation Dataset" and "All Datasets" Plots

1. Click *Datasets* and select the  (*All Datasets*) button to display all datasets.

The plot displaying all datasets appears.

2. Click *Datasets* and select the  (*Validation Only*) button to go back to the *Validation Dataset* plot.

To Switch between Curve and Bar Charts

1. Click *View Type* and select the  button to display the curve chart. The curve plot appears.
2. Click *View Type* and select the  button to go back to the bar chart.

> **i Note**
>
> You can combine the different types of plot. For example, you can display *All Datasets* in a curve chart or the *Validation Dataset* in a bar chart.

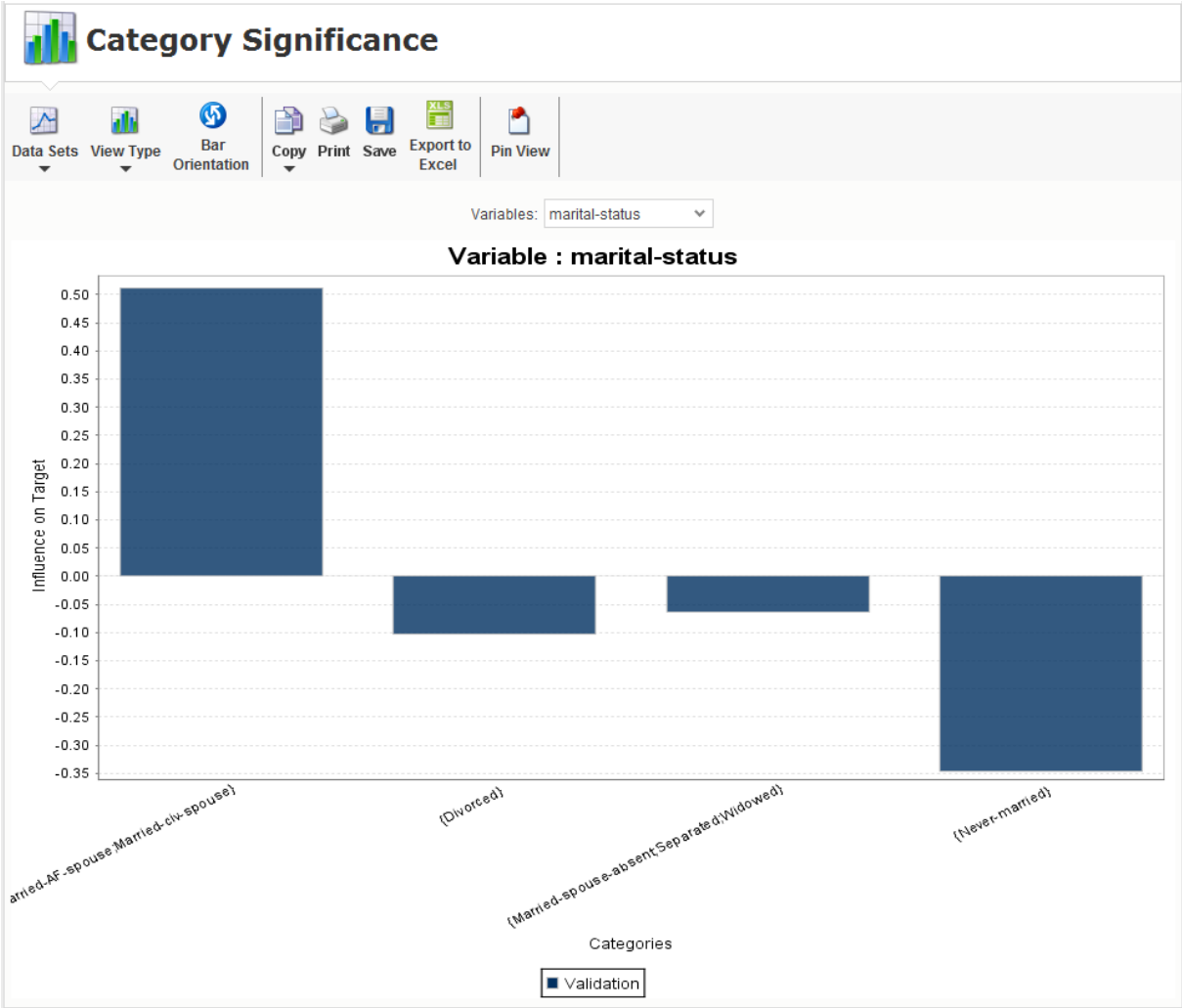
Understanding the Plots of Variables

For this scenario, select the variable `marital-status`, which is the explanatory variable that contributes the most to the target variable `Class`. This plot presents the effect of the categories of the `marital-status` variable on the target variable. For an explanation of the plot, see the topic on variable categories and profit.

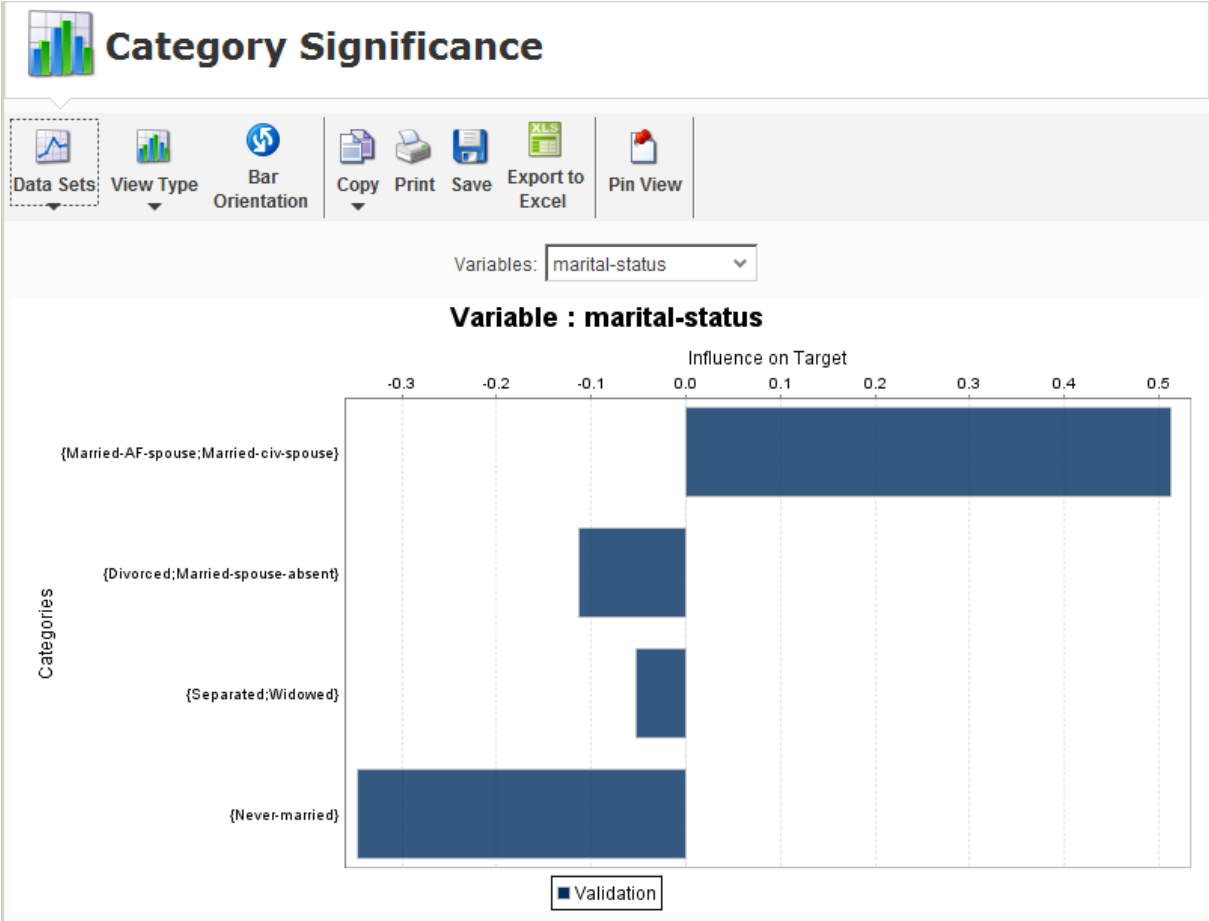
5.2.6.3.4.1 Variable Categories and Profit

The *Category Significance* plot shows the relative significance of the different categories of a given variable with respect to the target variable.

Each bar on the plot shows the amount of influence that category has on the target category (or hoped-for value) of the target variable. The bars are ordered by their amount of influence. In this example, the *Bar Orientation* option shows the bars vertically, so the bar the furthest to the left represents the category with the greatest positive effect, and the bar the furthest to the right, the category with the least effect.



If the *Bar Orientation* option is set to show the bars horizontally, the highest bar on the plot represents the category with the greatest positive effect. In other words, the higher a category appears on the plot, the more representative that category is of the target category of the target variable.



The length and direction of a bar correspond to the profit contributed by that category, in other words, whether the category has more or fewer observations belonging to the target category. A positive bar (*Influence on Target* greater than 0.0) indicates that the category contains more observations belonging to the target category than the mean (calculated on the entire dataset). A negative bar (*Influence on Target* less than 0.0) indicates that the category contains fewer observations belonging to the target category than the mean.

> **i Note**
>
> You can display the profit curve for the selected variable by clicking the  (*Display Profit Curve*) button located in the tool bar under the title.

The importance of a category depends on both its difference to the target category mean and the number of represented cases. High importance can result from any of the following:

- A high discrepancy between the category and the mean of the target category of the target variable
- A minor discrepancy combined with a large number of records in the category
- A combination of both

Use the *Variables* pull down menu to select and graph any of the variables in the model. Use the tool bar located under the title to copy the coordinates to the clipboard, print the plot, or save it in PNG format. The values are normalized and their sum always equals to "0". Depending on the chosen profit strategy, or on the continuous target variables value type, you can obtain all positive importance or negative and positive importance.

5.2.6.3.4.1.1 Axes

The X-axis shows the influence of the variable categories on the target. The significance of the different numbers on the X-axis are detailed in the following table:

Number on the X-axis	Indicates that the category has
positive number	a positive influence on the target
0	no influence on the target (the behavior is the same as the average behavior of the whole population)
negative number	a negative influence on the target

The Y-axis displays the variable categories. Categories sharing the same effect on the target variable are grouped. They appear as follow [Category_a;Category_b;Category_c]. Categories not containing sufficient numbers to provide robust information are grouped in the KxOther category. When a variable is associated with too many missing values, the missing values are grouped in the KxMissing category. Both categories are created automatically by the application.

5.2.6.3.4.1.2 Formulas

`Category Importance = NP*BF/{NC}` where `NP` is the *Normal Profit*, `BF` is the *Bin Frequency* and `NC` is the *Normalization Constant*.

The calculation of the normalization constant differs by target data type. The calculations for binary and continuous targets are detailed below.

For binary targets, it is calculated as follows:

```
(Target Frequency) * (1 - [Target Frequency]).
```

It can be approximated for non-pathological continuous targets (that is continuous targets without distribution peak (Dirac)) from:

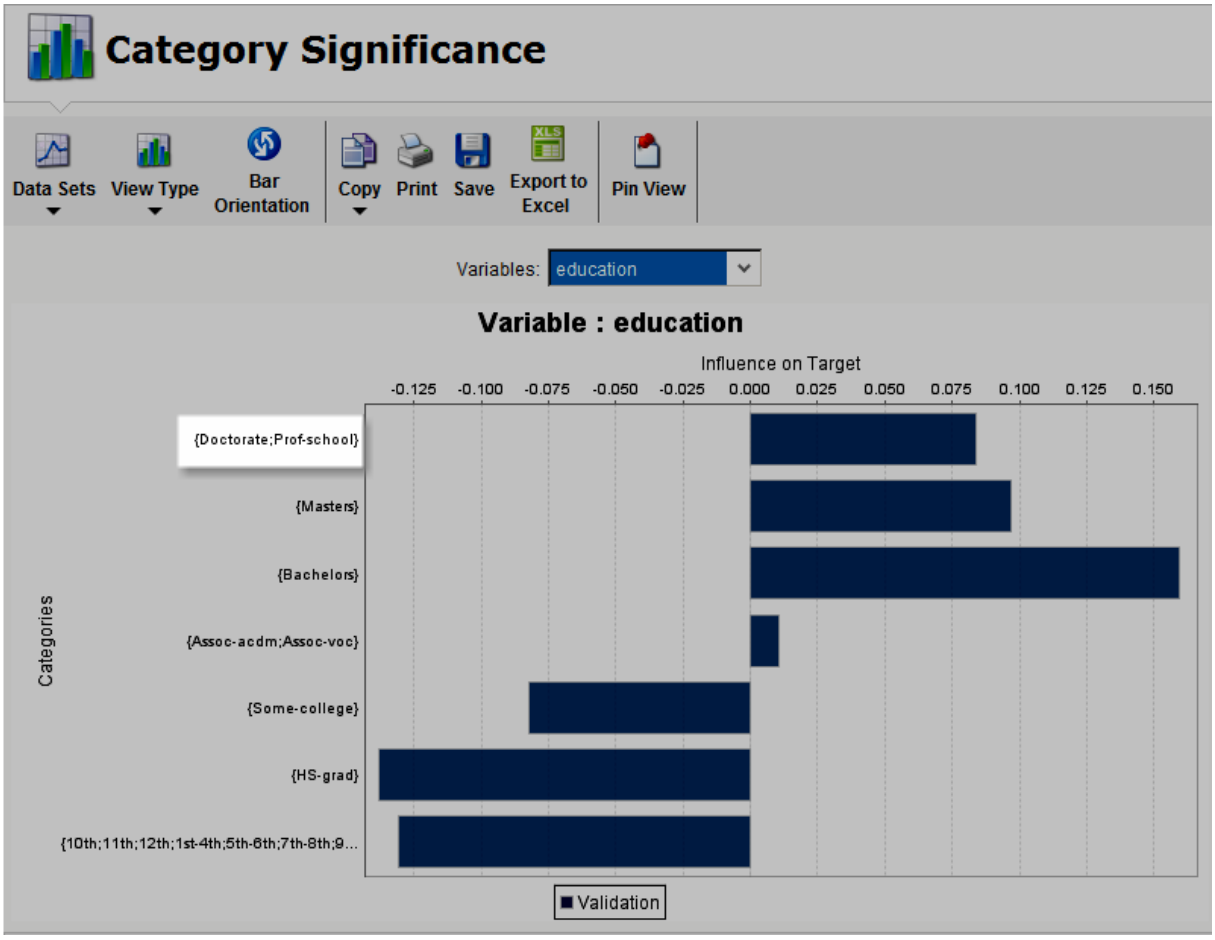
```
(Proportion above Median) * (1-Proportion above Median) = 0.5 * (1 - 0.5) = 0.25
```

5.2.6.3.4.2 Grouping Categories

On the plot of details of a variable, categories may appear grouped.

When the option *Optimal Grouping* is enabled, the application groups those categories sharing the same effect on the target variable. In the plot below (example in English), for the variable `education`, the categories `Doctorate` and `Prof-School` are grouped. If the explanatory variable is continuous, the application identifies the points where behavioral changes occur with respect to the target variable and automatically crops the variable into intervals exhibiting homogeneous behavior with respect to the target.

For more information, please see the related topic.



When categories do not contain sufficient numbers to provide robust information, they are grouped in the KxOther category, that is created automatically.

When a variable is associated with too many missing values, the missing values are grouped in the KxMissing category, that is also created automatically.

To understand the value of the categories KxOther and KxMissing, consider the following example. The database of corporate customers of a business contains the variable "web address". This variable contains the Web site address of the corporate customers contained in the database. Some companies have a Web site; others do not. In addition, each Web site address is unique. In this case, the application automatically transforms the "web address" variable into a binary variable with two possible values: KxOther (the firm has a Web site) and KxMissing (the firm does not have a Web site).

5.2.6.3.5 Cluster Summary

The following types of charts can be displayed:

- Bubble Charts: Bubble charts display the clusters by representing the relationship between three variables.
- Bar Charts (Cluster Plots): Bar charts display the three cluster plots that allow you to examine:
  - The proportion of observations of the dataset contained in each cluster (Frequencies plot),
  - The proportion of each cluster relative to the target variable (Target Means and Relative Target Means plots).

> **i Note**
>
> An additional cluster is always displayed in the cluster summary. This cluster contains all the data points that are not included in the clusters computed by the product.

5.2.6.3.5.1 Displaying Bubble Charts

1. On the screen *Using the Model*, click *Cluster Summary*.

   The panel *Cluster Summary* appears.
2. Use the options to define the variables you want to display in the bubble chart.

   The table below lists the available options:

The option...	allows you to...	Note that...
	select the variable to be used in the X-Axis.	only continuous and nominal numerical variables can be used.
	select the variable to be used in the Y-Axis.	only continuous and nominal numerical variables can be used.
	select the variable to be used for the bubble size.	only the variable Frequency and the target variable can be used.
	display cluster names.	cluster names can be customized in Cluster Profiles .

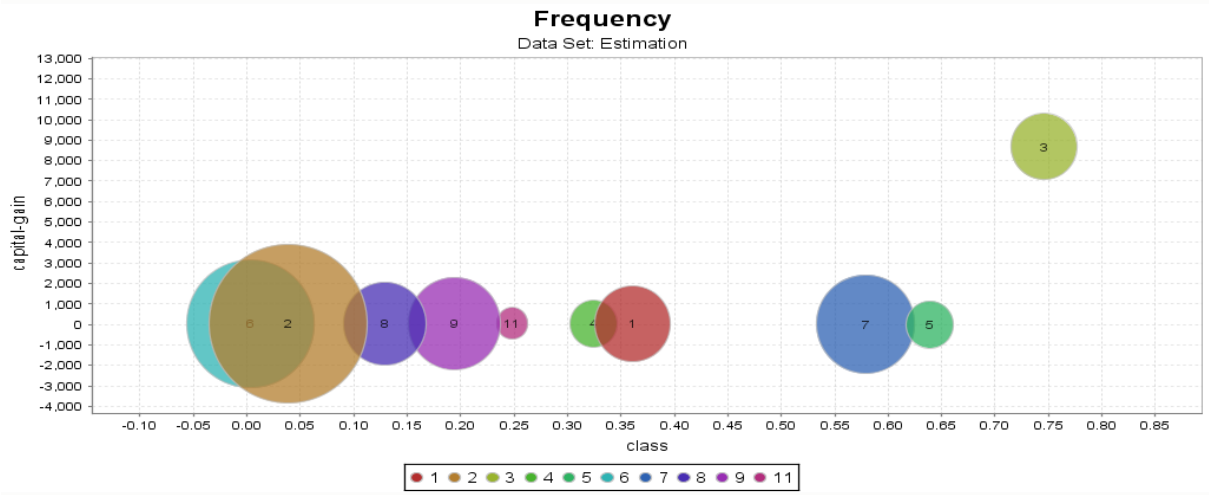
5.2.6.3.5.2 Understanding Bubble Charts

Bubble charts allow you to display the clusters representing the relationship between three variables. Thus, a bubble chart can provide 3 pieces of information on each cluster.

In addition, bubble charts provide a graphical representation of the segmentation, enabling you to easily visualize the clusters. For instance, it can be useful during a presentation.

The X-axis, the Y-axis and the bubble size represent one variable each. You can define the variables to use in the chart. Thus, you can create a bubble chart that separates distinctly the clusters from each other, enabling you to identify the clusters of interest for your marketing campaign.

The figure below represents the relationship between the variables class, capital-gain and frequency.



For instance, results show that the customers listed in cluster 3 are earning 8,682.24 dollars per year in average (capital-gain: 8,682.24) and represents 7% (Frequency: 0.07) of the population listed in the dataset. In addition, among these customers, 75% (class: 0.75) responded in a positive manner to the test phase of your marketing campaign.

In comparison, cluster 2 represents the biggest population listed in the dataset, namely 24% of the population (Frequency: 0.24), which is around three times bigger than the population listed in cluster 3. However, the customers listed in this cluster are earning less than the customers listed in cluster 3. They earn on average 38.40 dollars per year (capital-gain: 38.40). In other words, less than 0.5% of what customers listed in cluster 3 earn. Moreover, among the customers listed in cluster 2, only 4% (class: 0.04) responded in a positive manner to the test phase of your marketing campaign.

Consequently, compared to cluster 2, cluster 3 is more interesting because it shows better results to the test phase of your marketing campaign.

5.2.6.3.5.3 Displaying Cluster Plots

1. On the screen *Using the Model*, click *Cluster Summary*.

   The panel *Cluster Summary* appears.

2. Click the button (*View Type*) and select *Bar Chart*.
3. In the *Chart Type* list, select the type of chart that you want to display.

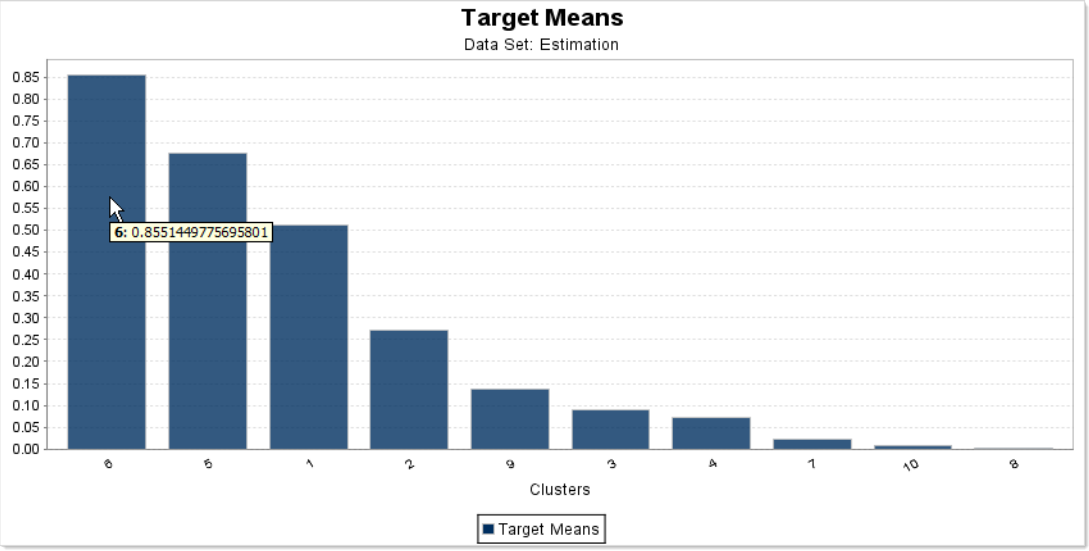
> **i Note**
>
> Select the option *Descending sort* to sort the plot bars in descending order. For instance, on the plot *Relative Target Means*, the descending sort allows quick examination of the most interesting clusters, that is, those which differ most from the mean behavior of the dataset taken as a whole.

5.2.6.3.5.4 Understanding Cluster Plots

The *Target Means* Plot

The Target Means plot presents the proportion of observations belonging to the target category of the target variable, present in each cluster.

The figure below presents the Target Means plot obtained in during this scenario. The bars have been sorted in descending order.



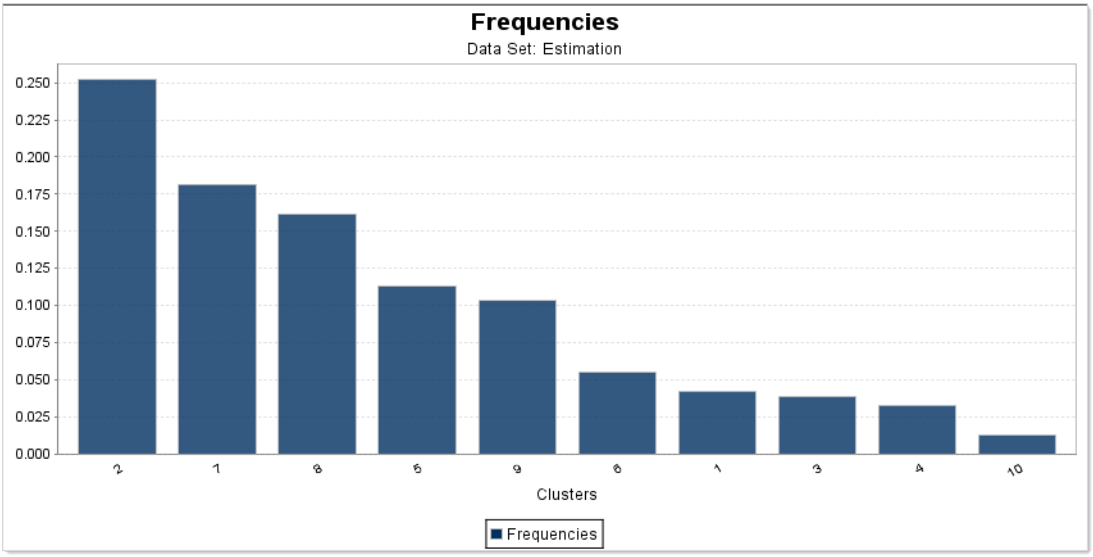
Among the five clusters, Cluster 6 is the one that has the greatest proportion of observations belonging to the target category of the target variable. In fact, 85.5% of the customers contained in cluster 6 belong to target category 1, target variable Class. In other words, 85.5% of the customers contained in cluster 6 responded in a positive manner to the test phase of your marketing campaign.

Cluster 8 is the cluster with the lowest density of observations belonging to the target category. Less than 1% of the customers contained in this cluster responded positively to the test phase of your marketing campaign.

The *Frequencies* Plot

The *Frequencies* plot presents the number of observations contained in each cluster relative to the total number of observations contained in the dataset. The figure below presents the Frequencies plot obtained during this scenario. The bars have been sorted in descending order.The Frequencies plot presents the number of observations contained in each cluster relative to the total number of observations contained in the dataset.

The figure below presents the Frequencies plot obtained during this scenario. The bars have been sorted in descending order.

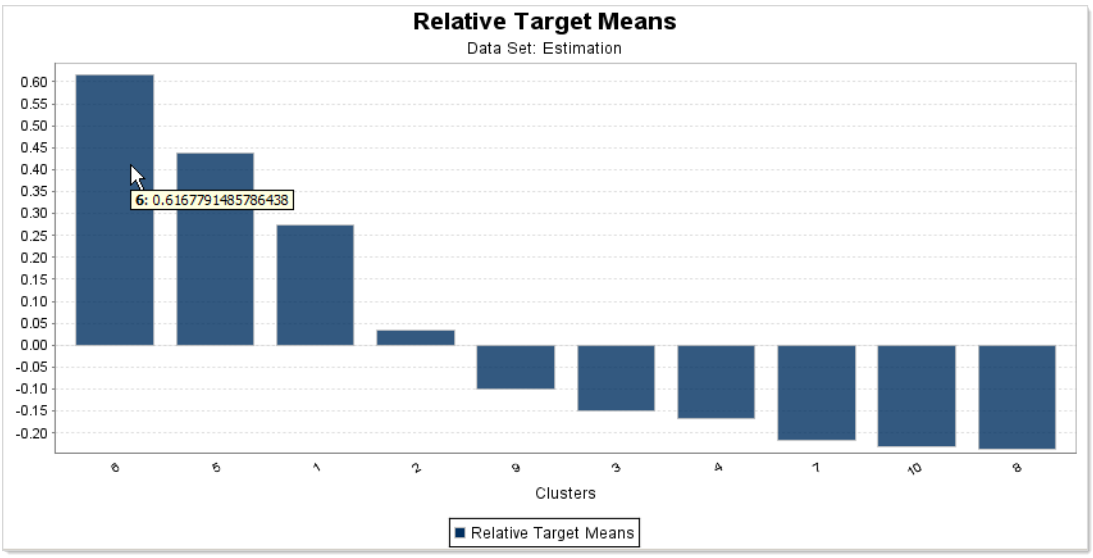


Among the five clusters, Cluster 2 is the one which contains the greatest number of observations, or 25,2% of the total number of customers contained in the entire dataset.

The *Relative Target Means* Plot

Similar to the Target Means plot, the Relative Target Means plot presents the proportion of observations, for each cluster, belonging to the target category of the target variable. The only difference between the two plots is the scale used on the Y axis. On the Relative Target Means plot, the proportion of observations belonging to the target category of the target variable relative to the entire dataset is re-expressed. In other words, the 0 value of the Y axis corresponds to the true percentage of observations belonging to the target category of the target variable in relation to the entire dataset.

The figure below presents the Relative Target Means plot obtained during this scenario. The bars have been sorted in descending order.



Among the ten clusters, Cluster 6 is the cluster that has the highest proportion of observations belonging to the target category of the target variable. Compared to the entire dataset, Cluster 6 contains 61.6% more customers belonging to the target category 1, of the target variable Class.

Cluster 2 contains less than 3,3% of customers belonging to the target category. In other words, Cluster 2 has almost the same customer density belonging to the target category as the dataset taken as a whole.

Cluster 8 is the cluster with the lowest density of observations belonging to the target category. Compared to the entire dataset, Cluster 8 contains 23.5% fewer of customers belonging to the target category. This cluster therefore has a density of customers belonging to the target category lower than that of the dataset.

5.2.6.3.6 Cluster Profiles

5.2.6.3.6.1 Cross-Statistics and Variable Profiles

The cluster profiles allow you to view for each cluster:

- the profile of each explanatory variable, with respect to their profile over the entire dataset,
- the SQL expression of the cluster when they have been calculated.

Variable Profile

The *Variable Profile* indicates the distribution of observations (belonging to a cluster of global dataset) within the categories of each variable. In other words, the profile indicates the proportion of observations contained in each of the categories of that variable.

Example of a Variable Profile

The variable "gender" of a dataset can be distributed as follows:

- 53% of observations belong to the category "male",
- 47% of observations belong to the category "female".

This distribution corresponds to the profile of the variable "gender" over this dataset.

Given a cluster A, taken from this dataset, the same variable "gender" may be distributed as follows:

- 80% of observations belong to the category "male",
- 20% of observations belong to the category "female".

This distribution corresponds to the profile of the variable "gender" over cluster A.

The cluster profiles allow you to view and compare the profiles of the variable "gender" over the dataset and the clusters taken from this dataset.

5.2.6.3.6.2 Displaying Cluster Profiles

1. On the screen *Using the Model*, click *Cluster Profiles*.

   The screen *Cluster Profiles* appears.
2. In the table, select the cluster for which you want to view the profile.

   > **i Note**
   >
   > If only the variable ranges for a specific cluster are displayed, click the black horizontal bar, a moving cursor is displayed. Drag the cursor down to display the list of clusters.
3. Below the table, from the drop-down list associated with the *Variable* field, select the variable for which you want to see the profile.

   The cross statistics will appear in the form of a plot in the lower part of the screen.

5.2.6.3.6.3 Understanding Cluster Profiles

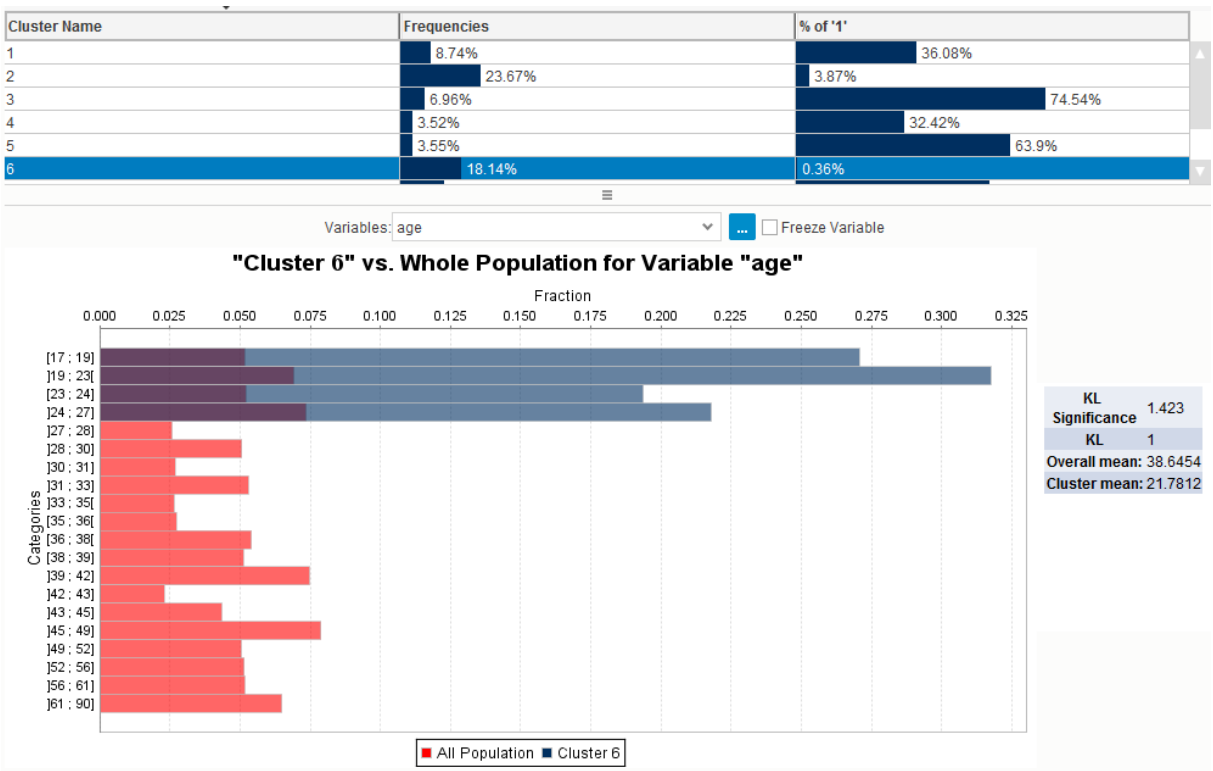
The screen *Cluster Profiles* can be broken down into three parts:

- In the upper part, a table summarizes the information for each cluster. This table allows you to select the cluster for which you want to view cross statistics.

Column ...	Indicates...	For instance...
Cluster Name	The name of the cluster.	1
Frequencies	The number of observations contained in the cluster relative to the total number of observations contained in the dataset.	The customers contained in cluster 1 represent 4.22% of the total number of customers contained in your entire training dataset.
% of '1'	The proportion of observations contained in the cluster belonging to the target category of the target variable.	51.17% of the customers contained in cluster 1 belong to the target category of the target variable `Class`. In other words, 51.17% of the customers contained in this cluster responded in a positive manner to the test phase of your marketing campaign.

- In the middle part, a drop-down list allows you to select the variable for which you want to see the cross statistics. Variables are presented in descending order of the significance of their contribution relative to the target category of the target variable. When a cluster is selected, the variables visible in the drop-down list are sorted according to the difference between their cluster profile and their population profile (the Kullback-Leibler divergence is used to measure this difference). The variable that appears first on the list is the variable exhibiting the greatest difference between its two profiles. This sorted list of variables provides the set of discriminatory variables required to describe a cluster.
- In the lower part, a plot presents either the cross statistics corresponding to the cluster and the variables selected, or; when it has been calculated, the SQL expression defining the cluster.

The figure below presents the screen Cluster Profiles, which appears as the default plot for this scenario. The plot presents the cross statistics for cluster 6.

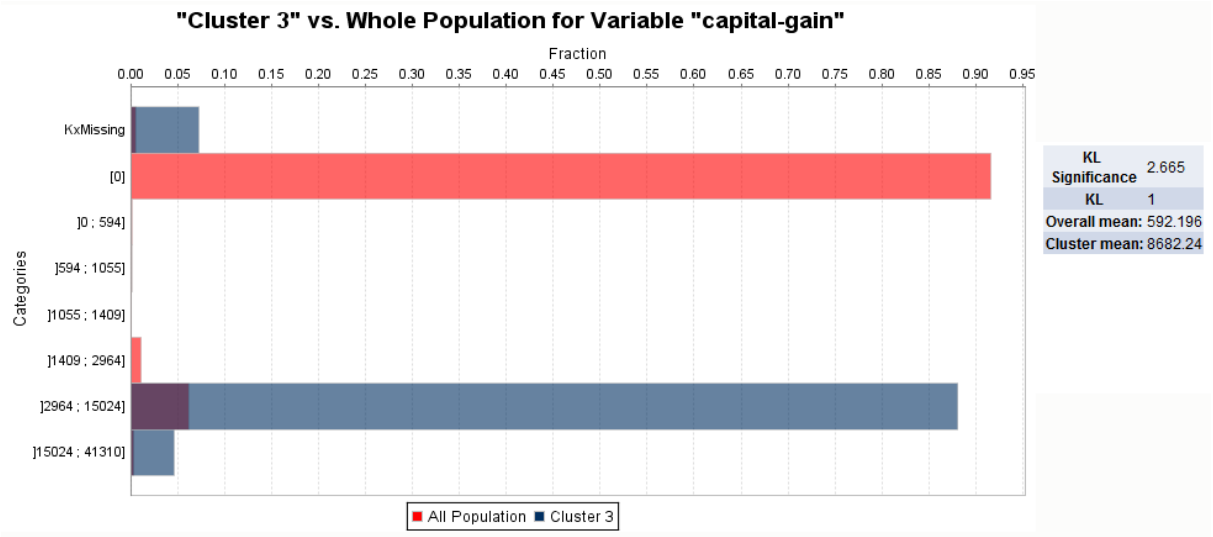


Cross Statistics Plots

Cross statistics plots contain two curves:

- The blue area corresponds to the profile of the variable selected over the cluster selected,
- The red area corresponds to the profile of the variable selected over the entire dataset.

The figure below presents the Cross Statistics obtained in this scenario for cluster 3.



In the figure above, the table allows you to identify cluster 3 as the cluster containing the highest density of observations belonging to the target category of the target variable: 75% of customers contained in this cluster belong to target category 1 of the target variable `Class`.

The cross statistics plot allows you to view and compare the profiles of the variable capital-gain over the entire dataset and over cluster 3. These profiles are repeated in the table below.

Categories of the variable "capital-gain"	Profile over the dataset	Profile over cluster 3
KxMissing	1%	7%
[0]	92%	0%
]1409 ; 2964]	1%	0%
] 2964 ; 15024]	6%	88%
] 15024; 41310]	0%	5%

The data distribution of the category] 2964 ; 15024] makes it clear that the majority of customers contained in cluster 3 realize significant annual capital gains relative to the entire set of customers contained in the dataset. In addition, the data distribution over the category [0] indicates that the majority of the customers contained in the dataset, or 92%, do not realize any annual capital gains, while none of the customers contained in cluster 3 fail to realize some annual capital gain.

Checking the *Fix Variable* box would allow you to compare the profiles of the variable capital-gain for all the segments.

5.2.6.3.6.4 Displaying SQL Expressions

The Cross Statistics screen also allows you to visualize the SQL Expression used to define each cluster.

> **i Note**
>
> SQL expressions are only available if you have selected the *Calculate SQL Expressions* option in the modeling parameters advanced screen before generating your model.

To Display SQL Expression for a Cluster

1. Select the cluster in the summary table.

   The plot for the selected cluster is displayed.

2. Click *View Type* and select the  *SQL* button.

   The SQL expression replaces the cross statistics plot in the lower part of the screen.

3. Click + or - to fold or unfold the tree representing the SQL expression.

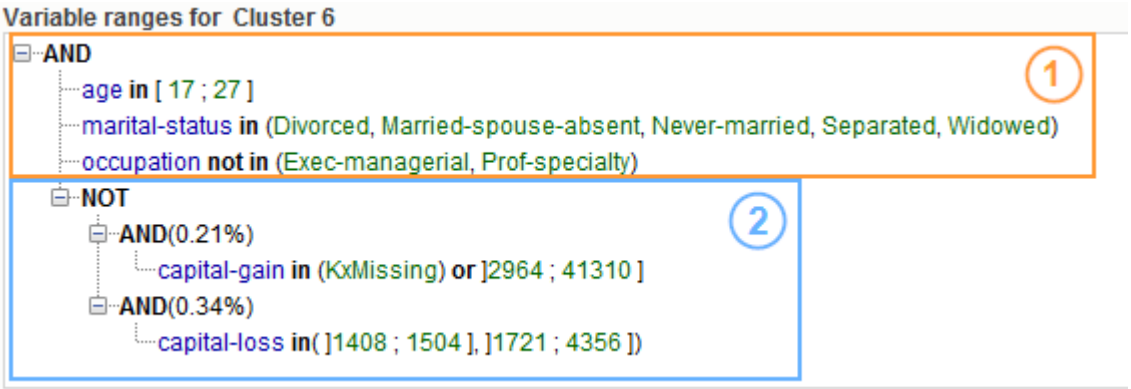
4. Click *View Type* and select  *Profiles* to go back to the *Cross Statistics* plot.

5.2.6.3.6.5 Understanding SQL Expressions

The SQL Expressions screen can be broken down into two parts:

- In the upper part, a table presents each cluster in a summarized fashion. It allows you to select the cluster for which you want to display a SQL expression.
- In the lower part, a tree presents the SQL expression corresponding to the selected cluster.

The following schema presents the SQL expression for Cluster 6.



The SQL expression can be broken down as follows:

- The first part (1) defines a cluster of observations where the variables equal the values displayed.
- The second part (2) defines clusters of observations that are excluded for the cluster found in part 1. The percentages displayed indicate the proportion of each cluster excluded with respect to the cluster found in part 1.
  In our example, the first excluded cluster corresponds to observations where the value of `capital-gain` ranges between 2964 excluded and 41310 (]2964 ; 41310]), or is missing (`KxMissing`). It represents 0.21% of the observations found in part 1.

> **i Note**
>
> The clusters are created by applying the SQL expressions in a specific order defined by the engine. If you apply the SQL rules randomly, you may not obtain exactly the same result.

5.2.6.3.6.6 Difference Between Standard Cross Statistics and SQL Expressions

When you ask for SQL expressions, the final segmentation is different from the one without. The goal of SQL is to have easy-to-understand and easy-to-apply segments. SQL expressions are built to describe as much as possible the basic segments (that is the ones you get when you do not ask for SQL). The SQL can be used both to have a better definition/understanding of the clusters and to deploy them on the full database or on new data (which is not usually trivial with other techniques).

The best way to understand the difference between centroid-based clusters and rule-based clusters is to use graphs.

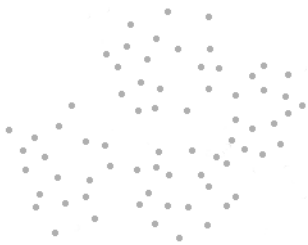
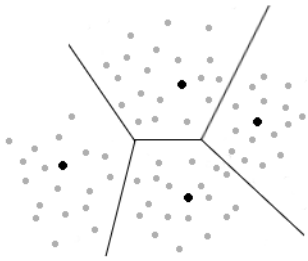
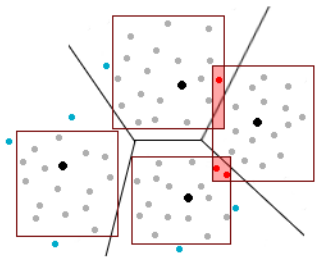
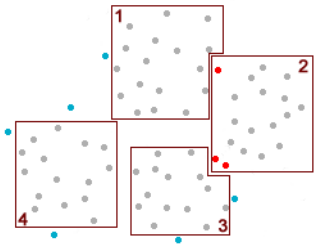
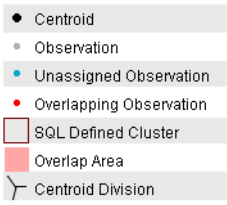
Diagram	Explanation
	This diagram represents a set of observations from a dataset.
	To create clusters, Modeler - Segmentation/Clustering engine uses the centroid approach. Centroids are the results of a clustering algorithm, meaning they are the barycenter of the points closest to them. When applying Modeler - Segmentation/Clustering on this dataset, the observations are grouped depending on their distance with each centroid. This graph represents the previous dataset observations grouped into four clusters. This diagram is known as a Voronoï diagram.
	To create the SQL expressions that define the clusters, Modeler - Segmentation/Clustering engine uses what is called Minimum Description Length (MDL). It means that after creating the initial clusters from the centroid approach then they are reshaped, cut to fit into the smallest possible expression thus trying to find the best compromise between length of the expression and the lost of information. This graph represents the SQL expressions of the clusters (in red) compared with the centroids. You can see on this graph that: • some observations that were in a cluster when using the centroid approach end up in another when using the SQL expressions. • some observations can not be described by the SQL expressions and are left outside the cluster. They are called the unassigned observations. • some observations are described by two different SQL expressions, thus appearing in two clusters. This is called the overlap.

Diagram	Explanation
	This graph presents the final result obtained with SQL expressions. An observation cannot appear in two different clusters, so when there is overlap between clusters, the observation concerned by the overlap is kept in the first cluster created. The second cluster that was also containing the observation is redefined to exclude it. In this schema, the numbers correspond to the order of creation of the clusters. You can see that the observations that were in two clusters are kept in only one. The choice of the cluster in which the overlapping observations are kept depends on the order in which the SQL rules are applied. In this case, the rule defining cluster 2 has been applied before the rules defining the clusters 1 and 3.
	Schema Key

How to decide which segmentation is better

As a side effect of the supervision, Segmentation/Clustering provides you with a KI and KR. It can be used to compare the two segmentations (especially because the number of segments is the same). If KI does not change significantly, then the one with SQL may be preferred because it is easier to understand. If there is a fall of KI, you may want to stick with the basic segmentation.

KI may not be the thing you want to optimize for segmentation. The target profile of each segment is available in the GUI. Out of the four clusters, maybe one or two are of real interest. In that case you have to focus on these interesting segments and see how they evolve with SQL generation.

5.2.6.3.7 Statistical Reports

Statistical Reports provide you with a set of tables that allow you a more detailed debriefing of your model. These reports are grouped in different levels of debriefing:

- The Descriptive Statistics, which provides the statistics on the variables, their categories and the datasets, as well as the variables cross-statistics with the target.

> **i Note**
>
> If your dataset contains date or datetime variables, automatically generated variables will appear in the statistical reports.

- The Model Performance, in which you will find the model performance indicators, the variables contributions and the score detailed statistics.
- The Control for Deviations, which allows you to check the deviations for each variable and each variable category between the validation and test datasets.
- The Expert Debriefing, in which you will find more specialized performance indicators, as well as the variables encoding, the excluded variables during model generation and the reason for exclusion, and so on.

5.2.6.3.7.1 Statistical Report Options

A toolbar is provided allowing you to modify how the current report is displayed, to copy the report, to print it, to save it or to export it to Excel.

Display Options

Menu	Option
View	Display the current report view in the graphical table that can be sorted by column.
	Display the current report view as an HTML table.
	Some reports can be displayed as a bar chart. This bar chart can be sorted by ascending or descending values, or by ascending or descending alphabetical order. You can also select which data should be displayed.
	Display certain reports as a pie chart.
	Display certain reports as a line chart.
Sort	When the current report is displayed as a bar chart, this option allows you to change the orientation of the bars (from horizontal to vertical and vice versa).
	Display the current report with no sorting.
	Sort the current report by ascending values.

Menu	Option
	Sort the current report by descending values.
	Sort the current report by ascending names.
	Sort the current report by descending names.
Series	Select which columns to display for current report.

Usage Options

Menu	Option
	Copy the data from the current view of the displayed report. The data can then be pasted in a text editor, a spreadsheet, a word processing software.
	Print the current view of the selected report depending on the chosen display mode (HTML table, graph, ...).
	Save under different formats (text, html, pdf, rtf) the data from the current view of the selected report.
	Save under different formats (text, html, pdf, rtf) the data from all the views of the selected report.
	Export to Excel.
	Save all reports.
	Save the customized style sheet.

5.2.6.4 Step 4 - Using the Model

Once generated, a clustering model may be saved for later use. A clustering model may be applied to additional datasets. The model thus allows you to assign observations to clusters. This part presents the option of applying the model to a new dataset for the Segmentation/Clustering feature. The other options for deployment of the clustering models are similar to those proposed for models generated using the Regression/Classification feature. For more information about these options, see the topics on saving and opening a model.

Applying the Model to a New Dataset

The currently open model may be applied to additional datasets. The model allows you to determine to which cluster the observations described in these datasets belong.

Constraints of Model Use

In order to apply a model to a dataset, the format of the application dataset must be identical to that of the training dataset used to generate the model. The same target variable, in particular, must be included in both datasets, even if values for the target variable are not contained in the application dataset.

5.2.6.4.1 Using the Option Direct Apply in the Database

This optimized scoring mode can be used if all the following conditions are met:

- the apply-in dataset (table, view, select statement, data manipulation) and the results dataset are tables coming from the same database,
- the model has been computed while at least one physical key variable was defined in the application,
- no error has occurred,
- the in-database apply mode is not deactivated,
- granted access to read and write (create table).

To use the in-database apply mode, check the option *Use the Direct Apply in the Database* and automatically the option *Add Score Deviation* is selected as well.

5.2.6.4.2 Advanced Apply Settings

5.2.6.4.2.1 General Outputs

Copy the Weight Variable

This option allows you to add to the output file the weight variable if it had been set during the variable selection of the model.

Copy Dataset Id

This option allows you to add to the output file the name of the sub-dataset the record comes from (*Training*, *Validation* or *Testing*).

> **i Note**
>
> This option cannot be used with the in-database apply feature.

Copy the Variables

This option allows you to add to the output file one or more variables from the dataset.

- To Add All the Variables
  - Check the *All* option.
- To Select only Specific Variables
  1. Check the *Individual* option.
  2. Click the >> button to display the variable selection table.
  3. In the *Available* list select the variables you want to add (use the `Ctrl` key to select more than one variable).
  4. Click the > button to add the selected variables to the *Selected* list.

User Defined Constant Outputs

This option allows you to add to the output file constants such as the apply date, the dataset name, or any other information useful for using the output file.

A user defined constant is made of the following information:

Parameter	Description	Value / Warnings
Visibility	indicates if the constant will appear in the output or not	*checked*: the constant appears in the output *unchecked*: the constant does not appear in the output
Name	the name of the user defined constant	• The name cannot be the same as the name of an existing variable of the reference dataset. • If the name is the same as an already existing user defined constant, the new constant will *replace* the previous one

Parameter	Description	Value / Warnings
Storage	the constant type (number, string, integer, date, ...)	*number* *string* *integer* *date* *datetime*
Value	the value of the constant	date format: YYYY-MM-DD datetime format: YYYY-MM-DD HH:MM:SS
Key	indicates if the constant is a key variable or identifier for the record. You can declare multiple keys. They will be built according to the indicated order (1-2-3-...).	*0*: the variable is not an identifier *1*: primary identifier *2*: secondary identifier ...

- To Define a Constant
  1. Click the *Add* button. A pop-up window opens allowing you to set the constant parameters.
  2. In the field *Output Name*, enter the constant name.
  3. In the list *Output Storage*, select the constant type.
  4. In the field *Output Value,* enter the constant value.
  5. Click the *OK* button to create the constant. The new constant appears in the list. You can choose whether to generate the defined constants or not by checking the *Visibility* box.

5.2.6.4.2.2 Outputs by Cluster Identifier, Cluster Rank, and Miscellaneous Outputs

5.2.6.4.2.2.1 For Clustering Models

Outputs by Cluster Ranks

Distance to Clusters

Use this option to add to the output file the distance of each observation from the clusters. The distances are generated in the columns named `kc_dist_cluster_<TargetVariable>_<ClusterId>`. For example if the target variable is *Age*, the distance from cluster *1* will appear in the column

`kc_dist_cluster_Age_1`.

- To Add the Distances from All Clusters
  - Select *All*.

- To Select Distances from Specific Clusters
  1. Select *Individual*.
  2. Click the >> button to display the cluster selection table.
  3. Select the clusters for which you want to add the distance.

> **i Note**
>
> When SQL mode is activated, the notion of nearest cluster does not exist. If a case belongs to a cluster, distance is set to 0. If a case does not belong to a cluster, distance is set to 1.

Probability for Clusters

Use this option to add to the output file the probability of each observation to belong to the various clusters. The probabilities are generated in the columns `kc_proba_cluster_<TargetVariable>_<ClusterId>`. For example, if the target variable is `Age`, the probability that the observation belongs to cluster 1 will be displayed in the column `kc_proba_cluster_Age_1`.

- To Add the Probabilities for All Clusters
  - Select *All*.
- To Select the Probabilities for Specific Clusters
  1. Select *Individual*.
  2. Click the >> button to display the cluster selection table.
  3. Select the clusters for which you want to add the probabilities.

> **i Note**
>
> When SQL mode is activated, the notion of nearest cluster does not exist. If a case belongs to a cluster, probability is set to 1. If a case does not belong to a cluster, probability is set to 0.

Outputs by Cluster Identifier

Top Ranking Cluster Indices

Use this option to add to the output file the number of the clusters whose centroids are the closest to the current observation. The closest cluster is the one the observation belongs to, its number is displayed in the column `kc_<Target variable>`. The next closest cluster is displayed in the column `kc_<Target Variable>_2`, and so on until the furthest cluster. You can choose to add all the clusters or only the closest.

- To Add All the Clusters
  - Select *All*.
- To Add Only the Closest Clusters
  1. Select *Top*.
  2. In the text field enter the number of clusters you want to add; for example, the two, three or four closest clusters.

Top Ranking Cluster Names

Use this option to add to the output file the names of the clusters whose centroids are the closest to the current observation. The closest cluster is the one the observation belongs to, its name is displayed in the

column `kc_name_<Target variable>`. The next closest cluster is displayed in the column `kc_name_<Target Variable>_2`, and so on until the furthest cluster. You can choose to add all the clusters or only the closest one.

- To Add All the Clusters
  - Select *All*.
- To Add Only the Closest Clusters
  1. Select *Top*.
  2. In the text field enter the number of clusters you want to add; for example, the two, three or four closest clusters.

> **i Note**
>
> The name of a cluster is its number by default. You can modify it in the column *User Name* of the panel *Clusters Profiles* accessible through the main menu.

Top Ranking Distances

Use this option to add to the output file the distances of each observation from the clusters centroids. The distance from the closest centroid is displayed int the column `kc_best_dist_<TargetVariable>`, the distance from the second closest centroid is displayed in the column `kc_best_dist_<TargetVariable>_2`, and so on until the furthest centroid. You can add the distances from all centroids or only the shortest.

- To Add All the Distances
  - Select *All*.
- To Add Only the Shortest Distances
  1. Select *Top*.
  2. In the text field enter the number of distances you want to add; for example, the two, three or four shortest distances.

> **i Note**
>
> When SQL mode is activated, the notion of nearest cluster does not exist. If a case belongs to a cluster, distance is set to 0. If a case does not belong to a cluster, distance is set to 1.

Probabilities

Use this option to add to the output file the probabilities that the observation belongs to each cluster. The probability for the observation to belong to the closest cluster is displayed in the column `kc_best_proba_<TargetVariable>`, this probability is usually the highest. The probability for the observation to belong to the second closest cluster is displayed in the column `kc_best_proba_<TargetVariable>_2`, and so on until the furthest cluster. You can add all the probabilities or only the ones corresponding to the closest clusters.

- To Add All Probabilities
  - Select *All*.
- To Add Only the Probabilities for the Closest Clusters
  1. Select *Top*.
  2. In the text field enter the number of probabilities you want to add; for example, the two, three or four best probabilities.

> **i Note**
>
> When SQL mode is activated, the notion of nearest cluster does not exist. If a case belongs to a cluster, probability is set to 1. If a case does not belong to a cluster, probability is set to 0.

Miscellaneous Outputs

Disjunctive Coding

Use this option to add to the output file the disjunctive coding of the clusters. A column is generated for each cluster and contains either 0 or 1 depending whether the observation belongs to the cluster or not. The columns created are named `kc_disj_<TargetVariable>_<ClusterId>`. For example if the target variable is Age and the model has five clusters, the five following columns will be generated `kc_disj_age_1`, `kc_disj_age_2`, `kc_disj_age_3`, `kc_disj_age_4`, `kc_disj_age_5`.

Target Mean / Target Key Probability

Use this option to add to the output file:

- for continuous targets:
  - the mean of the target for the cluster containing the observation (displayed in the column `kc_<TargetVariable>_Mean`),
  - the difference with the actual target value if the latter is known for the current observation (displayed in the column `kc_<TargetVariable>_Error`).
    Note that when the actual target value is not available, it is set to 0. Thus, the difference between the actual target value and the computed target mean equals to the same value, meaning that both `kc_<TargetVariable>_Mean` and `kc_<TargetVariable>_Error` columns display the same value.
- for nominal targets:
  - the proportion of the least frequent category of the target variable (key category) in the cluster containing the current observation (displayed in the column `kc_<TargetVariable>_Mean`).

5.2.6.4.3 Types of Results Available

The application of a model to a dataset allows you to obtain three types of results:

- The cluster index for each observation.
- The disjunctive encoding (or dummy coding) of the cluster indexes, which means that, for each cluster, a boolean variable is created indicating whether the current observation belongs to that cluster or not. For a given observation, the value "1" is assigned to the variable corresponding to the cluster containing the observation and the value "0" is assigned to the variables corresponding to the other clusters. The variable names are built according to the following pattern: `kc_<TargetName>_<ClusterIndex>`.
  Consider, as an example, that you have generated a five-clusters model. When applying this model, the application creates five variables corresponding to the five generated clusters. For an observation belonging to cluster 3, the result appears as shown below.

KxIndex	class	kc_class	kc_class_1	kc_class_2	kc_class_3	kc_class_4	kc_class_5
15	1	3	0	0	1	0	0

- The target mean for each cluster, that is, the percentage of observations belonging to the target category of the target variable contained in each cluster.

Depending upon the level of information desired, you can choose to generate:

- Only the cluster index to which each observation belongs (*Predicted Value Only* option).
- The cluster index and the disjunctive encoding of the cluster indexes (*Cluster Id Disjunctive Coding* option). You can also decide to include in the results file all input variables of the application dataset (*Cluster Id Disjunctive Coding (+copy dataset)* option).
- The cluster index and the target mean for each cluster (*Cluster Id Target Mean* option).

For this scenario, you will apply the model to the file `Census01.csv` that you used previously to generate the model.

In the procedure to apply a model to a new dataset:

- Select the format *Text files*.
- In the *Generate* field, select the option *Cluster Id Target Mean*.
- Select the folder of your choice in which to save the results file (*Results generated by the model*).

5.2.6.4.4 Analyzing the Results of the Application

For this scenario, open the results text file in Microsoft Excel, generated when you applied the model to the `Census01.csv` file.

To open the model application results file:

1. Depending upon the format of the results file generated, use Microsoft Excel or another application to open the file.

   The figure below presents the headings and columns of the results file obtained for this scenario.

	A	B	C	D
1	KxIndex	class	kc_clusterId	kc_TargetMeanClustId
2	1	0	3	0,017524
3	2	0	5	0,476401001
4	3	0	3	0,017524
5	4	0	2	0,237075999
6	5	0	5	0,476401001
7	6	0	4	0,308898985
8	7	0	3	0,017524
9	8	1	5	0,476401001
10	9	1	1	0,942696989
11	10	1	1	0,942696989
12	11	1	5	0,476401001
13	12	1	5	0,476401001
14	13	0	3	0,017524
15	14	0	4	0,308898985
16	15	1	2	0,237075999
17	16	0	2	0,237075999

2. You can now analyze the results obtained and use these results of your analysis to make the right decisions.

5.2.6.4.4.1 Description of the Results File

Depending upon which options you selected, the results file will contain some or all of the following information, in the same order as seen below:

- The key variable defined during data description at the model parameter settings step. If your dataset did not contain a key variable, the key variable `KxIndex` would have been generated automatically by the application.
- Possibly, the target variable given as known values if the latter appeared in the application dataset, as is the case in this scenario.
- The variable `kc_clusterId`, which indicates the number of the cluster to which each observation belongs.
- The variable `kc_TargetMeanClusterId`, which indicates the proportion of observations belonging to the target category of the target variable that are contained in each cluster.
- The variables corresponding to each cluster, and an indication of the encoding disjunction of the cluster numbers. The names of these variables correspond to cluster numbers, prefixed by `kc_cluster_` , for example `kc_cluster_1` for cluster 1.

5.3 Time Series

5.3.1 About Time Series

This section of the guide is addressed to people who want to evaluate or use Automated Analytics and in particular the time series analysis feature.

Before reading this section, you should read the sections *Classification/Regression* and *Segmentation/Clustering* that present respectively:

- An introduction to Automated Analytics
- The essential concepts related to the use of Automated Analytics features

This part of the guide introduces you to the main functionalities of the time series feature. Using the application scenario you can create your first models with confidence.

Modeler lets you build predictive models from data representing time series. With a time series analysis, you can:

- Identify and understand the phenomenon represented by your time series.
- Forecast the evolution of time series in the short and medium term, that is, predict their future values.

5.3.2 Before Beginning

5.3.2.1 Files and Documentation Provided with this Guide

Sample Data Files

Automated Analytics is supplied with sample data files. These files allow you to take your first steps using various features of the application, and evaluate them.

During installation of SAP Predictive Analytics, the following sample files for time series analysis are saved under the folder `<installation directory>/Samples/KTS/`:

- R_ozone-la.txt,
- CashFlows.txt,
- KxDesc_CashFlows.txt.

Documentation

Full Documentation

Complete documentation can be found on the SAP Help Portal at http://help.sap.com/pa.

Contextual Help

Most screens in the application are accompanied by contextual help that describes the options presented to you, and the concepts required for their application.

5.3.2.2 Displaying the Contextual Help

Each screen in Automated Analytics is accompanied by contextual help that describes the options presented to you, and the concepts required for their application.

1. To display the contextual help, press F1.
2. To acces a searchable version of all the help for the application, select *Help* > *Open Full Searchable Help*.

5.3.3 General Introduction to Scenarios

5.3.3.1 Scenario 1

This scenario demonstrates how to use a time series analysis for creating a standard model.

The data used in this scenario are monthly averages of hourly ozone (O3) readings in downtown Los Angeles from 1955 to 1972.

Ozone is a gas providing a protective shield against the ultraviolet radiation. When found in the lower atmosphere, it is a major component of smog. Thus ozone rate is a common measure for smog intensity.

Los Angeles municipality took three measures in order to reduce this level and so decrease the smog downtown:

- in 1960, the Golden State Freeway, which sails round downtown, opened,
- in the same year, the rule 63 came into effect, lowering the amount of allowable reactive hydrocarbons in gasoline,
- in 1966, emission regulations for new car engines were introduced.

The purpose of this scenario is to confirm the decreasing trend of the ozone rate by predicting the next 18 months and describing the different signal elements based on the ozone rate.

5.3.3.2 Scenario 2

This scenario demonstrates how to use a time series analysis to create a model with extra predictable inputs.

In this scenario, you are an executive of a financial entity that manages cash-flows. Your role is to make sure that credits are available with the correct amount at the correct date to provide the best management possible of your financial flows.

Time series provides you with two methods for reaching your objective:

- creating a standard model,
- creating a model with extra predictable variables.

5.3.3.3 Introduction to Sample Files

Automated Analytics provides sample data files allowing you to evaluate the time series analysis feature and take your first steps in using it.

The file `R_ozone-la.txt` is the sample data file that you will use to follow Scenario 1. It is an excerpt from the book Time Series Analysis: Forecasting and Control (G.E.P. Box and G.M. Jenkins), Third Edition, Prentice-Hall, 1994.

This file presents monthly averages of hourly ozone (O3) readings in downtown Los Angeles from 1955 to 1972. Each observation is characterized by 2 data items. These data, or variables, are described in the following table.

Variable	Description	Example of Values
`Time`	Month and year of the readings	A date in the format yyyy-mm-dd, such as 1955-01-28
`R_ozone-la`	Average of the hourly readings for the month	A numerical value with two decimals

The file `CashFlows.txt` is the sample data file used to follow Scenario 2 of the time series feature and use the extras predictable inputs.

This file presents daily measures of cash flows from January 2, 1998 to September, 30 1998. Each observation is characterized by 25 data items. The data or variables are described in the following table.

Variable	Description	Example of values
`Date`	Day, month and year of the readings	A date in the format yyyy-mm-dd such as 1998-01-02
`Cash`	Cash flow	A numerical value with n decimals
`BeforeLastMonday`, `LastMonday`, `BeforeLastTuesday`, `LastTuesday`, `BeforeLastWednesday`, `LastWednesday`, `BeforeLastThursday`, `LastThursday`, `BeforeLastFriday`, `LastFriday`	Boolean variables that indicates if the information is true or false	1 if the information is true.
`Last5WDays, Last4WDays`	Boolean variables that indicate if the date is in the 5 or 4 last working days of the month	1 if the information is true.
`LastWMonth`, `BeforeLastWMonth`	Boolean variables that indicates if the information is true or false	1 if the information is true.
`WorkingDaysIndices`, `ReverseWorkingDaysIndices`	Indices or reverse indices of the working days	An integer value
`MondayMonthInd`, `TuesdayMonthInd`, `WednesdayMonthInd`, `ThursdayMonthInd`, `FridayMonthInd`	Indices of the week days in the month	An integer value
`Last5WDaysInd`, `Last4WDaysInd`	Indices of the 5 or 4 last working days of the month	An integer value

The file `KxDesc_CashFlows.txt` is the description file corresponding to the data file `CashFlows.txt`.

5.3.3.3.1 Additional Sample Files

Additional sample files are provided to further test time series:

- Lag1AndCycles.txt
- Lag1AndCyclesAndWn.txt
- TrendAndCyclic.txt
- TrendAndCyclicAnd_4Wn.txt

- TrendAndCyclicAndWn.txt

These files are located in the folder `Samples\KTS`.

5.3.3.4 File Format Specifications

A training data file for time series must contain at least two columns:

- The date column,
- The signal column.

Three formats are supported for the Date column:

- datetime (ISO format: yyyy-mm-dd hh:mm), which has an hour-precision.
- date (ISO format: yyyy-mm-dd), which has a day-precision.
- number (number, for example "seconds").

The signal column (that is, the target variable) must be continuous

> **i Note**
>
> For a trend regressive modeling, an optional weight column can be used to tweak the modeling procedure. Setting the weight of some rows to 0 allows ignoring these rows during the modeling process. By default, when a weight variable is not provided, all rows have a weight of 1. Other modeling component (lag, cycle, period, auto regression, ...) will ignore the weight column.

For Extra Predictable Variables

Future values of the Extra Predictable Inputs have to be filled in order to use them in a modeling session. These variables have to be filled at least in the same range as the wanted forecasts. The file will have the appearance described by the table below (time series allows missing values in the extra-predictable variables).

Line Index	Date	Monday	Tuesday	Wednesday	Thursday	Friday	...	WorkingDayIndices		FlowsEmitted
...	...	...	...	...	...	...	...	...	...	...
177	1998-09-22	0	1	0	0	0	...	15	...	-242.1920724
178	1998-09-23	0	0	1	0	0	...	16	...	-395.9563714
179	1998-09-24	0	0	0	1	0	...	17	...	-381.9100825
180	1998-09-25	0	0	0	0	1	...	18	...	-769.5131684
181	1998-09-28	1	0	0	0	0	...	19	...	-1679.648843
182	1998-09-29	0	1	0	0	0	...	20	...	-1124.288375
183	1998-09-30	0	0	1	0	0	...	21	...	-601.915403
184	1998-10-01	0	0	0	1	0	...	1	...	
185	1998-10-02	0	0	0	0	1	...	2	...	
186	1998-10-05	1	0	0	0	0	...	3	...	
...	...	...	...	...	...	...	...	...	...	

The last known signal value is at the line 183 with the corresponding date 1998-09-30. This line corresponds to the end of the training dataset. The figures present after this line are the future values of the extras predictable variables: these figures are considered as predictive information.

Please note that the date variable has a special status. This variable is not considered as an extra predictable input; nevertheless it is possible to fill its future values. If you fill this variable in the predicted range, the application uses the values for forecasting. If you don't, it generates the future dates. This is true whether extra

predictable variables exist or not. This feature can be very useful if you are not satisfied by the dates automatically generated by the application.

The same file format has to be used for the training dataset and the application dataset.

> **i Note**
>
> If you want to use your own dates instead of the automatic date generation, please follow the same steps as the addition of extras predictable inputs.

5.3.3.5 Starting a Time Series Analysis

To accomplish the scenario, you will use the Java-based graphical interface of the application. SAP Predictive Analytics allows you to select the features with which you will work, and help you at all stages of the modeling process

To Start SAP Predictive Analytics

1. Select *Start* > *Programs* > *SAP Business Intelligence* > *SAP Predictive Analytics Desktop* > *SAP Predictive Analytics*

   SAP Predictive Analytics appears.
2. Select *Modeler*, and then*Create a Time Series Analysis*.

5.3.4 Scenario 1: Standard Modeling with Time Series

Data modeling with time series is subdivided into four broadly defined stages:

1. Defining the Modeling Parameters
2. Generating and Validating the Model
3. Analyzing and Understanding the Analytical Results
4. Using a Generated Model

5.3.4.1 Application Options

On every screen of the application, one or more of the following options may be available in a toolbar located under the screen title.

- Printing the screen,
- Saving the screen content,
- Copying the screen content,
- Displaying the contextual Help.

5.3.4.1.1 Printing the Screen

To Print the Screen

1. Click the *Print* button.

   A dialog box appears, asking you to select the printer to use.
2. Select the printer to use and set other print properties if need be.
3. Click *OK*.

   The screen content is printed.

5.3.4.1.2 Saving the Screen Content

1. Click the *Save* button.

   A dialog box appears, asking you to select the file properties
2. Type a name for your file.
3. Select the destination folder.
4. Click *OK*.

   If the screen displays:

   - A graph, it is saved as a PNG image
   - A report, it is saved as an HTML file

5.3.4.1.3 Copying the Screen Content

Click the *Copy* button.

If the screen displays:

- A report, the application copies its HTML code. You can paste it into a word processing program or into a spreadsheet program (such as Excel) and use it to generate your own graph.
- A graph, the application copies its parameters. You can paste it into a spreadsheet program (such as Excel) and use it to generate your own graph.

5.3.4.2 Step 1 - Defining the Modeling Parameters

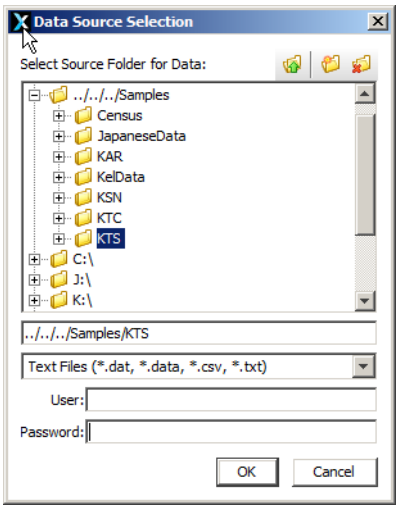
5.3.4.2.1 Selecting a Data Source

For this scenario, use the file `R_ozone-la.txt` as a training dataset.

To Select a Data Source:

1. On the screen *Select a Data Source*, select the option *Data Type* to select the data source format to be used.
2. Click the *Browse* button.

   The following dialog box appears.



3. Double-click the `Samples` folder, then the `KTS` folder.

   > **i Note**
   >
   > Depending on your environment, the Samples folder may or may not appear directly at the root of the list of folders. If you selected the default settings during the installation process, you will find the Samples folder located in `C:\Program Files\SAP Predictive Analytics`.

4. Select the file `R_ozone-la.txt`, then click *OK*.

   The name of the file appears in the *Dataset* field.

5.3.4.2.2 Selecting a Partition Strategy

To generate a time series model, you must select a partition strategy to cut your training dataset into the three sub-sets: training, validation and testing. Since the order of the observations in the dataset is important for the modeling, only two types of partition strategies are available in time series:

- Sequential with Testing
- Sequential without Testing

For more information on Partition Strategies, see the *Classification, Regression, Segmentation and Clustering Scenarios - Automated Analytics User Guide*.

5.3.4.2.2.1 Selecting a Partition Strategy

For this Scenario, do not change the partition strategy by default.

1. On the screen *Select a Data Source*, click the button *Partition Strategy*.

   The panel *Partition Strategy* is displayed.
2. In the *Predefined* list, select the partition strategy you want to use.
3. Click the *OK* button.
4. Back to the panel *Select a Data Source*, click the *Next* button.
5. The screen *Data Description* appears.
6. Go to the section Describing the Data Selected.

5.3.4.2.3 Describing the Data Selected

To describe your data, you can:

- Use an existing description file, that is, taken from your information system or saved from a previous use of Automated Analytics.
- Or create a description file using the Analyze option. In this case, it is important that you validate the description file obtained. You can save this file for later re-use.

> ⚠ Caution
>
> The description file obtained using the Analyze option results from the analysis of the first 100 lines of the initial data file. In order to avoid all bias, we encourage you to mix up your dataset before performing this analysis.

Why Describe the Data Selected?

In order for Automated Analytics features to interpret and analyze your data, the data must be described. To put it another way, the description file must specify the nature of each variable, determining their:

- Storage format: number (*number*), integer (*integer*), character string (*string*), date and time (*datetime*) or date (*date*).
- Type: *continuous*, *nominal*, *ordinal* or *textual*.

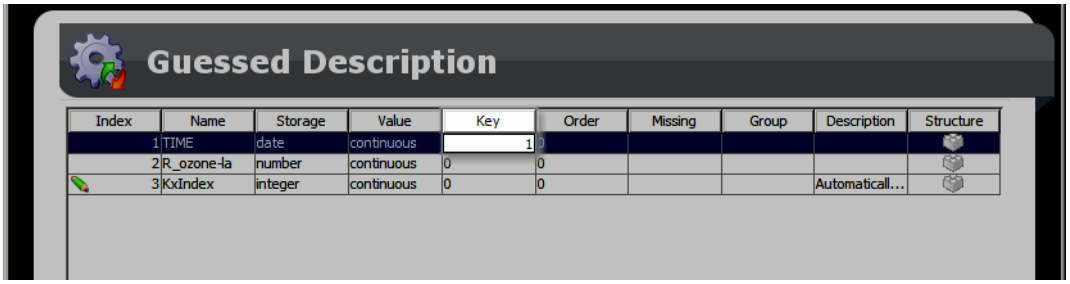
For more information about data description, see the Classification, Regression, Segmentation and Clustering Scenarios - Automated Analytics User Guide.

5.3.4.2.3.1 Specifying that a Variable is a Key

For data and performance management purposes, the dataset to be analyzed must contain a variable that serves as a key variable. Two cases should be considered:

- If the initial dataset does not contain a key variable, a variable index *KxIndex* is automatically generated. This will correspond to the row number of the processed data.
- If the file contains one or more key variables, they are not recognized automatically. You must specify them manually in the data description. See the procedure To Specify that a Variable is a Key. On the other hand, if your data is stored in a database, the key will be automatically recognized.

1. In the *Key* column, click the box corresponding to the row of the key variable.
2. Type in the value "1" to define this as a key variable.



5.3.4.2.3.2 Analyzing a Data File

For this Scenario:

- Select *Text Files* as the file type.
- Use the *Analyze* function to describe the `R_ozone-la.txt` data file.
- Set the *TIME* variable as the key.
- Set the *KxIndex* variable Key to 0.
- Set the *TIME* variable Order to 1.

1. On the screen *Data Description*, click the *Analyze* button.

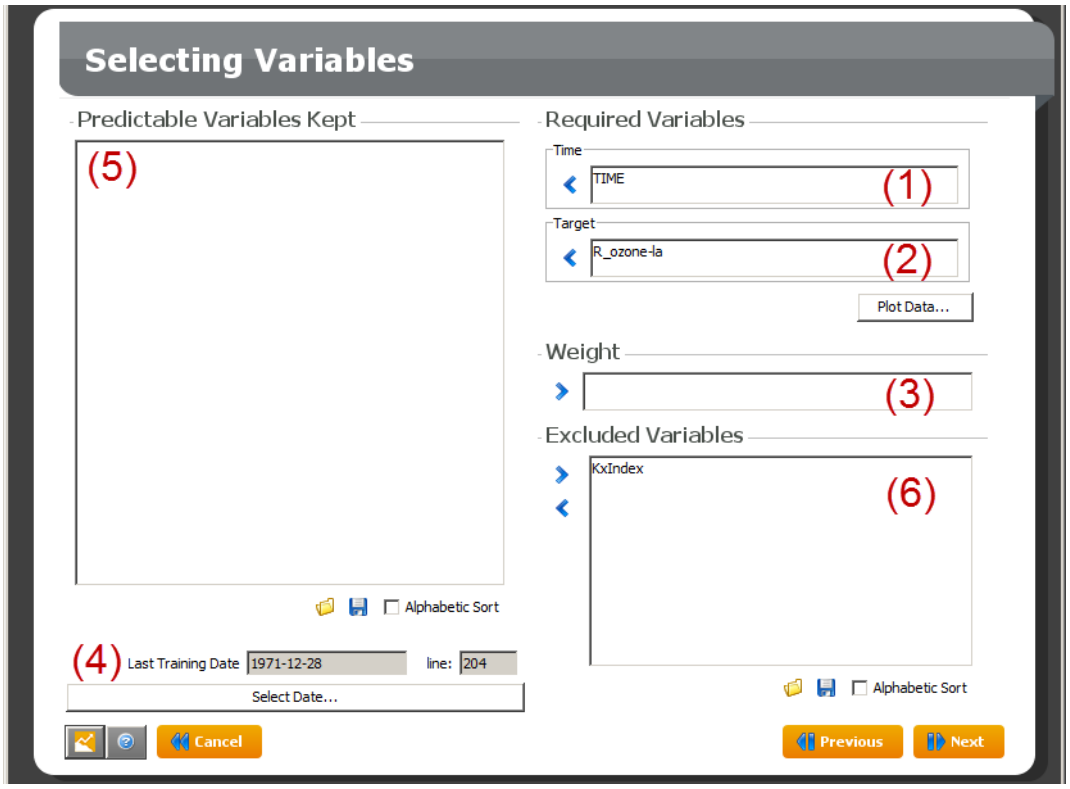
   The file description is displayed.
2. Validate the description (storage type and value).
3. Click the *Next* button.

5.3.4.2.4 Selecting Variables

The panel *Selecting Variables* allows you to:

- select the time variable (*1*),
- select the target variable (*2*),
- select a weight variable (optional) (*3*),

- set the last date to use for training the model (*4*),
- select which variables should be kept for the modeling (*5*) and which should be excluded (*6*).



5.3.4.2.4.1 Selecting a Time Variable

For this Scenario:

- Keep *TIME* as the time variable.
- Keep *R_ozone-la* as the target variable.
- Do not select a weight variable.
- Keep the last training date selected by default.

1. On the screen *Selecting Variables*, in the section *Predictable Variables Kept* (left hand side), select the variable you want to use as the time variable.
2. Click the button > located on the left of the *Time* field in section *Required Variable* (upper right hand side). The variable moves to the *Time* field.

> **i Note**
>
> To remove the time variable, select the variable in the *Time* field and click the button < to move the variables back to the screen section *Predictable Variables Kept*.

5.3.4.2.4.2 Selecting a Target Variable

1. On the screen *Selecting Variables*, in the section *Predictable Variables Kept* (left hand side), select the variable you want to use as the Target Variable.
2. Click the button > located on the left of the *Target* field in section *Required Variable* (upper right hand side).

   The variable moves to the *Target* field.

   > **i Note**
   >
   > To remove the target variable, select the variable in the *Target* field and click the button < to move the variables back to the screen section *Predictable Variables Kept*.

5.3.4.2.4.3 Selecting a Weight Variable

1. In the field *Predictable Variables Kept*, select the weight variable.
2. Click the > button located on the right of the *Weight* field.

   > **i Note**
   >
   > To remove the weight variable, select the *Weight* field and click the < button.

5.3.4.2.4.4 Excluding Variables

1. In the field *Predictable Variables Kept* (left hand side), select the variables you want to exclude.

   > **i Note**
   >
   > To select all the variables of a field, click inside the field and push the keys Ctrl and A at the same time.

2. Click the > button located near the top right corner of the field *Excluded Variable*.

5.3.4.2.4.5 Displaying the Signal

1. On the screen *Selecting Variables*, click the button *Plot Data*..., located in the section *Required Variables*. The screen *Display Signal* is displayed.
2. Click the *Time* list and select the variable containing the time information.

   > **i Note**
   >
   > By default, time series uses the first column of the dataset as time variable.

3. Click the *Signal* list and select the variable containing the signal information.

4. Click the *Previous* button to go back to the screen *Selecting Variables*.

5.3.4.2.4.6 Reducing the Training Dataset

The last date found in the dataset is automatically selected as the last training date. The second field, which indicates the number of the line in the dataset, is automatically updated depending on the date you have selected.

1. On the screen *Selecting Variables*, click the button *Select Date...*. The panel *Sample Data View* is displayed.
2. Uses the *First Row Index* and *Last Row Index* to display the line containing the date you want to select as the last training date.
3. Click the *Refresh* button to update the list of rows displayed.
4. Click the row corresponding to the date you want to select. The selected row is displayed in the section *Current Selection* at the bottom of the panel.
5. Click the *OK* button to validate your selection. The window closes and the Last Training Date information are updated in the main panel.

5.3.4.2.5 Checking Modeling Parameters

The screen *Summary of Modeling Parameters* allows you to check the modeling parameters just before generating the model.

- The name of the model is filled automatically. It corresponds to the name of the target variable (*R_ozone-la* for this scenario), followed by the underscore sign ("_") and the name of the data source, without its file extension (*R_ozone-la* in this case).
- The field *Number of Forecast*(s) allows you to select the number of forecasts to generate. The time unit used is determined by the data analyzed. For example, if the dataset observations are recorded monthly, the time unit will be one month. See section Defining the Number of Forecasts.
  The maximum number of forecasts allowed is indicated in the field *Maximum Forecast*. This number depends on the number of extra predictable variables available. If there is no extra predictable variables, the number of forecasts is unlimited.
- The *Autosave* button allows you to activate the feature that will automatically save the model once it has been generated. When the autosave option is activated, a green check mark is displayed on the Autosave button.

Defining the Number of Forecasts

For this Scenario, define the number of forecasts to 24, that is, two years.

To Define the Forecasts Number, on the screen *Summary of Modeling Parameters*, in the field *Number of Forecast*(s), enter the number of forecasts you want to obtain.

5.3.4.2.5.1 Defining the Advanced Parameters

The advanced parameters allow you to:

- limit the number of analyzed variables,
- define the modeling procedure.

1. Click the *Advanced* button. The panel *Specific Parameters of the Model* is displayed.
2. Set the parameters as explained in the following sections.
3. Click the *OK* button to save the new parameters. The panel *Summary of Modeling Parameters* is displayed.

5.3.4.2.5.2 Defining the Number of Analyzed Variables (optional)

Time series automatically generates the variables that are necessary to the modeling. Among these, it is possible to reduce the number of the cyclic variables by changing the parameter *Maximum Length of Analyzed Cycles* and the lagged variables created by changing the parameter *Maximum Order of the Autoregressive Model*.

- The *Maximum Length of Analyzed Cycles* controls the way that time series analyzes the periodicities in the signal. This is the length of the longest cycle time series will try to detect. The default value is 450. It is also limited by the size of the training dataset. You can disable the cyclic analysis by setting this parameter to zero.

  > **i Note**
  >
  > By reducing the default number of variables generated by time series, you are able to reduce the computation time. However it is strongly recommended to use the default settings otherwise the quality of modeling could be compromised.

- The *Maximum Order of the Autoregressive Model* controls the way that time series analyzes the random fluctuations in the signal. This parameter defines the maximum dependency of the signal on its own past values. You can set this parameter to zero to disable the fluctuations analysis.

5.3.4.2.5.3 Defining the Other Modeling Options

The *Ignore Outliers when Estimating the Trend* checkbox uses a strategy for reducing the effect of outliers when estimating the regressions in deterministic trends. This leads to an improvement in trend estimation.

The *Force Positive Forecasts* checkbox allows users to force time series to generate a positive model (with positive forecasts only).

5.3.4.2.5.4 Defining the Variables

This parameter groups some controls for the variable selection feature. When a variable selection is used, an automatic selection process is performed on trends or AR models during the competition and the result is kept only if it improves the final model.

- The *Percentage of Variable Contributions to Keep* is the percentage of contributions that are kept in the automatic selection process. The default value is 95%.

  > **i Note**
  >
  > You can change the *Percentage of Variable Contributions to Keep*. However, the engine will operate as follows:
  > - Remove all less contributing variables until reaching your specified threshold (for example: keep 90% of contribution).
  > - If the L2 (Root Mean Square Error) degradation on validation is greater than 5%, the variable selection is ignored.

- The *Activate for All Extra-predictable Based Trends* option performs a variable selection on all extra-predictable based trends. User variables are kept only if they have sufficient contributions in the trend regression. The checkbox is enabled by default.
- The *Activate for All Autoregressive Models* option performs an automatic variable selection on the past values of the signal for all autoregressive models. This leads to a more parsimonious AR model, that is a simpler model and a lower order. The box is not checked by default.

5.3.4.2.5.5 Available Modeling Procedures

If you want to modify the modeling procedure used to generate the model, you can select one of the procedures listed in the following table.

Modeling Procedure	Description
Default	Corresponds to the standard Time Series modeling. When using the standard Time Series modeling, the application tries all the models listed in **Table 3**. If none can be used, it defaults to the *Exponential Smoothing* procedure.
Only Based on Extra-Predictable Variables	Works as a classification or regression model built on the extra-predictable variables with the signal as the target. This mode can be used to refine and validate the extra-predictable variables or to identify useless ones.
Disable the Polynomial Trends	Generates all the models except those containing a polynomial trend.
Customized	Allows you to enable or disable the types of models generated by the application when analyzing the signal. Refer to Table 3 for the types of models that can be disabled.

Modeling Procedure	Description
Exponential Smoothing	Generates the forecasts using exponential smoothing. To select the better version of exponential smoothing, the application compares several smoothing procedures: • Double smoothing, which detects the trend, • Triple smoothing with various sizes of cycle, on the condition that the training dataset contains at least two cycles. The application determines the cycles based on the granularity of the time variable. Depending on the granularity, the application tests several cycles selected a priori. For example, if the time granularity is the month, the application tests two cycles: a cycle of 3, which corresponds to a quarter and a cycle of 12, which corresponds to the year. Refer to **Table 2** for the cycles tested by the application.
Linear Regression	Generates the forecasts using a simple linear regression.

Cycles tested by the application based on the time variable granularity

Granularity	Time Basis	Cycle Length	Other Tested Cycles
Second	Minute	60	360 (60*60) (hour)
Minute	Hour	60	1440 (60*24) (day)
Hour	Day	24	168 (week)
Day	Week	7	30 (month) - approximated
Week	Year	52	12 (quarter); 4 (month) - approximated
Month	Year	12	3 (quarter)
Quarter	Year	4	
Semester	Year	2	
Year	Decade	10	

The following table lists the types of models used by the *Default* mode, which can be disabled as needed when using the *Customized* modeling procedure.

Available types of models

Component	Model Type	Description
Trends	Lag1	Previous value of the signal
	Lag2	Value before previous
	Second Order Differencing	Trend using double-differencing to propagate the slope of the signal
	Linear in Time	Linear regression on the time
	Polynomial in Time	Polynomial regression on the time
	Linear in ExtraPredictables	Linear regression on the extra-predictable variables
	Linear in Time and Linear in ExtraPredictables	Linear regression on the time and extra-predictable variables

Component	Model Type	Description
	Polynomial in Time and Linear in ExtraPredictables	Polynomial regression on the time and linear regression on the extra-predictable variables
Periodicities	Cyclics	Detection of cyclic variables
	Seasonalities	Detection of seasonal variables
	Periodic Extrapredictables	Extra-predictable usage as Periodics
Fluctuations	Autoregressive	Autoregressive modeling

5.3.4.2.5.6 Modifying the Modeling Procedure

1. In the section *Modify the Modeling Procedure*, check the desired option.
2. If you have selected the Customized option, uncheck the types of models you want to disable.
3. Click the *OK* button. The panel *Summary of Modeling Parameters* is displayed.

5.3.4.2.5.7 Activating the Autosave Option

The *Model Autosave* panel allows you to activate the option that will automatically save the model at the end of the generation process and to set the parameters needed when saving the model.

To activate the option, proceed as follows:

1. In the *Summary of Modeling Parameters* panel, click the *Autosave* button.
   The *Model Autosave* panel is displayed.
2. Check *Enable Model Autosave*.
3. Set the parameters listed in the following table.

Parameter	Description
Model Name	This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
Description	This field allows you to enter the information you want, such as the name of the training dataset used, the polynomial degree or the performance indicators obtained. This information could be useful to you later for identifying your model. Note that this description will be used instead of the one entered in the *Summary of Modeling Parameters* panel.

Parameter	Description
Data Type	This list allows you to select the type of storage in which you want to save your model. The following options are available: ○ *Text Files*, to save the model in a text file. ○ *Data Base*, to save the model in a database. ○ *Flat Memory*, to save the model in the active memory.
Folder	Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
File/Table	This field allows you to enter the name of the file or table that is to contain the model. When saving the model as a text file, you must enter one of the following format extensions `.txt` (text file in which the data is separated by tabs) or `.csv` (text file in which the data is separated by commas).

4. Click *OK*.

5.3.4.3 Step 2 - Generating the Model

5.3.4.3.1 Generating the Model

Once the modeling parameters are defined, you can generate the model.

1. On the screen *Summary of Modeling Parameters*, click the *Generate* button. The screen *Training the Model* appears. The model is being generated. A progress bar allows you to follow the process.
2. If the Autosave option has been activated in the panel *Summary of Modeling Parameters*, a warning message is displayed at the end of the learning process confirming that the model has been saved.
3. Click Close.

5.3.4.3.2 Following the Progress of the Generation Process

There are two ways for you to follow the progress of the generation process:

- The Progress Bar displays the progression for each step of the process. It is the screen displayed by default.
- The Detailed Log displays the details of each step of the process.

To display the Progression Bar

Click *View Type* and select (Progress).

The progression bar screen appears.

To Display the Detailed Log

Click *View Type* and select the (Log) button.

The detailed log displays the details of each step of the process.

To Stop the Learning Process

1. Click the (Stop Current Task) button.
2. Click the *Previous* button.
   The screen *Summary of Modeling Parameters* appears.
3. Go back to the section on checking modeling parameters.

5.3.4.3.3 Visualizing the Model Results

At the end of the generation process, a summary of the model results appears.

If you have built more than one model in the same session, all model debriefing will be displayed on this screen sorted by Date of Build.

For more information on the model summary, go to section Understanding the Model Debriefing.

> ⚠ Caution
>
> In some cases, the message No Model Found is displayed instead of the signal information. It means that none of the models found can predict accurately the signal evolution.

5.3.4.4 Step 3 - Analyzing and Understanding the Generated Model

The suite of plotting tools within the application allows you to analyze and understand the model generated:

- The performance of the model,
- The forecasts generated by the model.

5.3.4.4.1 Model Debriefing

5.3.4.4.1.1 To Display the Model Debriefing

On the *Using the Model* menu, select the *Model Overview* option.

The screen *Model Overview* appears.

> **i Note**
>
> If you have built more than one model in the same session, all model debriefing will be displayed on this screen sorted by Date of Build.

5.3.4.4.1.2 Understanding the Model Debriefing

The Model Debriefing screen is composed of four sections detailing information on:

- The overview of the model
- The targets statistics
- The model components
- The model performance

5.3.4.4.1.3 The Model Overview

This section details the following information:

Name	Significance	For this scenario...
`Model Name`	Name of the model. It is generated by using the target variable name and the dataset name	R_ozone-la_R_ozone-la
`Dataset`	Name of the data source used for the model	R_ozone-la.txt
`Initial Number of Input Variables`	Total number of variables in the dataset	3
`Number of Selected Variables`	Number of variables used to generate the model	1
`Number of Records`	Number of observation in the data source file	204
`Building Date`	Date and time when the model was build	2015-03-10 10:51:26

`Learning Time`	Duration of the learning process	3s
`Engine Name`	Name of the component used to build the model	Kxen.TimeSeries

5.3.4.4.1.4 The Targets Statistics

For continuous targets:

Name	Significance	For this scenario...
`<target name>`	name of the target variable for which the statistics are stated	R_ozone-la
`Min`	Minimum value found in the dataset for the target variable	1.33
`Max`	Maximum value found in the dataset for the target variable	7.54
`Mean`	Mean of the target variable	3.662
`Standard Deviation`	Measure of the extent to which the target values are spread around their average	1.29

5.3.4.4.1.5 The Signal Components

This section details the model components, that is the components of the polynomial used to generated the forecasts.

A model is a combination of at least one of the three following types of elements:

- one trend,
- one or more cycles,
- one fluctuation.

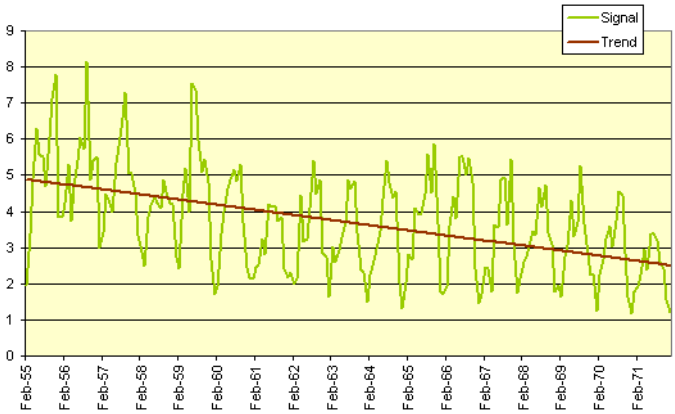
5.3.4.4.1.6 The Trend

The trend is the general orientation of the signal. The four types of trends are detailed in the following table:

Type of trend	Can be displayed as...

`Polynom(Time)`	a curve corresponding to the detected polynomial. A polynomial is modeled using Classification/Regression. The available functions for the polynomial are Time, square(Time), sqrt(Time), where Time equals the value of the *DateColumnName* parameter.
`Polynom(Time, ExtraPredictable)`	a curve corresponding to the detected polynomial. This is the same function using in addition the extra predictable variables as inputs.
`Linear(Time)`	a straight line. It is a special case of *Polynom(Time)* .
`Linear(Time, ExtraPredictables)`	a straight line. This is the same function using in addition the extra predictable variables as inputs.
`Polynom (ExtraPredictables)`	a curve corresponding to the detected polynomial. This function could be very next to a cyclic representation because Classification/Regression is only using the extra-predictable variables as inputs.
`L1`	the signal moved one step forward. This is the basic forecast where the predicted observation equals the latest signal observation.
`L2`	the signal moved two step forward.

For this scenario, the trend found in the signal is a *Linear(TIME)*. The following graph shows the trend compared with the signal.



5.3.4.4.1.7 The Cycles

The cycles are periodic elements that can be found at least twice in the *Training* dataset.

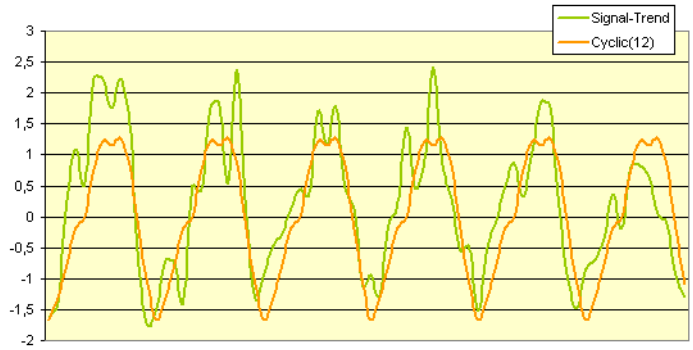
The two types of cycles are detailed in the following table:

Type of cycle	Description
`Periodic`	a cycle not depending on the date. A periodic is defined by the number of time units it covers.
`Seasonal`	a cycle depending on the date or time. For example, the day of the month, the hour of the day, and so on. **i Note** Time series can automatically detect the following seasonal variables: • dayOfYear • dayOfMonth • dayOfWeek • monthOfYear • halfMonthOfYear • weekOfYear • weekOfMonth • hourOfDay • minuteOfHour • secondeOfMinute

For this scenario, no periodic has been detected.

The following graph presents an example of two periodics found in the signal:

- A *monthOfYear* cyclic representing 12 months, that is a periodic of one year. The following graph presents this cycle compared with part of the signal once the trend has been extracted.



- A *Cyclic*(52) representing 52 months, that is a periodic of about four years.

5.3.4.4.1.8 The Fluctuation

The fluctuation is what is left when the trend and the cycles have been extracted. It is modeled with an auto-regression that uses a window of past data to model the current residue. The number of observations in the window is determined by time series depending on the total number of observations in the training dataset.

For this scenario, fluctuations have been detected.

The following graph presents the auto-regression. The orange area represents the window of past observations which are based on the past 37 months. The point in the red circle is the point calculated by the auto-regression.

5.3.4.4.1.9 The Residuals

The residuals is what is left when the trend, the cycles and the fluctuation have been extracted from the signal. This part called white noise and made of random elements that cannot be modeled.

5.3.4.4.1.10 The Outliers

An actual signal value is qualified as outlier once its corresponding forecasting error is considered to be abnormal relative to the forecasting error mean observed on the estimation dataset. The forecasting error indicator is the absolute difference between the actual and predicted values. This is also called the residue. The residue abnormal threshold is set to 3 times the standard deviation of the residue values on an estimation (or validation) dataset.

A value Yi is considered an outlier if |Residue(Yi) – Mean(Residue(Y), Estimation)| > 3 *StdDev(Residue(Y), Estimation).

Where:

- Residue (Yi) = Yi – Predict(Yi)

- Yi = value of the signal variable for a given observation i
- Predict(Yi) = predicted value of the signal variable for a given observation i
- Mean(ResidueY, Estimation): mean of the values of the residues on the estimation dataset
- |Residue(Yi) – Mean(ResidueY, Estimation)| : absolute difference between the terms Residue(Yi) and Mean(ResidueY, Estimation)|
- StdDev(Residue(Y), Estimation): standard deviation for the values of the residues on the estimation dataset

5.3.4.4.1.11 The Model Performance

This section gives the model performance.

Name	Significance	For this Scenario
`Horizon-Wide MAPE`	This quality indicator for the forecasting model is the mean of MAPE values observed over all the training horizon. A value of zero indicates a perfect model while values above 1 indicate bad quality models. A value of 0.09 means that the model takes into account 91% of the signal or , in other words, the forecasting error (model residues) is relatively of 9%. `MAPE - Mean Absolute Percentage Error` : The MAPE value is the average of the sum of the absolute values of the percentage errors. It measures the accuracy of the model's forecasts and indicates how much the forecasts differ from the real signal value.	0.178

5.3.4.4.2 Forecasts

5.3.4.4.2.1 Displaying the Forecast Plot

1. On the *Using the Model* menu, click the *View Forecasts* option.
2. A pop-up is displayed asking you to confirm or update the name and location of the training dataset file.
3. Update the information if you have renamed or moved the training dataset file, or if its type has been changed.

> **i Note**
>
> Steps 2 and 3 are required in case you open a saved model and the dataset used to train it has been moved, for example.

4. Click *OK*.

   The screen *View Forecasts* appears.

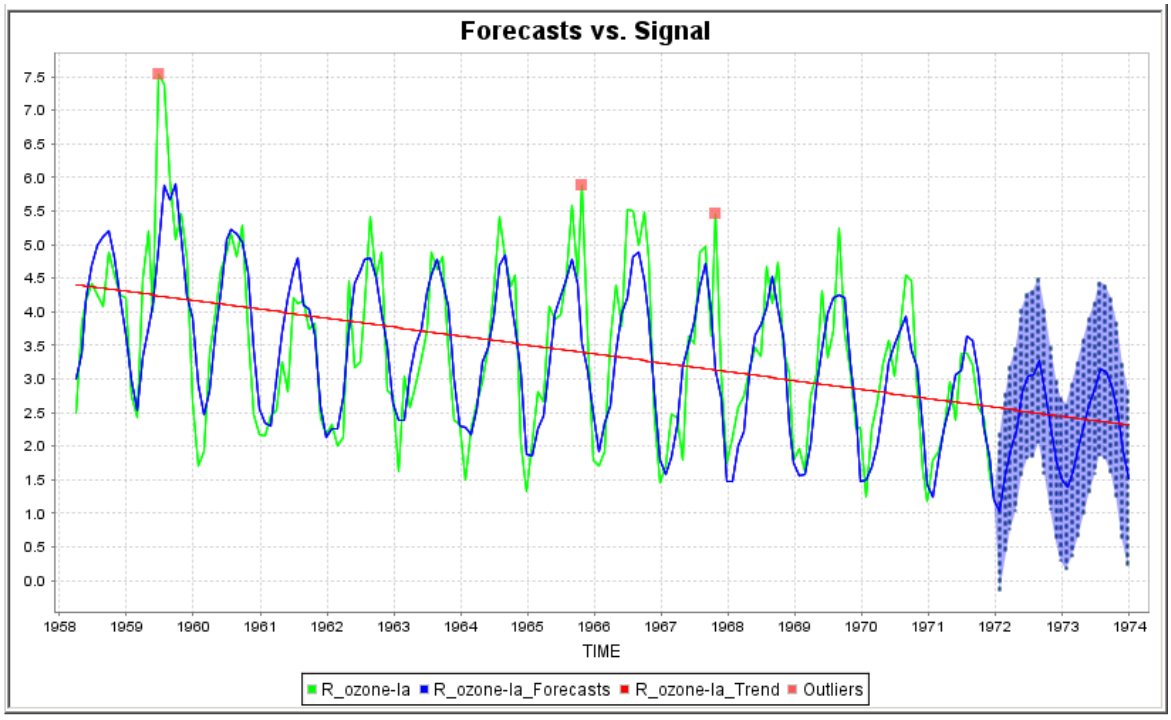
> **i Note**
>
> When you copy and paste the graph, the confidence interval information are not made available. To get this information, you can go to *Tables Debriefing* > *Error Bars*. The interval lower bound equals the signal value minus two times the value of the error bar and the upper bound equals the signal value plus two times the value of the error bar.

5.3.4.4.2.2 Displaying the Apply Results Graph

The panel *Apply Settings* allows you to graphically preview the results of an application of your model on a new dataset.

> **i Note**
>
> Be aware that this feature does not generate any output files apart from the graph. To generate an output file, see Step 4 - Applying the Model (see "Applying the Model").

1. Select the tab *Apply Setting*. The following panel is displayed.
2. Select the data source type in the *Data Type* drop-down list (Text Files, Data Base, Flat Memory, ...).
3. In the *Folder* field, select the folder or data base where the apply data is located.
4. In the *Data* field, select the file or table containing the apply data.
5. In the section *Forecast Parameters*, set the number of forecasts to use for the application. As an indication, the number of forecasts used to build the model is displayed in a non-modifiable field.
6. Click the *Display* button to visualize the graph resulting from the application of the model on the new data.



5.3.4.4.2.3 Understanding the Forecast Plot

The Forecast plot allows you to visualize five types of information:

- The signal
- The predicted signal
- The trend
- The error bars
- The outlier

The following table describes each graphical element and its signification:

Element	Symbolized by...	Represents...
`Signal`	The green curve	The information contained in the training dataset
`Predicted Signal`	The blue curve	The signal predicted by the generated model
`Trend`	The red curve	The signal trend. This curve is only displayed if the trend is a polynomial or a linear. For more information on the trend, see section The signal components (on page 34)

Element	Symbolized by...	Represents...
`Error Bars`	The blue area around the end of the blue curve	The zone of possible error where the predicted signal could be. The error bars are only displayed for the forecasts. Note - The error bars are equal to twice the standard deviation computed on the validation dataset.
`Outlier`	A red square	A point where the predictive curve is very distant from the real curve. Note - An outlier is detected when the absolute value of the residuals is over three times the value of the standard deviation computed on the training dataset.

As long as the original signal is displayed, you can measure the accuracy of the predicted signal against the original one. When the original signal stops, the error bars allow you to measure the level of confidence of the predicted signal. The error bars are not displayed further than the first forecast for which the model cannot guarantee the accuracy of the predicted signal. In other words, when the model cannot evaluate if its prediction is correct, it stops displaying the error bars.

> **i Note**
>
> If you have selected the *Sequential* partition strategy, the error bars are also displayed on the test part of the signal.

5.3.4.4.2.4 Zooming the Plot In/Out

1. Right-click the plot area you want to zoom in or out.

   A contextual menu appears.
2. Select the type of zoom you want to apply.
3. Select on which axes you want to zoom. Note that the point where you click is the central point of the zoom.

5.3.4.4.2.5 Displaying the Value of a Specific Element of a Signal

Place your cursor on a selected point of a signal curve.

A pop-up displays the information for this point of the signal.

5.3.4.4.3 Signal Components

5.3.4.4.3.1 Displaying the Signal Components

1. On the *Using the Model* menu, click the option *View Signal Components*.
2. A pop-up is displayed asking you to confirm or update the name and location of the training dataset file.
3. Update the information if you have renamed or moved the training dataset file, or if its type has been changed.

   > **i Note**
   >
   > Steps 2 and 3 are required in case you open a saved model and the dataset used to train it has been moved, for example.

4. Click *OK*.

   The screen *View Signal Components* appears.
5. In the list *Display Options*, select the component you want to display.

5.3.4.4.3.2 Understanding the Signal Components

The signal components are ordered as listed below:

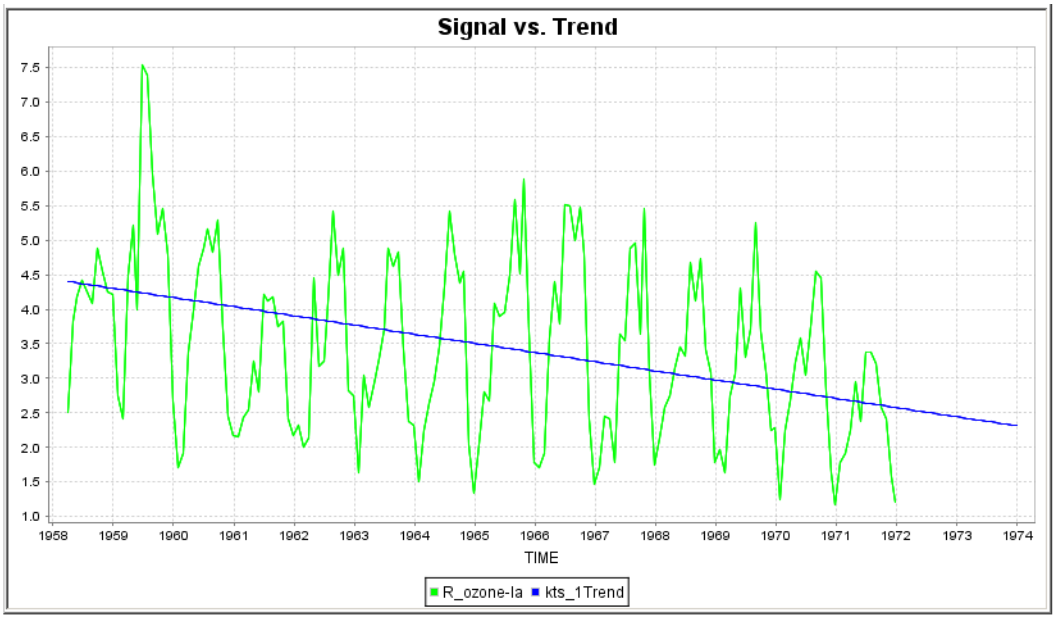
1. the trend,
2. the periodics,
3. the fluctuation
4. the residuals,
5. the outliers.

To display a component, time series removes the previous existing components from the signal. For example, to display the fluctuation, time series removes the trend and the periodics from the signal.

More information on the signal components available in Understanding the Model Debriefing: Signal Components.

Signal vs. Trend

For this Scenario, Time series has recognized a descending linear trend, it appears in blue on the plot *Signal vs. Trend*.



The detailed explanation of the trend can be found in *Understanding the Model Debriefing* > *Signal Components* > *The Trend* (see "The Trend").

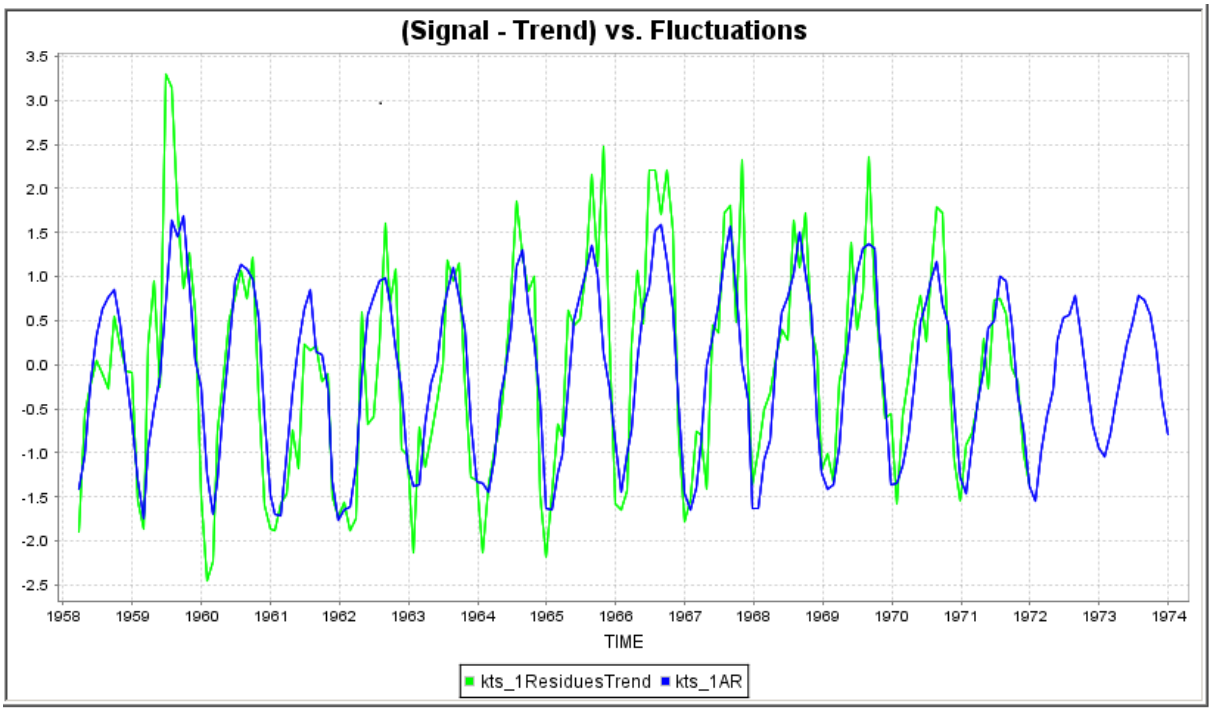
Signal vs. Periodics

For this scenario, the model has not found any periodic. A detailed explanation of the periodics can be found in section *Understanding the Model Debriefing* > *Signal Components* > *The Cycles*

Signal vs. Fluctuation

For this scenario, time series has detected fluctuations that are represented in blue in the plot `(Signal-Trend) vs. Fluctuation`.

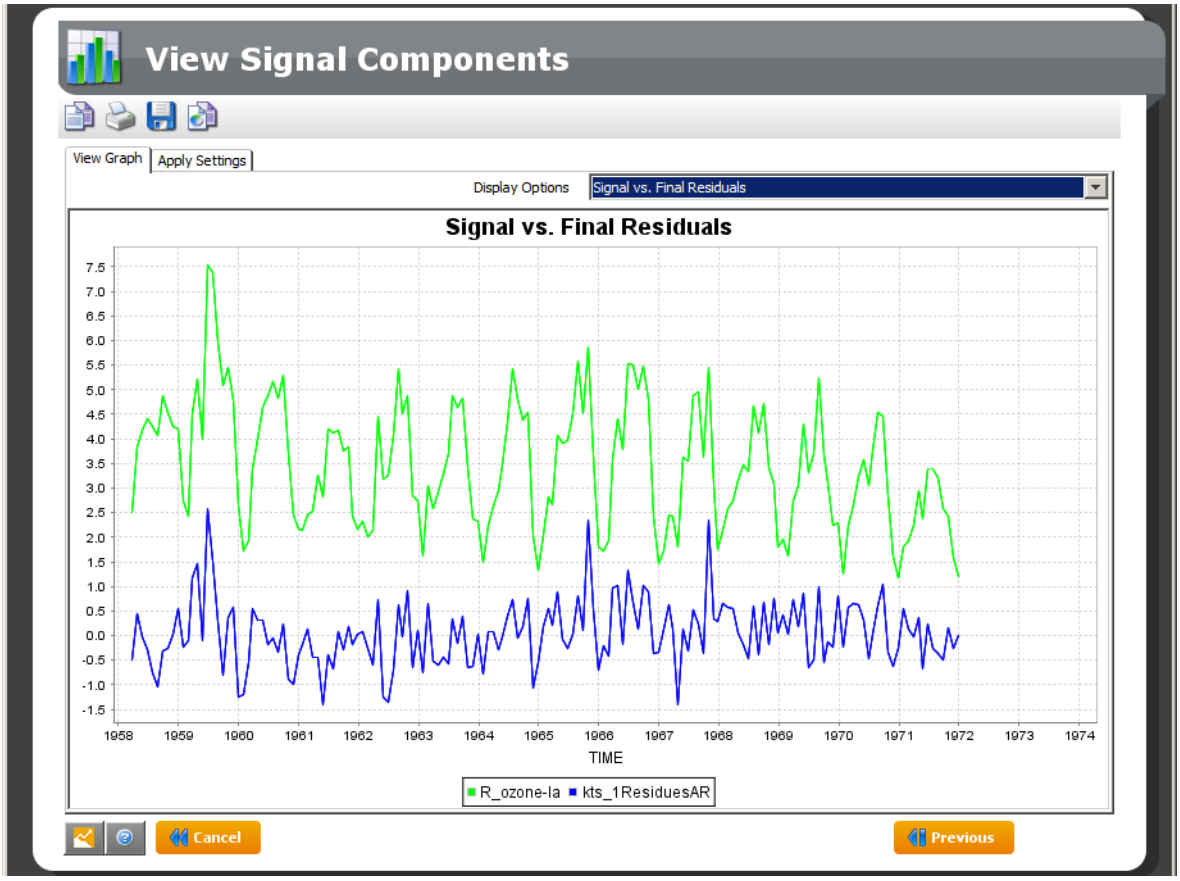
> **i Note**
>
> You can see that the trend has been removed from the signal to allow better visualization of the fluctuations.



A detailed explanation of the fluctuation can be found in section *Understanding the Model Debriefing* > *Signal Components* > *The Fluctuation* (see "The Fluctuation").

Signal vs. Residuals

For this scenario, the residuals found by the model appear in blue on the plot `Signal vs. Final Residuals`.



A detailed explanation of the residuals can be found in section *Understanding the Model Debriefing* > *Signal Components* > *The Residuals* (see "The Residuals").

5.3.4.4.4 Regressions: Contribution by Variables

The `Contributions by Variables` plot allows you to examine the relative significance of each of the variables in the regressions used in the model. The variables and the target variable used in the regression depends on the component being modeled. Two components can be modeled using a regression: the trend and the fluctuation.

The following table details which generated variables and which target can be used for the regressions:

Modeled Component	Possible Target Variable	Variables Used for the Regression
Trend	• signal	functions of the date, that is Time, square(Time), sqrt(Time) and the extra predictable variables.
Fluctuation	• signal • signal - trend • signal - trend - cycles	lag variables on the target

The following four types of plots allow you to visualize contributions by variables :

- Variable Contributions
- Variable Weights
- Smart Variable Contributions
- Maximum Smart Variable Contributions

The plot...	Presents...
`Variable Contributions`	The *relative importance* of each variable in the built model
`Variable Weights`	The *weights* (in the final polynomial) of the normalized variables
`Smart Variable Contributions`	The variables *internal* contributions
`Maximum Smart Variable Contributions`	The variables internal contributions including only the *maximum of similar variables*. For example, only binned encoding of the continuous variable age will be displayed

5.3.4.4.4.1 Displaying the Variable Contributions

In this scenario, the model contains only the regression *linear(TIME)* which defines the trend.

1. On the screen *Using the Model*, click the option *Regressions: Contributions by Variables*.

   The screen Contributions by Variable is displayed.
2. Select the type of plot you want to display in the drop-down list *Chart Type*. The plot *Maximum Smart Variables* is displayed by default.

   > **i Note**
   >
   > In case of a regression on one variable only, the plot *Maximum Smart Variables* is not available. Use the drop-down list *Chart Type* to select another plot.
3. Select the regression you want to analyze in the *Models* drop-down list. Note that if there is only one regression in the model, the *Models* drop-down list is not displayed.

5.3.4.4.5 Statistical Reports

To help you analyze your modeling results and to enable you to possibly share these results with your colleagues, managers, partners or PUBLICs, time series provides you with a set of statistical reports in various formats.

There are three categories of reports:

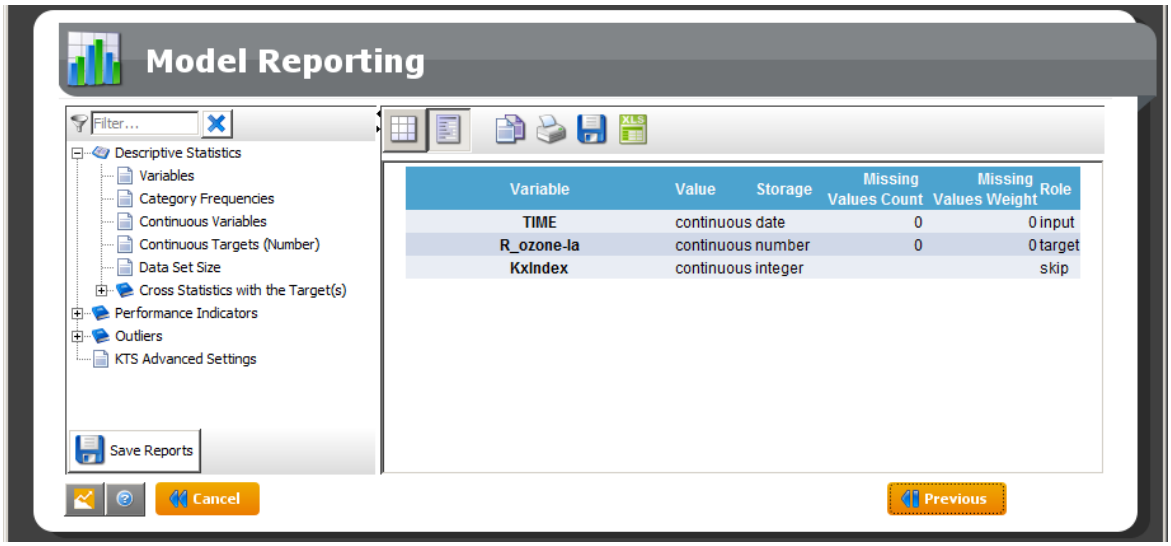
- the *Descriptive Statistics*, which provide information on the variables used to generate the model, such as the variables types and categories, the dataset size, the cross-statistics...
- the *Performance Indicators*, which provide information on the performance of the model thanks to various indicators such as the forecasts error bars and efficiency, the U2, the standard deviation, ...

- the *Cyclic Variables*, which provide you with an analysis of the seasonal and cyclic variables displayed as graphs.

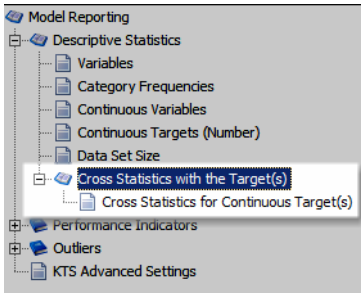
5.3.4.4.5.1 Displaying the Statistical Reports

1. On the screen *Using the Model*, click the option *Statistical Reports*.

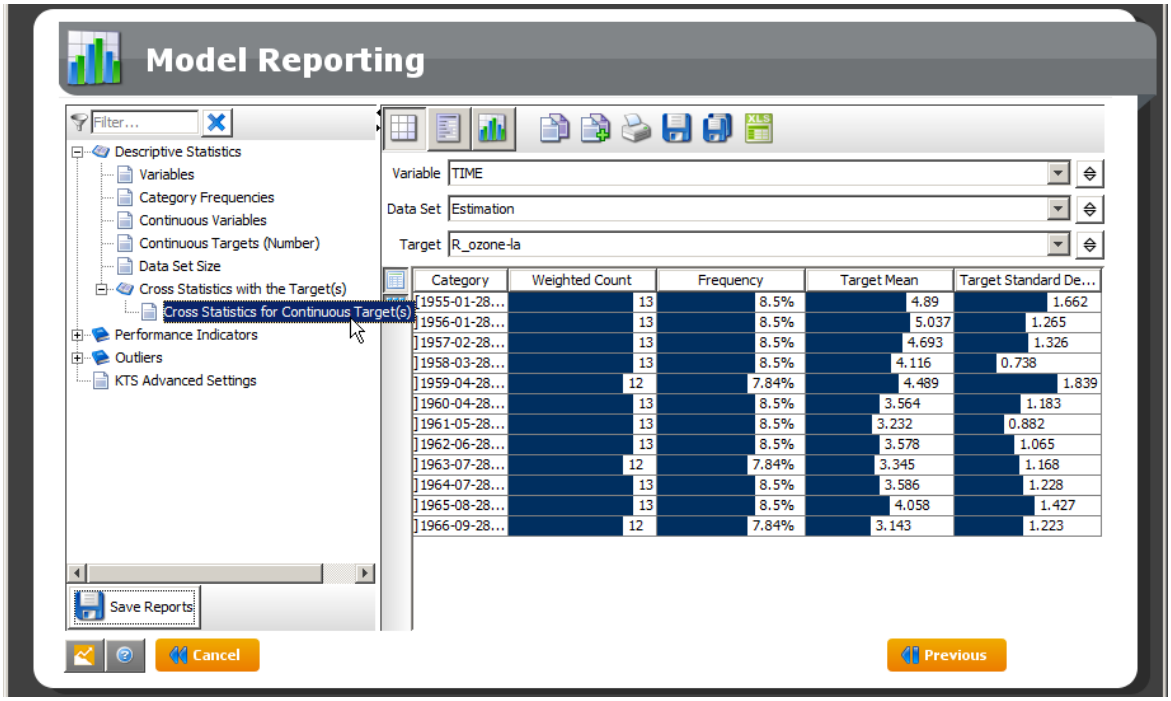
   The screen *Model Reporting* is displayed.



2. Click the + displayed in front of a category of reports to display the available reports.



3. Double-click the desired report to display it in the right frame.



5.3.4.4.5.2 Statistical Report Options

A toolbar is provided allowing you to modify how the current report is displayed, to copy the report, to print it, to save it or to export it to Excel.

Display Options

Menu	Option
View	Display the current report view in the graphical table that can be sorted by column.
	Display the current report view as an HTML table.
	Some reports can be displayed as a bar chart. This bar chart can be sorted by ascending or descending values, or by ascending or descending alphabetical order. You can also select which data should be displayed.
	Display certain reports as a pie chart.
	Display certain reports as a line chart.

Menu	Option
Sort	When the current report is displayed as a bar chart, this option allows you to change the orientation of the bars (from horizontal to vertical and vice versa).
	Display the current report with no sorting.
	Sort the current report by ascending values.
	Sort the current report by descending values.
	Sort the current report by ascending names.
	Sort the current report by descending names.
Series	Select which columns to display for current report.

Usage Options

Menu	Option
	Copy the data from the current view of the displayed report. The data can then be pasted in a text editor, a spreadsheet, a word processing software.
	Print the current view of the selected report depending on the chosen display mode (HTML table, graph, ...).
	Save under different formats (text, html, pdf, rtf) the data from the current view of the selected report.
	Save under different formats (text, html, pdf, rtf) the data from all the views of the selected report.
	Export to Excel.
	Save all reports.
	Save the customized style sheet.

5.3.4.5 Step 4 - Using the Model

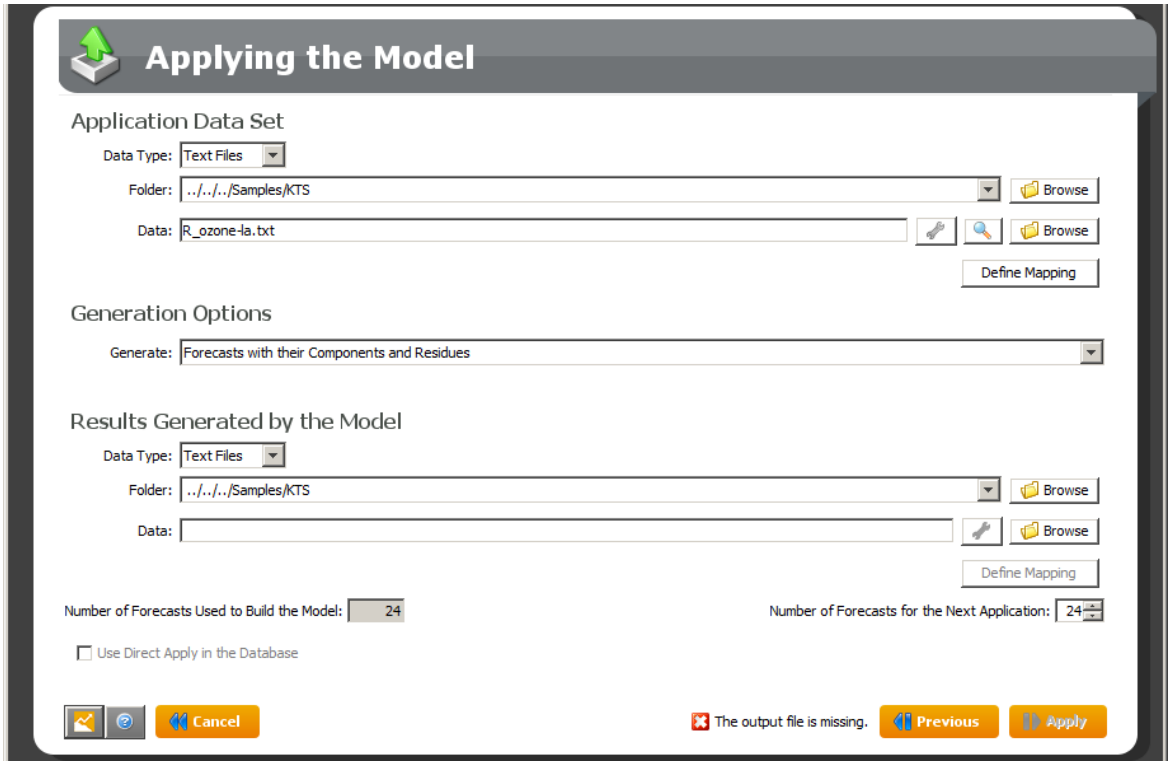
Once generated, the model may be:

- Applied to additional datasets. The model thus allows you to perform predictions on these application datasets, by predicting the values of a target variable.
- Saved for later use.

5.3.4.5.1 Applying the Model

1. On the screen *Using the Model*, click *Apply Model*.

   The screen *Applying the Model* appears.



2. In the section *Application Dataset*, the information concerning the dataset is already filled. See the section Application Dataset Requirements for further information.
3. In the section *Generation Options*, select the type of output you want (see section Type of Results Available (see "Understanding the Applying Mode")).
4. Check the option View Generated Outputs to also display the apply results in the interface.
5. In the section *Results Generated by the Model*, select the file format for the output file (*Text files, Database, Flat Files*).
6. Click the *Browse* button to select:
   - In the *Folder* field, the folder in which the result file will be saved,
   - In the *Data* field, the name of the result file.

7. Click the *Apply* button.
8. Click the Next button.

   The screen *Using the Model* appears. Once application of the model has been completed, the results files of the application are automatically saved in the location that you had defined from the screen *Applying the Model*.

5.3.4.5.1.1 Application Dataset Requirements

The dataset used to apply a time series model is generally the same used for training the model. Applying a time series model produces a similar output dataset with extra columns and/or rows containing the requested forecasts.

It is also possible to apply a model to a different dataset provided if the following conditions are fulfilled:

- the application dataset must contain:
  - the target variable.
  - all the input variables from the training dataset (that is, all the variables that have not been excluded during the variables selection step).
  - all the key variables from the training dataset (except for the key variables automatically generated by the application, such as KxIndex).
- the date column must be sorted (strictly increasing, Order Level = 1 for ODBC sources).
- the first date of the application dataset must be present in the time window defining the training dataset. For example, for the ozone model, a dataset ozone without the first 10 rows is a valid application dataset while a dataset starting with the date value '1973-03-01' is not (since this date is not contained in the training dataset, which ends on '1971-12-28').

Type of Results Available

In the *Generate pull-down* menu you can choose to generate three types of results:

If you choose the option...	The result file will contain...
`Predicted Values Only`	• all input variables, • the predicted variables, that is the forecasts for every date of the training dataset.
`Forecasts with their Components`	• all input variables, • the predicted variables, that is the forecasts for every date of the training dataset. • the components value (trend, cycles, fluctuation) for each forecast

If you choose the option...	The result file will contain...
`Forecasts with their Components and Residues`	• all input variables, • the predicted variables, that is the forecasts for every date of the training dataset. • the components value (trend, cycles, fluctuation) for each forecast • the remaining values (residue) obtained after extracting each component from each forecast
`Only First Forecasts Column and their Error Bars`	• all input variables, • the first predicted variable, that is the first forecast for every date of the training dataset. • the error bars for the predicted variable

5.3.4.5.1.2 Understanding the Application Mode

A time series model can only be applied on all or part of the training dataset.

The result file contains the input variables, that is, the time and the signal, and as many predicted variables, noted `KTS_x`, as the number of forecasts requested.

The following table describes a time series result file, where `TIME` is the time variable and `R_ozone-la` is the signal variable.

TIME	KxIndex	R_ozone-la	kts_1	kts_2	kts_3
...					
1959-01-28					
1959-02-28					
1959-03-28					
1959-04-28	52	5.210	4.320	4.250	5.229
1959-02-28	53	4.000			
1959-03-28	54	7.540			
1959-04-28					
...					

Legend: ← is an estimator of

- `KTS_1` is an estimator of the current value of the signal (at date t)
- `KTS_2` is an estimator for the next value of the signal (at date t+1)
- `KTS_3` is an estimator for the value of the signal at date t+2,
- and so on ...

A time series model uses its own previous predictions to compute the next ones. The values used to compute a forecast are presented below.

- KTS_1 is computed using the current date (known as t), the known extra predictable variables at this date, and past values of the signal:

TIME	KxIndex	R_ozone-la	kts_1	kts_2	kts_3
…					
1959-01-28	49	2.750			
1959-02-28	50	2.420			
1959-03-28	51	4.500			
1959-04-28	52	5.210	4.320	4.250	5.229
1959-02-28	53	4.000			
1959-03-28	54	7.540			
1959-04-28					
…					

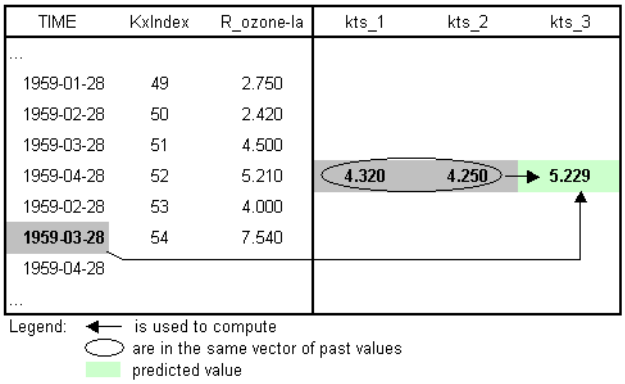
Legend: is used to compute
are in the same vector of past values
predicted value

- `KTS_2` is computed using date t+1, known values for extra predictable variables at this date, and past values of the signal. However since the signal value at date t is unknown, time series uses the last prediction, that is `KTS_1`, to compute `KTS_2`. The following table shows how `KTS_2` is computed using KTS_1:

TIME	KxIndex	R_ozone-la	kts_1	kts_2	kts_3
…					
1959-01-28	49	2.750			
1959-02-28	50	2.420			
1959-03-28	51	4.500			
1959-04-28	52	5.210	4.320	4.250	5.229
1959-02-28	53	4.000			
1959-03-28	54	7.540			
1959-04-28					
…					

Legend: is used to compute
are in the same vector of past values
predicted value

- `KTS_3` is computed using date t+2, known values for extra predictable variables at this date, and past values of the signal. However since the signal value at date t and t+1 are unknown, time series uses the last two predictions, that is `KTS_1` and `KTS_2`, to compute `KTS_3`. The following table shows how `KTS_3` is computed using `KTS_1` and `KTS_2`:

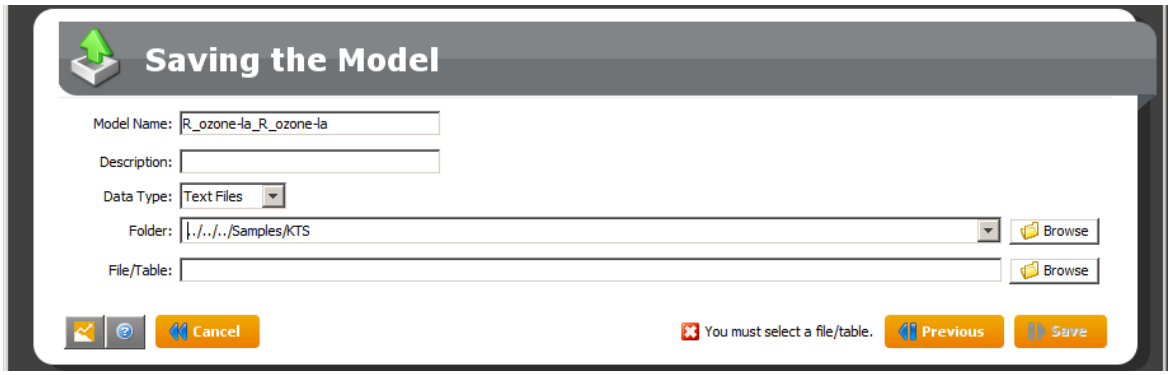


5.3.4.5.2 Saving the Model

Once a model has been generated, you can save it. Saving it preserves all the information that pertains to that model, that is, the modeling parameters, the forecasts view, and so on.

1. On the screen *Using the Model*, click the option *Save Model*.

   The screen *Saving the Model* appears.



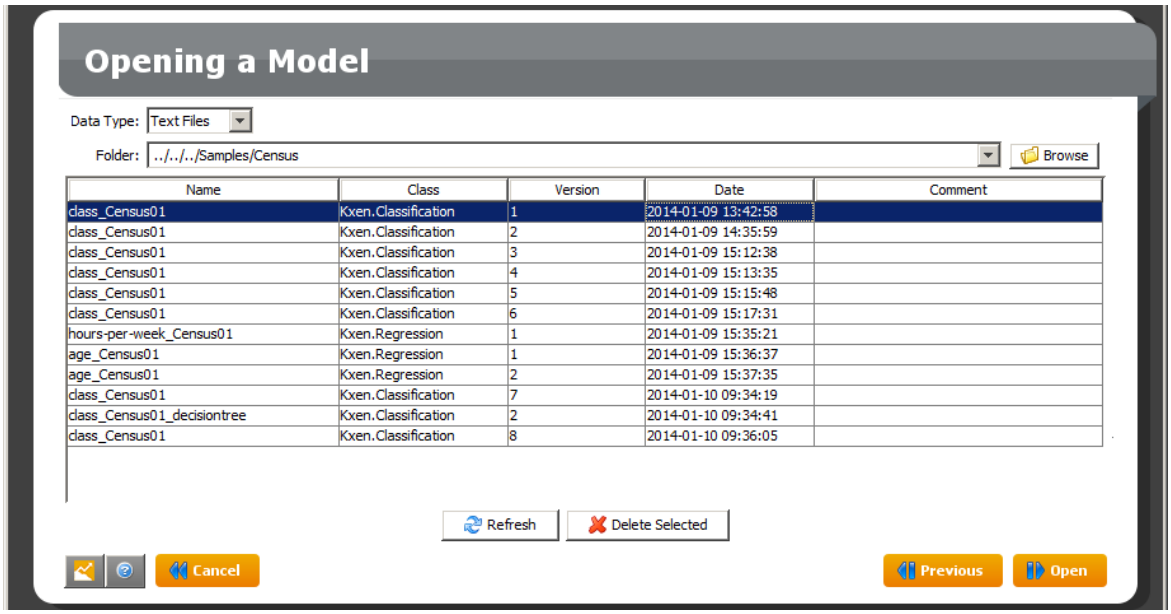
2. Above the *Browse* button, select one of the following options:
   - *Text files*, to save the model in a text file.
   - *Database*, to save the model in a database.
   - *Flat Memory*, to save the model in the active memory.
3. Complete the following fields:
   - *Model Name*: This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
   - *Description*: This field allows you to enter the information of your choosing, such as the name of the training dataset used or the number of forecasts calculated. This information will help you identify your model for a later use.
   - *Folder*: Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
   - *File/Table:* This field allows you to enter the name of the file or table that is to contain the model. The name of the file must contain one of the following format extensions .txt (text file in which the data is separated by tabs) or .csv (text file in which the data is separated by commas).

5.3.4.5.3 Opening a Model

Once saved, models may be opened and reused in Automated Analytics.

1. On the main screen of SAP Predictive Analytics, select *Modeler*, and then *Load a Model*.

   The screen *Opening a Model* appears.



Name	Class	Version	Date	Comment
class_Census01	Kxen.Classification	1	2014-01-09 13:42:58	
class_Census01	Kxen.Classification	2	2014-01-09 14:35:59	
class_Census01	Kxen.Classification	3	2014-01-09 15:12:38	
class_Census01	Kxen.Classification	4	2014-01-09 15:13:35	
class_Census01	Kxen.Classification	5	2014-01-09 15:15:48	
class_Census01	Kxen.Classification	6	2014-01-09 15:17:31	
hours-per-week_Census01	Kxen.Regression	1	2014-01-09 15:35:21	
age_Census01	Kxen.Regression	1	2014-01-09 15:36:37	
age_Census01	Kxen.Regression	2	2014-01-09 15:37:35	
class_Census01	Kxen.Classification	7	2014-01-10 09:34:19	
class_Census01_decisiontree	Kxen.Classification	2	2014-01-10 09:34:41	
class_Census01	Kxen.Classification	8	2014-01-10 09:36:05	

2. Select one of the following Data Type options:
   - Text files
   - Database,
   - Flat files,

   depending upon the format of the model that you want to open.
3. Click the *Browse* button.

   A selection dialog box appears.
4. Select the folder that holds the model that you want to open.

   The list of models contained in that folder appears.

   The following table lists the information provided for each model allowing to identify the model you want to reload.

Column	Description	Values
`Name`	Name under which the model has been saved	Character string

`Class`	Class of the model, that is the type of the model	○ Kxen.Classification: Classification/Regression with nominal target ○ Kxen.Regression: Classification/Regression with continuous target ○ Kxen.Segmentation : Clustering with SQL Mode ○ Kxen.Clustering : Clustering without SQL Mode ○ Kxen.TimeSeries: Time Series ○ Kxen.AssociationRules: Association Rules ○ Kxen.SimpleModel : Classification/Regression and Clustering multi-target models, any other model
`Version`	Number of the model version when the model has been saved several times	Integer starting at 1
`Date`	Date when the model has been saved	Date and time in the format yyyy-mm-dd hh:mm:ss
`Comment`	Optional user defined comment that can be used to identify the model	Character string

5. Select a model from the list.
6. Click the *Open* button.

   The *Using the Model* menu appears.

5.3.5 Scenario 2: Modeling with Extra Predictable Variables

This section details:

- what are the extra inputs variables for time series,
- how to use these extra variables,
- what is the impact of this feature on the modeling.

5.3.5.1 Presentation

In Forecasting modeling, extra variables are exogenous factors that may have an influence on the modeling. These variables can be ordinal, binary or continuous.

These extra variables are Predictable variables as their future values are known (like the first Friday of the month, the first working day of the month, and so on). This type of variable can contain additional information, which can be very useful for the trend and/or the cyclic analysis.

The predictable variable is the subject of this section.

5.3.5.2 Standard Modeling

Summary of the Modeling Settings to Use

In this step, you will follow the default scenario without using any extras predictable variables (see section Standard Modeling with Time Series).

The table below summarizes the modeling settings that you must use. It should be sufficient for users who are already familiar with the KJWizard.

For detailed procedures and more information, see the following sections.

Replace the options given in the scenario by the following ones:

Task(s)	Screen	Settings
• Specifying the data source • Selecting a partition strategy	`Data to be Modeled`	• In the field *Dataset* , select the file `CashFlows.txt` • Partition strategy : sequential without testing
Specifying a description file	`Data Description`	In the field *Description* , select the file `KxDesc_CashFlows.txt`
Defining the extra predictable inputs number	`Selecting Variables`	Select and exclude all the variables from the field *Predictable Variables Kept*
Defining the Forecasts Number	`Summary of Modeling Parameters`	In the field Number of Forecasts , enter 20

Selecting a Data Source and a Partition Strategy

For this Scenario:

- Use the file CashFlows.txt as the training dataset.
- Select the partition strategy Sequential without testing.

For the detailed procedures, refer to sections Selecting a Data Source and Selecting a Partition Strategy.

Describing the Data

For this Scenario, use the description file `KxDesc_CashFlows.txt`.

For the detailed procedure, refer to section Describing the Data Selected.

Selecting Variables

For this Scenario:

- Keep *Date* as the time variable.
- Keep *Cash* as the target variable.
- Exclude all extra predictable variables.
- Do not select a weight variable.
- Check that the last training line is set at 251.

On the screen *Displaying the signal*, select the option *Date in the Time list*.

For detailed procedures, refer to section Selecting Variables.

Defining the Forecasts Number

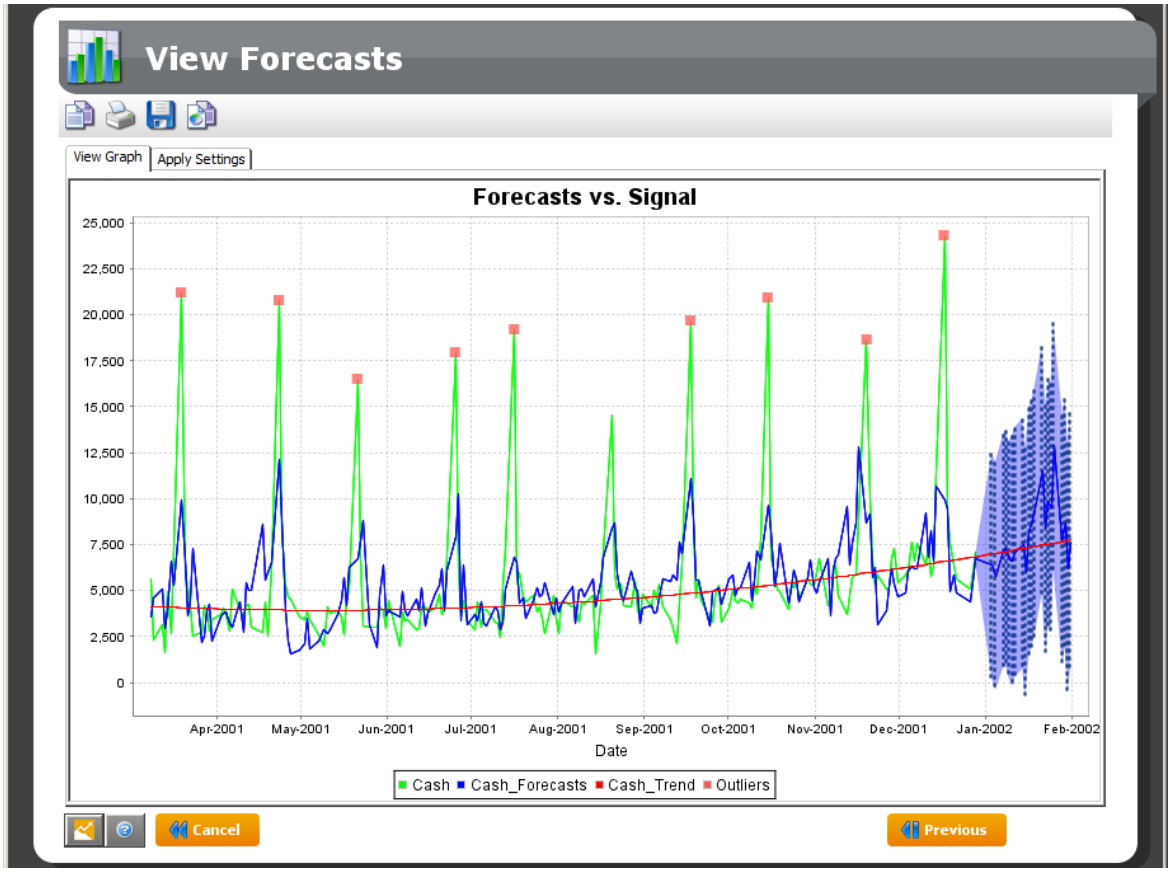
For this Scenario, define the number of forecasts to 21. This number corresponds to the average number of days worked in one month.

For the detailed procedure, refer to section Defining the Number of Forecasts.

5.3.5.2.1 Viewing the Corresponding Forecasts

For this Scenario, the model obtained has the following form: `Model = AR(37) + polynom( Date) + R`.

1. On the menu *Using a Model*, click the option *View Forecasts*.
2. Click the *Next* button, the screen *View Forecasts* appears.



5.3.5.3 Modeling with Extra Predictable Inputs

The following section will show how extra predictable variables can increase the performances on the current dataset.

Summary of the Modeling Settings to Use

In this step, you will execute Scenario 2 using extra predictable variables. The table below summarizes the modeling settings that you must use. It should be sufficient enough for users who are already familiar with SAP Predictive Analytics.

For detailed procedures and more information, see the following sections.

Replace the options given in the scenario by the following ones:

Task(s)	Screen	Settings

• Specifying the data source • Selecting a partition strategy	*Data to be Modeled*	• In the field *Dataset* , select the file `CashFlows.txt` • *Partition strategy* : sequential without testing
Specifying a description file	*Data Description*	• In the field *Description* , select the file `KxDesc_CashFlows.txt`
Defining the extra predictable inputs	*Selecting Variables*	• Keep all the variables in the field *Predictable Variables Kept*
Defining the Forecasts Number	*Summary of Modeling Parameters*	• In the field *Number of Forecasts* , enter 21

Selecting a Partition Strategy and a Data Source

For this Scenario:

- Use the file `CashFlows.txt` as the training dataset.
- Select the partition strategy *Sequential without testing*.

For the detailed procedures, refer to sections Selecting a Data Source and Selecting a Partition Strategy.

Describing the Data

For this Scenario, use the description file `KxDesc_CashFlows.txt`.

For the detailed procedure, refer to section Describing the Data Selected.

Selecting Variables

The panel *Selecting Variables* allows you to:

- select the time variable,
- select the target variable,
- select a weight variable (optional),
- set the last date to use for training the model,
- select which variables should be kept for the modeling.

For this Scenario:

- Keep *Date* as the time variable.
- Keep *Cash* as the target variable.
- Keep all the extra predictable variables.
- Do not select a weight variable.
- Check that the last training line is set at *251*.

On the screen *Displaying the Signal*:

- Select the option *Date* in the Time list.

For the detailed procedures, refer to section Selecting Variables.

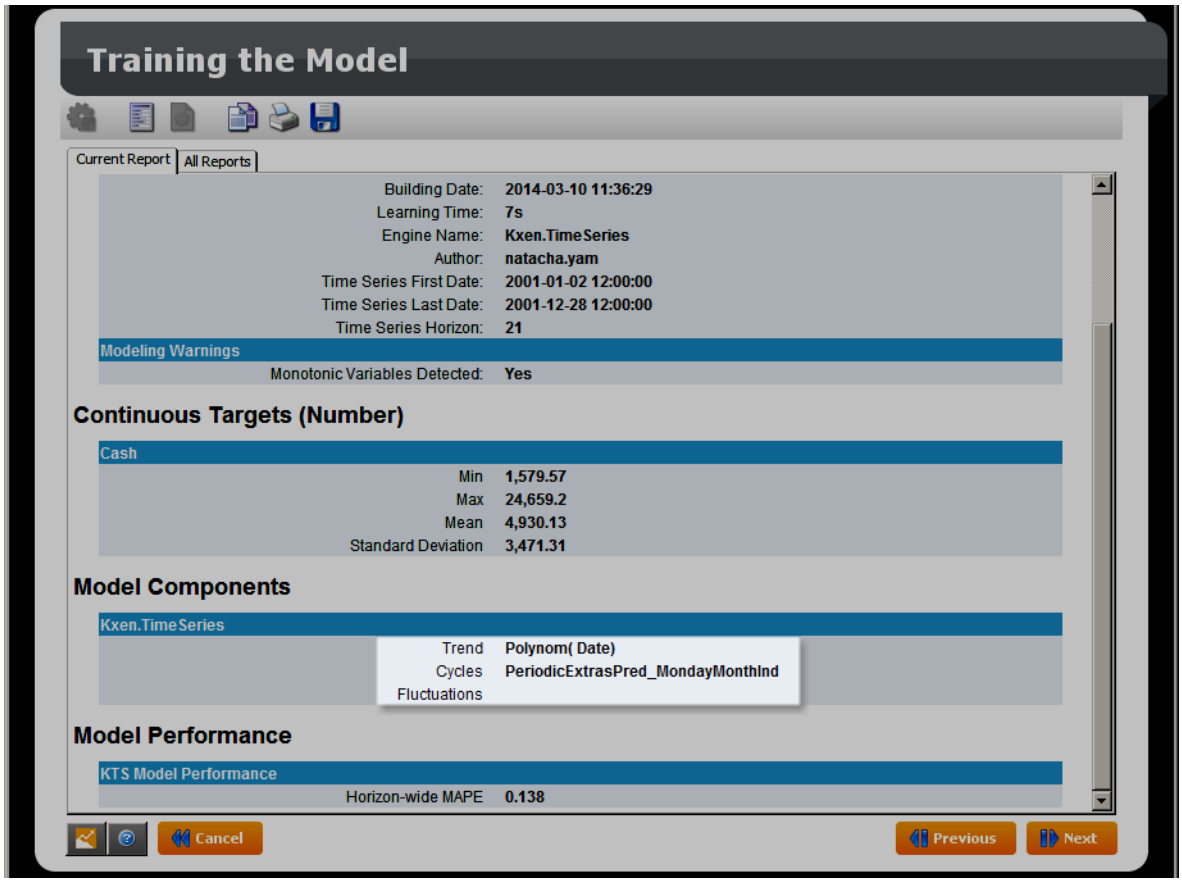
Defining the Forecasts Number

For this Scenario, define the number of forecasts to *21*. This number corresponds to the average number of days worked in one month.

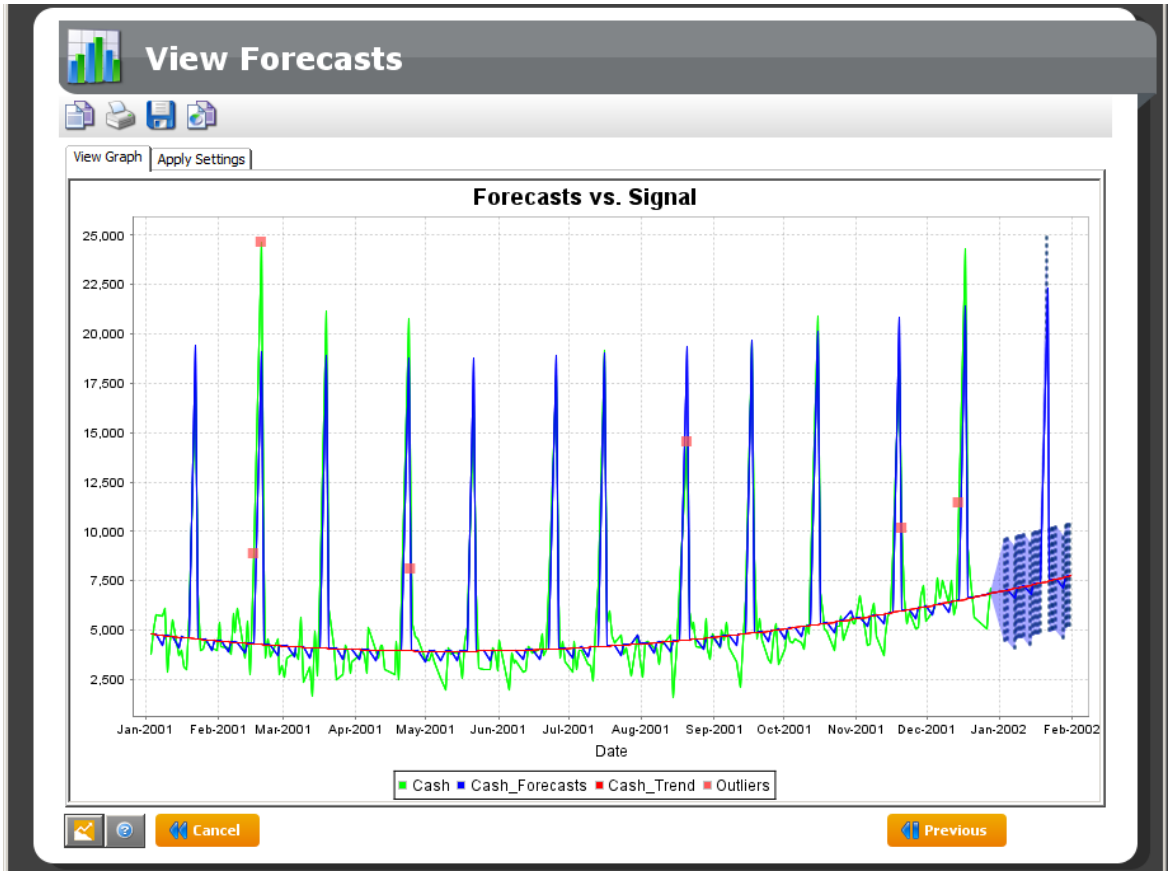
For the detailed procedure, refer to section Defining the Number of Forecasts.

5.3.5.3.1 Viewing the Generated Forecasts

For this Scenario, you get a model which have the following form: *Model = Polynom(Date) + Cyclic(PeriodicExtrasPred_MondayMonthInd) + R*



1. On the menu *Using a Model*, click the option *View Forecasts*.
2. Click the *Next* Button, the screen *View Forecasts* appears.

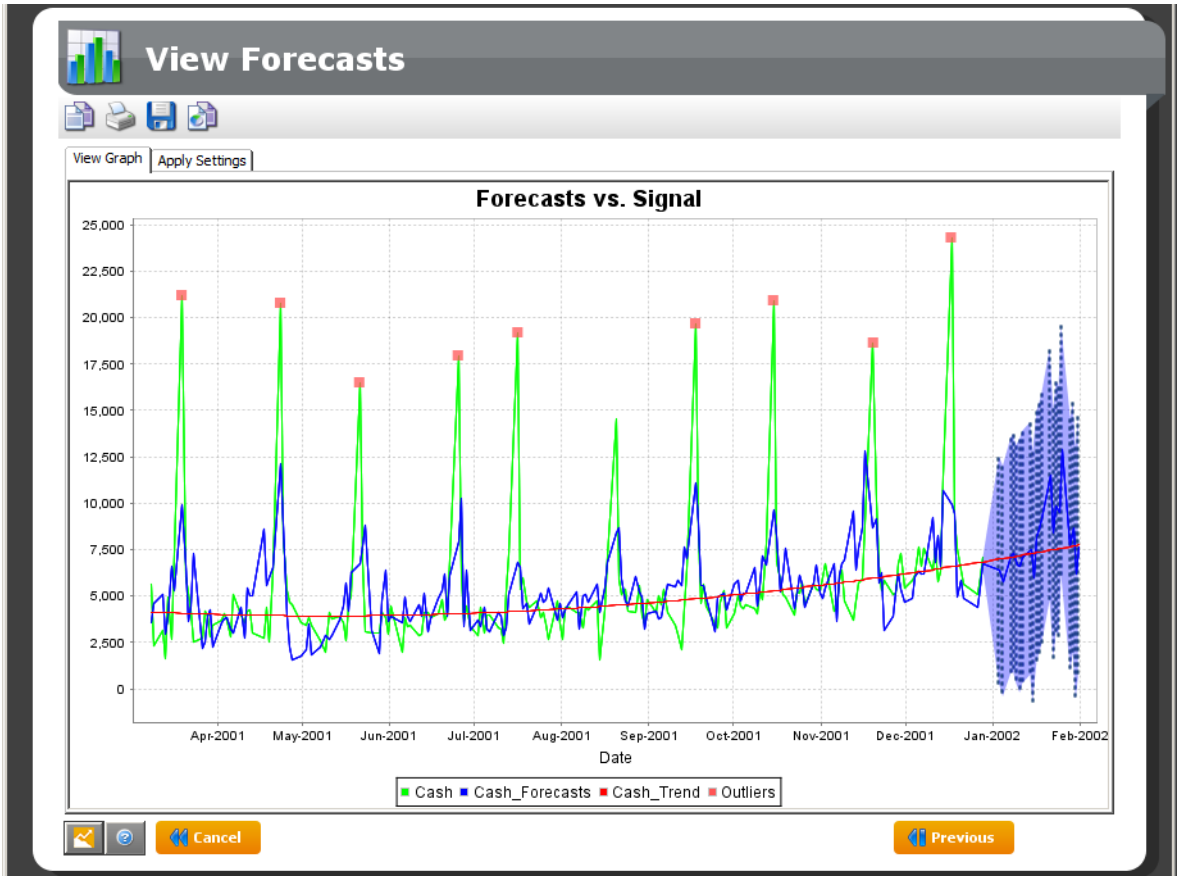


5.3.5.4 Comparing the Forecasts With and Without Extra Predictable Variables

To better understand how extra predictable variables can improve a model, this section will analyze and compare the forecasts obtained with both a standard modeling and a modeling with extra predictable variables.

Understanding Forecasts without Extra Predictable Variables

The following screen displays the forecasts generated by time series when the extra predictable variables have been excluded.



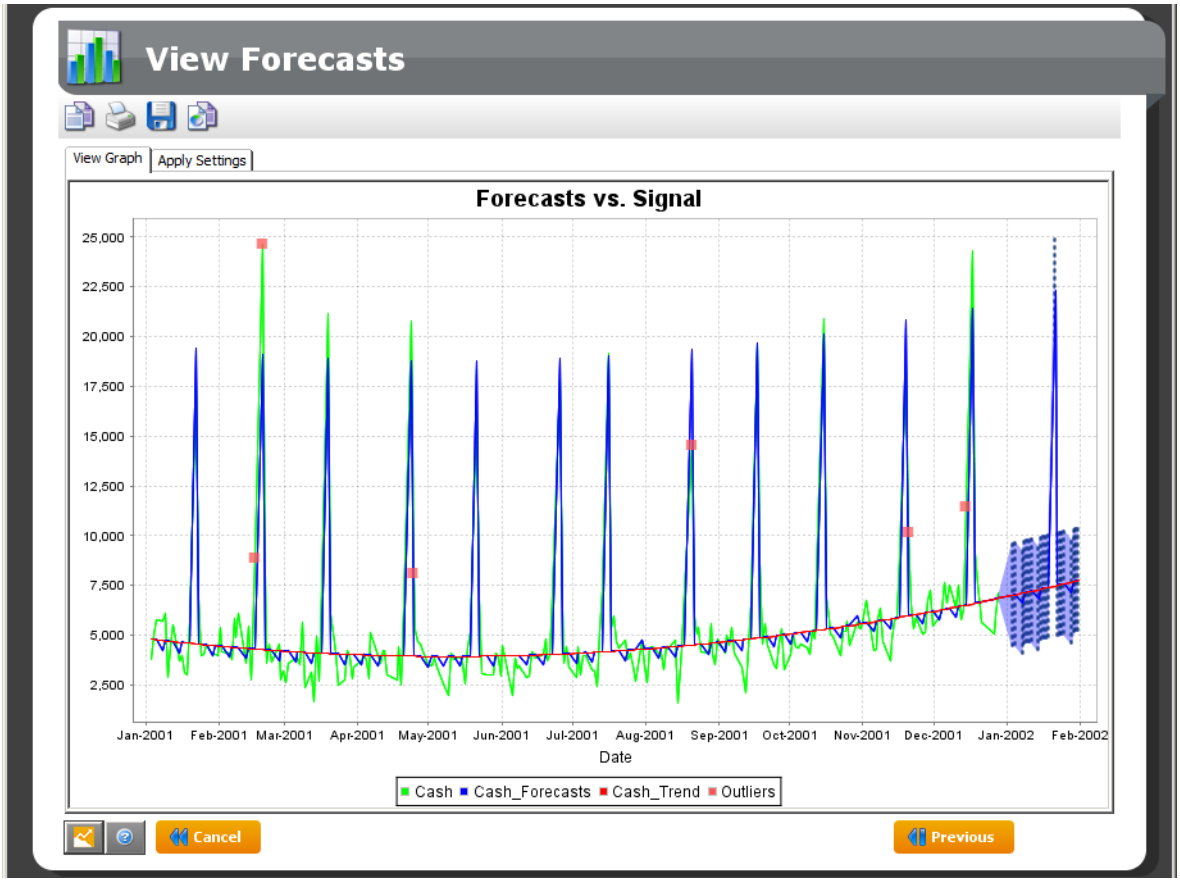
In this model, the engine uses its own variables (cyclics, trend, fluctuations,...) to generate the more predictive model possible. The trend and the picks position are correctly detected but the following points could be improved:

- the error bars are very extended, meaning that the confidence of the model is low,
- the picks amplitude is not correctly forecasted,
- the model appears noisy.

The solution to refine this model is to add extra predictable variables.

Understanding Forecasts with Extra Predictable Variables

The following screen displays the forecasts generated by time series when using extra predictable variables.



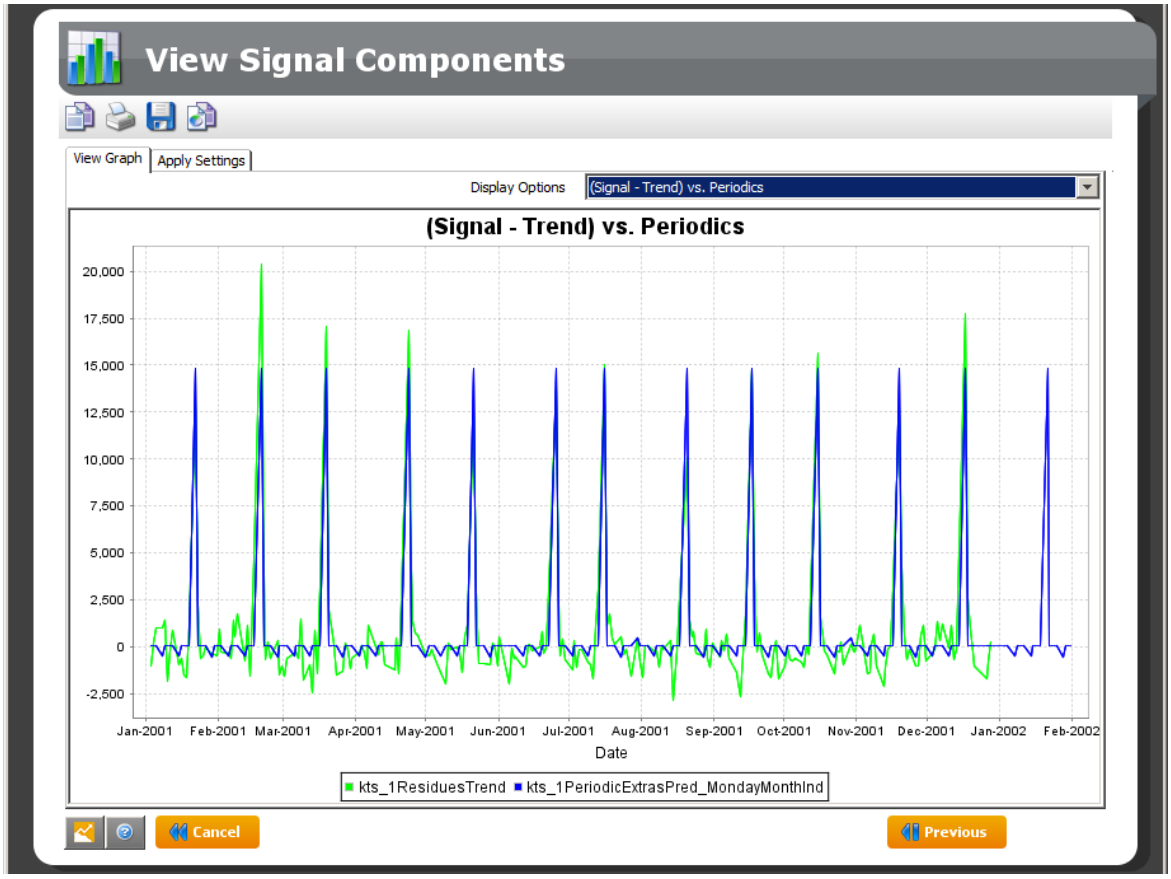
In this scenario, the addition of extra predictable variables has improved the trend detection and therefore the model quality. The three points that needed to be refined in the previous section are improved:

- the error bars are reduced, especially for the forecasted pick,
- the picks amplitude is correctly forecasted,
- the noise has been almost completely attenuated.

The extra predictable variable selected by the application to refine the model is the Monday in month index. It is found in the model definition: Cyclic(PeriodicExtrasPred_MondayMonthInd) + polynom(Date) + R.

5.3.5.4.1 Displaying the Periodics

1. On the *Using the Model*, select the option *View Signal Components*.
2. In the list *Display Options*, select the option *(Signal-Trend) vs Periodics*. The following screen appears.



5.3.5.4.2 Modeling using KxShell scripts

This section details the KxShell script corresponding to Scenario 2.

Creating the time series model

The following code describes the method to create a *Forecasting* model. A default model is created including a Forecasting transform.

For this scenario

```
#creating a model with a Forecasting transform
createModel Kxen.SimpleModel model
model.pushTransformInProtocol Default Kxen.TimeSeries
```

Setting the model parameters

The partition strategy is the only one model parameter, which has to be set.

For this scenario

```
#setting the model's parameters
model.getParameter ""
model.changeParameter "Parameters/CutTrainingPolicy" "sequential with no testing"
model.validateParameter
```

Opening the Training Dataset

In this step, the training dataset `CashFlows.txt` is opened with its description `KxDesc_CashFlows.txt`. As the file contains training information and predictive information, the line index of the end of training is fixed.

For this scenario

```
#open the training dataset
model.openNewStore Kxen.FileStore .
model.newDataSet Training CashFlows.txt
model.readSpaceDescription Training KxDesc_CashFlows.txt
```

```
#set the line index of the end of training
bind model DataSet Training myDataset
myDataset.getParameter ""
myDataset.changeParameter "Parameters/LastRow" 251
myDataset.validateParameter
delete myDataset
```

5.3.5.4.2.1 Setting the time series transform parameter

The Forecasting transform parameters are the following:

- *AutoFeedCount* saves the number of forecasts asked by the user (default = 1).
- *MaxCyclics* indicates the maximum number of cyclicalities, that will be analyzed by time series (default = 450).
- *DateColumnName* saves the name of the date variable (required parameter).
- *ForecastsConnection* gives the format of the forecasts in the output of time series (default = 1). If its value is 1, then the forecasts will be transposed at the end of the KTS_1 variable with the corresponding dates. If its value is 0, then the forecasts stay in the last line of the file.
- *LastRowWithForecastingInformation* saves the index of the last line of the file. This parameter is required if you want to use extras predictable inputs.
- *PredictableExtras* saves the names of the extras predictable variable.
- *ExtraMode*, gives the the format of the output of time series (default = No Extra). `<No Extra>` value is the default format with the KTS_ variables. `<Signal Components>` value is the format which includes with the previous cited variables, each component of each variables: trend, cycles/seasonality, fluctuations.

`<Component Residues>` value is the format, which includes with the previous format the residues after each variable component.

For this scenario

```
#setting KTS basic parameters
bind model TransformInProtocol Default 0 myKTS
myKTS.getParameter ""
myKTS.changeParameter "Parameters/DateColumnName" Date
myKTS.changeParameter "Parameters/AutoFeedCount" 20
myKTS.changeParameter "Parameters/LastRowWithForecastingInformation" 271
myKTS.validateParameter
```

Setting the PredictableExtras parameter

All the variables, except *Date* and *FlowsEmitted*, are added to the *PredictableExtras* parameter.

For this scenario

```
#setting KTS PredictableExtras parameter
myKTS.bindParameter "Parameters/PredictableExtras" extra1
extra1.insert ReverseWorkingDaysIndices dummyPar_1
extra1.insert MondayMonthInd dummyPar_2
extra1.insert TuesdayMonthInd dummyPar_3
extra1.insert WednesdayMonthInd dummyPar_4
extra1.insert ThursdayMonthInd dummyPar_5
extra1.insert FridayMonthInd dummyPar_6
extra1.insert BeforeLastMonday dummyPar_7
extra1.insert LastMonday dummyPar_8
extra1.insert BeforeLastTuesday dummyPar_9
extra1.insert LastTuesday dummyPar_10
extra1.insert BeforeLastWednesday dummyPar_11
extra1.insert LastWednesday dummyPar_12
extra1.insert BeforeLastThursday dummyPar_13
extra1.insert LastThursday dummyPar_14
extra1.insert BeforeLastFriday dummyPar_15
extra1.insert LastFriday dummyPar_16
extra1.insert Last5WDaysInd dummyPar_17
extra1.insert Last5WDays dummyPar_18
extra1.insert Last4WDaysInd ummyPar_19
extra1.insert Last4WDays dummyPar_20
extra1.insert LastWMonth dummyPar_21
extra1.insert BeforeLastWMonth dummyPar_22
delete extra1
myKTS.validateParameter
```

Learning the model

Once all the parameters have been set, the learning phase is launched.

For this scenario

```
#learning the model
model.sendMode learn
```

5.3.5.4.2.2 Saving the model

Once a model has been generated, you can save it. Saving it preserves all the information that pertains to that model, that is, the modeling parameters, its profit curves, and so on. Note that the directory in which the model is saved must exist.

For this scenario

```
#saving the model in the current directory
model.saveModel DefaultBankFlows_Model.txt "With Extras Predictable Inputs and
20 forecasts"
```

Opening an Existing Model

Once saved, models may be opened and reused in Automated Analytics.

For this scenario

```
createStore Kxen.FileStore myRestoreStore
myRestoreStore.openStore .
setDefaultUserPassword "" ""
myRestoreStore.restoreLastModelD model
```

Applying the model

A model generated by time series can be applied ONLY to datasets for which the first date of the "time" variable is located between the first date and the last date of the "time" variable of the training dataset. By default, a model generated by time series is applied to the training dataset.

To apply the model, you have to open a dataset containing the data to use: the ApplyIn dataset. You have to open as well a dataset, that will contain the output of the apply session: the ApplyOut dataset. As the training dataset specifications, you have to set the end of the training section, i.e. the end of the known values of the signal.

By default, the model is applied with the same horizon as the horizon used for training. The user can however apply with a different horizon by setting the parameter AutoFeedCountApplied.

For this scenario

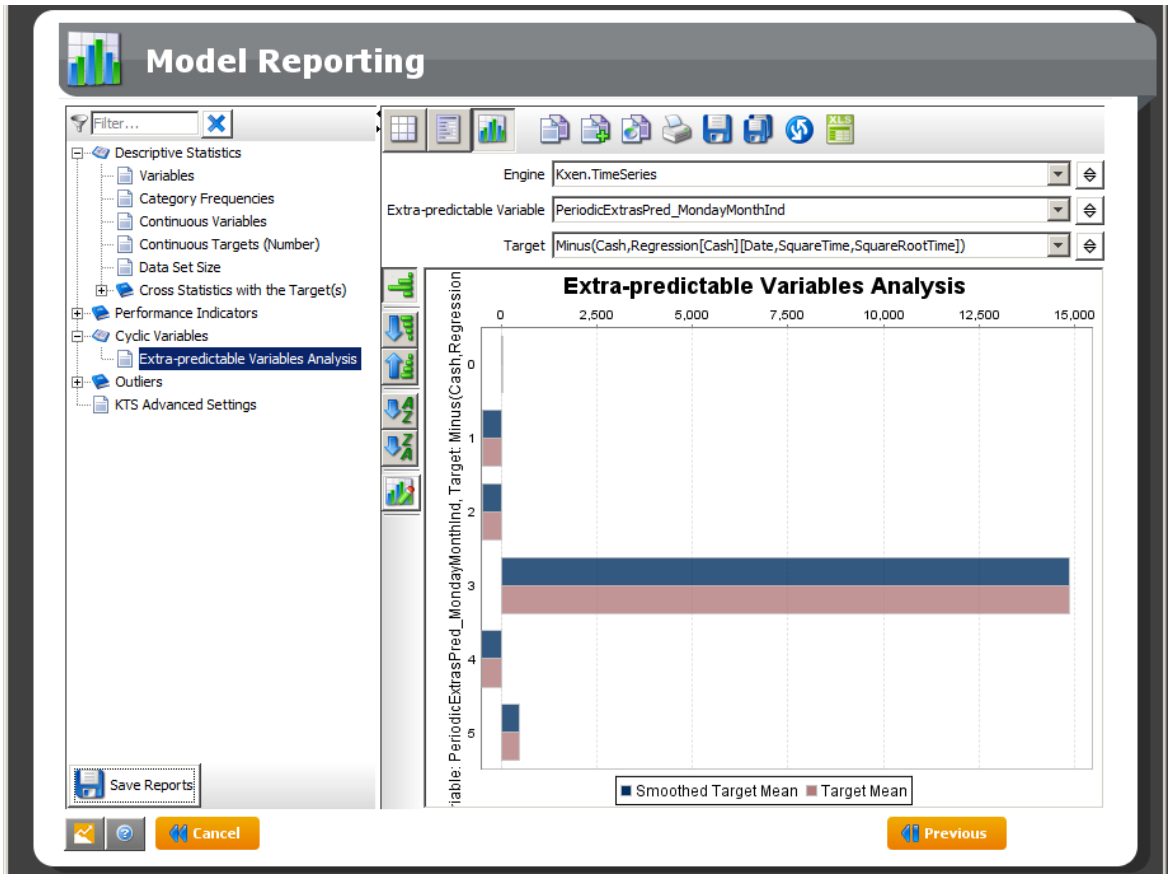
```
#Applying the model on the training dataset: default use.
#open the ApplyIn dataset
model.newDataSet ApplyIn CashFlows.txt
bind model DataSet ApplyIn myDataset
myDataset.getParameter ""
myDataset.changeParameter "Parameters/LastRow" 251
myDataset.validateParameter
delete myDataset
#open the ApplyOut dataset
model.newDataSet ApplyOut out_CashFlows.txt
#apply the model
model.sendMode apply
```

```
#open the ApplyOut dataset
model.openNewStore Kxen.FileStore "Saved"
model.newDataSet ApplyOut out_CashFlows.txt
#apply the model
model.sendMode apply
# apply with a different horizon (=6)
bind model TransformInProtocol Default 0 myKTS
myKTS.getParameter ""
myKTS.changeParameter "Parameters/AutoFeedCountApplied" 6
myKTS.validateParameter
model.sendMode apply
```

5.3.5.4.2.2.1 Displaying Cyclic Details

1. On the panel *Using the Model*, select the option *Statistical Reports*.
2. On the left menu, select the item *Cyclic Variables* > *Extra Predictable Variables Analysis* .

   The following screen appears.



This screen presents the *cyclic PeriodicExtrasPred_MondayMonthInd*, as shown in the list *Extra Variable*. The numbers *1, 2, 3, 4 and 5* represent the index of the Mondays in a month. This plot shows a pick on the index 3, that is on the third Monday of the month.

5.4 Association Rules

5.4.1 About Association Rules

This section of the guide is addressed to people who want to evaluate or use the Automated Analytics Modeler and in particular the Association Rules feature.

Before reading this section, you should read the sections *Classification/Regression* and *Segmentation/Clustering* that present respectively:

- An introduction to Automated Analytics
- The essential concepts related to the use of Automated Analytics features

No prior knowledge of SQL is required to use data manipulation -only knowledge about how to work with tables and columns accessed through ODBC sources. Furthermore, users must have “read” access on these ODBC sources.

To use the Java graphical interface, users need write access on the tables KxAdmin and ConnectorsTable , which are used to store representations of data manipulations.

This part of the guide introduces you to the main functionalities of the Association Rules feature.

Association Rules generates association rules. Association rules provide clear and useful results, especially for market basket analysis. They bring to light the relations between products or services and immediately suggest appropriate actions. Association rules are used in exploring categorical data, also called items.

The strengths of Association Rules are:

- to produce clear and understandable results,
- to support unsupervised data mining (no target attribute),
- to explore very large datasets thanks to its ability to first generate rules on parts of the dataset before aggregating them (exploration by chunks),
- to generate only the more relevant rules (also called primary rules).

How to Use this Section

This section is subdivided into five chapters:

- The current chapter, About Time Series Scenarios, serves as an introduction to the section.You will find information on how to read this section.
- Chapter 2, Association Rules Definitions, provides definitions relative to the Association Rules that will allow you a better understanding of the feature functionalities.
- Chapter 3, Introduction to Sample Files, presents the data files provided to start with Association Rules.
- Chapter 4, Modeling with Association Rules, presents the Association Rules feature. This chapter is organized in five parts:
  - The first part presents the first step of the modeling process, that is the data selection and description.
  - The second part presents the standard use of the Association Rules feature.
  - The third part describes how to use sequences when modeling with Association Rules .

    - The fourth part describes how to save or apply an Association Rules model.
    - The last part provides you with a KxShell script based on the sample files described in chapter 3.
- Chapter 5, Modeling Using KxShell Script, details a KxShell script that can be used with the provided sample files.

5.4.2 Association Rules Definitions

This part gives some definitions relative to the Association Rules.

Session

A session is a set of transactions identified by a unique key; for example, all the purchases done at one time by a single customer.

Transaction

A transaction is defined by:

- A unique key,
- The key of the related session ,
- An attribute, called an item.

Itemset

A group (or a set) of items, is called an itemset.

Association Rule

An Association Rule is an implication relation of the form X => Y. The rule means: if the attribute X is present in a session, then the attribute Y is present too. Two measures allow qualifying the quality of the rule: the Support and the Confidence.

Antecedent

X is called the antecedent of the rule. The antecedent can be composed of an item or an itemset, for example X can be the set {A,B,C}.

Consequent

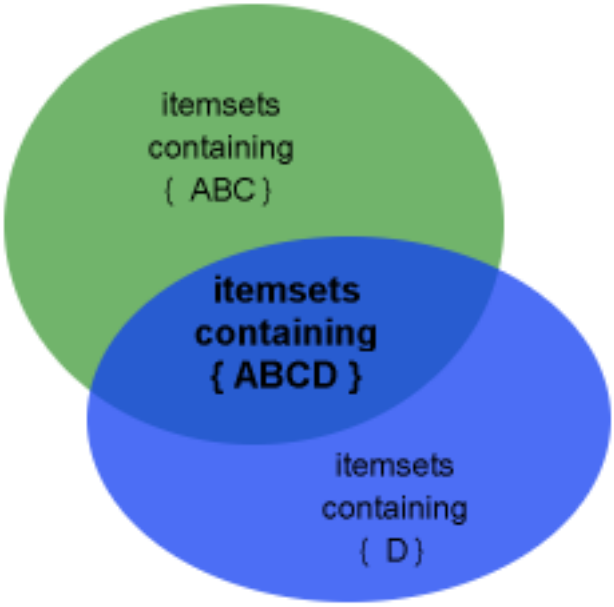
Y is called the consequent of the rule. The consequent is composed of only one item, for example Y can be the item {D}.

Support

The Support of a rule is a measure that indicates the number of sessions that verify the rule. For instance the number of sessions, that contains the itemset {A,B,C} and the item D.

Confidence

The Confidence of a rule is a measure that indicates the percentage of sessions verifying the consequent among those verifying the antecedent. For instance the number of sessions containing the item D, among the ones containing the itemset {A,B,C} (see graph below).



The formula used to calculate the confidence is:

Confidence = Support(ABCD) / Support(ABC)

Lift

The Lift of a rule is a measure that indicates the chances of finding the consequent by using the antecedent compared with the chances of randomly finding the consequent. A value greater than 1 indicates that using the antecedent increases your chances to find the consequent.

Predictive Power (KI)

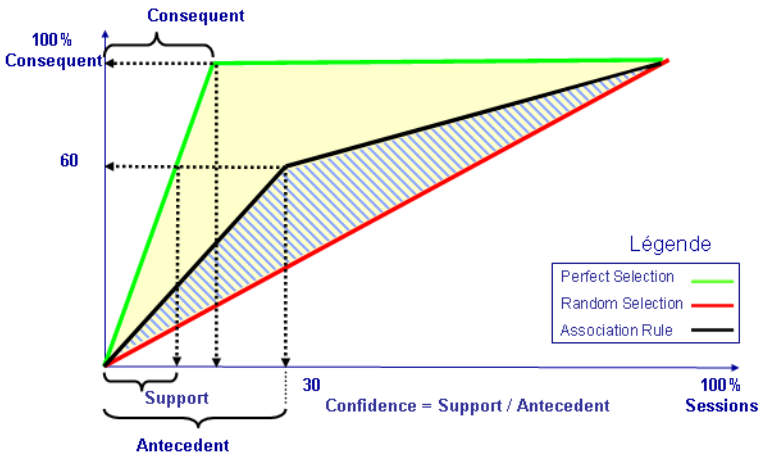
The predictive power (KI) indicator is the quality indicator of the models generated using Automated Analytics. This indicator corresponds to the proportion of information contained in the consequent that the antecedent is able to explain.

The following diagram graphically represents:

- an association rule (black curve), that is the chances of the antecedent rule to allow finding the expected consequent,

compared with:

- a random selection (red curve), that is the chances of a randomly selected item in the dataset to be the expected consequent,
- and a perfect selection (green curve), that is the case when all the first selected items are the expected consequent.

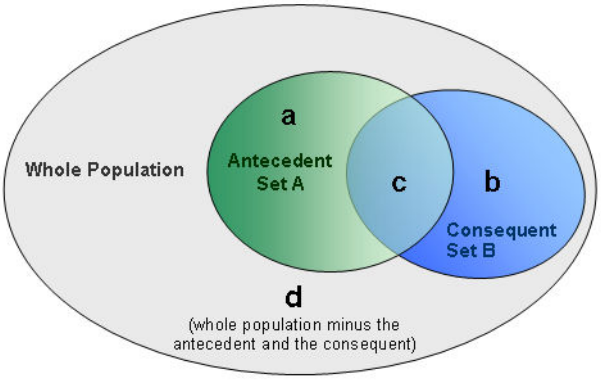


In this diagram, the KI indicator corresponds to "the area found between the curve of the rule generated and that of the random selection" divided by "the area found between the curve of the perfect selection and that of the random selection". As the curve of the generated rule approaches the curve of the perfect selection, the value of KI approaches 1.

Predictive Power in Relation to the Classic Indicators

This section defines the mathematical relation of the predictive power to other standard indicators in the Association Rules Mining area: Rule Support, Confidence and Lift.

Assuming we have an association rule R with the shape: R: A => B (EQ 1) where A is the antecedent and B the consequent. The following figure, named Venn diagram, represents the association rule R defined above.



Associated with the association rule R, we make the following assumptions about the variety of the supports to simplify the following computations (all support measures are represented as proportions).

$$Supp(A) = a + c$$
$$Supp(B) = b + c$$
$$Supp(R) = c$$
$$Supp(\overline{A}) = 1 - (a + c)$$
$$Supp(\overline{B}) = 1 - (b + c)$$
$$Supp(\overline{A \cup B}) = d = 1 - (a + b + c)$$
$$a + b + c + d = 1 \qquad (EQ\ \ 2)$$

where

$$0 \le a \le 1$$
$$0 \le b \le 1$$
$$0 \le c \le 1$$
$$0 \le d \le 1$$

The Venn diagram presented above could be mapped to the EQ 2 in a confusion matrix:

Actual Consequent	Consequent Predicted (from Antecedent) 1	0	
1	c	b	Supp(B)
0	a	d	Supp(Not B)
	Supp(A)	Supp(Not A)	1

> **i Note**
>
> The different supports represent the support of a set with respect to the whole population.

Using EQ 2 and the Confusion Matrix representation of the KI, we can extract the following definition.

$$KI = \frac{\frac{c-(a+c)(b+c)}{(b+c)}}{1-(b+c)}$$

$$KI = \frac{c-(a+c)(b+c)}{(b+c)(a-(b+c))} \qquad (EQ\ \ 3)$$

Using EQ 3, predictive power (KI) could be expressed in terms of:

Rule Support:

$$KI = \frac{Supp(R)\ - Supp(A)\ Supp(B)}{Supp(B)\ (1 - Supp(B))} \qquad (EQ\ 4)$$

Confidence:

$$KI = \frac{Supp(A)\ [Conf(R)\ -\ Supp(B)]}{Supp(B)\ (1 - Supp(B))} \qquad (EQ5)$$

Lift:

$$KI = \frac{Supp(A)\ (Lift(R)\ -\ 1)}{1 - Supp(B)} \qquad (EQ6)$$

5.4.3 Introduction to Sample Files

You can test the Association Rules feature using the sample files available in the folder `Samples/KAR` located:

- for Windows , in the folder `Program Files\SAP Predictive Analytics\Desktop <version number>\Automated\`
- for Linux, in the folder where you have decompressed the archive file (that is .tar.Z or .tar.gz).

The dataset contains a single day of Web traffic from an E-commerce site in December 1999. The site content was served by a Broadvision server, but no "cookies" or login was required, making the sessions effectively anonymous.

File	Description
`website_references.csv`	list of sessions and binary purchase target (50581 rows)
`website_references_desc.csv`	description for website_references.csv
`website_transactions.csv`	log of files requested from Broadvision server (532860 rows)
`website_transactions_desc.csv`	description for website_transactions.csv

The parameters of the model used in this guide are based on these sample files.

> **i Note**
>
> A Read-Me file is provided with the sample files describing the datasets and how to use them to test Association Rules feature.

Additional Sample Files

Another set of sample files is available in the folder `Samples\KAR` with a readme text file describing how to use the following samples:

- customers_references.txt
- customers_references_desc.txt
- customers_transactions.txt
- customers_transactions_desc.txt

5.4.4 Modeling with Association Rules

5.4.4.1 Creating Association Rules

Association Rules generates association rules. Association rules provide clear and useful results, especially for market basket analysis. They bring to light the relations between products or services and immediately suggest appropriate actions. Association rules are used in exploring categorical data, also called items.

The strengths of Association Rules are:

- to produce clear and understandable results,
- to support unsupervised data mining (no target attribute),

- to explore very large datasets thanks to its ability to first generate rules on parts of the dataset before aggregating them (exploration by chunks),
- to generate only the more relevant rules (also called primary rules).

1. Select *Start* > *Programs* > *SAP Business Intelligence* > *SAP Predictive Analytics Desktop* > *SAP Predictive Analytics*.

   SAP Predictive Analytics opens up.
2. Select the *Association Rules* feature in the *Modeler* section.

5.4.4.2 Step 1 - Selecting the Data

5.4.4.2.1 Selecting a Data Source

After selecting the type of model that you want to generate, you must select the data source that you want to use as the training dataset.

An Association Rules Reference dataset must have a single variable unique primary key. If the primary key is non-unique or spread out over several variables, Association Rules will not function properly.

1. On the screen *Select a Data Source*, select the *Data Type* to be used (Text file, ODBC, ...).

   For our example, select the option *Text File*.
2. Click the *Browse* button.

   The *Data Source Selection dialog* opens.
3. Select the file you want to use, then click *OK*.

   The name of the file will appear in the *Dataset* field.

   For our example, select the file `website_references.csv`.
4. Click *Next*.

5.4.4.2.1.1 Data Sources Supported

Automated Analytics supports the following data sources:

- Text files (also called flat files) in which the data are separated by a delimiter, such as commas in `.csv` (Comma Separated Value) files.

  > **! Restriction**
  >
  > When accessing data in `.csv` files, Automated Analytics only supports `CR` + `LF` (common on Microsoft Windows) or `LF` (common on Linux) for line breaks.

- Database management systems that can be accessed using ODBC.

> **i Note**
>
> For the list of supported ODBC-compatible sources, see the SAP Product Availability Matrix (PAM) at http://service.sap.com/sap/support/pam.
>
> For more information about using SAP HANA, see the related information below.
>
> To configure Automated Analytics modeling tools to access data in your database management system, refer to the guide *Connecting your Database Management System on Windows* or *Connecting your Database Management System on Linux*.

- SAS files

Related Information

SAP HANA as a Data Source [page 8]

5.4.4.2.1.2 SAP HANA as a Data Source

You can use SAP HANA databases as data sources in Data Manager and for all types of modeling analyses in Modeler: Classification/Regression, Clustering, Time Series, Association Rules, Social, and Recommendation.

SAP HANA tables or SQL views	found in the *Catalog* node of the SAP HANA database
All types of SAP HANA views	found in the *Content* node of the SAP HANA database. An SAP HANA view is a predefined virtual grouping of table columns that enables data access for a particular business requirement. Views are specific to the type of tables that are included, and to the type of calculations that are applied to columns. For example, an analytic view is built on a fact table and associated attribute views. A calculation view executes a function on columns when the view is accessed. **! Restriction** • Analytic and calculation views that use the variable mapping feature (available starting with SAP HANA SPS 09) are not supported. • You cannot edit data in SAP HANA views using Automated Analytics.

Smart Data Access virtual tables	Thanks to Smart Data Access, you can expose data from remote sources tables as virtual tables and combine them with HANA regular tables. This allows you to access data sources that are not natively supported by the application, or to combine data from multiple heterogeneous sources. **⚠ Caution** To use virtual tables as input datasets for training or applying a model or as output datasets for applying a model, you need to check that the following conditions are met: • The in-database application mode is not used. • The destination table for storing the predicted values exists in the remote source before applying the model. • The structure of the remote table, that is the column names and types, must match exactly what is expected with respect to the generation options; if this is not the case an error will occur. **⚠ Caution** In Data Manager, use virtual tables with caution as the generated queries can be complex. Smart Data Access may not be able to delegate much of the processing to the underlying source depending on the source capabilities. This can impact performance.

Prerequisites

You must know the ODBC source name and the connection information for your SAP HANA database. For more information, contact your SAP HANA administrator.

In addition to having the authorizations required for querying the SAP HANA view, you need to be granted the `SELECT` privilege on the `_SYS_BI` schema, which contains metadata on views. Please refer to SAP HANA guides for detailed information on security aspects.

5.4.4.2.1.3 To Select an SAP HANA Table or SQL View as Data Source

1. On the *Select a Data Source* screen, select *Data Base* in the *Data Type* list.
2. Click the *Browse* button left of the *Folder* field.
3. In the *Data Selection* dialog, select the SAP HANA database you want to use in the *Select Source Folder for Data* list.

4. If this is the first connection to this database, enter the login information in the *User* and *Password* fields and click *Connect*.

   The content of the database is displayed in the left part of the window.
5. Open the *Catalog* node, select the schema containing the table or SQL view you want to use as the data source for your model and click *OK*.
6. Click the *Browse* button left of the *Dataset* field.
7. In the *Data Selection* dialog, select the table or SQL view you want to use.

   You can filter the table list by entering part of the table or SQL view name in the field located below the list.
8. Click *OK*.

5.4.4.2.1.4 To Select an SAP HANA View

You can select an SAP HANA analytic or calculation view as a data source for your model.

1. On the *Select a Data Source* screen, select *Data Base* in the *Data Type* list.
2. Click the *Browse* button left of the *Folder* field and select the SAP HANA database you want to use in the *Select Source Folder for Data* list.
3. If this is the first connection to this database, enter the login information in the *User* and *Password* fields and click *Connect*.

   The content of the database is displayed in the left part of the window.
4. Open the *Content* node. Use the tree to browse the packages and select the one containing the SAP HANA view you want to use as the data source for your model and click *OK*.
5. Click the *Browse* button left of the *Dataset* field.
6. In the *Data Selection* dialog, select the view you want to use.

   You can filter the view list by entering part of the view name in the field located below the list.
7. Click *OK*.

   If the SAP HANA view you have selected requires you to specify some values, the *User Values* window opens when you click *View Data* or *Next* on the *Select a Data Source* screen. The variables for which you need to enter a value are listed on the left side of the panel. The following table details how the different variables are identified in the list. In this topic, the term variable is used for both variables and input parameters.

Type of variable	Noted as	Signaled by
Mandatory variable	mandatory	A red star
Optional variable	optional	
Variable with a default value	default value	A green check mark
Variable with a value set by the user		A green check mark

   Check the box *Show mandatory only* to display only mandatory variables in the list.

8. In the list, select the variable you want to set a value for.

   The field allowing you to set the value for the variable is displayed on the right side of the panel. Depending on the type of value required, you may see different types of fields. For example, if a single value is required,

you see a list of operators to choose from and a field to enter a value. If a range of values is required, you see two fields for the range bounds and a list of operators to choose from.

9. Some variables have a list of preset possible values. In that case, the list of values for the current variable is displayed on the right side of the panel. Select the value you want to set for this variable; it will be displayed in the value field.
10. Use the *Restore* button to reset the value of the current variable to its default value.
11. Once you have entered all required values, click *OK*.

    Once you have accessed an SAP HANA view, the values you have set will be remembered and used as default when you access this view again during the modeling process.

5.4.4.2.2 Describing the Data

In order for the application features to interpret and analyze your data, the data must be described. To put it another way, the description file must specify the nature of each variable, determining their:

- Storage format: number (*number*), character string (*string*), date and time (*datetime*) or date (*date*).
- Type: *continuous*, *nominal*, *ordinal* or *textual* .

For more information about data description, see the Types of Variables and Storage Formats in Classification, Regression, Segmentation and Clustering Scenarios – Automated Analytics User Guide.

To describe your data, you can:

- Either use an existing description file, that is, taken from your information system or saved from a previous use of the application features,
- Or create a description file using the Analyze option, available to you in the application. In this case, it is important that you validate the description file obtained. You can save this file for later re-use.

> ⚠ Caution
>
> The description file obtained using the *Analyze* option results from the analysis of the first 50 lines of the initial data file. In order to avoid all bias, we encourage you to mix up your dataset before performing this analysis.

Using a description file:

1. On the screen *Data Description*, use the *Browse* button to select the description file. For our example, select the file `website_references_desc.csv`.
2. Click the button *Open Description*.
3. Select the description file and click *OK*.
4. Click the *Next* button.

Analyzing the dataset:

5. On the screen *Data Description*, click the *Analyze* button.
6. To verify the description you obtained, click the button *View Data* and take a look at the dataset beginning.
7. Click the *Next* button.

5.4.4.2.3 Selecting Events Data

The screen *Events Data Source* lets you specify the data source to be used as the *Transaction dataset*.

1. Select the *Data Type* of your data source (Text Files, ODBC, ...).

   For our example, select the option Text Files.
2. In the *Folder* field, specify the folder where your data source is stored.
3. In the *Events* field, specify the name of your data source.

   In our example, select the file `website_transactions.csv`.
4. Click the *Next* button.

5.4.4.2.4 Describing Events Data

The screen *Events Data Description* lets you describe your Transaction data, offering you the same options as the screen *Data Description* . For Association Rules to function properly, there must be a variable in the Transaction dataset that is the same as the primary key declared for the Reference dataset, referred to as a "Join Column". The name of the variable can be different, but the storage and value must be the same. The values of this variable need not be unique, since each Reference key can have 0, 1, or several associated transactions.

In addition to a suitable join column, the Transaction dataset must have at least one item variable. The item variable will be used by Association Rules to build the association rules.

For detailed procedures on how to set parameters on this screen, see Describing the Data.

There are two ways to describe the events data, either by using a description file, or by letting the application analyze the first hundred lines of the dataset.

Using a description file :

1. On the screen *Events Data Description*, select the description file.

   For our example, select the file `website_transactions_desc.csv`.
2. Click the button *Open Description*.

   The data description is displayed.
3. Select the description file and click *OK*.
4. Click *Next*.

Analyzing the dataset:

5. On the screen *Events Data Description*, click the *Analyze* button.
6. To verify the description you obtained, click the button *View Data* and take a look at the dataset beginning.
7. Click *Next*.

5.4.4.3 Scenario 1: Standard Modeling with Association Rules

5.4.4.3.1 Step 2 - Defining the Modeling Parameters

5.4.4.3.1.1 Setting ASSOCIATION RULES Parameters

The screen Association Rules Extraction Parameters enables you to set Association Rules parameters by:

- Joining your reference data with your transaction data
- Setting the Specific Modeling Parameters

1. On the screen *Association Rules Extraction Parameters*, select the join column for both the log and reference datasets.

   For our example, select the column SessionID in both datasets.
2. Select the *Item Column*.

   For our example, select the *column Page*.
3. Set the *Minimum Support*.

   For our example, enter 1.
4. Click *Next*.

5.4.4.3.1.2 Understanding Association Rules Parameters

The Association Rules parameters allow you to set the format of the rules you want to get by setting the following indicators:

- the minimum support
- the minimum confidence
- the maximum length

Minimum Support

The support of a rule is the number of records verifying the rule. With a rule of the form X=>Y, the support is the count of records containing the itemset X and the attribute Y.

The default value is 10. If the value is superior to *10*, it represents the number of sessions. If the value is between 0 and 10, it represents a percentage of the number of sessions (default = 10, required value > 0).

Minimum Confidence

The confidence of a rule is the percentage of records verifying the consequent of the rule among those verifying the antecedent of the rule. With a rule of the form X=>Y, the confidence is the count of records containing the itemset X and the attribute Y in relation to the count of records containing only the itemset X.

The default value is *0.5*. This parameter requires a value between 0 and 1.

Maximum Length

The maximum length of a rule is its total number of items, including the antecedent and the consequent. It is called the cardinality of the rule.

The default value is *4*. This parameter requires a value higher or equal to 2.

5.4.4.3.1.3 Understanding Advanced Parameters

You can set the association rules advanced parameters by clicking the *Advanced* button located on the right bottom corner of the panel *Association Rules Extraction Parameters*.

Chunk Size

The option *Chunk Size* allows you to set a minimal number of sessions for which Association Rules will generate temporary rules.

In a standard use, that is when the chunk size is 0, Association Rules imports all the transactions in memory before generating the association rules corresponding to the model parameters (minimum support, minimum confidence, maximum rule length). With large datasets, this method can be very expensive in time and memory.

To gain speed and memory space, Association Rules offers the option to import and generate rules from pieces of the events file. In details, Association Rules will load in memory a number of sessions equivalent to the value of the “Chunk Size” parameter. Temporary Association Rules are generated on this set of sessions. Then, memory is cleaned from these sessions and Association Rules will load the next sessions with a number equivalent to the value of the “Chunk Size” parameter. This operation will be repeated until the end of the Events data file. At the end of the learning step, these rules are joined and their statistics are updated for the entire dataset.

The more sessions you have to analyze, the more interesting the option *Chunk Size* becomes; although the number of rules and of frequent found itemsets can differ depending on the method you select. However, these differences are not significant if the chunks size is large enough. The table below describes the utility of this option depending on the datasets specifications.

Chunk Size Use Recommendation	Low Number of Sessions	High Number of Sessions
Low Number of Transactions in the Session	Not Needed	Strongly Recommended
High Number of Transactionsin the Session	Not Needed	Strongly Recommended
	Prefer the use of the parameter *Rules Length* to limit the number of items combination.	

As an indication, the chunk size should be roughly 10% of the total number of sessions in your dataset (that is 10 to 15 chunks depending on the dataset size). You can start with 10% and modify this number to improve your results. If your transactions file contains less than 5000 rows, you should use a no-chunk strategy.

> ⚠ Caution
>
> The transactions in the events dataset must be grouped by sessions, meaning that the Events file must be sorted by sessions.

However, the task of splitting the sessions in chunks can be delegated to the application through the option *Guess Chunk Size* which will automatically compute the chunks' size in order to obtain ten of them.

Skip Derived Rules / Skip Reducible Itemsets

The option *Skip Derived Rules* indicates to Association Rules to generate only the primary rules and thus skip the derived rules. For instance, given R1 = X => Z and R2 X,Y =>Z two association rules, R1 is called a primary rule and R2 a derived rule. The default value is True.

Filters on Consequent Items

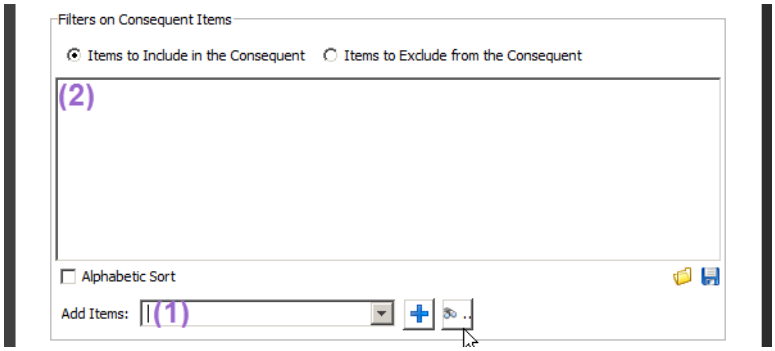
Before starting the learning process, you can select which items you want to see as consequents in the rules generated by Association Rules by defining either the consequents to keep or the ones to exclude from the modeling.

> ⚠ Caution
>
> Before creating either list, you need to know the exact value of each item you want to include or exclude.

5.4.4.3.1.3.1 Selecting the Items to Keep

1. Select the option *Items to Include in the Consequent*.

   This option is selected by default.



2. In the text field located at the bottom of the panel (1) enter the value of the item you want to see appear in the rules.
3. Click the + button located on the right of the field. The item appears in the list above (2).
4. Repeat steps 2 and 3 for all the items you want to include.
5. Click the *Validate* button to save the list and go back to the panel Association Rules Parameters *Settings*.

5.4.4.3.1.3.2 Excluding Items from the Consequents

1. Select the option *Items to Exclude in the Consequent*.
2. In the text field located at the bottom of the panel enter the item you do not want to see appear as consequent in the rules.
3. Click the + button located on the right of the field. The item appears in the list above.
4. Repeat steps 2 and 3 for all the items you want to exclude.
5. Click the *OK* button to save the list and go back to the panel *Association Rules Parameters Settings*.

5.4.4.3.1.3.3 Removing Items

1. Select the items you want to remove (use the Ctrl key to select several items).
2. Right-click the list. A contextual menu is displayed.
3. Select the option *Remove this (these) item(s)*.

5.4.4.3.1.3.4 Renaming an Item

1. Select the item you want to modify.
2. Right-click the list.

   A contextual menu is displayed.
3. Select the option Rename this item.

   The item value is displayed in the text field.

4. Modify the value as needed.
5. Click the + button to validate the change.

   To sort a list alphabetically, check the box *Alphabetic Sort* if you want to see the consequent listed in an alphabetical order.

5.4.4.3.1.3.5 Checking Modeling Parameters

The *Summary of Modeling Parameters* screen allows you to check the modeling parameters just before generating the model.

- The name of the model is filled automatically. It corresponds to the name of the item variable, followed by the underscore sign ("_") and the name of the Reference file without its file extension.
- The *Autosave* button allows you to activate the feature that will automatically save the model once it has been generated. When the *autosave* option is activated, a green check mark is displayed on the *Autosave* button.

5.4.4.3.1.3.6 Activating the Autosave Option

The *Model Autosave* panel allows you to activate the option that will automatically save the model at the end of the generation process and to set the parameters needed when saving the model.

To activate the option, proceed as follows:

1. In the *Summary of Modeling Parameters* panel, click the *Autosave* button.
   The *Model Autosave* panel is displayed.
2. Check *Enable Model Autosave*.
3. Set the parameters listed in the following table.

Parameter	Description
Model Name	This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
Description	This field allows you to enter the information you want, such as the name of the training dataset used, the polynomial degree or the performance indicators obtained. This information could be useful to you later for identifying your model. Note that this description will be used instead of the one entered in the *Summary of Modeling Parameters* panel.
Data Type	This list allows you to select the type of storage in which you want to save your model. The following options are available: ○ *Text Files*, to save the model in a text file. ○ *Data Base*, to save the model in a database. ○ *Flat Memory*, to save the model in the active memory.

Parameter	Description
Folder	Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
File/Table	This field allows you to enter the name of the file or table that is to contain the model. When saving the model as a text file, you must enter one of the following format extensions `.txt` (text file in which the data is separated by tabs) or `.csv` (text file in which the data is separated by commas).

4. Click *OK*.

5.4.4.3.2 Step 3 - Generating the Model

Once the modeling parameters are defined, you can generate the model.

1. On the *Summary of Modeling Parameters* screen, click the *Generate* button.

   The *Training the Model* screen opens. The model is being generated. A progress bar allows you to follow the process.
2. If the *Autosave* option has been activated in the *Summary of Modeling Parameters* panel, a warning message is displayed at the end of the learning process confirming that the model has been saved.
3. Click *Close*.

5.4.4.3.2.1 Following the Generating Process

There is two ways for you to follow the progress of the generation process:

- The Progress Bar displays the progression for each step of the process. It is the screen displayed by default.
- The Detailed Log displays the details of each step of the process.

To display the detailed log, click the *Show Detailed Log* button.

To display the progression bar, click the *Show Progression* button. The progression bar screen appears.

To stop the learning process:

1. Click the *Stop Learning Process* button.
2. Click *Previous*. The *Summary of Modeling Parameters* screen appears.
3. Go back to the *Checking Modeling Parameters* section.

Visualizing the Model Results

At the end of the generation process, a summary of the model results appears.

For more information on the model summary, go to step 4, Analyzing and Understanding the Model (see "Step 4 - Analyzing and Understanding the Generated Model").

5.4.4.3.3 Step 4 - Analyzing and Understanding the Generated Model

Automated Analytics allows you to analyze and understand the model generated thanks to a suite of plotting tools describing:

- all the items treated by the model,
- the rules generated by the model.

Model Debriefing

To display the *Model Debriefing*, on the *Using the Model* menu, select the *Model Overview* option. The *Model Overview screen* opens.

> **i Note**
>
> If you have built more than one model in the same session, the debriefing for all the models will be displayed on this screen, sorted by *Date of Build*.

Understanding the Model Debriefing

The *Model Overview* screen is composed of four sections detailing information on:

- the model,
- the transformation,
- the parameters,
- the modeling results.

The Model

This section details the following information:

Name	Significance	For example...
Model	Name of the model. It is generated by using the item variable name and the reference dataset name.	Page_website_references
Dataset	Name of the Reference dataset used for the model.	`website_references.csv`
Initial Number of Variables	Total number of variables in the Reference dataset.	1
Number of Selected Variables	Number of variables used to build the model.	

Name	Significance	For example...
Building Date	Date and time when the model was built.	2012-04-20 15:45:23
Learning Time	Time needed to build the model.	1s
Engine Name	Name of the feature used to build the model.	Association Rules

The Training Parameters

This section summarizes the parameters used to build the model. The following table details the provided information:

Name	Significance	For example...
Minimum Support	The minimum number of sessions to validate a rule.	10%
Minimum Confidence	The minimum percentage of sessions verifying the consequent among those verifying the antecedent.	50%
Maximum Length	The maximum length of the rule.	4

The Datasets Parameters

This section summarizes the parameters of the datasets used to build the model. The following table details the provided information:

Name	Significance	For example...
Transaction ID Column	name of the join key for the events dataset.	SessionID
Reference ID Column	name of the join key for the reference dataset.	SessionID
Item Column	name of the column containing the items.	Page

The Results

This section displays information on the rules generated.

The following table details the provided information:

Name	Significance	For example...
Number of Sessions Processed	The number of sessions treated by the model.	245
Number of Rules Found	The number of rules found by the model.	61
Number of Items Found	The number of items found by the model.	481

Name	Significance	For example...
Number of Frequent Item Sets	The number of item sets whose support is superior to the minimum support. ⚠ Caution The Frequent Item sets information should not be taken into account if the *Chunk* learning strategy has been used. In this particular case, the information may be under estimated.	731
Transactions Found	The number of transactions found by the model	2500

Association Rules Exploration

On the *Using the Model* menu, select the *Association Rules Exploration* option. The *Association Rules Exploration* screen opens. This screen allows you to search through all the rules found by the association rules engine thanks to filtering options and allows you to display the rules as a graph.

Setting the Search Parameters

There are three types of filters available:

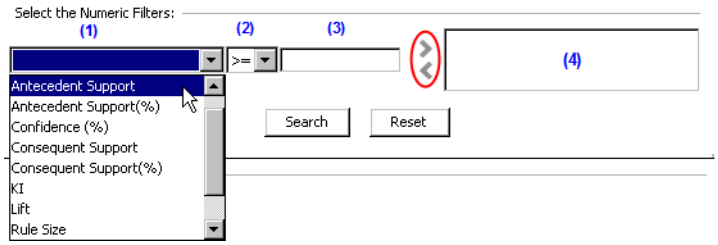
- by Antecedent, where you select the antecedents you want to see in the rules
- by Consequent, where you select the consequents you want to see in the rules
- by numerical filters:
  - Rule Support (in number of sessions or in percentage)
  - Antecedent Support (in number of sessions or in percentage)
  - Consequent Support (in number of sessions or in percentage)
  - KI (predictive power)
  - Confidence
  - Lift

You can either apply only one filter or combine several to refine your search.

> **ℹ Note**
>
> To Set a Filter by Antecedent and/or Consequent, set the filter(s) you want to apply:
>
> - by antecedent: select one or more antecedents in the list,
> - by consequent: select one or more consequents in the list.
>
> If you filter both by antecedent and consequent be careful not to select the same item in both lists.

5.4.4.3.3.1 Setting a Numerical Filter

1. Select an indicator in the list (Confidence, Rule Support, Size, Lift, KI, Antecedent Support). (1)
2. Select a comparison operator. (2)
3. Enter the value of comparison in the text field. (3)
4. Click the > button to add the filter to the filter list (4).



Example: KI >= 0.8

> **i Note**
>
> 1. To start the search, once the filters are set, click the *Search* button. The tab rule(s) found displays the number of rules found and the list of these rules. You can sort them by any column: just click the select column header.
>
> **⚠ Caution**
>
> If you click the *Search* button without selecting a filter, all the rules will be displayed. When the number of rules is very high, displaying all of them can take a very long time.
>
> 2. To reset the search engine, cick the *Reset* button.

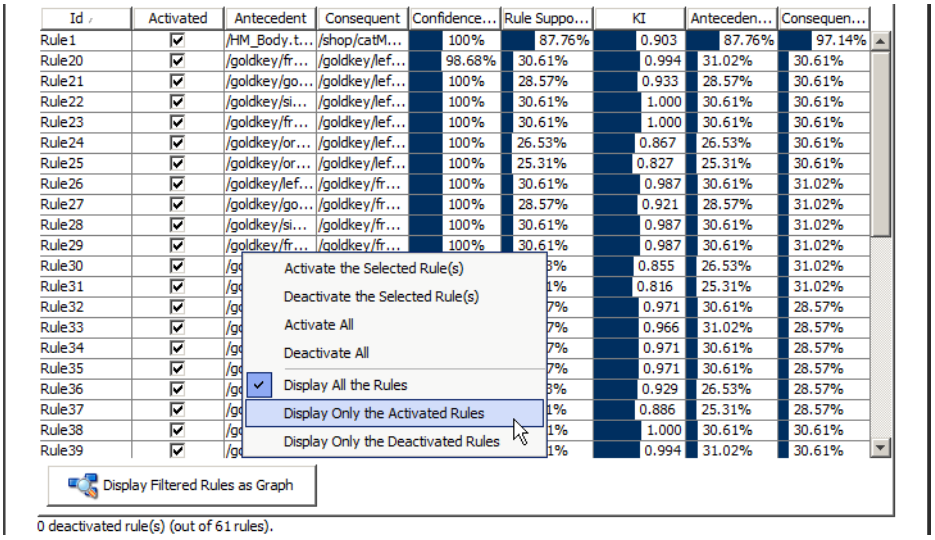
5.4.4.3.3.2 Displaying the Search Results

The search results are displayed in the tab ..*rules found*.

To Select the Rules to Display

1. Check or uncheck the *Activated* box next to the rules you want to display/hide.
2. Right-click one of the table fields.

   A contextual menu is displayed.

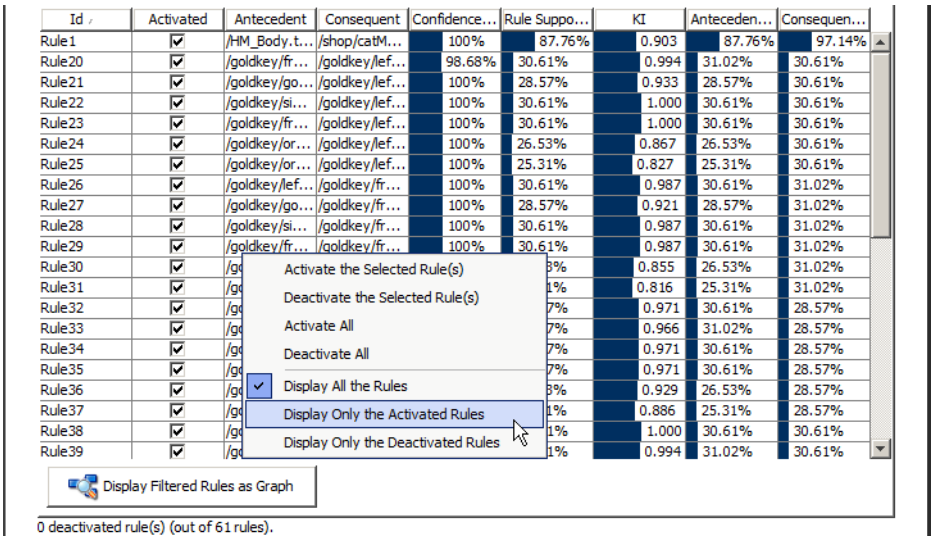


3. Choose one of the offered options.

To Select the Columns to Display

4. Right-click the table header.

   A contextual menu is displayed.



5. Check the columns you want to see displayed in the result list.

   The selected columns automatically appear in the list.

> **i Note**
>
> To sort the results by column, click the selected column header.

5.4.4.3.3.3 Displaying the Rules as a Graph

The tab *Graph View* allows you to display the rules as a graph.

To display the Association Rules as a graph:

- Either filter the rules found by *Association Rules* and then display the corresponding graph: On the tab .. rule(s) found, click the button *Display Filtered Rules as a Graph*.
- Or select the antecedent(s) and/or consequent(s) you want to see in the graph (see section Understanding the Graph below).

To Add an Item to the Graph:

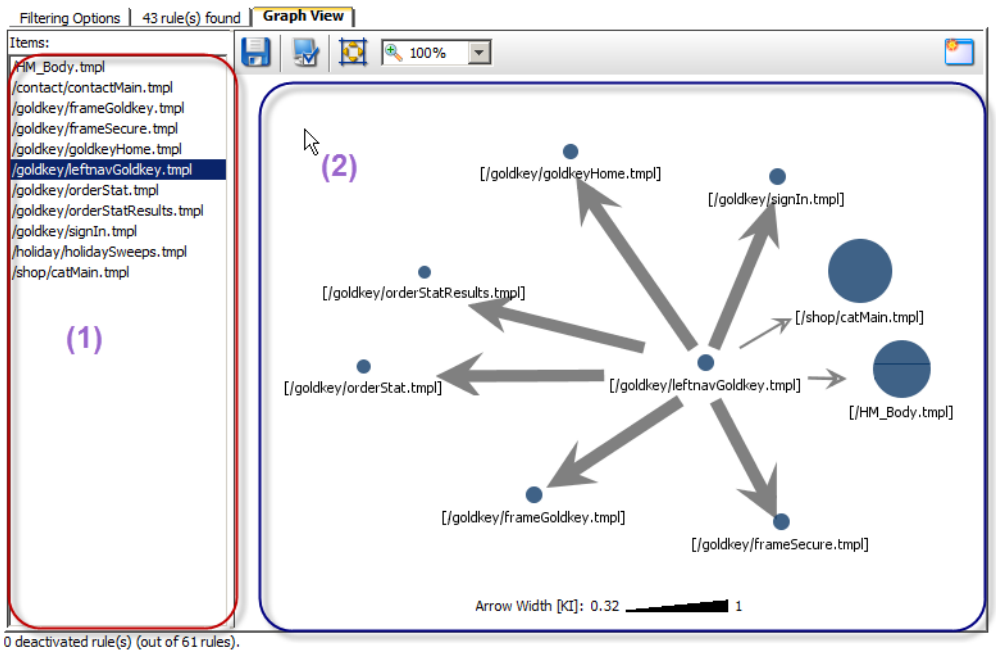
1. Drag and drop the selected item from the list of items to the graph section.

   A contextual menu is displayed.
2. Select if you want to display the rules where the item is part of the rule antecedent, the rule consequent or both:
   - if you select *Antecedent*, only the rules where the antecedent contains the selected item will be displayed,
   - if you select *Consequent*, only the rules where the consequent contains the selected item will be displayed,
   - if you select *Antecedent* or *Consequent*, only the rules where the antecedent or the consequent contains the selected item will be displayed.

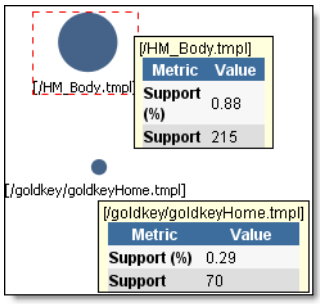
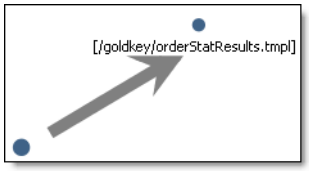
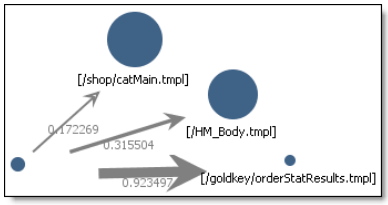
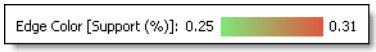
5.4.4.3.3.4 Understanding the Graph Panel

The tab *Graph View* is split in two parts:

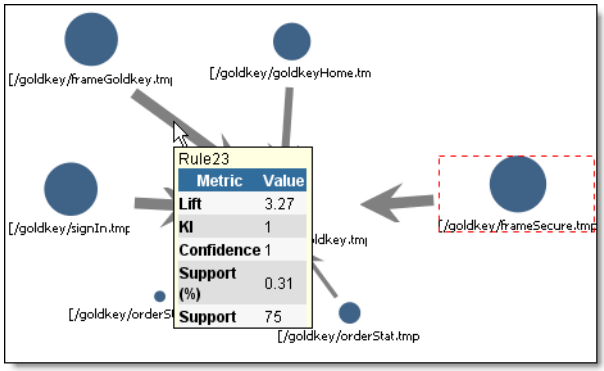
- (1) on the left, the list of the items used in at least one rule, either as an antecedent or as a consequent,
- (2) on the right, the graph.



The following table sums up the visual elements used in the graph:

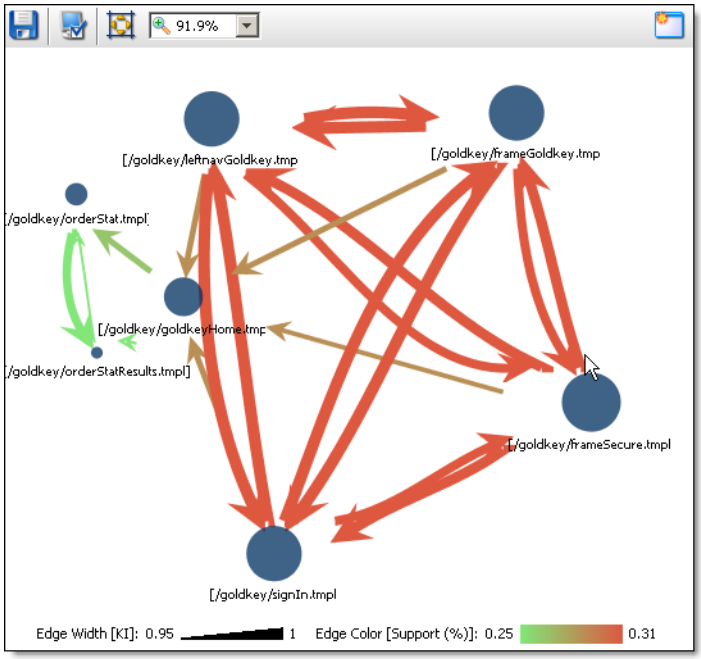
The element...		...represents
	a blue circle	an itemset
	the circle size	the itemset support
	an arrow	a rule
	the width of the arrows	By default, the width of the arrows represents the predictive power (KI) of the rule, but you can customize this parameter (see Customizing the Graph)
	the color of the arrows	By default, the color of the arrows is not set to represent any information, however you can customize this parameter (see Customizing the Graph).

You can display the information about a rule (KI, Lift, Support,...) by pointing a rule with the cursor.



5.4.4.3.3.4.1 Customizing the Graph

You can customize the width and/or the color of the arrows connecting the itemsets.



To Customize the Rules Display:

1. Click the *Customize Rules Display* button.

   A dialog opens allowing you to set the graphics visual parameters.

To Customize the Arrows Width:

2. In the *Edge Width list*, select the metrics you want to use.

To Customize the Arrows Color:

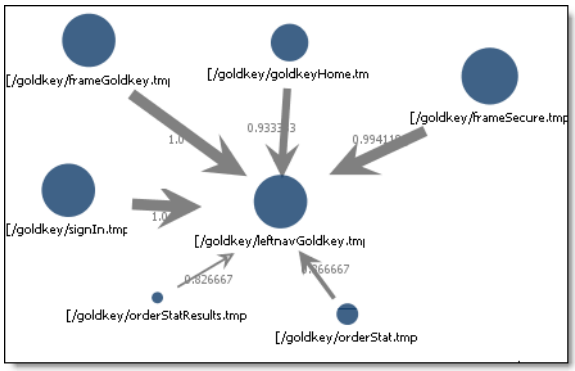
3. In the *Edge Color list*, select the metrics you want to use.

   The color varies from green to red, green being the lowest value and red the highest value.

To Display the Edge Width Value:

4. Check the *Display Edge Width Value as Label* option.

   The corresponding value is displayed for each arrow.



The toolbar located above the graph offers the following options:

	Save as a png image
	Customize the graph
	Fit the graph to the screen size
100%	Zoom in or out
	Clear the current graph

5.4.4.3.3.4.2 Statistical Reports

To help you analyze your modeling results and possibly share these results with your colleagues, managers, partners or customers, Association Rules provides you with a set of statistical reports that can be saved in a variety of formats (html, pdf, rtf).

The following table presents the four types of tables available to you.

The screen ...	Presents the following information ...
Rules Details (Support in Percentage)	The association rules are displayed in the form: *item_1&item_2&...&item_n->item_consequent*. The following information are given for each rule: confidence, KI, RuleSupport, ConsequentSupport, AntecedentSupport.
Rules Details (Support in Sessions Count)	This presents the same table as the previous one, but the support is displayed in number of sessions.
Items Statistics	This presents the list of the items treated by the model. The items frequency is given with respect to the transactions and not the sessions. For instance, if one item is present n times in one session, it will be counted n times and its percentage will be computed on the total count of the transactions.

5.4.4.4 Scenario 2: Modeling Using Sequence Mode

Presentation

Usually, the transactions have a temporal dimension, meaning that transactions occur one after another. In this case, the use of a standard Association Rules generation engine is not sufficient since it does not take into account the sequence of the transactions. To make the most of this temporal dimension, the association rules for which the antecedent occurs before the consequent need to be identified: this is the notion of Sequence.

Summary of the Modeling Settings to Use

In this step, you will follow the default scenario without using the Sequence mode (see "Scenario 1: Standard Modeling with Association Rules").

The following table summarizes the modeling settings to be used. It should be sufficient for users who are already familiar with SAP Predictive Analytics.

Task(s)	Screen	Settings
Specifying the data source	*Select a Data Source*	In the field *Dataset*, select the file `website_references.csv` .
Selecting a partition strategy	*Partition Strategy*	Partition strategy: random
Specifying a description file	*Data Description*	In the field *Description*, select the file `website_references_desc.csv` .
Specifying an Events file	*Events Data Source*	In the field *Events*, select the file `website_transactions.csv`.
Specifying a description file	*Events Data Description*	In the field *Description*, select the file `website_transactions_desc.csv`.
Activate the Sequence mode	*Association Rules Parameters Settings*	• Check the field *Activate Sequence Mode*. • In the field *Date*, select the *variable Time*.

5.4.4.4.1 Step 2 - Defining the Modeling Parameters

5.4.4.4.1.1 Setting Association Rules Parameters

The screen *Association Rules Extraction Parameters* enables you to set Association Rules parameters by:

- Joining your reference data with your transaction data.
- Setting the Specific Modeling Parameters.

To set the parameters:

1. On the *Association Rules Extraction Parameters* screen, select the join column for both the log and reference datasets.

   For our example, select the column `<SessionID>` in both datasets.
2. Select the *Item Column*.

   For our example, select the column `<Page>`.
3. Set the *Minimum Support*.

   For our example, enter `<1>`.

4. Check the option *Activate Sequence Mode*.
5. Select Time as the *Date column*.
6. Click *Next*.

5.4.4.4.1.2 Understanding Association Rules Parameters

This section describes only the *Sequence Mode*. The parameters Minimum Support, Minimum Confidence and Maximum Length are described in the section Understanding Association Rules Parameters of Scenario 1.

Sequence Mode

When you activate the *Sequence mode*, *Association Rules* computes specific information relative to this notion. The following parameters are available in the contextual menu *Select the Columns to Display* on the panel *Association Rule Exploration*:

- the *Sequence Support* indicates the number of sessions that verify the rules and in which the antecedent occurs before the consequent.
- the *Sequence Confidence* indicates the percentage of sessions verifying the consequent among those verifying the antecedent but only for the antecedents occurring before their consequent.
- the *Sequence KI* is only measured on the rules for which the antecedent occurs before the consequent.
- the *Sequence Lift* is only measured on the rules for which the antecedent occurs before the consequent.
- the *Sequence Ratio* measures the percentage of sessions where the antecedent occurs before the consequent in relation to all the sessions verifying the rule.

Also additional temporal measures will be computed:

- *Minimum Duration* indicates the minimum amount of time observed between an antecedent and its consequent. This value is expressed in seconds if the date is in a date or a datetime format.
- *Maximum Duration* indicates the maximum amount of time observed between an antecedent and its consequent. This value is expressed in seconds if the date is in a date or a datetime format.
- *Average Duration* indicates the average amount of time observed between an antecedent and its consequent. This value is expressed in seconds if the date is in a date or a datetime format.

Finally, only the Association Rules for which the *Sequence Support* is superior to the *Minimum Support* and the *Sequence Confidence* is superior to the *Minimum Confidence* will be generated.

> **i Note**
>
> Computing and Filtering Association Rules taking into account the temporal dimension requires an additional sweep on the transaction file.

Understanding Advanced Parameters

The Advanced Parameters are described in Scenario 1 (see "Understanding Advanced Parameters").

5.4.4.4.2 Step 3 - Generating the Model

Once the modeling parameters are defined, you can generate the model.

To generate the model, on the screen *Summary of Modeling Parameters*, click the *Generate* button. The screen *Training the Model* will appear. The model is being generated. A progress bar allows you to follow the process.

There is two ways for you to follow the progress of the generation process:

- The Progress Bar displays the progression for each step of the process. It is the screen displayed by default.
- The Detailed Log displays the details of each step of the process.

To Display the Detailed Log:

1. Click the *Show Detailed Log* button.

   The log will be displayed.

To display the Progression Bar:

2. Click the *Show Progression* button.

   The progression bar screen appears.

To Stop the Learning Process:

3. Click the *Stop Learning Process* button.
4. Click *Previous*.

   The *Summary of Modeling Parameters* screen appears.
5. Go back to the *Checking Modeling Parameters* section.

> **i Note**
>
> At the end of the generation process, a summary of the model results will appear.
>
> For more information on the model summary, go to step 4 (see "Step 4 - Analyzing and Understanding the Generated Model").

5.4.4.4.3 Step 4: Model Debriefing

Automated Analytics allows you to analyze and understand the model generated thanks to a suite of plotting tools describing:

- all the items treated by the model,
- the rules generated by the model.

To Display the Model Overview:

1. On the *Using the Model* menu, select the *Model Overview* option.
2. Click *Next*. The *Model Overview* screen will appear.

> **i Note**
>
> If you have built more than one model in the same session, all model debriefing will be displayed on this screen sorted by *Date of Build*.

5.4.4.4.3.1 Understanding the Model Debriefing

The screen *Model Debriefing* is the same as described in Scenario 1 (see "Understanding the Model Debriefing"), except for two added parameters corresponding to the *Sequence Mode*.

The following table details the additional information provided in section *Parameters*:

Name	Significance	For example...
Sequence Mode	indicates if the Sequence Mode has been activated.	Activated
Date Variable	indicates which variable has been used as Date Variable for the sequences.	Time

Association Rules Exploration

The panel *Association Rules Exploration* is the same as described in Scenario 1, except for added filters corresponding to the *Sequence Mode*:

- *Sequence Support*, that is the number of sessions in which the antecedent occurs before the consequent.
- *Sequence Ratios*, that is the percentage of session where the consequent occurs after the antecedent for a specified rule.
- *Sequence Confidence*, that is the rule confidence _in the sequence mode.
- *Sequence KI*, that is the rule KI in the sequence mode.
- *Sequence Lift*, that is the rule Lift in the sequence mode.
- *Minimum Duration*, that is the minimum amount of time observed between an antecedent and its consequent. This value is expressed in seconds if the date is in a date or a datetime format.
- *Maximum Duration*, that is the maximum amount of time observed between an antecedent and its consequent. This value is expressed in seconds if the date is in a date or a datetime format.
- *Average Duration*, that is the average amount of time observed between an antecedent and its consequent. This value is expressed in seconds if the date is in a date or a datetime format.

Debriefing Tables

For details on the Debriefing Tables, go to the section Understanding the Model Debriefing of Scenario 1.

Model Parameters

For details on the Model Parameters, go to the section Understanding Association Rules Parameters of Scenario 1.

5.4.4.4.4 Step 5: Using the Model

5.4.4.4.4.1 Saving the Model

Once a model has been generated, you can save it. Saving it preserves all the information that pertains to that model, that is, the modeling parameters, the rules generated, and so on.

1. On the *Using the Model* screen, click *Save the Current Model*.

   The *Saving the Mode*l screen opens.
2. Above the *Browse* button, select the type of file to create.
3. Complete the following fields:
   - *Model Name*: this field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
   - *Description*: this field allows you to enter the information of your choosing, such as the name of the training dataset used. This information will help you identify your model for a later use.
   - *Folder*: depending on which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
   - *File/Table*: this field allows you to enter the name of the file or table that will contain the model. I you have selected the option Text File, the name of the file must contain one of the following format extensions .txt (text file in which the data is separated by tabs) or .csv (text file in which the data is separated by commas).

5.4.4.4.4.2 Applying a Model

The currently open model may be applied to additional datasets. The model allows you to perform predictions using the application datasets, and specifically, to make recommendations based on the datasets used to generate the model.

To Apply the Model to a New Dataset:

1. On the *Using the Model* panel, select the *Apply Model* option.
2. Click *Next*.

   The *Applying the Model* screen is displayed.
3. In the *Application Dataset* section, click the *Browse* button to select:
   - In the *Folder* field, the folder which contains your dataset,
   - In the *Data* field, the name of the file corresponding to your dataset.
4. In the *Results Generated by the Model* section, enter:
   - In the *Folder* field, the name of the folder where you want to save the output of the apply,
   - In the *Data* field, the name of the file containing the output of the apply.
5. In the *Generate list*, select the type of output you want to generate:
   - *Predicted Value Only*: generates basic outputs, that is, the session key, the ID of the rule used to find the consequent and the consequent itself.
   - *Consequents Optimized by KI*: generates basic outputs. If more than one rule give the same consequent for a session, the rule presenting the best KI (predictive power) will be selected.

- *Consequents Optimized by Confidence*: generates basic outputs. If more than one rule give the same consequent for a session, the rule presenting the best Confidence will be selected.
- *Predicted Value with Full Rules Description*: generates the extended outputs, that is, the session key, the rule ID, the consequent, the antecedent, the KI, the confidence and the rule support.
- *Consequents Optimized by KI with Full Rules Description*: generates the extended outputs. If more than one rule give the same consequent for a session, the rule presenting the best KI will be selected.
- *Consequents Optimized by Confidence with Full Rules Description*: generates the extended outputs. If more than one rule give the same consequent for a session, the rule presenting the best Confidence will be selected.

6. Click *Next*. The *Apply Events Data* screen is displayed.
7. Click the *Browse* button to select:
   - In the *Folder* field, the folder which contains your events dataset,
   - In the *Data* field, the name of the file corresponding to your events dataset.
8. Click *Next*.

   A dialog box is displayed, asking if you want to replace the current events dataset by the one you just selected.
9. Click *Apply*.

   A progress bar is displayed allowing you to follow the apply process.
10. At the end of the applying process, click *Next* to go back to the *Using the Model* screen.

5.4.4.4.4.2.1 Constraints of Model Use

In order to apply a model to a dataset, it must contain the following variables:

- all the variables with the "input" role, meaning all variables that have not been excluded in the variable selection step (even if all are not used by the model)

  > ⚠ Caution
  >
  > Even if an auto selection has been used, all the initial input variables are mandatory.

- all model keys even if they have been excluded (except the KxIndex generated by Automated Analytics).

Also the dataset generated as a result of the application must contain the following variables:

- the estimator variable (prefixed rr_) if the *Predicted* value is requested in the *Advanced Apply Settings* (which is the default behavior),
- the target variables, to retrieve their values if they exist in the application dataset,
- all model keys.

To make sure that the datasets are consistent, use the mapping feature provided in the panel *Applying the Model*.

5.4.4.4.4.2.2 Deactivating Association Rules

Not all the association rules have the same importance when applying: some of them are relevant whereas other ones do not give any information. That is why Association Rules allows you to deactivate association rules before applying them.

1. Before applying the model, go to the *Association Rules Exploration* panel. For a complete description of this panel, see section Association Rules Exploration.
2. Set a filter.
3. Click the *Search* button to display a set of association rules in the *Association Rules Found* section.
4. Right-click the association rules list. A contextual menu is displayed.

   The first section of the menu, above the separator, allows you to change the application state of the association rules:

The Option...	Allows You to..
Activate the Selected rule(s)	activate the selected rules for the next application
Deactivate the selected rules	deactivate the selected rules for the next application
Activate All	activate all association rules found by Association Rules
Deactivate All	deactivate all association rules found by Association Rules

   The second section of the menu, below the separator, allows you to control the association rules displayed in the table:
   - *Display all the rules* is the default display. All the association rules are displayed; the activated ones as well as the deactivated ones.
   - *Display only the activated rules*
   - *Display on the Deactivated rules*

   The deactivated association rules will not be used in the next application of the model.

5.4.4.4.4.3 Opening a Model

Once saved, models may be opened and reused in the application.

1. On the main screen, click on *Load a Model*.

   The *Opening a Model* screen will appear.
2. Select the file type depending on the format of the model you want to open.
3. Click *Browse*.
4. Select the folder that holds the model that you want to open. The list of models contained in that folder will appear.

   The following table lists the information provided for each model.

Column	Description	Values
Name	Name under which the model has been saved	Character string
Class	Class of the model, that is the type of the model	○ Kxen.Classification: Classification/Regression with nominal target ○ Kxen.Regression: Classification/Regression with continuous target ○ Kxen.Segmentation: Clustering with SQL Mode ○ Kxen.Clustering: Clustering without SQL Mode ○ Kxen.TimeSeries: Time Series ○ Kxen.AssociationRules: Association Rules ○ Kxen.SimpleModel: Classification/Regression and Clustering multi-target models, any other model
Version	Number of the model version when the model has been saved several times	Integer starting at 1
Date	Date when the model has been saved	Date and time in the format yyyy-mm-dd hh:mm:ss
Comment	Optional user defined comment that can be used to identify the model	Character string

5. Select a model from the list
6. Click *Open*.

   The *Using the Model* screen will appear.

5.4.5 Modeling Using KxShell Script

This section details a KxShell script that can be used with the sample files.

Creating the Association Rules Model

The following code describes the method to create an Association Rules model. A default model is created including an association rules transform.

```
#creating a simple model with an association rules transform
createModel KXEN.SimpleModel myModel
myModel.pushTransformInProtocol Default KXEN.ASSOCIATION RULES myASSOCIATION
RULES
```

Setting the Simple Model Parameters

The partition strategy is the only model parameter that has to be set.

```
#setting the model's parameters
myModel.getParameter ""
myModel.changeParameter Parameters/CutTrainingPolicy random
myModel.validateParameter
```

Opening the Reference and the Events Dataset

In this step, the Reference dataset `website_references.csv` is opened with its description `website_references_desc.csv`. The Events dataset `website_transactions.csv`, containing the transactions, is also opened with its description `website_transactions_desc.csv`.

```
#opening the datasets
myModel.openNewStore KXEN.FileStore C:\Program Files\KXEN\KxenComp\Samples
\ASSOCIATION RULES
myModel.newDataSet Training website_references.csv
myModel.readSpaceDescription Training website_references_desc.csv
myModel.newDataSet Events website_transactions.csv
myModel.readSpaceDescription Events website_transactions_desc.csv
```

Setting the Association Rules Transform Parameters

The association rules transform parameters are the following:

- the Transaction parameters, which are used to set the information relative to the Events data source,
- the Reference parameters, which are used to set the information relative to the Reference data source,
- the Association Rules engine parameters. The algorithm used to generate the association rules is Frequent Pattern Vertical (FPV), this is why the Association Rules parameters are located under a branch called FPV.
- ExtraMode [read-write] is a special flag allowing to set the type of outputs that Association Rules will generate. It can take the following six values:
  - No Extra: generates basic outputs, that is the session key, the ID of the rule used to find the consequent and the consequent itself.
  - Optimized by KI: generates basic outputs. If more than one rule give the same consequent for a session, the rule presenting the best KI will be selected.
  - Optimized by Confidence: generates basic outputs. If more than one rule give the same consequent for a session, the rule presenting the best Confidence will be selected.
  - Full Description: generates the extended outputs, that is the session key, the rule ID, the consequent, the antecedent, the KI, the confidence and the rule support.
  - Full Description and Optimized by Confidence: generates the extended outputs. If more than one rule give the same consequent for a session, the rule presenting the best Confidence will be selected.
  - Full Description and Optimized by KI: generates the extended outputs. If more than one rule give the same consequent for a session, the rule presenting the best KI will be selected.
- DateColumnName [Sequence mode only] is the column in which the date is stored.

- SequencesMode [Sequence mode only] is a flag specifying if the Sequence mode of Association Rules is activated. *True* (or *1*) means that the Sequence mode is activated, *False* (or *0*) means that the Sequence mode is deactivated.

Transactions Section

- Transactions/LogSpaceName indicates Automated Analytics role set for the Transactions space name.
- Transactions/TIDColumnName indicates the name of the transactions key variable.
- Transactions/ItemIDColumnName indicates the name of the item variable.

References Section

References/TIDColumnName indicates the name of the reference key variable.

Association Rules Engine

The Association Rules engine can be used to generate general association rules or specific association rules to replace the missing values. This second section can only be used by other features.

Default Use

ARulesEngineParameters/FPV/Activated indicates whether the FPV algorithm is activated or not. This value should always be set to true (default = true).

ARulesEngineParameters/FPV/MinimumSupport gives the minimum support required for a rule. With a value > 1, we consider the number of sessions. With a value between 0 and 1, we consider a percentage of the number of sessions (default = 1, required value > 0).

ARulesEngineParameters/FPV/MinimumConfidence gives the minimum threshold for the confidence of a rule (default = 0.5, required a value between 0 and 1).

ARulesEngineParameters/FPV/MaxLength saves the maximum size of a rule. A rule has a minimum size equal to 2: the antecedent plus the consequent (default = 4, required value >= 2).

ARulesEngineParameters/FPV/SearchMethod indicates the search method the FPV has to use. When the value equals 1, the Association Rules engine uses the basic FPV algorithm. When the value equals 2, the Association Rules uses a modified FPV algorithm: the rules are generated regarding the frequence order of their consequent item and the memory used is limited (default = 2, required value equals 1 or 2).

ARulesEngineParameters/FPV/SkipDerivedRules indicates to FPV to generate only the primary rules and thus skip the derived rules. For instance, given R1: X => Z and R2: X & Y => Z two association rules, R1 is called a primary rule and R2 a derived rule. The option SkipDerivedRules is only available with a SearchMethod parameter equal to 2 (default = true).

ARulesEngineParameters/FPV/SkipReducibleItemsets indicates if FPV has to generate the reducible itemsets (default = false). An itemset is non reducible if one of its items cannot be removed without changing the transaction space.

ARulesEngineParameters/FPV/TidSet indicates the storage method of the transactions associated to an itemset (default = 2, required a value equal to 1 or 2).

ARulesEngineParameters/FPV/ChunkSize saves the size of the chunks (in number of sessions) use by FPV to import and generate the rules. With a value equal to 0, the chunk strategy will not be used and ALL the sessions will be imported before generating the rules (default = 0, required value >= 0).

Description of the Related Script

```
#setting the transform parameters
myASSOCIATION RULES.getParameter ""
myASSOCIATION RULES.changeParameter "Parameters/Transactions/LogSpaceName" Events
myASSOCIATION RULES.changeParameter "Parameters/Transactions/TIDColumnName"
SessionID
myASSOCIATION RULES.changeParameter "Parameters/Transactions/ItemIDColumnName"
Page
myASSOCIATION RULES.changeParameter "Parameters/References/TIDColumnName"
SessionID
myASSOCIATION RULES.changeParameter "Parameters/ARulesEngineParameters/FPV/
Activated" true
myASSOCIATION RULES.changeParameter "Parameters/ARulesEngineParameters/FPV/
MinimumSupport" 0.01
myASSOCIATION RULES.changeParameter "Parameters/ARulesEngineParameters/FPV/
MinimumConfidence" 0.5
myASSOCIATION RULES.changeParameter "Parameters/ARulesEngineParameters/FPV/
MaxLength" 4
myASSOCIATION RULES.changeParameter "Parameters/ARulesEngineParameters/FPV/
SkipDerivedRules" true
myASSOCIATION RULES.validateParameter
```

Specify Filters on the Association Rules

This section is used to specify filters on the items that should or should not appear as consequents in the generated association rules.

ARulesEngineParameters/RulesGenerationFilters/ConsequentFilters/IncludedList contains the list of the items that should appear as consequents.

ARulesEngineParameters/RulesGenerationFilters/ConsequentFilters/ExcludedList contains the list of the items that should not appear as consequents.

The list of the included items is taken into account before the list of the excluded items. The list of the excluded items is used if the list of the included items is empty.

Description of the Related Script

```
#_____________________________________________________________
#set the list of the consequents to keep
myInfiniteInsight™ Modeler - Association Rules.getParameter ""
myInfiniteInsight™ Modeler - Association Rules.bindParameter "Parameters/
ARulesEngineParameters/RulesGenerationFilters/ConsequentFilters/IncludedList"
ConsParam
ConsParam.insert "/HM_Body.tmpl" fakeParam_0
myInfiniteInsight™ Modeler - Association Rules.validateParameter
```

```
delete ConsParam
```

Missing Values Specific Additional Parameters

ARulesEngineParameters/FPV/DefaultTargetEnable indicates whether to generate the rules that have the most frequent item of a group as the consequent part. This parameter should be set to false, because this parameter is useless to deduce the rules used to fill the missing values (default = false).

ARulesEngineParameters/FPV/UseMissInfo indicates that the confidence of the rules will be computed taking into account the missing information for each variable. This parameter is only used when the Association Rules engine is used to replace the missing values.

Activate the Sequence Mode

```
#Activating the Sequence Mode
myKAR.getParameter ""
myKAR.changeParameter "Parameters/SequencesMode" true
myKAR.changeParameter "Parameters/DateColumnName" Time
myKAR.validateParameter
```

5.4.5.1 Learning the Model

Once all the parameters have been set, the learning step is launched.

```
#learning the model
myModel.sendMode learn
```

5.4.5.2 Applying the Model

Once the model has been generated, you can apply it. Before applying the model you can change the value of the ExtraMode parameter (see above in section Setting the Association Rules Transform Parameters).

```
#_____________________________________________________________
#apply the model with default ExtraMode
mymodel.newDataSet ApplyIn website_references.txt
mymodel.openNewStore Kxen.FileStore Saved
mymodel.newDataSet ApplyOut out_default_website.csv
mymodel.sendMode apply
```

Several association rules can be deactivated before applying the model because they are not relevant. There are two ways to realize that:

- Specify a list of association rule identifiers to deactivate in the path *Parameters/ARulesEngineParameters/ApplyActivationOptions/RulesExcludedList*
- Specify a list of interest items, that are items for which all the association rules having one of these items as a consequent should be activated, in the path *Parameters/ARulesEngineParameters/ApplyActivationOptions/ActivatedConsequentsList*. The Association Rules that have a consequent not included in this list will not be activated.

> **i Note**
>
> The Association Rules having their consequent item included in the interest items list but having their identifier registered in the excluded list, these association rules will be excluded for the next application.

Related Script

```
#_____________________________________________________________________
#deactivate several rules before applying again
#First specify a list of association rules to activate by specifying the interest
# consequent items
myKAR.getParameter ""
myKAR.bindParameter "Parameters/ARulesEngineParameters/ApplyActivationOptions/
ActivatedConsequentsList" ConsParam
ConsParam.insert "/shop/order1.tmpl" fakeParam_0
ConsParam.insert "/shop/order2.tmpl" fakeParam_1
ConsParam.insert "/shop/order3.tmpl" fakeParam_2
ConsParam.insert "/shop/order4.tmpl" fakeParam_3
ConsParam.insert "/shop/order5.tmpl" fakeParam_4
#Then, specify a list of association rules to exclude by specifying their
identifiers
myKAR.bindParameter "Parameters/ARulesEngineParameters/ApplyActivationOptions/
RulesExcludedList" RulesParam
RulesParam.insert "Rule850" fakeParam_5
RulesParam.insert "FakeRule" fakeParam_7
myKAR.validateParameter
delete ConsParam
delete RulesParam
```

5.4.5.3 Saving the Model

Once a model has been generated, you can save it. Saving it preserves all the information that pertains to that model that is the modeling parameters, the association rules and so on.

> **i Note**
>
> The directory, in which the model is saved, must preexist.

```
#saving the model in a Saved directory
myModel.openNewStore KXEN.FileStore Saved
myModel.saveModel ASSOCIATION RULES.txt "ASSOCIATION RULES on website_references
with <Page> as the item variable."
```

5.4.5.4 Opening an Existing Model

Once saved, models may be opened and reused in SAP Predictive Analytics.

For this scenario:

```
#restoring the last saved model
createStore KXEN.FileStore myRestoreStore
myRestoreStore.openStore "Saved"
setDefaultUserPassword "" ""
myRestoreStore.restoreLastModelD myModel
```

5.4.5.5 Summarizing the KxShell script

Below is an example of a complete script you can build.

```
#creating a simple model with an association rules transform
createModel KXEN.SimpleModel myModel
myModel.pushTransformInProtocol Default KXEN.ASSOCIATION RULES myASSOCIATION
RULES
#setting the model parameters
myModel.getParameter ""
myModel.changeParameter Parameters/CutTrainingPolicy random
myModel.validateParameter
#opening the data sets
myModel.openNewStore KXEN.FileStore C:\Program
Files\KXEN\KxenCompV3\Samples\ASSOCIATION RULES
myModel.newDataSet Training website_references.csv
myModel.readSpaceDescription Training
website_references_desc.csv
myModel.newDataSet Events website_transactions.csv
myModel.readSpaceDescription Events
website_transactions_desc.csv
#setting the transform parameters
myASSOCIATION RULES.getParameter ""
myASSOCIATION RULES.changeParameter "Parameters/Transactions/LogSpaceName"
Events
myASSOCIATION RULES.changeParameter
"Parameters/Transactions/TIDColumnName" SessionID
myASSOCIATION RULES.changeParameter
"Parameters/Transactions/ItemIDColumnName" Page
myASSOCIATION RULES.changeParameter "Parameters/References/TIDColumnName"
SessionID
myASSOCIATION RULES.changeParameter
"Parameters/ARulesEngineParameters/FPV/Activated" true
myASSOCIATION RULES.changeParameter
"Parameters/ARulesEngineParameters/FPV/MinimumSupport" 0.01
myASSOCIATION RULES.changeParameter
"Parameters/ARulesEngineParameters/FPV/MinimumConfidence"
0.5
myASSOCIATION RULES.changeParameter
"Parameters/ARulesEngineParameters/FPV/MaxLength" 4
myASSOCIATION RULES.changeParameter
"Parameters/ARulesEngineParameters/FPV/SkipDerivedRules"
true
myASSOCIATION RULES.validateParameter
#Activating the Sequence Mode
myASSOCIATION RULES.getParameter ""
myASSOCIATION RULES.changeParameter "Parameters/SequencesMode" true
myASSOCIATION RULES.changeParameter "Parameters/DateColumnName" Time
```

```
myASSOCIATION RULES.validateParameter
#learning the model
myModel.sendMode learn
#APPLY the association rules
#apply the model with default ExtraMode
mymodel.newDataSet ApplyIn website_references.txt
mymodel.openNewStore Kxen.FileStore Saved
mymodel.newDataSet ApplyOut out_default_website.csv
mymodel.sendMode apply
#apply only some of the association rules
#deactivate several rules before applying again
#First specify a list of association rules to activate by specifying the interest
# consequent items
myASSOCIATION RULES.getParameter ""
myASSOCIATION RULES.bindParameter "Parameters/ARulesEngineParameters/
ApplyActivationOptions/ActivatedConsequentsList" ConsParam
ConsParam.insert "/shop/order1.tmpl" fakeParam_0
ConsParam.insert "/shop/order2.tmpl" fakeParam_1
ConsParam.insert "/shop/order3.tmpl" fakeParam_2
ConsParam.insert "/shop/order4.tmpl" fakeParam_3
ConsParam.insert "/shop/order5.tmpl" fakeParam_4
#Then, specify a list of association rules to exclude by specifying their
identifiers
myASSOCIATION RULES.bindParameter "Parameters/ARulesEngineParameters/
ApplyActivationOptions/RulesExcludedList" RulesParam
RulesParam.insert "Rule850" fakeParam_5
RulesParam.insert "FakeRule" fakeParam_7
myASSOCIATION RULES.validateParameter
delete ConsParam
#applying with the new options
mymodel.newDataSet ApplyIn website_references.txt
mymodel.openNewStore Kxen.FileStore Saved
mymodel.newDataSet ApplyOut out_OrdersAsConsequent_ExceptRule850_website.csv
mymodel.sendMode apply
#saving the model in a Saved directory
myModel.openNewStore KXEN.FileStore Saved
myModel.saveModel ASSOCIATION RULES.txt "ASSOCIATION RULES on website_references
with
<Page> as the item variable."
```

6 Social

6.1 About Social

This section is addressed to people who want to evaluate or use the Social component of Automated Analytics.

Before reading this section, you should read the sections *Classification/Regression* and *Segmentation/Clustering* that present respectively:

- An introduction to Automated Analytics
- The essential concepts related to the use of Automated Analytics features

6.2 Social

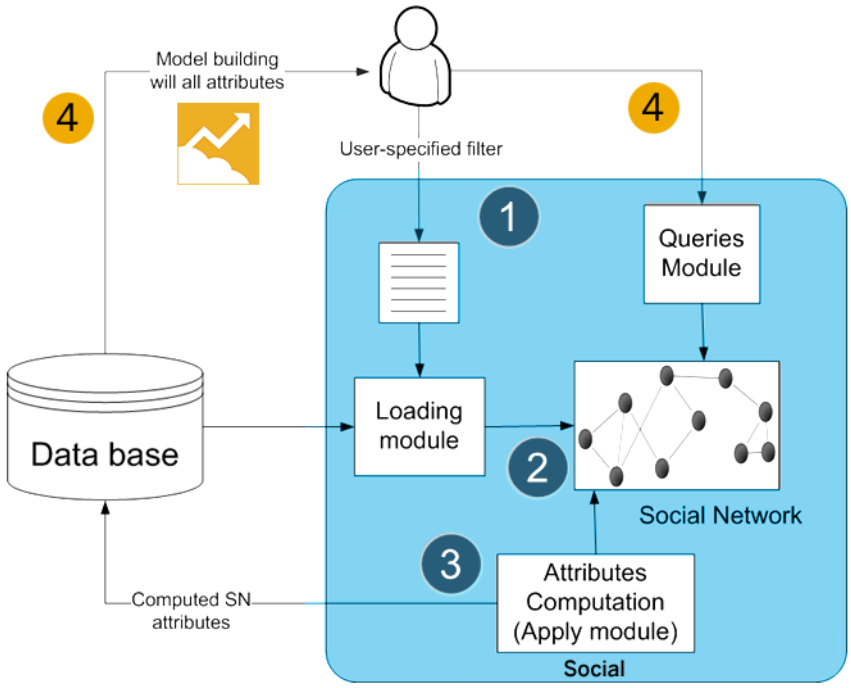
6.2.1 Scalability

Source databases are often very large (in telecom, call detail records can reach terabytes of data each month, containing billions of entries) and a great variety of social networks can be extracted from them. Since the user might want to work with several social networks at the same time, it is important to have a scalable and light-weighted representation.

Huge volumes also imply time issues. It must be possible to access a node from its key or Id in linear time. Metrics computations on a set of input nodes also need to be computed in reasonable time.

6.2.2 Overview of Social Functionalities

The following scheme describes the overall Social functionalities:



The loading module has to be configured by the user, indicating where to find the nodes list in the database, and the graphs to be extracted must be defined. The application uses these loading specifications to create a graph set during the learn process and stores them in the model. The graphs are then available for the computation of social variable.

A number of attributes to be generated by the application can be specified during the apply process. The social attributes derived from the graph structure complement the view of entities from the initial database.

6.2.3 Data Sources Supported

Automated Analytics supports the following data sources:

- Text files (also called flat files) in which the data are separated by a delimiter, such as commas in `.csv` (Comma Separated Value) files.

  > **! Restriction**
  >
  > When accessing data in `.csv` files, Automated Analytics only supports `CR` + `LF` (common on Microsoft Windows) or `LF` (common on Linux) for line breaks.

- Database management systems that can be accessed using ODBC.

> **i Note**
>
> For the list of supported ODBC-compatible sources, see the SAP Product Availability Matrix (PAM) at http://service.sap.com/sap/support/pam.
>
> For more information about using SAP HANA, see the related information below.
>
> To configure Automated Analytics modeling tools to access data in your database management system, refer to the guide *Connecting your Database Management System on Windows* or *Connecting your Database Management System on Linux*.

- SAS files

Related Information

SAP HANA as a Data Source [page 8]

6.2.3.1 SAP HANA as a Data Source

You can use SAP HANA databases as data sources in Data Manager and for all types of modeling analyses in Modeler: Classification/Regression, Clustering, Time Series, Association Rules, Social, and Recommendation.

SAP HANA tables or SQL views	found in the *Catalog* node of the SAP HANA database
All types of SAP HANA views	found in the *Content* node of the SAP HANA database. An SAP HANA view is a predefined virtual grouping of table columns that enables data access for a particular business requirement. Views are specific to the type of tables that are included, and to the type of calculations that are applied to columns. For example, an analytic view is built on a fact table and associated attribute views. A calculation view executes a function on columns when the view is accessed. **! Restriction** • Analytic and calculation views that use the variable mapping feature (available starting with SAP HANA SPS 09) are not supported. • You cannot edit data in SAP HANA views using Automated Analytics.

Smart Data Access virtual tables	Thanks to Smart Data Access, you can expose data from remote sources tables as virtual tables and combine them with HANA regular tables. This allows you to access data sources that are not natively supported by the application, or to combine data from multiple heterogeneous sources. **⚠ Caution** To use virtual tables as input datasets for training or applying a model or as output datasets for applying a model, you need to check that the following conditions are met: • The in-database application mode is not used. • The destination table for storing the predicted values exists in the remote source before applying the model. • The structure of the remote table, that is the column names and types, must match exactly what is expected with respect to the generation options; if this is not the case an error will occur. **⚠ Caution** In Data Manager, use virtual tables with caution as the generated queries can be complex. Smart Data Access may not be able to delegate much of the processing to the underlying source depending on the source capabilities. This can impact performance.

Prerequisites

You must know the ODBC source name and the connection information for your SAP HANA database. For more information, contact your SAP HANA administrator.

In addition to having the authorizations required for querying the SAP HANA view, you need to be granted the `SELECT` privilege on the `_SYS_BI` schema, which contains metadata on views. Please refer to SAP HANA guides for detailed information on security aspects.

6.2.4 Data Security for Local Model Storage

.

Social nework and Recomendation models can contain personal data. While on the modelling server, this data is subject to the server security, however, if models are copied to a folder on a local machine, then access to this data can present a security issue.

It is recommended that models copied to a local machine are stored in a password protected location, for example a directory path in Windows could be: `c:\Users\<your user name>`.

6.2.5 Concepts

The main idea behind the Social component is to extract and use implicit structural relational information stored in different kinds of datasets, and thus improve decision and prediction capacities of the models.

The following sections present the different concepts used in the application.

6.2.5.1 Directed / Undirected

A graph can be undirected (representing a symmetric link: A is friend of B, and B is friend of A) or directed (representing a non-reflexive relation: A called B; client A bought product C). Note that the number of links can vary between a directed and an undirected graph.

6.2.5.2 On-/Off-net

Off-net nodes are nodes in the graph for which no additional information (age, marital status, income...) is available. These nodes cannot be joined with any database to add descriptive attributes, but represent themselves a bit of information. In the case of a telephone company, for instance, off-net nodes represent customers from another company and their presence in a node neighborhood can help to predict the churn rate.

On the contrary, on-net nodes are those for which extra information is available.

On/off-net information can be seen as meta attributes for the nodes.

6.2.5.3 Node Degree

The degree of a node is the number of connections it has to other nodes.

6.2.5.4 Mega-hub

A node can be considered as a mega-hub when its degree is above a certain threshold and hence has a negative contribution regarding the graph global modularity and its clustering potential. A node connected to a big

amount of neighbors tends to regroup distinct communities in one big community, hiding the actual underlying clusters.

Removing these highly connected nodes can dramatically improve processing time, and put away irrelevant nodes from the analysis. Because of their high degree, such nodes are likely to create 'artificial cluster' in the network, making it harder to detect communities, and hiding the real structure of the graph.

Such mega-hub nodes can be found in telecommunication networks (service number, call center, taxis), transactional networks (best sellers, head of tail products). These nodes have a large amount of links, but are semantically insignificant.

Removing mega-hubs can have positive impacts on:

- increasing the resulting modularity
- reducing the community processing time (because community assignment involve looking at neighbors)
- reducing the pairing detection time (because neighbors of mega-hub will have very large second circle)
- reducing the size of the largest community (wrongly induced by the fact that some nodes are connected to 'many' nodes)

Related Information

Communities Detection [page 314]
Node Pairing [page 314]

6.2.6 Dataset Types

Three types of datasets can be used in Social:

- Link Dataset (mandatory)
- Descriptive Attributes Dataset (optional)
- Identifiers Conversion Dataset (optional)

Related Information

Link Dataset [page 309]
Descriptive Attributes Dataset [page 309]
Identifiers Conversion Dataset [page 309]

6.2.6.1 Link Dataset

This dataset is used to specify where to find nodes and what information is to be used to create links from raw data.

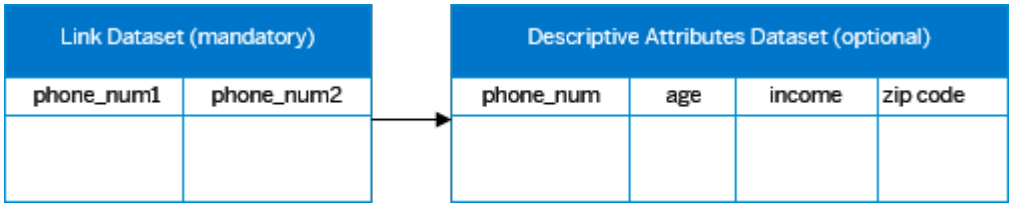
One of the key features of the component is its ability to generate graphs from different sorts of data stored in operational databases. This can be done in one sweep with the raw data, applying user-defined filters. This means that for each case of the input dataset, the application checks if the row can be used to add a link in one of the specified graphs, or if it has to be ignored.

6.2.6.2 Descriptive Attributes Dataset

The application offers the possibility to add descriptive attributes to the nodes by associating additional variables, such as age, marital status, gender or profession. This can be done by specifying the column containing this information to be later retrieved during the loading process.

Note that this information is joined via the Ids found in this dataset and the ones used in the graph (found in the link dataset, or indirectly through the identifiers conversion dataset).

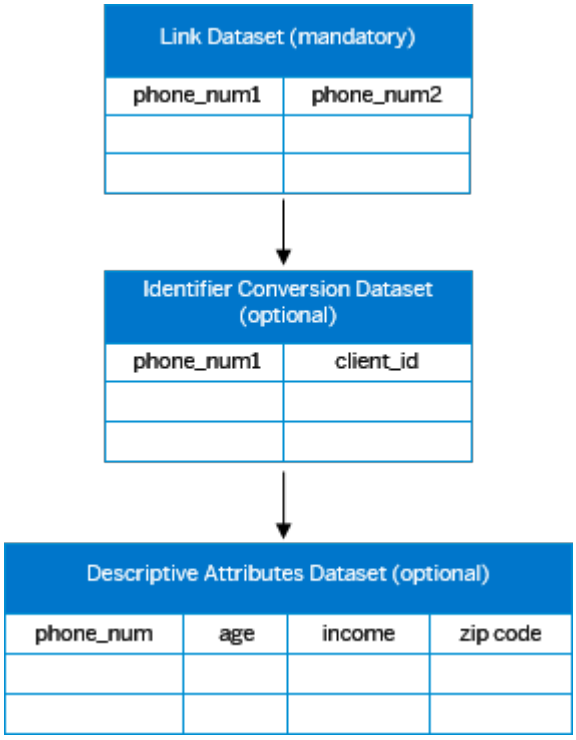
If the Descriptive Attributes Dataset does not contain the same Ids as the Link Dataset, you need an Identifiers Conversion Dataset in order to join these two tables.



6.2.6.3 Identifiers Conversion Dataset

This is a two-column dataset employed to map an Id from the link dataset to an Id used as node key. It can happen that an Id found in the link dataset is not the one desired in the graph.

For instance, if a graph is built from a call detail record (phone line numbers), you might want to be able to translate it automatically to client Ids. This is why the application provides a convenient way to do such a translation (conversion) during the loading process. Moreover, this table allows inferring the on-/off-net information: an Id found in the identifiers conversion dataset is considered as on-net, otherwise it is considered as off-net.



6.2.7 Graph Types

The following sections present the different graph types used in the application.

6.2.7.1 Contacts

This graph type describes communication contacts between Node1 and Node2 (in this case phone numbers):

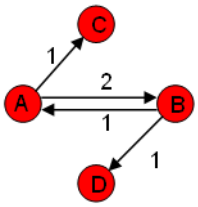
Node 1	Node 2	Class	Date	Duration
0143525691	0631527812	VOICE	12/03/2007	0 :01 :12
0143525691	0631527812	VOICE	13/03/2007	0 :04 :13
0499472604	0325363819	SMS	14/03/2007	0 :12 :52
0143525691	0631527812	VOICE	15/03/2007	0 :05 :42
0186076517	0631527812	SMS	16/03/2007	0 :05 :09
0189340041	0154876556	MMS	17/03/2007	0 :36 :04

The same filtering condition as for the option Links Only (see page 14) can be used, but an additional condition is available: nodes are linked, if (and only if) an event between them occurs at least `n` times (see the blue examples in the table).

Examples of gaphs:

- Graph of clients with at least 3 voice phone calls
- Graph of clients with at least 5 sms communications between 03/01/2007 and 04/01/2007
- Graph of clients with at least 2 voice communications of 120 seconds between 03/01/2007 and 04/01/2007

Node 1	Node 2
A	B
A	C
B	D
A	B
B	A



(directed graph)

Related Information

Links Only [page 313]

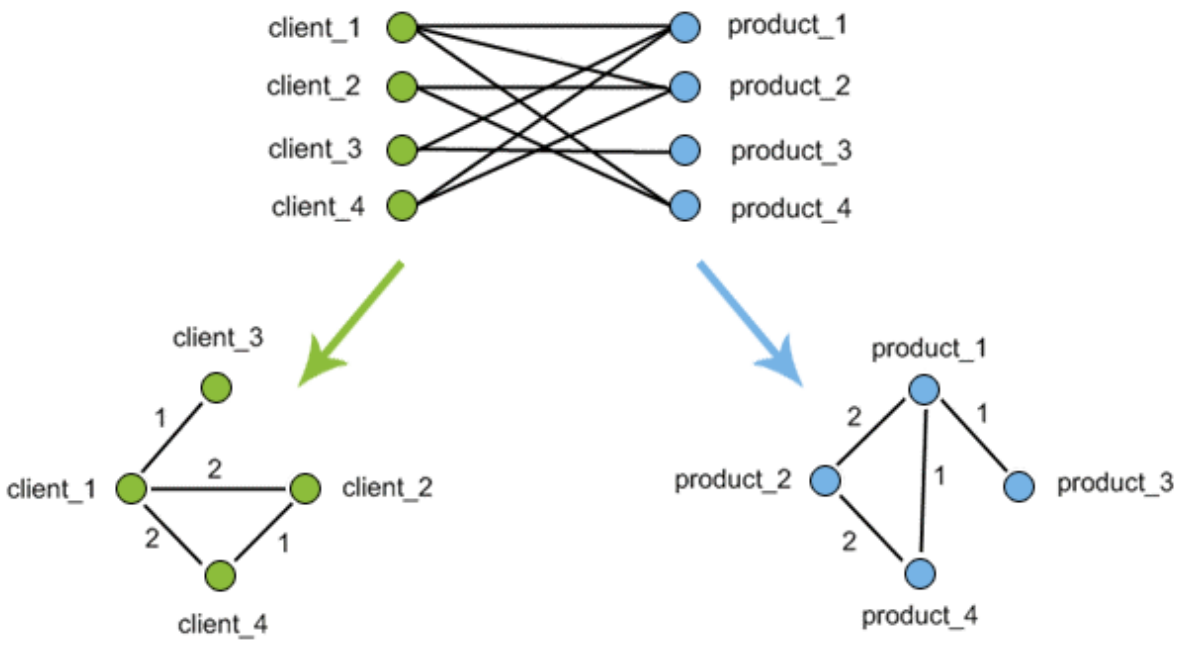
6.2.7.2 Transactions

To build graphs of the type Transactions, the link dataset needs to describe links between two distinct populations, for example clients and products:

Client_Id	Product_Id	Date
1	1	12/03/2007
1	2	13/03/2007
1	4	14/03/2007
2	2	15/03/2007
2	4	16/03/2007
...	...	...

The load process builds a bipartite graph, where all the links are between the two distinct populations. This graph allows deriving two unipartite graphs for each population, such as the graph of clients who bought `n`

products in common and the graph of products having more than `n` clients in common. The links of such graphs can be labeled with the count of common products/clients, as shown below:



Examples of graphs:

- Graph linking the customers who bought at least 3 products in common
- Graph linking the products having at least 8 clients in common

6.2.7.3 Nearest Neighbors

This graph type is built from a dataset containing entities and their respective distance. Given a parameter `k`, the aim is to obtain a graph linking every node with its `k`-nearest neighbors:

Node 1	Node 2	Distance	Date
53123	63152	10	12/03/2007
68129	15487	12	13/03/2007
43261	32536	6	14/03/2007
12993	22866	2	15/03/2007
60765	63152	75	16/03/2007
18934	48765	23	17/03/2007

The same filtering condition as for the option Links Only can be used, plus the `k` parameter.

Note that this type of graph is always directed. This can be easily understood with the following example, where a graph is built and each node is connected to its single nearest neighbor:

A is the nearest neighbor for B, and is also the nearest neighbor for C. In an undirected graph, A would be linked to two different neighbors. That is to say, A would have two nearest neighbors, which would be a contradiction. This is due to the fact that "is the nearest neighbor of" is not a symmetric relation.

Example of graphs:

- Graph linking every node to its 3 nearest neighbors

Related Information

Links Only [page 313]

6.2.7.4 Links Only

This is the simplest graph type, where all links information is already computed and explicitly specified:

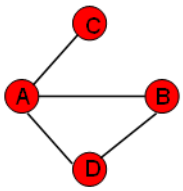
Node 1	Node 2	Class	Date
0143525691	0631527812	VOICE	12/03/2007
0467892154	0154876556	VOICE	13/03/2007
0499472604	0325363819	SMS	14/03/2007
0282812993	0228627344	SMS	15/03/2007
0186076517	0631527812	MMS	16/03/2007
0189340041	0154876556	VOICE	17/03/2007

Several graphs can be built on the fly from the same dataset by creating different filters, such as a filter by date, a filter on columns with discrete values (sms, voice) or a filter on columns with continuous values.

Examples of graphs:

- Graph of clients with only voice phone calls
- Graph of clients with SMS communications between 03/01/2007 and 04/01/2007

Node 1	Node 2
A	B
A	C
B	D
D	A



6.2.8 Graph Set Customization

The described graphs can be easily defined and several graphs can be extracted, from the same dataset and in one single sweep.

6.2.9 Communities Detection

This visualization mode displays the nodes as members of communities. Each community is shown with a shape of a different form and a different color for an easy distinction. According to their connections to other nodes of the same or another community, the nodes can play different roles inside their communities. Social distinguishes between four roles:

- *Bridge Node*: these nodes connect communities, thus acting like a bridge. They have many connections with other communities, but only few in their own community.
- *Local Node*: these nodes have many connections inside a given community and only few connections to other communities.
- *Social Node*: these nodes have many connections inside their own community and to other communities.
- *Passive Node*: these nodes have few connections inside their own community and to other communities.

6.2.9.1 Supercommunities

The application links smaller sub-communities with similar interests or behaviors to create larger “supercommunities”. In many social networks, an individual or node may only be connected to a limited number of other nodes (for example, the average phone number is highly connected to about 10 other phone numbers). If you are interested in spotting trends or implementing a viral marketing program like friend-get-friend, you need to understand the big picture.

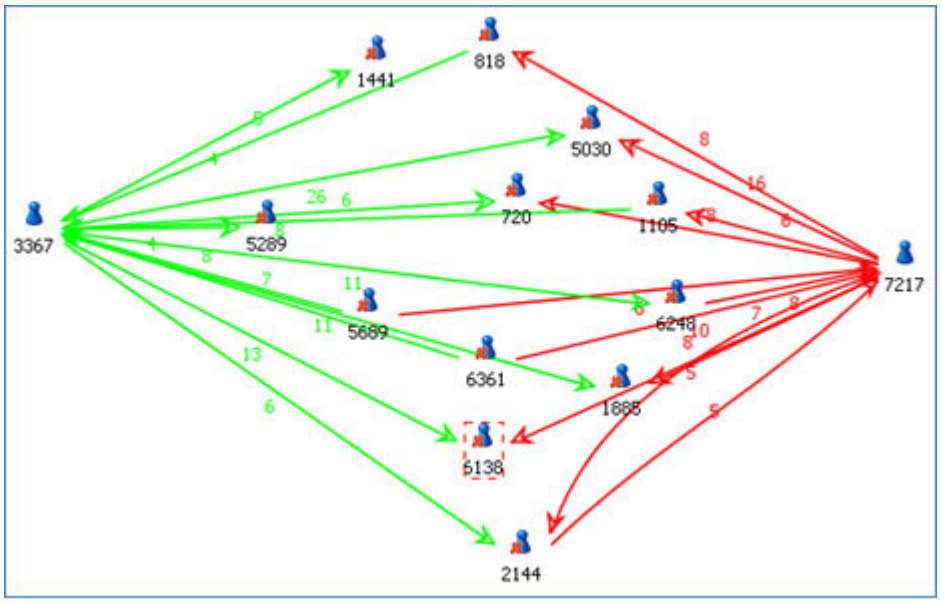
With supercommunity detection, business users can visually explore social networks, easily identify supercommunities with strong social influence and introduce social attributes (community and supercommunity statistics, centrality measures, and so on) that can strengthen predictive models for viral marketing initiatives.

6.2.10 Node Pairing

Node pairing allows identifying an item (customer, product, etc.) present in two different graphs (thus with two different nodes), by means of its common neighbors. This technique can be used, for instance, by a telephone company to detect "rotational churn" (locate clients who change their phone subscription on a regular basis to benefit from welcome offers).

To retrieve an item present in two different graphs: links between two nodes can either be labeled with the count of their common neighbors (count mode) or the ratio common_neighbors/average_neighbors_count

(ratio mode). This ratio value can be seen as a match probability: a ratio of 1 means that the two considered nodes have 100% of neighbors in common and are very likely to represent one and the same item.



Related Information

Nearest Neighbors [page 312]

6.2.11 Derived Attributes

Once the graphs are generated, Social can apply them to a new dataset in order to derive social variable from the generated graph set.

Neighbors Mode

The derived attribute is a simple count of neighbors for the considered node. It is also possible to know the sum/ratio of off-net neighbors in the direct neighborhood. A directed graph generates two columns, one for the in-degree and one for the out-degree.

Circle Mode

The aim is to extract the information contained in the direct neighborhood of a node by aggregating properties from its direct neighbors. The neighborhood can then be profiled by computing the mean of continuous

variables (age, income) or the ratio/count of discrete properties (marital status, sex). Directed graphs generate two columns, one for the in-degree and one for the out-degree.

Centrality Analysis

This indicator measures the role and the importance of a node in the graph and its influence on its neighborhood.

Community Mode

The objective is to detect existing communities: the derived attributes mark out potential communities and allow roles identification (nodes behavior inside and outside their community).

Node Pairing

The generated attributes can be used to identify nodes in two different graphs that represent the same item (customer, product, etc.). Node pairing mode is useful to detect rotational churn or multi-SIM users for instance.

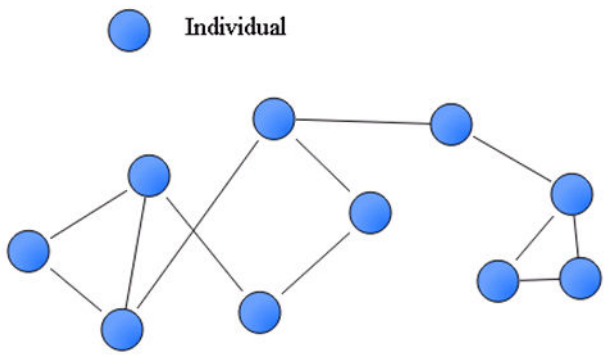
Related Information

Communities Detection [page 314]
On-/Off-net [page 307]
Nearest Neighbors [page 312]
Directed / Undirected [page 307]

6.3 Introduction to Social Networks

6.3.1 Definition

A social network is a social structure represented in the form of a graph, made of nodes and links.



The nodes are the actors/items within a network (individuals, customers, products, organizations...). The links are the relations, or social interactions, between them.

A node can have several variables associated to it (name, address, profession, age...). There can be many kinds of ties between nodes, depending on the context and what the graph actually represents, such as ideas, friendship, business collaboration, diseases, hypertext links, phone calls, financial exchange, etc.

These links can be directed or undirected depending on the type of relation, symmetric or not. So in its simplest form, a social network is a map of all the relevant links between the nodes being studied.

Social networks can be used to represent many kinds of networks: informational (web, blogs), communicational (phone calls, emails), social (social networking, illness), technological (power grid, roads, internet router), financial (transactions), etc.

6.3.2 Application and Data Mining

The graph representation offers a visual and mathematical analysis of its structure and interactions. Social network analysis has been used to approach a lot of problems such as community identification, diffusion in graphs (product adoption, epidemiology), graph evolution or influence of an individual within a community (leader vs. follower).

Social network analysis produces an alternate view, where the attributes of individuals are less important than their relationships and ties with other actors within the network. In a data mining context, it can be seen as a way to derive new attributes from the social network, to enrich existing data and obtain a better predictive model. These additional variables can be descriptive statistics from a node direct neighborhood or more complex metrics, related to the centrality of the node in the graph or the community.

6.4 Modeling with Social

6.4.1 Starting Social

1. Click *Start > Programs > SAP Business Intelligence > SAP Predictive Analytics Desktop > SAP Predictive Analytics*.
2. Click *Social > Create a Social Network Analysis*.

6.4.2 Step 1 - What Type of Graph to Create ?

There are three ways to create a social network analysis graph:

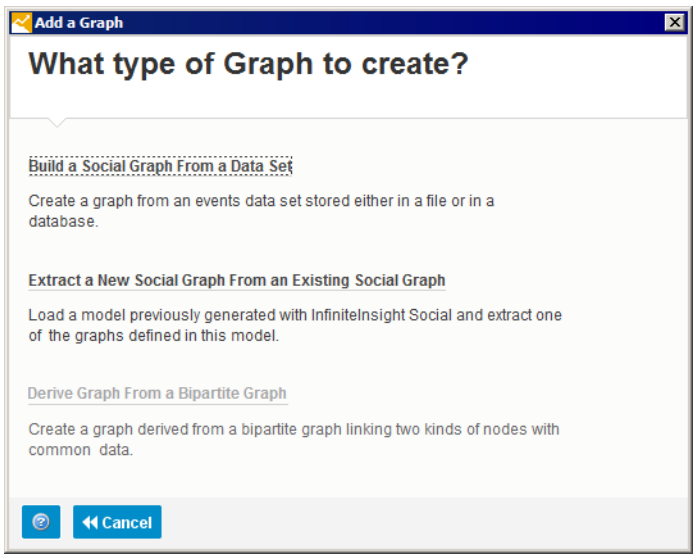
- *Build a Social Graph from a Dataset*: this option allows you to select a dataset and use it to generate a new graph. The source dataset may be stored either in a file or in a database.
- *Derive Graph from a Bipartite Graph*: this option allows you to derive a graph from an existing bipartite graph, that is, to create a graph linking the members of one population based on the links they have with the other population. For example, in a bipartite graph linking customers and products, you can derive a graph linking the products that have been bought by the same customer, as shown in the schema below.

  > **i Note**
  >
  > If you want to create recommendations using Social, you need to define a bipartite graph linking customers and products and then derive a graph linking the products together.

- *Extract a New Social Graph from an Existing Social Graph*: this option allows you to load a model previously generated with Social and extract one of the graphs defined in this model.

> **⚠ Caution**
>
> This option is only available when a bipartite graph has already been created in the current model, either from a dataset or loaded from an existing model.

To select the type of graph you want to create, click the selected option in the pop-up window.



Related Information

Building a Graph from a Dataset [page 319]
Deriving a Graph from a Bipartite Graph [page 322]
Extracting a Graph from an Existing Graph [page 323]

6.4.2.1 Building a Graph from a Dataset

You can either select a new dataset or select a dataset that you have already defined. The original data can be in tab or comma delimited text format, extracted from an ODBC source, or in proprietary format (SPSS, Excel...).

> **i Note**
>
> The second option is only available when clicking the *Add Graph* button in the *Graph Definitions* panel.

6.4.2.2 Selecting a Link Dataset

The link dataset has to be a table with at least two columns: a source node and a target node. Additionally, there might be a date column, a distance column (indicating the relative distance between the nodes) and other columns giving further information on the nodes.

To select a new dataset, proceed as follows:

1. Check *New Dataset*.
2. In *Data Type*, select the type of data you want to access.

   > **i Note**
   >
   > Some type of data require specific licenses.

3. Click the *Browse*button to select the folder or database containing the data.

   > **i Note**
   >
   > In case of a protected database, you will need to enter the user name and password when selecting the database.

   Whenever allowed by the data format, the following options are available in the *Data Source Selection* panel:

   -  : go to parent directory.
   -  : create a new folder.
   -  : delete the selected folder (possible for empty folders only).

4. Click the *Browse* button to select the dataset you want to use.

   Analytical datasets created with the modeling assistant are displayed at the top of the list.

   The magnifier button allows you to view the data and compute statistics over the selected dataset.

   Advanced Settings allows you to set advanced settings such as the sub-sampling or the file settings.
5. Click *Next*.
6. Check *Existing Dataset*.
7. In the list, select the dataset you want to use.

   Each dataset is defined by the name of the file or table, the source location and the type of data.
8. Click *Next*.

6.4.2.3 Describing the Link Dataset

You can describe the data by automatically analyzing the dataset structure or opening a description file.

- *Analyze*: scans the first hundred lines of data and provides the user with an initial guess of the data file description. For text data files, it is recommended that the first row contain the variable names. In order to avoid all bias, we encourage you to mix up your dataset before performing this analysis. After generating a description, you can save it for future use by clicking *Save Description*.
- *Open description*: loads a previously saved description.These metadata can be loaded in a tab or comma delimited text format, from an ODBC source, or in a proprietary format.
- *View Data*: displays the first hundred lines of the dataset to help you validate the description.

> **i Note**
>
> When using an existing dataset, the description step is skipped since the data has already been described for a previous graph.

> **⚠ Caution**
>
> The value of a node identifier must always be set as nominal and the storage type must be a string.

Each variable or attribute is described by the following fields:

Name	Description	Possible Values	Value Description
Name	variable name		
Storage	type of values stored in the variable	*number*	Figures, or numerical values on which operations may be performed (a phone number or an account number should not be considered numbers)
		string	Alphanumeric character strings

Name	Description	Possible Values	Value Description
		integer	Figures, or numerical integer values on which operations may be performed
		datetime	Dates and times expressed in the following formats: • YYYY-MM-DD HH:MM:SS • YYYY/MM/DD HH:MM:SS
		date	Dates expressed in the following formats: • YYYY-MM-DD • YYYY/MM/DD
		angle	Angles refer to latitude and longitude, which are angles used as geographic coordinates that allow defining a point in a map.
Value	type of value	*continuous*	variables whose values are numerical, continuous, and sortable. They can be used to calculate measures (such as mean or variance)
		nominal	Variables with discrete values that are not sortable. Zip codes and phone numbers should be nominal.
		ordinal	Variables with discrete values that are sortable.
		textual	A type of nominal variable containing phrases, sentences or complete texts. Textual variables are used for text analyses.
Key	key variable or identifier for the record	0	the variable is not an identifier
		1	primary identifier
		2	secondary identifier
Order	identifies whether a variable represents a natural order. If the value is set at 1, the variable is used in SQL expressions in an "order by" condition.	0	the variable does not represent a natural orde

Name	Description	Possible Values	Value Description
	⚠ Caution Marking a variable as an order variable will have an impact on the choice of a partition strategy. Only the Sequential strategies can be used with an order variable.	1	the variable represents a natural order
Missing	string used in the data description file to represent missing values	Examples: • 999 • #Empty	
	additional description label for the variable		

6.4.2.4 Why Describe the Dataset ?

In order for Automated Analytics features to interpret and analyze your data, the data must be described. To put it another way, the description file must specify the nature of each variable, determining their:

- *Storage format*: number (*number*), character string (*string*), date and time (*datetime*) or date (*date*).
- *Type*: *continuous*, *nominal* or *ordinal*.

For more information about data description, see *Types of Variables* of *Storage Formats* in *Classification, Regression, Segmentation and Clustering Scenarios – Automated Analytics User Guide*.

6.4.2.5 Deriving a Graph from a Bipartite Graph

> **ℹ Note**
>
> This option is only available when a bipartite graph has been created for the current model.

1. In *Graph to Derive from*, select the transactional graph from which you want to derive a new graph.
2. In *Entity*, select the entity to use in the new graph, that is, the population you want to see appear in the new graph.
3. In *Minimum Support*, enter the minimum number of links necessary for a node to be added to the graph.

   For example, when deriving a graph on a product population from a bipartite graph linking customers and products, if you set the *Minimum Support* to 5, the new graph will only contain the products that have been bought by at least 5 customers.
4. Click *OK*.

6.4.2.6 Extracting a Graph from an Existing Graph

1. In *Data Type*, select the type of data you want to access. For some type of data you will need a specific license.
2. Use the *Browse* button to select the folder or database containing the model. In case of a protected database, you will need to enter the user name and password when selecting the database.

   Whenever allowed by the data format, the following options are available in the *Data Source Selection* panel:

   - : go to parent directory.
   - : create a new folder.
   - : delete the selected folder (possible for empty folders only).

   > ⚠ Caution
   >
   > When displaying the content of a database, you may need to click *Refresh* to see all the models, especially those created recently.

3. Select the model from which you want to extract a graph.
4. Click *Load graphs*.
5. Select the graph you want to load.
6. Click *OK*.

6.4.3 Step 2 - Graphs Parameters Configuration

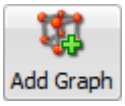
6.4.3.1 Graph Definitions

When creating a graph or selecting one in the *Graph List*, the graph settings are displayed on the right part of the panel. The settings depend on the type of graph you have selected in the previous panel.

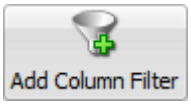
This panel allows you to define the information you need, for example:

- Many identical graphs with different start and end dates give you the possibility to visualize the evolution of a population.
- In telecommunications, one graph per type of call can be created by putting a filter on the column "call type".

6.4.3.1.1 Adding Graphs and Filters

1. Click 

to create a new graph in the *Graph List*.
A default graph is automatically created.
2. Click 

and select the desired option to copy a graph with all its settings and filters.
3. Click 

to create a new flter for the selected graph.
4. Click 

to delete the selected element (graph or filter).
5. Once all graphs and filters have been defined, click *Next*.

6.4.3.1.2 Defining the General Graph Settings

1. In the *Graph list*, select the graph you want to define settings for. When creating a new graph, it is selected by default.
2. In *Graph Name*, enter the name of the graph.
3. In *Graph Creation Type*, select the type of graph you want to create.

   The graph creation type has to be defined according to your dataset:

This option...	Generates graphs...	With the Following Information on the Link...
Contact	Containing only the links found at least a specified number of times.	number of occurrences of the link
Transaction	Containing the relations between objects of a different nature, such as customers and products for example.	
Nearest Neighbors	Containing a specific number of nearest neighbors for each node. The nearest neighbors are determined by their relative distance, which must be indicated in an additional column.	distance
Links Only	Linking the identifiers from the source column with the identifiers from the target column.	none

6.4.3.1.3 Setting the Column Roles

1. In the *Column Roles* section, select the column for the *Source Node*:

Option	Description
For directed graphs	Select the identifier
For bipartite graphs	Select the first population

2. Select the column for the *Target Node*:

Option	Description
For directed graphs	Select the identifier
For bipartite graphs	Select the second population

> **i Note**
>
> The *Source Node* and the *Target Node* must be different columns. For directed graphs, their order determines the way the links are displayed when you visualize the graph. For example, when creating a graph linking phone calls, in a directed graph the source node would be the caller and the target node would be the reciever.

3. **Optional:** If needed, select the column containing the date information in the list *Date Column*. This column can be used to filter the data to be used in the graph.
4. For the graph types *Contact* and *Links Only*, in the *Links Type* list check whether the direction of the links is relevant to your business issue.

   Select *Directed* if you want only the links going from the source nodes to the target nodes to be taken into account, otherwise select *Undirected*. There can be a difference in the events count for directed and undirected graphs. In the previous example, let's consider that Paul calls Lucy twice and Lucy calls Paul three times. In a directed graph, there will be two types of events: Paul->Lucy with a count of 2 and Lucy->Paul with a count of 3. In an undirected graph, there will be only one type of event: Paul <->Lucy with a count of 5.
5. For the graph type *Neighbors*, select the column that represents the relative distance between the nodes in the *Distance Column* list.

   For example a relative distance might be a communication time, a number of items purchased, or a physical distance.
6. For the graph types *Transactions* and *Proximity*, you can filter your results depending on the value of a specific variable in your dataset. For example, if you are creating rules to suggest additional products to customers, you can choose to only suggest products that are in stock.
   a. In the *Target* list, select the variable you want to use to filter the population graph.
   b. In the *Target Key* field, enter the value you want to use to filter the results. If the variable you have selected as *Target* can have several values, only the rules corresponding to the value indicated in the *Target Key* field will be provided.

6.4.3.1.4 Setting the Conditions for Link Creation

Depending on the type of graph you are creating the conditions for creating links between the nodes vary.

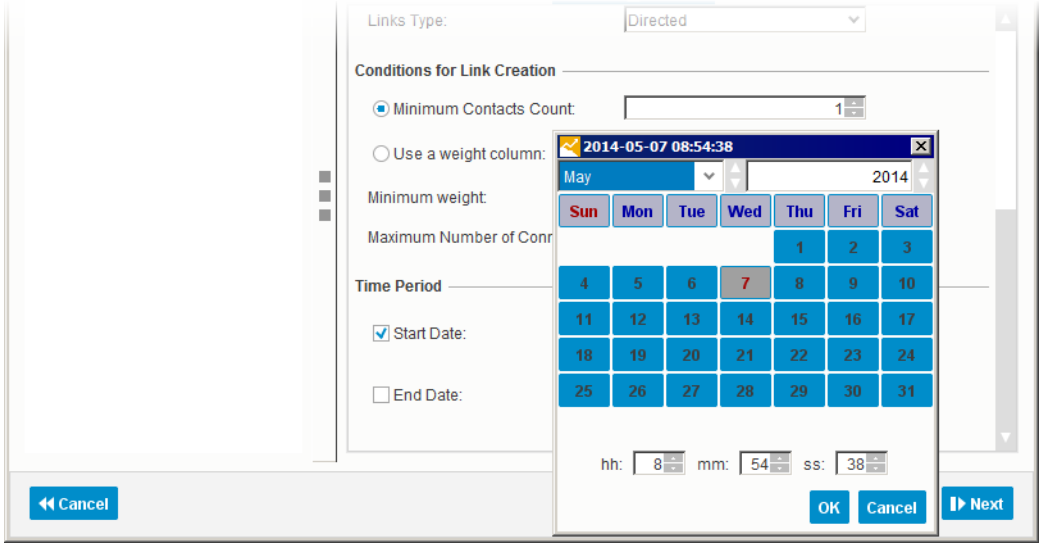
In the *Conditions for Link Creation* section, set the appropriate options as described in the following table.

Setting	Description	Graph Type
Minimum Contacts Count	Enter the minimum number of occurrences of the couple source node-target node needed in the dataset to create a link.	Contact
Use a weight column	Select the column containing the weight to be attributed to each link. In case of a Contact graph, the link weight in conjunction with the minimum weight can be used to determine which links are going to be created.	• Contact • Links Only • Transactions
Minimum Weight	Enter the minimum weight value needed for a link to be created.	Contact
Maximum Number of Connections	Enter the maximum number of connections to create for one node. Once a node has reached this number of connections, no other connection will be created. If you have a threshold for mega-hub detection lower than the default value, enter it here as well as in the mega-hub detection settings in the next panel.	• Contact • Links Only • Nearest Neighbors • Transactions
Maximum Number of Neighbors	Enter the maximum number of nearest neighbors that should be kept in the graph.	Nearest Neighbors

6.4.3.1.5 Setting the Time Period

The section *Time Period* allows you to filter the events on a specific time period. You can define a start date, an end date or both.

> **i Note**
>
> The start date is included in the defined period, whereas the end date is excluded from it.



1. If you want to use only events occurring after a specific date:
   - Check *Start Date*.
   - Click the calendar button located on the right of the date field to select the date.
2. If you want to use only events occurring before a specific date:
   - Check *End Date*.
   - Click the calendar button located on the right of the date field to select the date.

6.4.3.1.6 Adding Column Filters

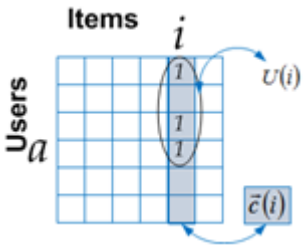
1. Choose the graph you want to apply a filter on from the *Graph List* in the left part of the panel.
2. Click *Add Column Filter*.
   The *Filter Settings* are displayed in the right part of the panel.
3. In *Filtered Column*, select the column you want to filter by.
4. If the selected column is a number or a time you can define a range of values as filter. In that case:
   a. Check the radio button *Accept Value in this Range*.
   b. In the *Minimum* field indicate the minimum value for the range you want to filter by.
   c. In the *Maximum* field indicate the maximum value for the range you want to filter by.
5. Or you can enter one or several values to accept or to discard:
   a. Check the radio button *Accept* or *Discard*.
   b. Click to add a new value.
   c. Type the value or use the arrows.
   d. Click to remove the selcted value.
   e. Click to extract variable categories from a defined number of lines or for the entire dataset.

6.4.3.1.7 Setting the Projection

When deriving a graph from a bipartite graph, you must define the projection for an entity (source or target column) of this graph.

The following table sums up the available parameters for the projection:

Settings	Description
Graph to derive from	The name of the bipartite graph from which the current graph is derived.
Entity	The entity used as node for the current graph. A bipartite graph creates links between two types of entities, only one can be used in a standard graph.
Keep top N	The number of pairings you want to keep among the highest ranking ones.

Settings	Description
Weight	Indicate which value to assign as a weight for the links. It is recommended to use the *Jaccard Index* or the *Independence Probability*, the latter especially when working with communities. *Support* : number of links found for each node. *Jaccard Index*: It measures similarity between sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets: $J(A,B) = \frac{\lvert A \cap B \rvert}{\lvert A \cup B \rvert}$ *Independence Ratio* : Two events A and B are independent if the probability that both events occur equals the probability of event A times the probability of event B: $\Pr(A \cap B) = \Pr(A).\Pr(B)$ The following formula is used to compute the weight: $w = \frac{\Pr(A \cap B)}{\Pr(A).\Pr(B)}$ A weight of 1 indicates completely independent events. The higher is the weight, the stronger is the correlation between the events. *Cosine*  where: a: the active user i and j: two items $U(i)$: set of users who purchased i $\vec{c}(i)$: column of matrix R for item i $Sim(i,j) = cos[\vec{c}(i),\vec{c}(j)] = \frac{\vec{c}(i) \bullet \vec{c}(j)}{\lVert \vec{c}(i) \rVert_2 \times \lVert \vec{c}(j) \rVert_2} = \frac{\sum_{u=1}^{L} w_{ui} \times w_{uj}}{\sqrt{\sum_{u=1}^{L} (w_{ui})^2} \times \sqrt{\sum_{u=1}^{L} (w_{uj})^2}}$

Settings	Description
Maximum Iterations	Maximum number of iterations after which the algorithm should stop if the Epsilon criteria has not been reached.
Optimize Graph Memory Consumption	Use this option reduce memory use. Note that it will slow down the process.
Optimize Graph Computation Speed	Use this option to speed up the process. Note that this will increase memory use.
Minimum Support	The support is the number of items two entities have in common. For example, when deriving a graph from a bipartite graph linking products and customers, the support of the link between product A and product B is the number of customers having bought both products. A link which support is below the *Minimum Support* will not be created.

6.4.3.1.8 Setting the Filters

The following table sums up the available filters that you can set: You can choose to filter the links that will be created with the following parameters:

Filter Settings	Description
Maximum Confidence	The confidence of a link is the percentage of records verifying the consequent of the rule among those verifying the antecedent of the rule. With a rule of the form X=>Y, the confidence is the count of records containing the itemset X and the attribute Y in relation to the count of records containing only the itemset X. If you have not checked the *Directed* box and the Confidence of a link is above the Minimum Confidence only for one direction, the link will still be kept.
Minimum KI	Only links for which the KI (that is, the predictive power) is above this number are kept. If you have not checked the *Directed* box and the KI of a link is above the Minimum KI only for one direction, the link will still be kept.
Directed	This filter enables to differentiate the links depending on their direction and provides directed rules.

6.4.3.1.9 Filtering the Population for a Derived Graph

For derived graphs, you can use the options provided in the *Candidate List* tab to filter the population you want to see in the graph.

- To use the *Target Event Required* option, you need to first indicate a *Target* variable and a *Target Key* value in the settings of the transactions graph you want to use to create the derived graph (see Setting the Column Roles [page 325] above). If the *Target* variable and the *Target Key* value are not set beforehand, no rules will be generated.
- The *Policy* option and the *Next Transition Limit* option are only available if the derived graph has been created based on a proximity graph using a weight column.

1. Check the *Target Event Required* option to filter the population of the current graph based on the value of the target variable defined for the bipartite graph.
2. To set the *Policy* option:
   a. Select how to filter the population depending on the timeframe in the *Policy* list.

Policy	Description
All	Selects the whole population without a defined time
Synchronous	Selects people who were at the same location (the same tile) at the same time
Sequence	Selects people who followed the same path. For the *Sequence* policy the *Maximum Time Delta* corresponds to the time lag between the beginning and the end of the path

   b. If you have selected the *Synchronous* policy or the *Sequence* policy, click *Time Scale* to select the time unit to use.

3. Enter the maximum number of transitions to take into account after a pair has been found in the *Next Transitions Limit* field.

6.4.3.1.10 Setting the Final Selection

To define the maximum number of nodes to keep, proceed as follows.

Check the *Max Top Nodes* option.

6.4.3.2 Setting the Post Processing Tools

6.4.3.2.1 Detecting Communities

This option is used to identify communities in the dataset. A community can be seen as a subgraph in which the density of internal connections is larger than the connections with the rest of nodes in the network. The

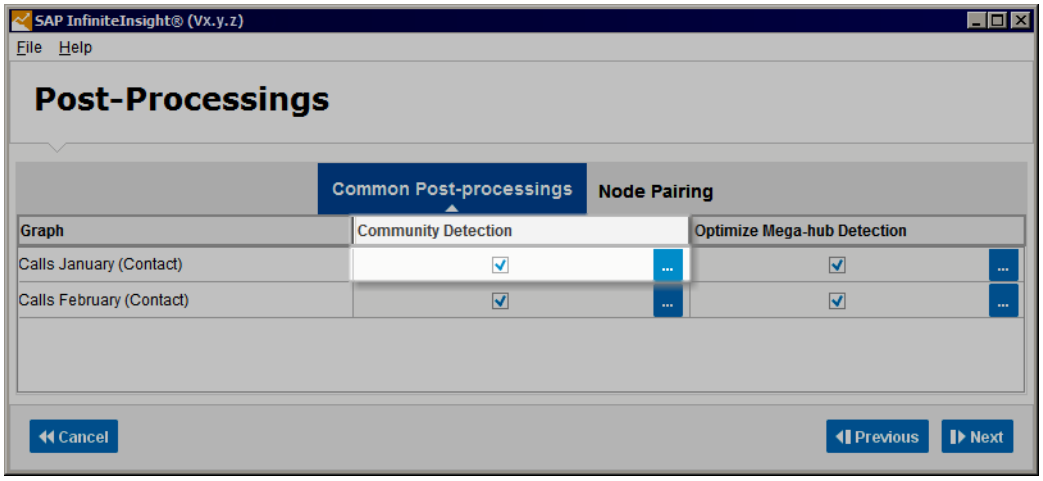
community detection is based on an iterative algorithm for which the stop criteria are the maximum number of iterations and the modularity gain between one iteration and the next, which is represented by Epsilon.

> **i Note**
>
> For more information on how the community detection is computed, you can check information on Louvain algorithm (http://en.wikipedia.org/wiki/Community_structure#The_Louvain_method).

You can also base the communities from one graph on the communities from another graph. For example February's communities could be initialized with January's ones allowing you to visualize their evolution from one month to the other.

6.4.3.2.2 Enabling Community Detection

Check the *Community Detection* box corresponding to the graph for which you want to enable the detection.



6.4.3.2.3 Configuring the Community Detection

1. Click the ... button located to the right of the checkbox *Community Detection*.
   The *Configuration of Communities Detection* window opens.
2. Check *Enable Communities Detection*.
3. Set the options you want to apply.

   The following table describes the available options.

Option	Description
Max Iterations	Maximum number of iterations after which the algorithm should stop if the Epsilon criteria has not been reached.

Option	Description
Epsilon	Value below which the modularity gain is no longer interesting enough to justify another iteration of the algorithm.
Seed from Graph	Graph from which you want to use the communities as seed for the current graph. The list *Seed from graph* contains only the names of graphs for which community detection has been enabled. Community seeding is not available for transactional graphs.
Ignore Self-Loops	Reduces the number of communities and gives a better insight in the graph structure.
Disable Superco-munities	Disables the creaion of supercommunities, which speeds up the creation when dealing with large graphs.

4. Click *OK* to validate the parameters.

6.4.3.2.4 Mega-hub Detection Optimization

A node can be considered as a mega-hub when its degree is above a certain threshold and hence has a negative contribution regarding the graph global modularity and its clustering potential. A node connected to a big amount of neighbors tends to regroup distinct communities in one big community, hiding the actual underlying clusters.

Removing these highly connected nodes can dramatically improve processing time, and put away irrelevant nodes from the analysis. Because of their high degree, such nodes are likely to create 'artificial cluster' in the network, making it harder to detect communities, and hiding the real structure of the graph.

Such mega-hub nodes can be found in telecommunication networks (service number, call center, taxis), transactional networks (best sellers, head of tail products). These nodes have a large amount of links, but are semantically insignificant.

Removing mega-hubs can have positive impacts on:

- increasing the resulting modularity
- reducing the community processing time (because community assignment involve looking at neighbors)
- reducing the pairing detection time (because neighbors of mega-hub will have very large second circle)
- reducing the size of the largest community (wrongly induced by the fact that some nodes are connected to 'many' nodes)

The *Optimize Mega-hub Detection* option removes any nodes for which the degree is over the threshold.

There are two ways to set this threshold:

- use the automatic threshold, which is based on the distribution of the nodes depending on their degree of connectivity. The automatic threshold is calculated with the following formula:

```
mean + standard deviation factor * standard deviation
```

> **i Note**
>
> By default the standard deviation factor is set to 4.

- set a fixed value.

6.4.3.2.5 Configuring the Mega-hub Detection

1. Click the ... button located to the right of the checkbox *Optimize Mega-hub Detection*.
2. Check *Optimize Mega-hub Detection* to activate the option.
3. Select the type of threshold you want to use:
   - Automatic threshold
   - Manual threshold
4. Depending on the type of threshold you have selected:
   - use the field Standard deviation factor to configure the formula used to calculate the automatic threshold,
   - use the field Threshold for exclusion to set the manual threshold.
5. Click *OK* to validate the parameters.

6.4.3.2.6 Pairing Nodes

Node pairing allows identifying an item (customer, product, etc.) present in two different graphs (thus with two different nodes), by means of its common neighbors. This technique can be used, for instance, by a telephone company to detect "rotational churn" (locate clients who change their phone subscription on a regular basis to benefit from welcome offers).

1. Select the *Node Pairing* tab.
2. Click the located in the top right corner of the tab to add a new pairing definition.
3. Select the *First Graph* for the node pairing.
4. Select the *Second Graph*.
5. Set the options you want to apply to the node pairing.

   The following table lists the available options.

Option	Description	Setting
Output Graph Name	Displays the paired nodes as a new graph.	1. Check the corresponding box. 2. Enter a name in *Output Graph Name*.
Minimum Common Neighbors Count	Allows you to define the minimum number of neighbors two nodes must have in common to be paired.	1. Enter the minimum number of neighbors.

Option	Description	Setting
Keep Top N	Allows you to define the number of pairings you want to keep among the highest ranking ones. The order of importance of the pairings is determined by the selected *Pairing Type*. When set to 0, all the pairings are kept. **i Note** The graphs keep the N links out with the biggest weights, so when the user wants to generate graphs with both ratio and counts, links with different pairs may be kept.	1. Enter the number of pairings to be kept among the highest ranking ones.
Minimum Common Neighbors Ratio	Keeps only the pairings for which the Jaccard index in higher than or equal to a specific value.	1. Check the corresponding box. 2. Enter the value in *Minimum Common Neighbors Ratio*.
Pairing Type	Allows you to select the type of pairing to be used to compute the *Top N* pairings. The available types are: ○ Ratio ○ Jaccard ○ Independence Ratio (only when paring nodes in a same graph) ○ Confidence ○ Clustering Coefficient	1. Select the type of pairing to be used.
Weight Radio	Applies the weight variable declared when setting column roles to be used to compute the *Pairing Type*.	1. Check the corresponding box.
Include Common Neighbors Count Graph	Generates an additional graph where the links are labeled with the number of common neighbors.	1. Check the corresponding box.

6. Click *OK* to validate the pairing definition.
7. **Optional:** If you want to define another pairing, repeat steps 2 to 6.

6.4.3.2.7 Pairing Type Formulas

Ratio

Unweighted: $$ratio(A,B) = \frac{\#\,common\_neighbors(A,B)}{(\#\,neighbors(A) + \#\,neighbors(B))/2}$$

Weighted: $$ratio(A,B) = \frac{w\_degree\_common(A,B)}{(w\_degree(A) + w\_degree(B))}$$

Jaccard

Unweighted: $$jaccard(A,B) = \frac{\#\,common\_neighbors(A,B)}{\#\,neighbors(A) + \#\,neighbors(B) - \#\,common\_neighbors(A,B)}$$

Weighted: $$jaccard(A,B) = \frac{w\_degree\_common(A,B)}{w\_degree(A) + w\_degree(B)}$$

Independence Ratio

Unweighted: $$independence\_ratio(A,B) = \frac{P(A\,\&\,B)}{P(A)P(B)} = \frac{\#\,common\_neighbors(A,B) * \#\,links}{\#\,neighbors(A) * \#\,neighbors(B)}$$

Weighted: $$independence\_ratio(A,B) = \frac{P(A\,\&\,B)}{P(A)P(B)} = \frac{w\_degree\_common(A,B) * total\_weight}{w\_degree(A) * w\_degree(B)}$$

Confidence

Unweighted: $$confidence(A,B) = \frac{P(A\,\&\,B)}{P(A)} = \frac{\#\,common\_neighbors(A,B)}{\#\,neighbors(A)}$$

Weighted: $$confidence(A,B) = \frac{P(A\,\&\,B)}{P(A)} = \frac{w\_degree\_common\_from(A)to(A,B)}{w\_degree(A)}$$

Clustering Coefficient

Unweighted: $$clustering(A,B) = \frac{2 * (\#\,triangles(A) + \#\,common\_neighbors(A,B))}{\#\,neighbors(A) * (1 + \#\,neighbors(A))}$$

Weighted: $$clustering(A,B) = \frac{2 * (\#\, triangles(A) + w\_degree\_common\_from(A) to(A,B))}{w\_degree(A) * (1 + w\_degree(A))}$$

6.4.3.2.8 Notations

The following table lists the notations used in the formulas.

Notation	Definition
`#neighbors (N)`	Number of distinct neighbors of N.
`degree (N)`	Sum of number of links going out N or coming in N. • In directed graphs, one link may be going out and coming in from the same neighbors • Two links can be going to the same neighbor
`w_degree (N)`	Sum of weights going out N or coming in N.
`#common_neighbors(A,B)`	Number of common distinct neighbors of A and B.
`w_degree_common (A,B)`	Sum of the weights going from A to common and B to common in ALL directions.
`w_degree_common_from (A) to (A,B)`	Sum of the weights going from A to common between A and B in ALL directions.

6.4.3.3 Selecting Additional Datasets

You can select optional datasets to add more information to the graph. Two types of additional datasets are possible:

- the *Identifiers Conversion Dataset* that can be used to relate the descriptive attributes dataset to the link dataset. If you do not have a decoration dataset, you can use this kind of dataset to indicate whether you have additional information on the node or not. This dataset should contain two identifier columns.
- the *Descriptive Attributes Dataset* that can be used to give additional information for the variables of the event dataset. If it does not have the same identifier column as the link dataset, you need to select an identifiers conversion dataset as well.

6.4.3.3.1 Adding an Identifiers Conversion Dataset

1. If you have defined a *Transaction* graph, select the tab corresponding to the dataset for which you want to add an identifiers conversion dataset.
2. Check *Add Identifiers Conversion Dataset*.

3. Click *Select Dataset*.
   A pop-up opens allowing you to select the dataset.
4. In the *Data Type* list, select the type of the data source.
5. Use the *Browse* button corresponding to the *Folder* field to select the location of the data source.
6. Use the *Browse* button corresponding to the *Dataset* field to select the dataset.
   a. Use the (Advanced Parameters) button to reduce the dataset or define specific settings.
   b. Use the (View Data) button to display your dataset and compute its statistics.

You now have to map the identifiers.

7. In the *Original Identifier* list, select the column containing the node identifier corresponding to the one existing in the link dataset.
8. In the *Converted Identifier* list, select the column containing the identifier corresponding to the one existing in the node decoration dataset.

> ⚠ Caution
>
> Both identifiers must have the same storage format.

Related Information

Describing the Link Dataset [page 320]

6.4.3.3.2 Adding a Descriptive Attributes Dataset

1. If you have defined a *Transaction* graph, select the tab corresponding to the dataset for which you want to add an identifiers conversion dataset.
2. Check *Add Descriptive Attributes Dataset*.
3. Click *Select Dataset*.
   A pop-up opens allowing you to select the dataset.
4. In the *Data Type* list, select the type of the data source.
5. Use the *Browse* button corresponding to the *Folder* field to select the location of the data source.
6. Use the *Browse* button corresponding to the *Dataset* field to select the dataset.
   a. Use the (Advanced Parameters) button to reduce the dataset or define specific settings.
   b. Use the (View Data) button to display your dataset and compute its statistics.

You can now select the descriptive attributes.

7. In the *Node Identifier* list, select the column containing the identifier to be used to link the descriptive attributes to the main dataset.

> ⚠ Caution
>
> The identifiers from the two datasets must have the same storage format.

8. In the *Available Columns* list, select the columns you want to use in the graph and click the > button. The selected columns moves to the *Selected Descriptive Attributes* list.
9. Sort the list of attributes by using the  and  buttons.

> **i Note**
>
> You can also search for specific attributes by their names by clicking  and entering all or part of the name in the displayed field.

10. Select a column name and click < to remove it from the list.

Related Information

Describing the Link Dataset [page 320]

6.4.4 Step 3 - Model Generation

In the *Summary of Modeling Parameters* panel you can see the datasets that you have selected for the model and the graph type you have chosen. These parameters have a gray background and cannot be modified.

You can change the model name or add a description.

- The name of model is filled automatically. It corresponds to the name of the target variable (*class* for this scenario), followed by the underscore sign ("_") and the name of the data source, minus its file extension (*Census01* for this scenario).
- The *Autosave* button allows you to activate the feature that will automatically save the model once it has been generated. When the autosave option is activated, a green check mark is displayed on the *Autosave* button. 

6.4.4.1 Generating the Model

1. Click *Generate* to start the learning process.

   If the *Autosave* option has been activated in the panel *Summary of Modeling Parameters*, a message is displayed at the end of the learning process confirming that the model has been saved.
2. Click *Close*.

6.4.4.2 Activating the Autosave Option

The *Model Autosave* panel allows you to activate the option that will automatically save the model at the end of the generation process and to set the parameters needed when saving the model.

To activate the option, proceed as follows:

1. In the *Summary of Modeling Parameters* panel, click the *Autosave* button.
   The *Model Autosave* panel is displayed.
2. Check *Enable Model Autosave*.
3. Set the parameters listed in the following table.

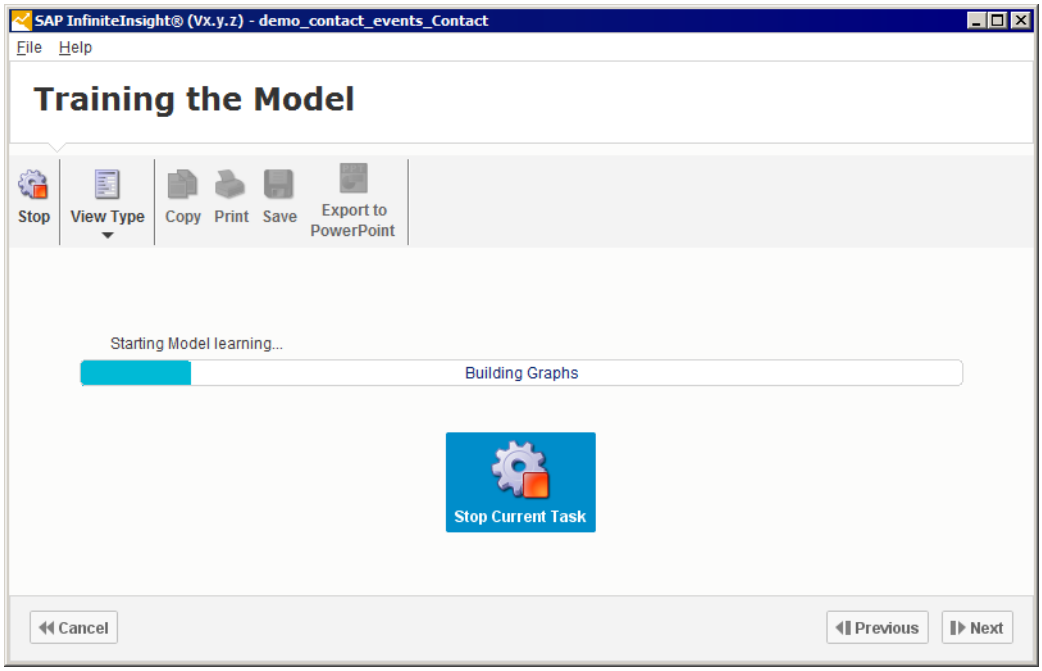
Parameter	Description
Model Name	This field allows you to associate a name with the model. This name will then appear in the list of models to be offered when you open an existing model.
Description	This field allows you to enter the information you want, such as the name of the training dataset used, the polynomial degree or the performance indicators obtained. This information could be useful to you later for identifying your model. Note that this description will be used instead of the one entered in the *Summary of Modeling Parameters* panel.
Data Type	This list allows you to select the type of storage in which you want to save your model. The following options are available: ○ *Text Files*, to save the model in a text file. ○ *Data Base*, to save the model in a database. ○ *Flat Memory*, to save the model in the active memory.
Folder	Depending upon which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.
File/Table	This field allows you to enter the name of the file or table that is to contain the model. When saving the model as a text file, you must enter one of the following format extensions `.txt` (text file in which the data is separated by tabs) or `.csv` (text file in which the data is separated by commas).

4. Click *OK*.

6.4.4.3 Following the Training Process

There are two ways to follow the progress of the generation process:

- The Progress Bar displays the progression for each step of the process. This is the screen displayed by default.



- The Detailed Log displays the details of each step of the process.

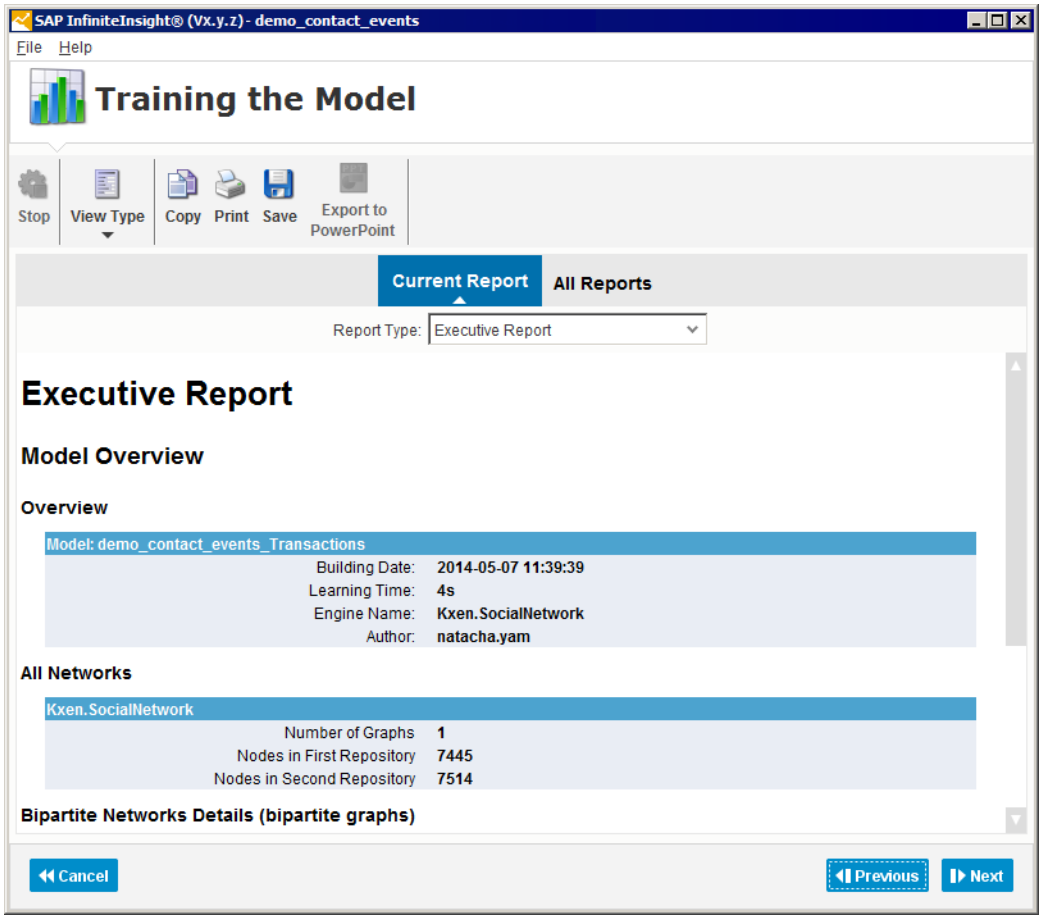
1. Click (*Show Detailed Log*) to display the detailed log.
2. Click (*Show Progression*) to display the progress bar.

To stop the training process:

3. Click (*Stop Current Task*).
4. Click *Previous*.
   The *Summary of Modeling Parameters* screen appears.

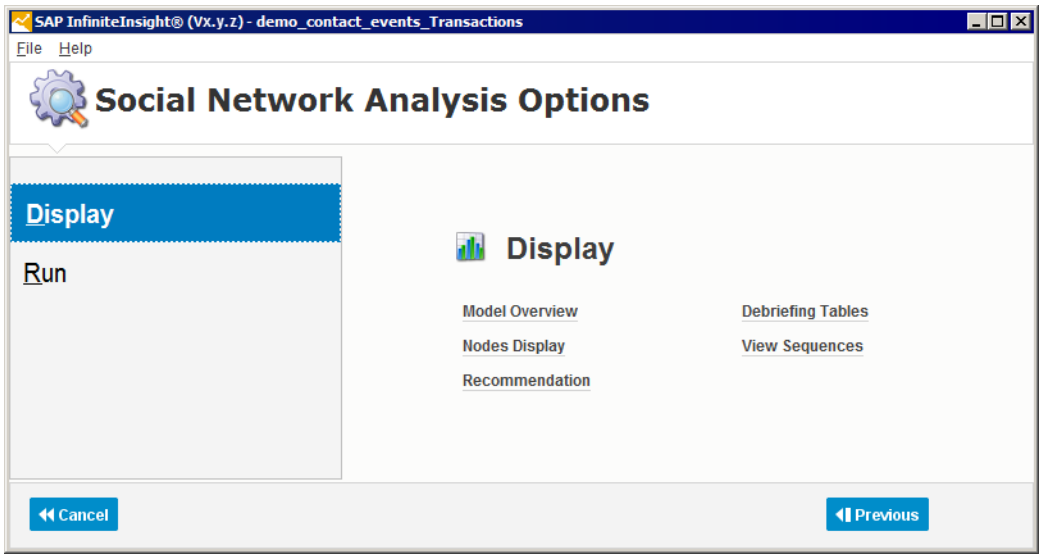
6.4.4.4 Model Results Vizualization

At the end of the generation process, a summary of the model results appears.



6.4.5 Step 4 - Using the Model

This panel offers the different options for displaying and running the model.



Display Options

- *Model Overview*: displays information on the model generated such as the number of variables, the performance indicators, the target information, and so on.
- *Debriefing Tables*: Automated Analytics provides you with a set of tables that allows you a detailed debriefing of your model.
- *Node Display*: displays the graph for a specific node.
- *View Sequences*: selects the specific sequences you want to view on Google Earth.
- *Recommendation*: allows you to generate recommendations based on a transactions graph for a specific customer or product.

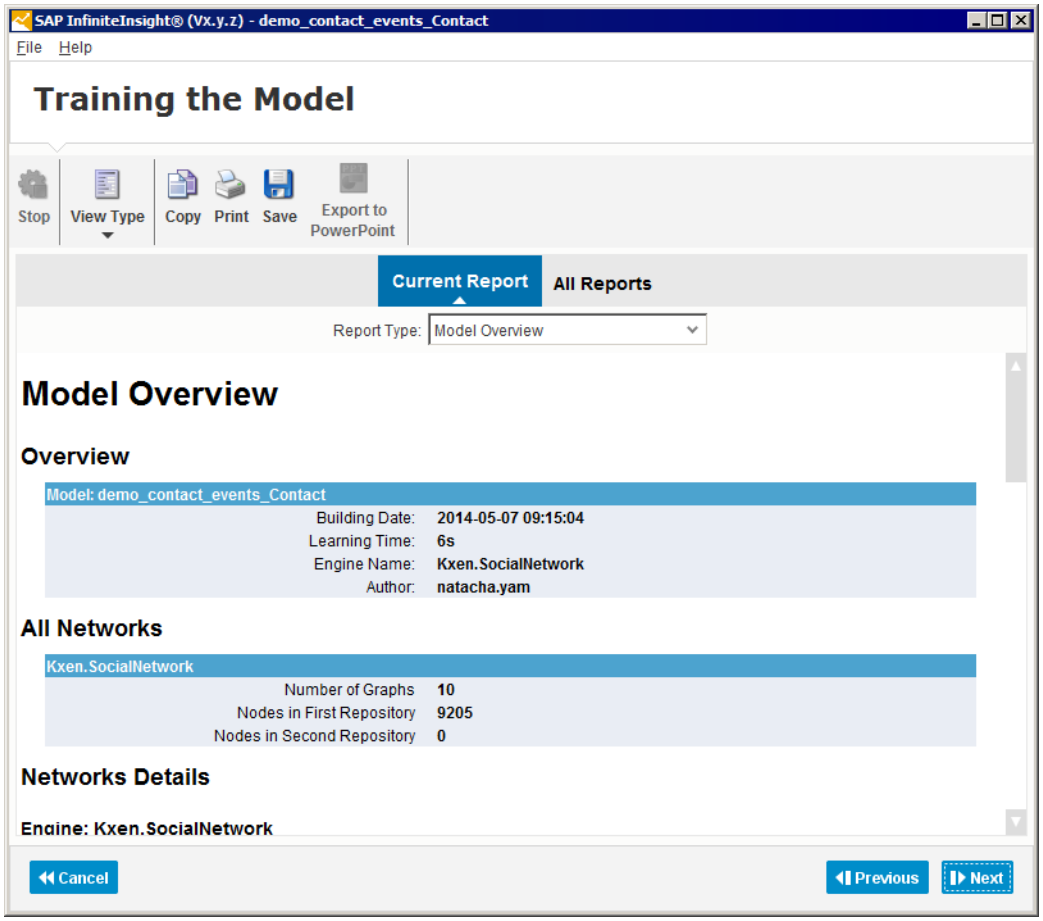
Run Options

- *Apply Model*: a new dataset can be scored using the model built. Scores can also be applied to the data used to build the model.
- *Save Model*: the model built can be saved and loaded at a later time.
- *Export KxShell Script*: generates a KxShell script that can re-create the model.
- *Batch Recommandation*: generates a list of recommended products for each of the customers in your dataset.

6.4.5.1 Model Overview

The *Model Overview* screens displays the model debriefing.

> **i Note**
>
> To access the *Model Overview* panel, click *Model Overview* on the *Social Network Analysis Options* panel.



Below are tables that break down the information of each graph created.

Overview

Model: <Name>	Name of the model, created by default from the target variable name and the dataset name.
Building Date	Date and time when the model was built.
Learning time	Total learning time.
Engine name	Depending on the feature used, in this case *Kxen.SocialNetworks*.
Author	Name of the user who has created the model (from the computer information).

All Networks

Number of Graphs	The number of graphs that have been created.
Nodes in First Repository	The number of nodes that have been found or in the case of a bipartite graph, the number of nodes in the first population.
Nodes in Second Repository	The number of nodes in the second population (only for bipartite graphs).

Network Details

For each graph that has been created:

<Name>	Name of the graph.
Link Type	The type of the links in the graph (directed or undirected).
Repository	Indicates the repository containing the nodes used in the current graph. The possible values are: • First • Second • First Community Nodes • Second Community Nodes
Number of Nodes	The number of nodes that have been found.
Number of Links	The number of links between the nodes.
Density	Ratio between the number of existing links and the total number of possible links.

Bipartite Networks Details

For each graph that has been created:

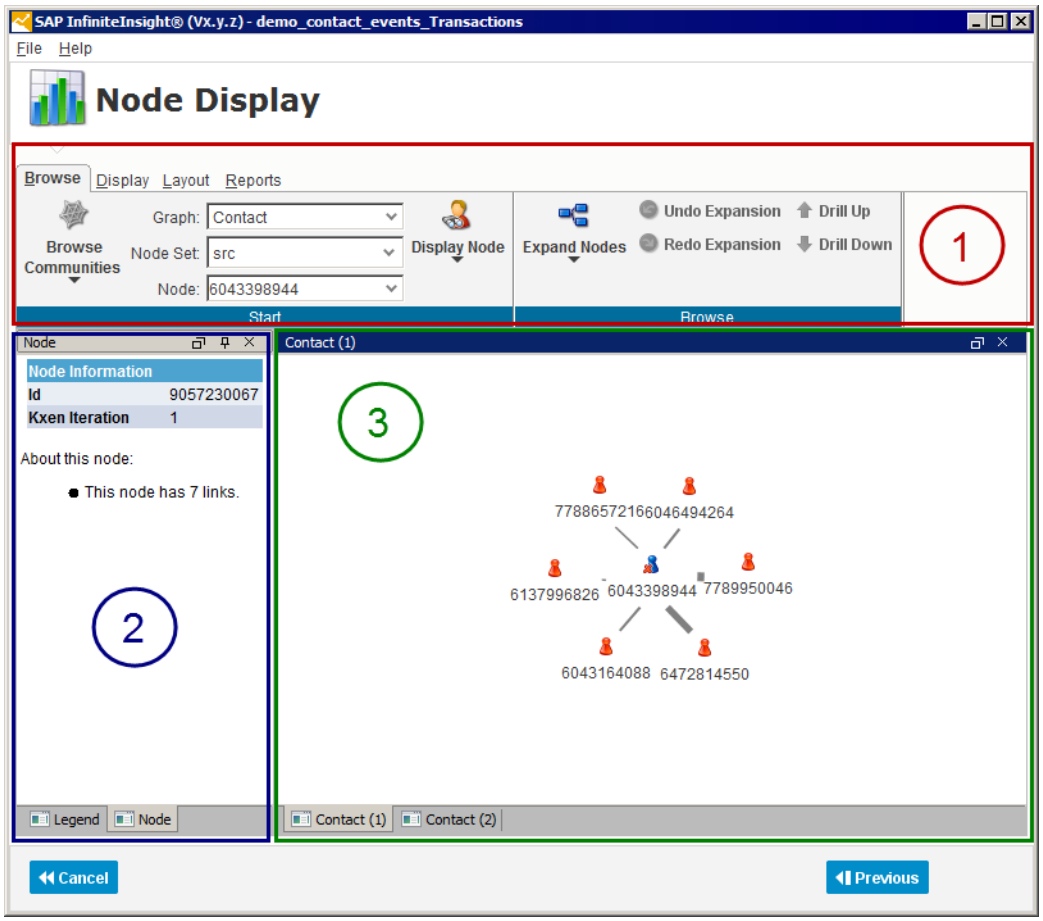
<Name>	Name of the graph.
Link Type	The type of the links in the graph (directed or undirected).
Nodes in First Repository	The number of nodes in the first population.
Nodes in Second Repository	The number of nodes in the second population.
Number of Links	The number of links between the nodes.
Filtered Line Count	The number of filtered lines.
Density	Ratio between the number of existing links and the total number of possible links.

Abnormally Connected Nodes in the First Repository	The number of nodes in the first population that are abnormally connected.
Abnormally Connected Nodes in the Second Repository	The number of nodes in the second population that are abnormally connected.

6.4.5.2 Graphs Visualization

The *Node Display* panel is divided in three sections:

- (1): The ribbon, which displays all the options that are available in this panel.
- (2): The *Reports* section that displays all types of reports and information such as *Node*, which allows you to select the node you want to display and lists the information related to the displayed node; or *Legend*, which contains the caption related to the graph. Each requested reports is displayed in a new tab.
- (3): The graph itself is displayed on the right part of the panel.



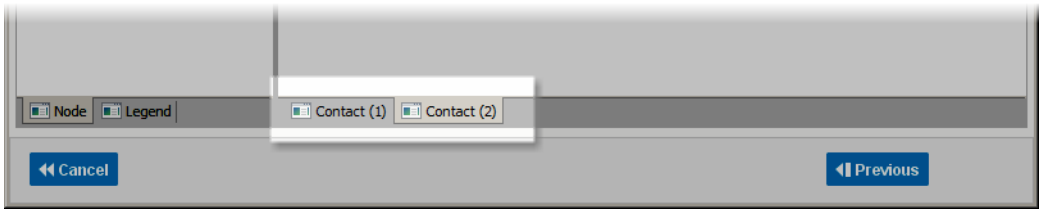
6.4.5.2.1 Displaying the Graph

> **i Note**
>
> By default, the graph is displayed in the on-/off-net mode. The only information provided for the nodes is whether additional information is available for a given node or not.

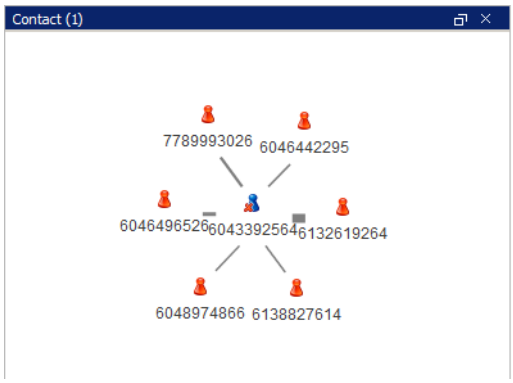
1. Select the graph you want to display, in the *Graph* list located in the *Browse* tab of the ribbon.
2. In case of a Transactions graph, select the column containing the node you want to display in the *Node Set* list.
3. In the *Node* list, select the identifier of the node you want to display.

   The nodes are listed in descending order of weight, that is of number of neighbors.
4. Click *Display Node*.
5. Select the *Display Node* option.

   Each requested graph is displayed in a new tab on the right side of the panel.



6. Click *Display Node* located in the Browse tab of the ribbon to display the community it belongs to.
7. Select *Display Community*.
8. Double-click a node to expand it and display all its neighbors.



> **i Note**
>
> By clicking once on a node, you can see its number of links on the left of the screen without having to display them. This can be useful for nodes with a high number of neighbors.

Use the *Reports* tab to display information:

9. In the *Reports* tab, click *Reports*.
10. Select the information of you want to display.

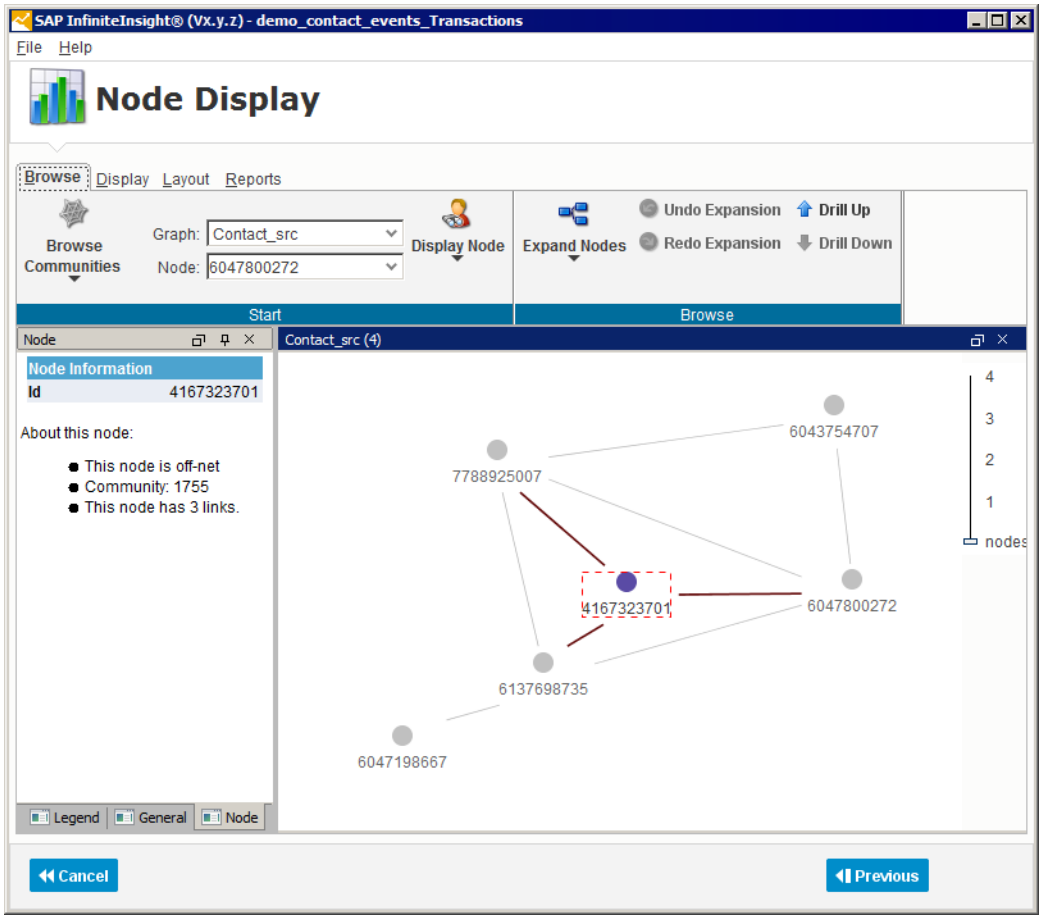
> **i Note**
>
> - The *General* tab gives overall information about the graph (Nodes Count, Links Count).
> - The *Metrics* tab give details metrics about the graph (In Median Degree, Out Median Degree, In Power Law Exponent, In Power Law Exponent Fit, Out Power Law Exponent, Out Power Law Exponent Fit).

You can now export the graph to a .csv file or save it as a .png file:

11. In the *Reports* tab, click the *CSV* button to export the graph to a .csv file.
12. In the *Reports* tabs, click the *Copy* button to copy the graph to the clipboard.
13. In the *Reports* tab, click the *Image* button to save the graph as a .png file.

6.4.5.2.2 Communities Mode

This visualization mode is only available if the *Community Detection* option has been activated in the *Post-processings* panel when setting the model parameters. When switching from the default mode to this mode, a conditional formatting is automatically applied to the nodes in order to show the communities they belong to. The shapes and colors applied by default have no particular meaning and can be customized.



An important feature of this mode is the possibility to detect the roles the nodes play inside, or outside, their community:

- *Bridge Node*: these nodes connect communities, thus acting like a bridge. They have many connections with other communities, but only few in their own community.
- *Local Node*: these nodes have many connections inside a given community and only few connections to other communities.
- *Social Node*: these nodes have many connections inside their own community and to other communities.
- *Passive Node*: these nodes have few connections inside their own community and to other communities.

6.4.5.2.3 Customizing the Communities

1. Right-click a node belonging to the community for which you want to modify the display format.
2. In the contextual menu, select the option *Edit Community Formatting for this Node*.
   The window *Define Conditional Formatting* opens.
3. Check the option corresponding to the type of display you want, that is, either a *colored shape* or an *icon*.
4. For a colored shape:
   a. Select the shape in the drop-down list.
   b. Click the button located on the right to select the color.
5. For an icon:
   a. Select an icon in the drop-down list.
6. Click *OK*.
   The graph is now updated.

6.4.5.2.4 Displaying Several Graphs

1. Click one of the icons representing the tabs and move it elsewhere (drag and drop) to see more than one graph or filter at the same time.

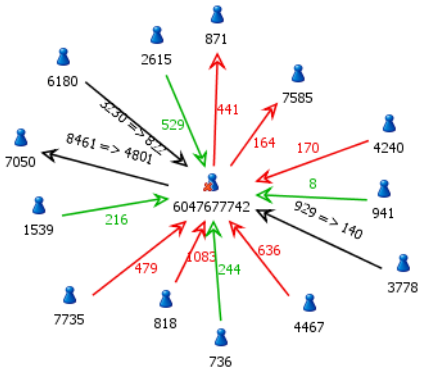
   Social offers the possibility to compare two graphs and visualize the differences between them. This feature can be useful, for example, to see the evolution of the nodes, if the graphs have been filtered on the date variable.

You can also compare two graphs. To do so:

2. Click  *(Compare graphs)*.
3. In the *Select Graph to Compare* pop-up window, select a graph in the *First Graph* list.
4. Select the graph you want to compare it to in the *Second Graph* list.
5. Click *OK*.

   The comparison graph is displayed in a new tab and the colors give you additional information on the links:
   - green: the link exists only in graph 1
   - red: the link exists only in graph 2

- black: the link exists in both graphs and the weight is displayed as `<weight in graph1> => <weight in graph2>`



6.4.5.2.5 Expanding the Nodes Using the Ribbon

1. In the *Browse* tab of the ribbon, click *Expand Nodes*.
2. Select *Expand All Nodes*.

You can also expand only the nodes you have selected. To do so:

3. On the graph, select the nodes you want to expand by using the Shift key of your keyboard.
4. Click *Expand Nodes* and select *Expand Selected Nodes*.

To select only the nodes belonging to the community:

5. On the graph, select the desired node.
6. Click *Expand Nodes* and select *Expand Community of Selected Node*.
   The graph corresponding to the community the node belongs to is displayed in a new tab.

You can also cancel an action:

7. In *Browse* tab of the ribbon, select one of the following options:
   - Undo expansion: you can also use the keyboard shortcut Ctrl+ Z.
   - Redo expansion: you can also use the keyboard shortcut Ctrl + Y.

To browse the sibling communnities graph of a selected node:

8. Select the desired node.
9. Click *Drill Up* to browse the sibling communities of the selected node in a new graph.
10. Select the community and choose one of the following options:
    - *Drill Up* to browse the sibling community of the selected node in a new graph.
    - *Dril Down* to explore the selected community in a new graph.

> **i Note**
>
> You can define the zoom level by using the list in the  field under the *Display tab*. To fit the graph size to the screen, click the icon  .

6.4.5.2.6 Expanding the Nodes Using the Contextual Menu

To expand all the nodes:

1. Right-click the graph background to display the contextual menu.
2. Select the option *Expand Nodes* > *Expand All Nodes.*.

You can also expand only the nodes you have selected. To do so:

3. On the graph, right-click the nodes you want to expand.
4. Select the option *Expand Nodes* > *Expand Selected Nodes.*

To select only the nodes belonging to the community:

5. Right-click the node to display the contextual menu.
6. Select the option *Expand Nodes* > *Expand Selected Nodes.*.
   The graph corresponding to the community the node belongs to is displayed in a new tab.

You can also cancel an action:

7. In *Browse* tab of the ribbon, select one of the following options:
   - Undo expansion: you can also use the keyboard shortcut Ctrl+ Z.
   - Redo expansion: you can also use the keyboard shortcut Ctrl + Y.

To browse the sibling communnities graph of a selected node:

8. Select the desired node.
9. Click *Drill Up* to browse the sibling communities of the selected node in a new graph.
10. Select the community and choose one of the following options:
    - *Drill Up* to browse the sibling community of the selected node in a new graph.
    - *Dril Down* to explore the selected community in a new graph.

> **i Note**
>
> You can define the zoom level by using the list in the  field under the *Display tab*. To fit the graph size to the screen, click the icon  .

6.4.5.2.7 Customizing the Links

1. In the *Display* tab of the ribbon, click  .
2. Select one of the following options:

Display Mode	Description	Example
Do not display weights	No information about the link is displayed.	
Display weight as link width	The higher is the number of links between nodes, the thicker is the line.	
Display weight as link color	The different weights are represented by different colors.	
Display weight as link labels	A tag indicates the number of links between the nodes.	

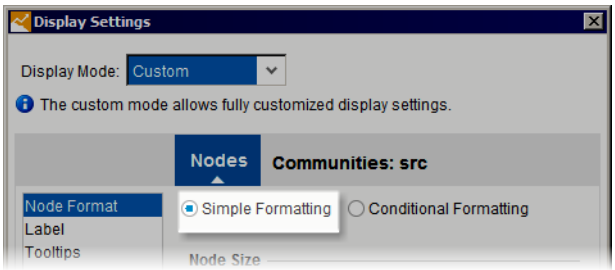
3. To select the links you want to display, use the cursors on the *Edge Visibility Threshold* slide to define the visibility of the links and to point out the links to display.

6.4.5.2.8 Customizing the Node Display: Setting a Simple Format

When customizing the Node Display, you can set either a simple or conditional format. The simple format allows you to use the size of the nodes to visualize the values of a continuous variable and the color of the nodes to visualize the values of a continuous or nominal variable.

To define a simple format:

1. In the *Display* tab of the ribbon, click *Node Format* to open the *Display Settings* window.
2. Check *Simple Formatting.*



You now need to define the nodes size.

3. In the *Size Based on* list, select the variable that will be represented by the size of the nodes. If you do not want to use this feature, select *Nothing.*

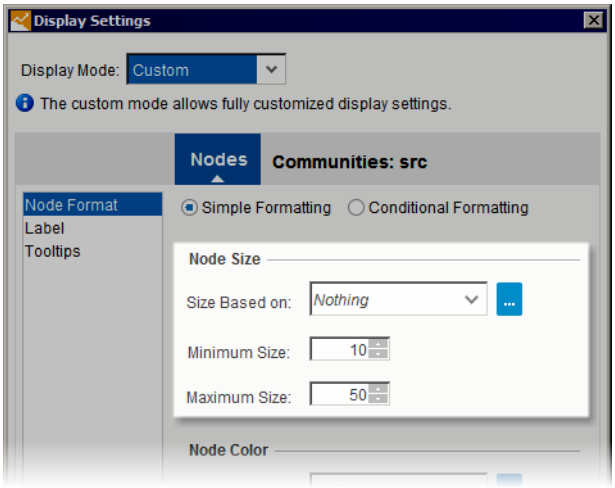
> **i Note**
>
> Only continuous variables are displayed in this list.

4. In *Minimum Size*, enter the diameter to use for the smallest value.

   This size is set in pixels and can range from 5 to 150.

5. In *Maximum Size*, enter the diameter to use for the highest value.

   This size is set in pixels and can range from 5 to 150.



You can now define the nodes color.

6. In the *Color Based* on list, select the variable that will be represented by the color of the nodes.

   If you do not want to use this feature, select *Nothing*.

   > **i Note**
   >
   > Only continuous and nominal variables are displayed in this list.

7. If you select a continuus variable:
   - Click the button located next to the label *Color for Minimum Value*.
   - Select the color corresponding to the variable smallest value.
   - Click the button located next to the label *Color for Maximum Value*.
   - Select the color corresponding to the variable highest value.

   The color of each node will be extracted from the color gradation going from the color selected for the minimum value to the color selected for the maximum value, depending on the variable value for the node.

8. If you select a nominal variable, the *Category* list is automatically filled with the selected variable categories. One color is associated to each category of the selected variable.
   - If need be, click the + button to add a value.
   - Modify the color of each category by clicking the associated color.

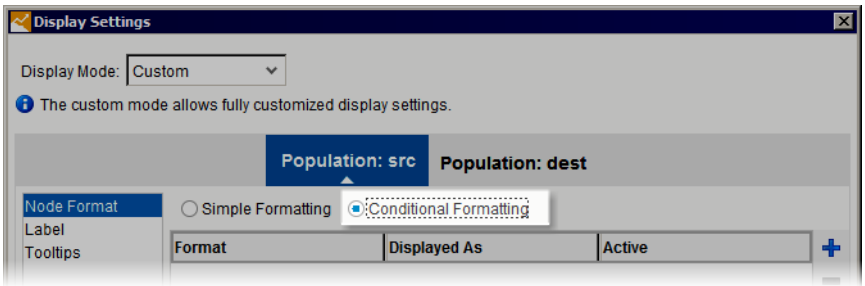
If you want to modify a simple format:

9. Double-click the format you want to modify in the *Graph Caption* list located on the left of the *Node Display* panel.

6.4.5.2.9 Customizing the Node Display: Setting a Conditional Format

Conditional formats allow you to apply a specific format to each node depending on a set of rules you have defined.

1. In the *Display* tab of the ribbon, click *Node Format* to open the *Display Settings* window.
2. Check *Conditional Format*.



3. Click + to add another conditional format to open the *Define Conditional Format* window.
4. Enter a name for the new format, which will appear in the *Graph Caption*.
5. Select how the node will be displayed: as a colored shape or as an icon.
6. Click + to add the conditions that will define to which nodes this format will apply.
7. Use the radio buttons located below the *Conditions* list to indicate when to apply this format.
8. Click *OK* to validate.

If you want to modify a conditional format:

9. Double-click the format you want to modify in the *Graph Caption* list located on the left of the *Node Display* panel.

6.4.5.2.10 Modifying the Labels Displayed on the Graph

You can choose which information to display on the graph for each node.

1. Click the button to open the *Display Settings* window.
2. Select *Labels* in the list located on the left.
3. In the *Available Attributes* list, select the information you want to display on the graph.

   To select and add several attributes at the same time:

   - Keep the Ctrl key pressed.
   - Select the attributes you want to add.
   - Click the > button to add the attributes to the list.

   To find and select attributes by their name or part of their name:

   - Click the button.
   - Enter the text to look for.
     If a set of attributes uses the same prefix, you can select them all by entering the prefix in the text field.

- Click the > button located on the right of the text field to select the attributes.
- Click the > button located between the lists to add the attributes to the list.
- Click the >> button to add all the attributes available.

6.4.5.2.11 Modifying the Attributes Displayed in the Tooltips

The tooltips allow you to display additional information about the nodes when you hover the cursor over a node. The tooltips are activated by default and all available attributes are displayed.

You can disable the tooltips.

1. Click the  button to open the *Display Settings* window.
2. Select *Tooltips* in the list located on the left.
3. Uncheck *Enable Tooltips*.

You can modify the attributes displayed in the tooltips. To select and remove several attributes at a time:

4. Keep the Ctrl key pressed.
5. Select the attributes you want to remove.
6. Click the < button to remove the attributes from the list.

To find and remove attributes by their name or part of their name:

7. Click the  button.
8. Enter the text to look for.

   If a set of attributes uses the same prefix, you can select them all by entering the prefix in the text field.
9. Click the > button located on the right of the text field to select the attributes.
10. Click the < button located between the lists to remove the attributes from the list *Selected Attributes*.
11. Click the << button to remove all the attributes from the list.

6.4.5.3 Sequences Visualization

The *Sequences Visualization* panel allows you to select the sequences you want to view on Google Earth.

In the table, the *Id* column corresponds to the path identifier. The *Sequences* column corresponds to the identifiers of the start and finish tiles.

Filtering the Sequences

- To Filter the Sequences
  1. In the list *Projection*, select the graph containing the geolocalization data you want to view.
  2. In the list *Sequence Length*, select an operator and a value. The operator and the value determine the length of your path.
     - The operator can be one of the following: equal to, lower than, higher than or in range.
     - The value corresponds to the number of tiles present in the path. The higher the number of tiles, the longer the path.

3. Click the *Filter Sequences* button to validate what you have defined.

- **Viewing the Sequences**

  The *Selection* button allows you to do specific selections, you can choose to:

  - make a toggle selection,
  - select all the rows,
  - unselect all the rows.

  You will then be able to view the selected sequences on Google Earth.

  - To preview the sequences
    1. Select the sequences you want to view,
    2. Click *Selection Preview*. A Google Earth window opens. The sequences are displayed with their latitude and longitude.

- **Exporting Sequences**

  - To Export Sequences
    1. In the **Export Type** list, select to which type of file you want to export the sequences data.

       Types of files available for export

Type of file	Description
KML	Keyhole Markup Language (KML) is an XML notation for expressing geographic annotation and visualization within Internet-based, two-dimensional maps and three-dimensional Earth browsers.
GML	Geography Markup Language (GML) is the XML grammar defined by the Open Geospatial Consortium (OGC) to express geographical features.
SHAPE_FILE	Shape File is a popular geospatial vector data format for geographic information system software.

    2. Click the *Export* button.
    3. Select either *Selected Sequences* or *All Sequences* depending on what you want to export.

- **GIS**

  The *Sequences Visualization* panel features a GIS (Geographic Information System). A GIS is a system that captures, manages, analyzes and displays all types of geographical data. Social enables you to add sequences to GIS, and thus, to manage and to display them.

  - To Add Sequences to GIS

  Note that to add a sequence to GIS, you need to define an address in the *Geographic Information System Protocol* field in the options of the application.

  1. In the table, select the sequences you want to add.
  2. Click *Add Sequence*. The field next to this button is automatically filled with the address you defined in the options of the application, accessible through the menu item *File* > *Preferences...*, followed by the centroid coordinates of the sequence.
  3. Click *Open*.

6.4.5.4 Recommendations

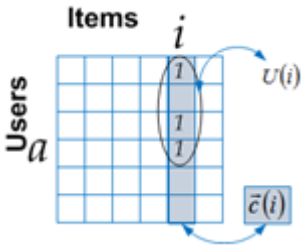
With Social, you can generate recommendations based on a graph derived from a bipartite graph. Before generating your model, you need to define a bipartite graph linking customers and products and then derive a graph linking the products together. You can then create your model and use it to generate product recommendations for a specific customer.

The recommendations are generated from association rules of the type X => Y. The rule means: if X (antecedent) is present, then Y (consequent) is also present. Two indicators, the Support and the Confidence, measure the quality of the rule. Several rules can lead to a same recommendation, however only the metrics corresponding to the rule with the best predictive power (KI) are provided for each recommendation.

6.4.5.4.1 Recommendation Types

There are several recomender: *Confidence*, *Support*, *Predictive Power*, *Combined Confidence* and *Cosine*. When getting recommendations, you can select the recommender you want to use. Below is a table that lists the differences between each recommender.

Label	Description
Recommendation	Recommended item (node).
Support	Number of times the rule used to recommend this item has been found in the dataset. For example, the number of times two products have been bought together.
Confidence	Number of times a node A has a node B as neighbor over the total number of times the node A appears in the dataset. For example, the number of times product A has been bought with product B over the total number of times product A has been bought.
Predictive Power	Predictive Power of the rule used to recommend this item.
Combined Confidence	$Sim(i,j) = P(i/j)^{\alpha} P(j/i)^{1-\alpha}$ where α is the Combination Factor. When the Combination Factor equals 1, the Combined Confidence is the Confidence. When the Combination Factor equals 0.5, the Combined Confidence is the Cosine count.

Label	Description
Cosine	 where a is the active user i and j: two items $U(i)$: set of users who purchased i $\vec{c}(i)$: column of matrix R for item i $Sim(i,j) = cos[\vec{c}(i), \vec{c}(j)] = \frac{\vec{c}(i) \bullet \vec{c}(j)}{\lVert \vec{c}(i) \rVert_2 \times \lVert \vec{c}(j) \rVert_2} = \frac{\sum_{u=1}^{L} w_{ui} \times w_{uj}}{\sqrt{\sum_{u=1}^{L} (w_{ui})^2} \times \sqrt{\sum_{u=1}^{L} (w_{uj})^2}}$

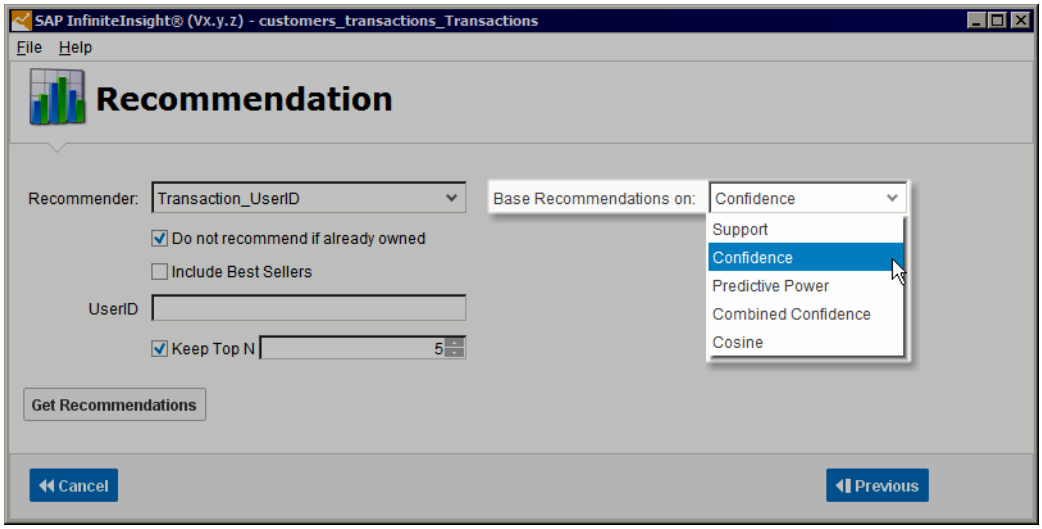
6.4.5.4.2 Getting Recommendations

1. In the *Recommender* list, select the graph containing the items you want to recommend. In the current example, this graph will contain products.

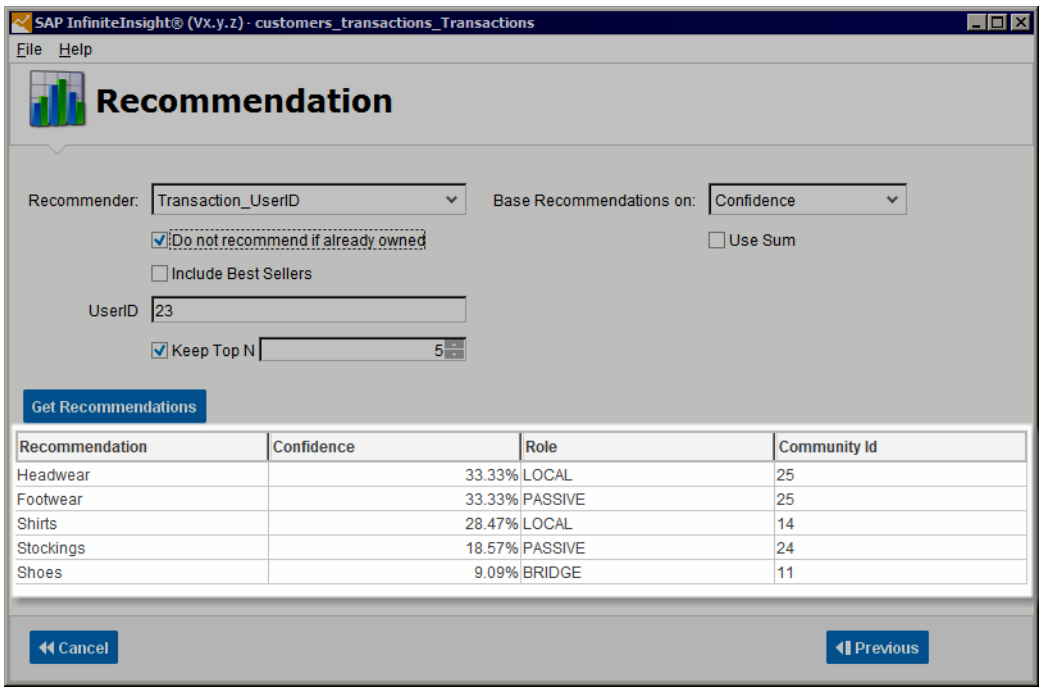
The name of the next field, which varies depending on the *Recommender*, is the name of the variable containing the identifier of the item for which you want to make recommendations. In the current example, it will be the name of the variable containing the customers identifiers.

2. Enter the value corresponding to the specific item for which you want to make the recommendations. For example, a customer's identifier.
3. **Optional:** If you want to exclude the items already owned from the recommended items, check *Do not recommend if already owned*.
4. **Optional:** If you want to include the best sellers, check *Include Best Sellers*.
5. **Optional:** If you want to keep the indicated number of top pairings, check *Keep Top N*.

   The top pairings are determined by the pairing type. When set to 0, no filter is applied. The graphs keep the N links out with the biggest weights, so when the user wants to generate graphs with both ratio and counts, links with different pairs may be kept.
6. In the *Base Recommendations on* list, select the label you want to base your recommendations on.



7. Click *Get Recommendations* to list the recommended items.



You can sort the recommended items by clicking the heading of the column used as sorting criteria.

6.4.5.4.3 Batch Recommendations

When creating recommendations with Social, this panel allows you to generate a list of recommended products for each of the customers in your dataset.

- Setting the Generation Options [page 360]
- Selecting the Application Dataset [page 361]
- Saving the Results Generated by the Model [page 362]

- Using the In-Database Apply Mode [page 362]

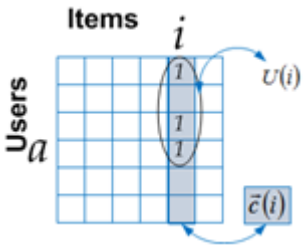
6.4.5.4.3.1 Setting the Generation Options

- To Set the Generation Options
  1. In the *Recommender* list, select the recommender you want to use to generate the recommendations.
  2. The following options are available to customize your output dataset.
     - *Do not recommend if already owned*
       By default, products that have already been bought by the customer will not be recommended.

     Uncheck this option if you want to include these products in the recommendations.
     - *Include Best Sellers:*
       By default, best sellers are not included in the recommendations. The recommendation of Best Seller products is not linked to specific users but depends on a specific analysis.

     Check this option if you want best sellers to be added to the recommended products.
     - *Keep Top N*
       By default, only the top five recommendations are provided in the results dataset.
       - To change the number of recommendations included in the results, change the number in the field *Keep Top N*.
       - To get all the generated recommendations, uncheck this option. Be aware that there may be a high number of recommendations, so do be careful when deactivating this option.
     - *Base Recommendations on*
       This option allows you to select the metrics to be used to sort the recommendations. By default, the recommendations are sorted by descending confidence.
     - *Use Sum*
       When checked, this option uses the sum of the metrics of the rules having the same consequent, that is, recommending the same product.
     - *Use Weight*
       This option is only available when *Use Sum* is activated. When checked, it uses the weight column defined when setting the model parameters to weight the sum.
     - *Combination Factor*
       Is a parameter of the *Combined Confidence*- see Metrics and Formulas below.

Metrics and Formulas

Metrics	Formula and Comments
Combined Confidence	$Sim(i,j) = P(i/j)^{\alpha} P(j/i)^{1-\alpha}$ where α is the Combination Factor . • When the Combination Factor equals 1, the Combined Confidence is the Confidence. • When the Combination Factor equals 0.5, the Combined Confidence is the Cosine count.
Cosine	 where • a : the active user • i and j: two items • $U(i)$: set of users who purchased i • $\vec{c}(i)$: column of matrix R for item i $Sim(i,j) = cos[\vec{c}(i),\vec{c}(j)] = \frac{\vec{c}(i) \bullet \vec{c}(j)}{\lVert\vec{c}(i)\rVert_2 \times \lVert\vec{c}(j)\rVert_2} = \frac{\sum_{u=1}^{L} w_{ui} \times w_{uj}}{\sqrt{\sum_{u=1}^{L}(w_{ui})^2} \times \sqrt{\sum_{u=1}^{L}(w_{uj})^2}}$
KI	$Sim(i,j) = (P(i \& j) - P(i) * P(j)) / (1 - P(i)) * P(i)$

6.4.5.4.3.2 Selecting the Application Dataset

- To Select the Application Dataset
  1. In the section *Application Dataset*, click the *Browse* button on the *Folder* line to select the subdirectory of the source file.
  2. Click the *Browse* button of the *Data* line to select the text file or ODBC source.The input data format must be the same as the one used to previously build the model.

> **i Note**
>
> - A warning may be displayed indicating that there are more fields in the dataset than needed by the model. This is an informational message that does not need any action from your part. Click *Close* to continue with the application process.

- If you have activated the *Community mode* when setting the parameters of the model, the application dataset needs to include the variable named kxComIndex that contains communities identifiers. It should be a nominal integer.
  This variable allows you to apply on the community graphs (asking for neighbors, for nodes list in community or for aggregated statistics). You can use this column to provide the list of community identifiers on which you want to compute metrics. If you are not applying on community graphs, you should leave it empty or, in case of a database, fill it with a dummy value as it will not be used.

6.4.5.4.3.3 Using the In-Database Apply Mode

This optimized scoring mode allows applying the model directly into the database. It avoids extracting the data from the database, and speeds up the writing process of the model outputs. This mode can be used if all the following conditions are met:

- the input application dataset (table, view, select statement, data manipulation) and the results dataset are tables from the same database
- the input dataset used to create the model contains at least one variable declared as a key
- the *In-database Apply* mode is not deactivated
- you have the necessary rights to read and write in the database
- there is a valid code generator license for the database
- no error has occurred
- themodel must be saved in the database that contains the application datasets

> **i Note**
>
> If the model has not been saved, a message box is displayed allowing you to save the model.

6.4.5.4.3.4 Saving the Results Generated by the Model

- To Save the Results Generated by the Model
  - In the section *Results generated by the model*, select the type of the output dataset in the list *Data Type*.
  - specify the output directory by clicking the *Browse* button on the *Folder* line.
  - Enter a name for the output file on the *Data* line or click the *Browse* button to select a file.
  - Click *Apply*.

> **i Note**
>
> A warning may be displayed indicating that there are more fields in the dataset than needed by the model. This is an informational message that does not need any action from your part. Click *Close* to continue with the application process.

6.4.5.5 Applying the Model

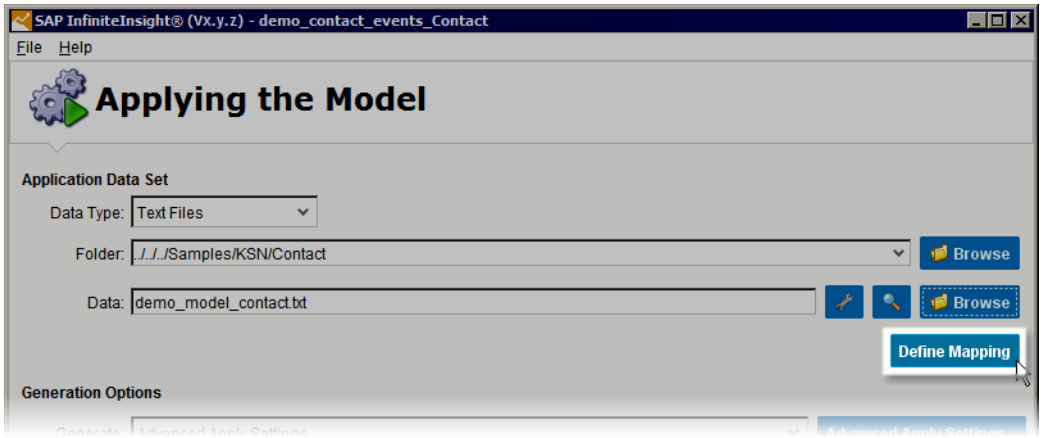
The currently open model may be applied to additional datasets. The model allows you to perform predictions using the application datasets, and specifically, to make recommendations based on the datasets used to generate the model.

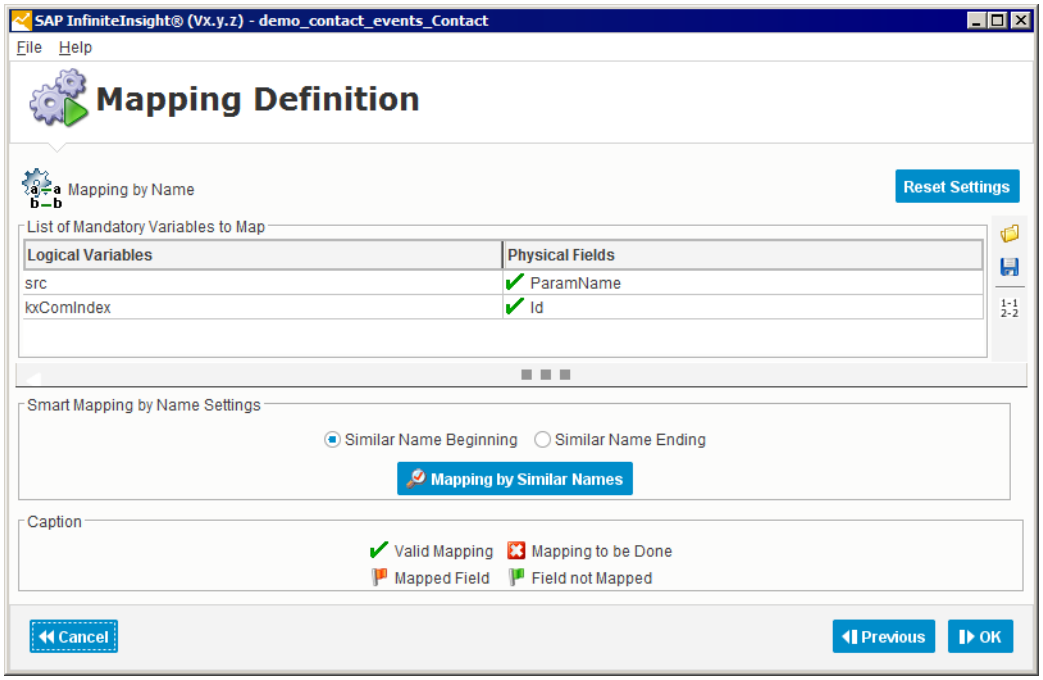
6.4.5.5.1 Constraints of Model Use

In order to apply a Social model to a dataset, it must contain the following variables:

- one variable for each population; for example, one for customers and one for products.
- the `kxComIndex` variable, which contains communities identifiers. It should be a nominal integer. This variable allows you to apply on the community graphs (asking for neighbors, for nodes list in community or for aggregated statistics). You can use this column to provide the list of community identifiers on which you want to compute metrics. If you are not applying on community graphs, you should leave it empty or, in case of a database, fill it with a dummy value as it will not be used.

To make sure that the datasets are consistent, use the mapping feature provided in the panel *Applying the Model*.





6.4.5.5.2 Applying the Model to a New Dataset

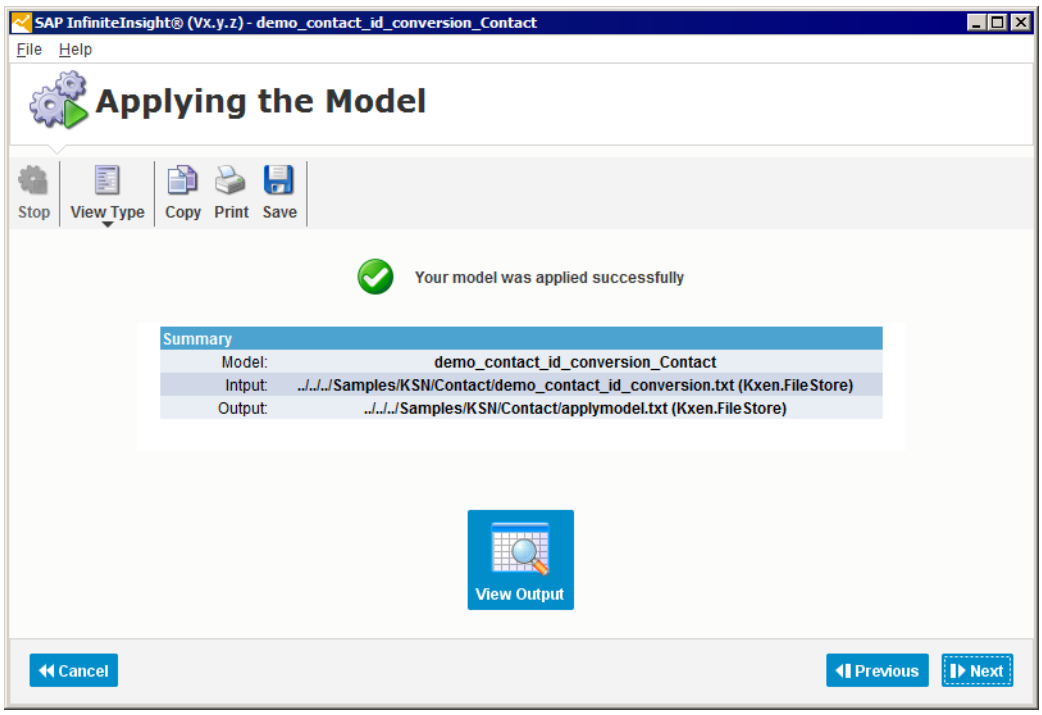
1. Select the option *Apply the Model* to open the *Using the Model* panel.
2. In the section *Application Dataset*, click the *Browse* button to select:
   a. the folder which contains your dataset,
   b. the name of the file corresponding to your dataset.
3. In the *Generate* list, select the type of output you want to generate:

If you select the option...	the generated data will contain for each node...
Default Mode	All the information available on the node and its neighbors.
Circle Mode	○ The number of neighbors. ○ Additional information on the neighbors, if available. ○ The average of the neighbors' attributes.
Centrality Mode	An evaluation of its centrality by analyzing its local clustering and its number of neighbors.
Neighbors Mode	A list with all its neighbors and additional information on them.
Describe Mode	The list of all information available for this node.
Community Mode	Information concerning the community to which it belongs as well as the role played.

If you select the option...	the generated data will contain for each node...
Node Pairing Mode	Information about its presence in the graph(s) and its neighborhood (neighbors count, ratio common_neighbors/average_neighbors_count).
Advanced settings	Information selected by the user (node class, triangle count, neighborhood information, community index, node role in community, etc.).

> **i Note**
>
> At the end of the apply process a button will allow you to display the outputs in the wizard.

4. In the section *Results Generated by the Model*, enter:
   a. the name of the folder where you want to save the output of the apply,
   b. the name of the file containing the output of the apply.
5. Click *Apply*.
6. At the end of the applying process, click *View Output* to display the resulting table.

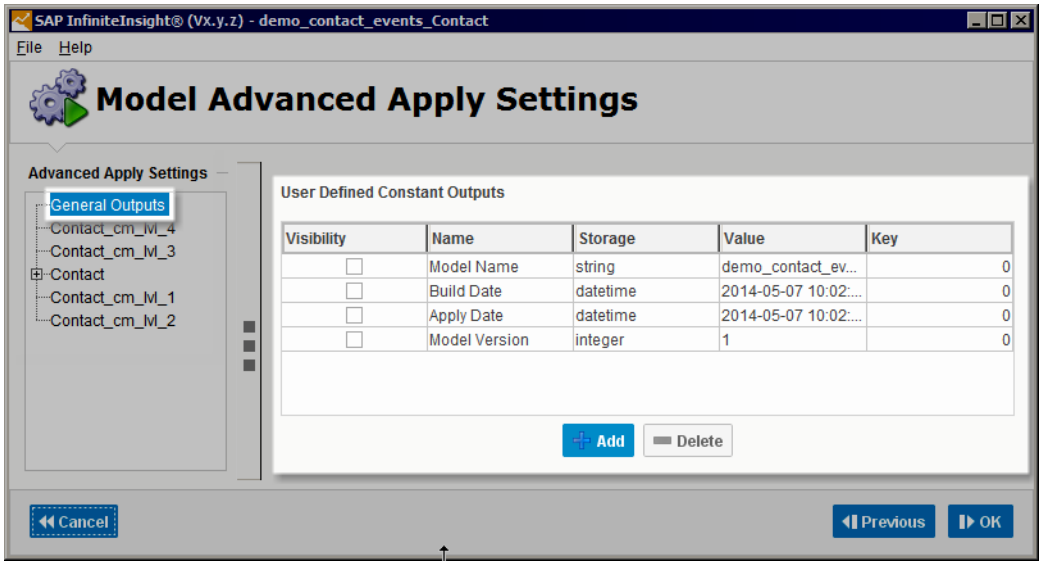


7. Click *Next* to go back to the *Using the Model* panel.

6.4.5.5.3 Advanced Apply Settings: Defining the Content of the Output File

1. Click *Advanced Apply Settings...* to select which content should appear in the output file resulting from the model apply on a new dataset.

The navigation tree located on the left of the panel allows you to define the different output types: the general parameters, the parameters for each graph in the model being applied, the parameters specific to the node pairing, and so on.



You can also define constant outputs.

2. Check the *Visibility* box of one of the suggested constants or add a new constant.

To add a new constant:

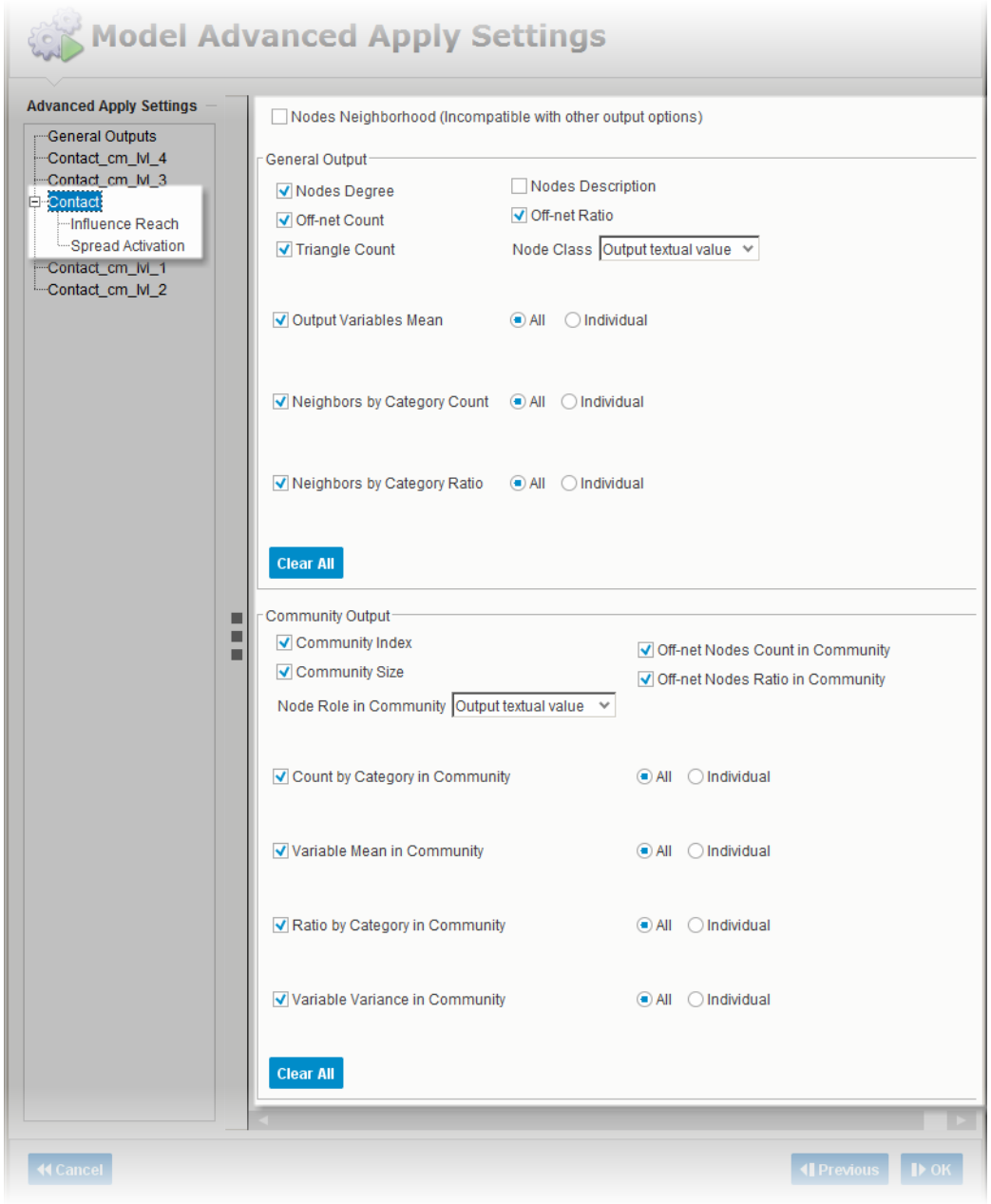
3. Click *Add*.
   A pop-up window opens allowing you to set the constant parameters.
4. In *Output Name*, enter the constant name.
5. In the *Output Storage* list, select the constant type (number, string, integer, datetime or date).
6. In *Output Value* enter the constant value.

> **i Note**
>
> Dates should be entered with the format `YYYY-MM-DD`.

7. Click *OK* to create the constant.
   The new constant appears in the list. You can choose whether to generate the defined constants or not by checking the *Visibility* box.

6.4.5.5.4 Graph Outputs Customization

In addition to the *General Outputs*, you can define the output settings for each graph you have created.



The different output options are checked according to the generation mode selected on the *Applying the Model* panel. In the navigation tree, select the graph for which you want to modify the settings.

Option	Output	Generated Variable Names
Nodes Neighborhood	All additional information on the neighbors of the node, with link information such as direction and link weight. Note that if you activate this option, it is not possible to activate another one.	

Option	Output	Generated Variable Names

Nodes Degree	The number of neighbors and also in the case of a directed graph the link direction.
	Directed graphs have distinct count or weight for incoming and outgoing link.
		`sn_<graphname>_dg (degree)`
		`sn_<graphname>_w_dg (weighted degree)`
		`sn_<graphname>_w_dg (weighted degree)`
		`sn_<graphname>_i_w_dg (incoming weighted degree)`
		`sn_<graphname>_o_w_dg (outgoing weighted degree)`

Option	Output	Generated Variable Names

Nodes Description	All additional information on the node itself.	Activating this flag generates the following variables depending on the input node population:
		
		Basic Nodes
		
		`sn_d_degree_total (total degree)`
		`sn_d_is_offNet (is the node off net?)`
		`sn_d_role (role in community)`
		`sn_d_intra_cm_link_count (number of links inside its community)`
		`sn_d_extra_cm_link_count (number of links outside its community)`
		`sn_d_<var_from_dataset> (descriptive attribute from data)`
		
		Community Nodes
		
		`sn_d_size (size of the community)`
		`sn_d_internal_weight (sum of weights inside the community)`
		`sn_d_external_weight (sum of weights outside the community)`
		`sn_d_offnet_count (count of off-net nodes inside the community)`
		`sn_d_offnet_ratio (ratio of off-net nodes in the community)`
		`sn_d_density (density of the community)`
		`sn_d_mean_<continuous_var> (mean of a continuous var in the community)`
		`sn_d_count_<nominal_var>_<category> (count of a nominal var category in the community)`
		`sn_d_ratio_<nominal_var>_<category> (ratio of a nominal var category in the community)`
		
		Some of the community nodes descriptive outputs are the same proposed in the "Community Outputs" section.

Option	Output	Generated Variable Names

Off-Net Count	The number of neighbors that are off-net.	`sn_<graphname>_c_off (off-net count)`
		`sn_<graphname>_i_c_off (incoming off-net count)`
		`sn_<graphname>_o_c_off (outgoing off-net count)`

Off-Net Ratio	The ratio between the off-net neighbors and the total number of neighbors	`sn_<graphname>_r_off (off-net count)`
		`sn_<graphname>_i_r_off (incoming off-net count)`
		`sn_<graphname>_o_r_off (outgoing off-net count)`

Triangle Count	The number of triangles found for the node.	`sn_<graphname>_tc`
Node Class	Only for directed graphs: indicates whether a node is a sink, a source, or neither (repeater).	`sn_<graphname>_sg`
Community Index	The number of the node's community.	`sn_<graphname>_cm_idx_<community_level>`
Off-Net Nodes Count in Community	The number of neighbors in the community that are off-net.	`sn_<graphname>_cm_c_off_<community_level>`
Community Size	The total number of nodes in the community.	`sn_<graphname>_cm_sz_<community_level>`
Off-Net Nodes Ratio in Community	The ratio between the off-net neighbors and the total number of neighbors in the community.	`sn_<graphname>_cm_r_off_<community_level>`
Node Role in Community	The role of the node inside its community (bridge node, local node, social node or passive node).	`sn_<graphname>_cm_rl_<community_level>`

> **i Note**
>
> The outputs Nodes Degree Info, Off-Net Count and Off-Net Ratio generate two columns for directed graphs, one column for incoming and one for outgoing links.

For some of these options, you can choose if you want to apply them to all variables (*All*) or only to the ones you select (*Individual*). To select individual variables, use the > button.

Option	Variable Type	Output	Generated Variable Names

Output Variables Mean	continuous	The mean value of the variable	`sn_<graphname>_u_m_<var_name> (undirected)`
			`sn_<graphname>_i_m_<var_name> (incoming)`
			`sn_<graphname>_o_m_<var_name> (outgoing`

Output Variables Count	continuous	A column per value and the number of neighbors for each column	
Output Variable Ratio	continuous	A column per value and the ratio between the neighbors with the value of the column name and the total number of neighbors	
Count by Category in Community	continuous	A column per value and the number of neighbors in the community for each column	`sn_<graphname>_cm_c_<community_level>_<var_name>_<cat_name>`
Variable Mean in Community	continuous	The mean value of the variable in the community	`sn_<graphname>_cm_m_<community_level>_<var_name>`
Ratio by Category in Community	continuous	A column per value and the ratio between the neighbors in the community with the value of the column name and the total number of neighbors in the community	`sn_<graphname>_cm_r_<community_level>_<var_name>_<cat_name>`
Variable Variance in Community	continuous	The variance of the variable in the community	`sn_<graphname>_cm_v_<community_level>_<var_name>`

> **i Note**
>
> If the graph has weighted links the ratio, the count and the degrees take these weights into account.

Influence Reach

The metric Influence Reach detects and outputs the size of the cascades collection originated from a given node based on a date attribute. So it tells us what nodes seem to have adopted a given behavior after a source node did. It also computes the maximum depth of the cascade, which is the shortest path length to the farthest node in the contagion chain.

For each node ID given in the application dataset, two fields are computed:

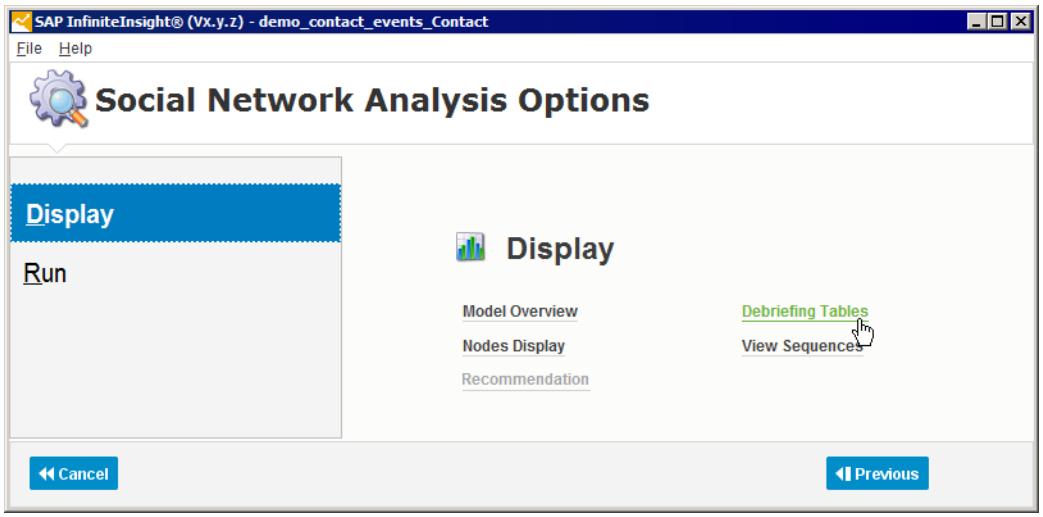
- `sn_<graphname>_ir_count_<var_name>` : Influence Reach total size
- `sn_<graphname>_ir_depth_<var_name>` : shortest path length to the farthest node in the cascade (maximum influence depth)

Spread Activation

Spreading activation is a method for searching associative networks, neural networks, or semantic networks. The search process is initiated by labeling a set of source nodes (for example, concepts in a semantic network) with weights or "activation" and then iteratively propagating or "spreading" that activation out to other nodes linked to the source nodes. Tthese "weights" are real values that decay as activation propagates through the network. When the weights are discrete this process is referred to as marker passing. Activation may originate from alternate paths, identified by distinct markers, and terminate when two alternate paths reach the same node.

6.4.5.6 Displaying the Statistical Reports

On the *Social Network Analysis Options* panel, click the button *Debriefing Tables*.



The messages displayed depends on the model:

Descriptive Statistics

- Variables
- Category Frequencies
- Dataset Size

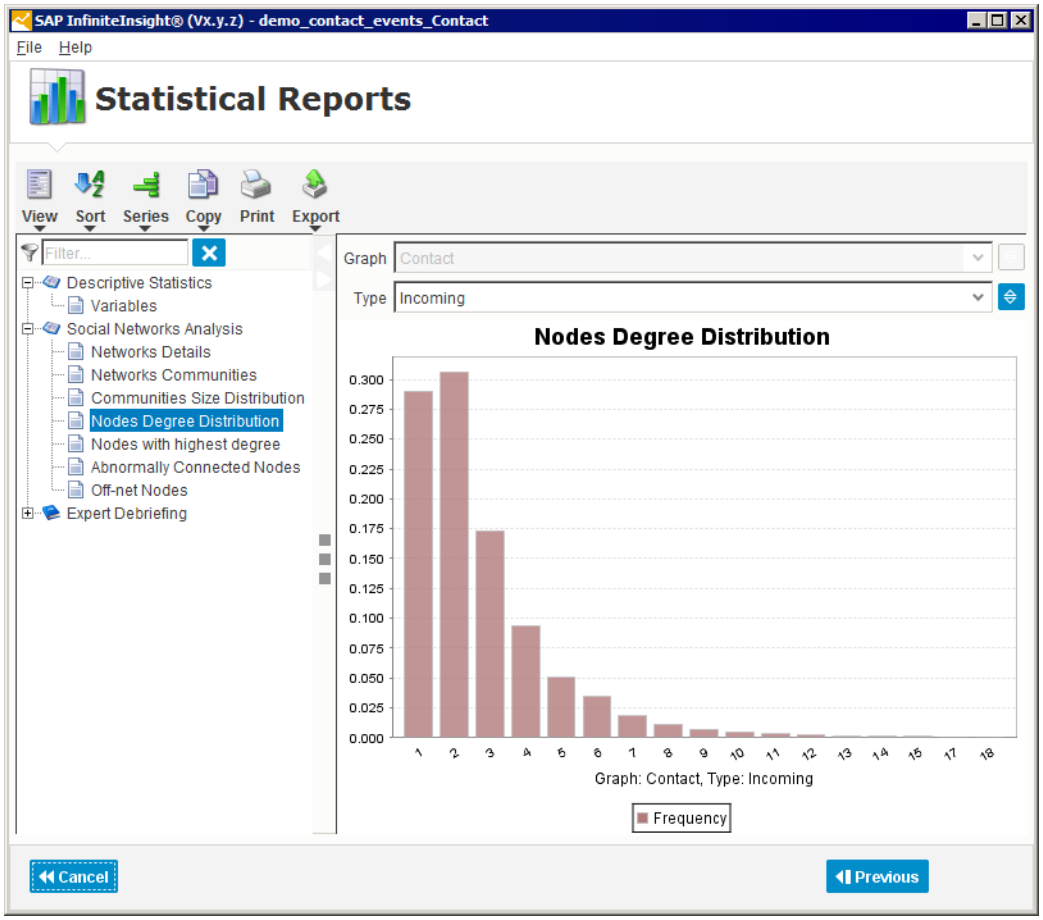
Social Network Analysis

- Network Details
- Bipartite Network Details

- Nodes Degree Distribution

Expert Debriefing

- Model Settings
- Graph Load Type
- Nodes Settings



6.4.5.7 Exporting the KxShell Script

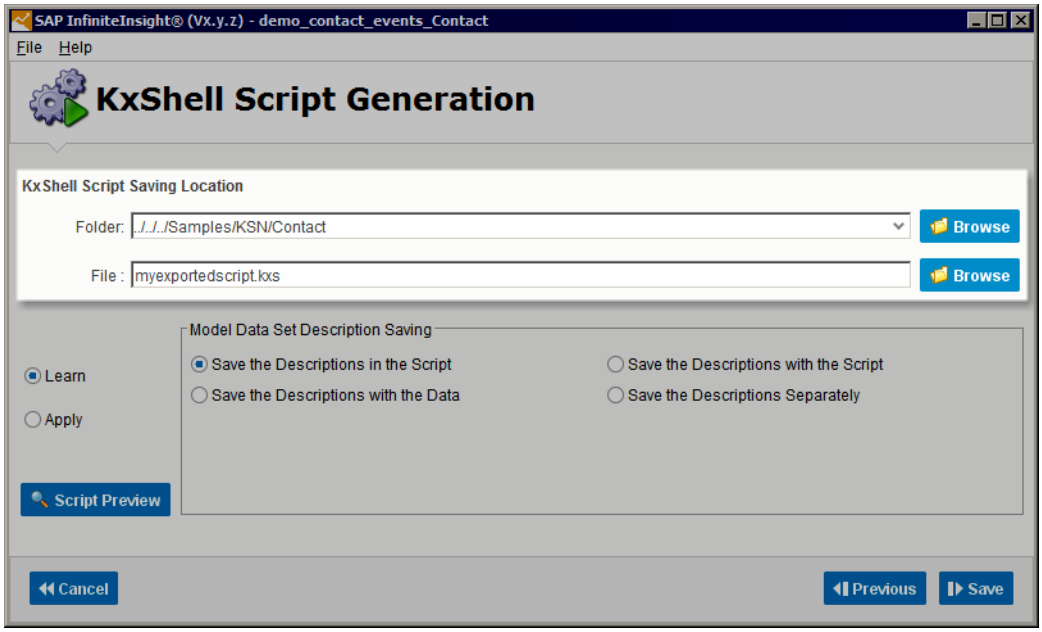
This feature allows you to generate a KxShell script reproducing the current model. This script can be used to run models in batches.

One easy way to get special settings in exported KxShell scripts is to first do the corresponding operation in the graphical user interface. For example, if you run an auto-selection of variables before exporting the shell script, then the exported script will include the code needed to do the auto-reduction.

6.4.5.7.1 Selecting the Saving Location

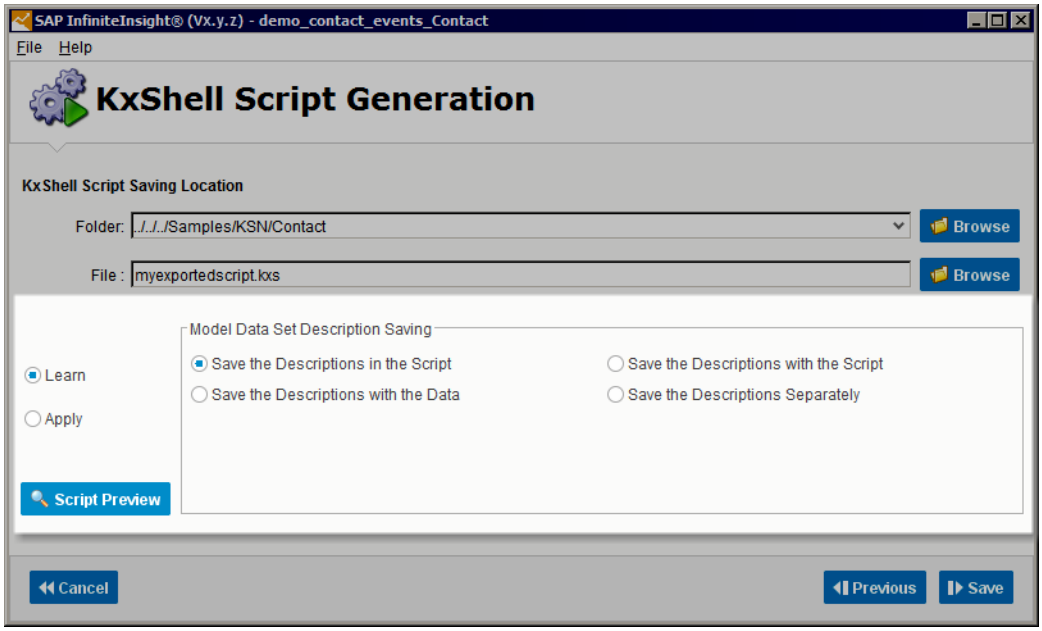
1. Click the *Browse* button located on the right of the *Script File* field.
2. Select the folder where you want to save the script.

   If the file already exists, select it. Else enter **/** followed by the new file name after the folder name.
3. Click *OK*.



6.4.5.7.2 Exporting the Script to Train the Model

1. Check the *Learn* option.



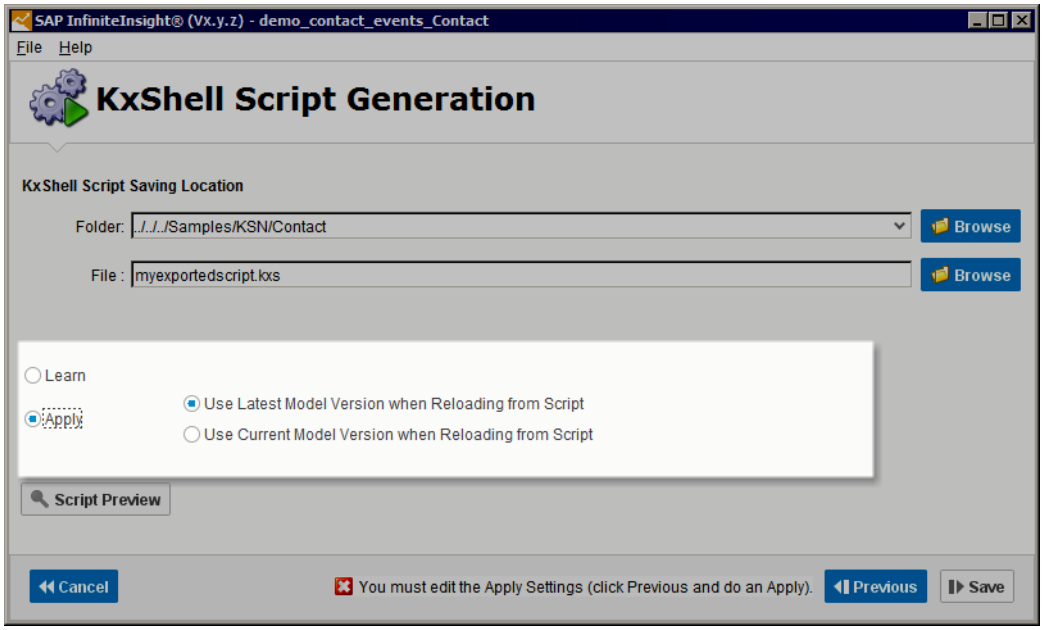
2. In the *Model Dataset Description Saving* frame, select the location where you want to save the data description. The following options are available:
   - *Save the Description in the Script*: the data description is added in the KxShell script. Only one file is generated.
   - *Save the Description with the Script*: the data description is saved in an additional file in the same folder as the KxShell script.
   - *Save the Description with the Data*: the data description is saved in an additional file in the same folder as data used for the model.
   - *Save the Description Separately*: the data description is saved in an additional file. The user indicates the type of the description (text file, data base, flat memory, ...) and the location where the data description should be saved.

> **i Note**
>
> When saving the description in an additional file, the file is named according to this syntax: `KxDesc_<Dataset Role>_<Dataset Name>`. For example, for a training dataset named Census.csv, the description file name will be `KxDesc_Training_Census.csv`.

3. Click *Next* to start the generation process.

6.4.5.7.3 Exporting the Script to Apply the Model

1. Check the *Apply* option.



> ⚠ Caution
>
> You need to apply the model once in the wizard before exporting the script.

2. Select the model version you want to use:
   - *Use Latest Model Version when Reloading from Script*: the script will always apply the latest created version of the model.
   - *Use Current Model Version when Reloading from Script*: the script will always apply the version that has been used to export the script.
3. Click *Next* to start the generation process.

6.4.5.8 Saving The Model

Once a model has been generated, you can save it. Saving it preserves all the information that pertains to that model, that is, the modeling parameters, its profit curves, and so on.

1. On the screen *Using the Model*, click *Save the Current Model* to open the *Saving the Model* window.
2. Enter a name for the model.

   This name will then appear in the list of models to be offered when you open an existing model.
3. Give a description of the model.

   Use this field to enter additional information that could help identify your model, such as the name of the training dataset, the polynomial degree or the predictive power and the prediction confidence.
4. Select the type of storage in which you want to save your model in the *Data Type* list. The following options are available:
   - *Text Files*, to save the model in a text file.
   - *Data Base*, to save the model in a database.
   - *Flat Memory*, to save the model in the active memory.

5. Specify a folder.

   Depending on which option you selected, this field allows you to specify the ODBC source, the memory store or the folder in which you want to save the model.

6. Enter a name for the file or the table that is to contain the model.

   The name of the file must contain one of the following format extensions .txt (text file in which the data is separated by tabs) or .csv (text file in which the data is separated by commas).

6.4.5.8.1 Files Created When Saving a Model

When saving a model, the application creates a set of files/tables in the specified store. Some of these files are specific to the type of model. The following table lists the files or tables created when saving a model and in which case.

File	Description	Used By
`KxAdmin`	Lists all the models contained in the folder/database with additional information (date, version, name of the model, comments).	All models created with Automated Analytics
`<Model_name>`	File named after the model and containing all the model data, except graphs information. Graphs are stored in additional tables (see below).	All models created with Automated Analytics
`KxInfos`	Indicates which additional tables are needed by the model.	All models created with Automated Analytics
`KxOlapCube`	Stores the OLAP Cube used by the decision tree when the option Regression/Classification as Decision is activated.	Regression/Classification models with decision tree
`KxLinks`	Contains the links from the graphs of the model.	Social model only
`KxNodes`	Lists all the nodes from all the graphs and their attributes.	Social model only
`KxCommunities`	Matches the nodes to their communities, if the community detection was enabled.	Social model only

> ⚠ Caution
>
> When sharing or sending a model, all these files must be joined to the model or the recipient will not be able to open the model.

6.4.6 Step 5 - Opening a Model

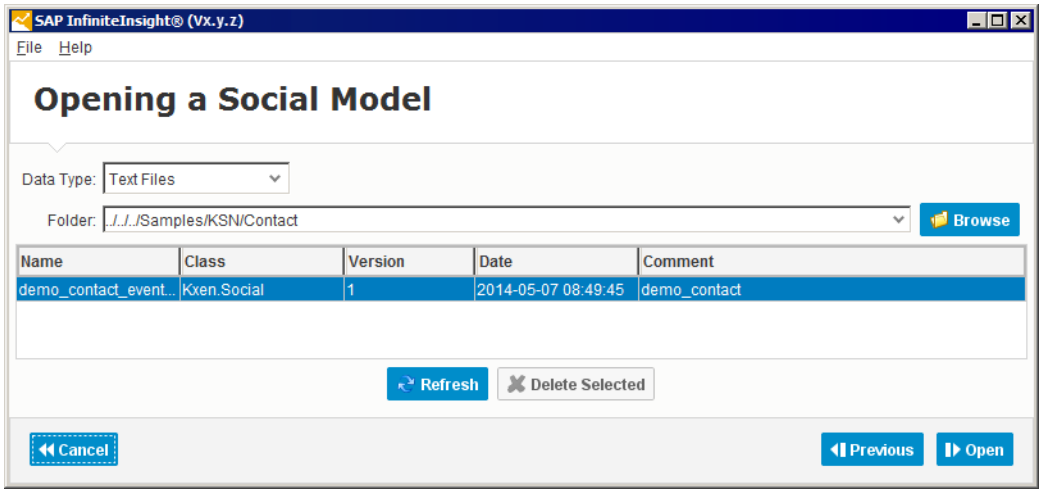
Once saved, models may be opened and reused in SAP Predictive Analytics.

> **⚠ Caution**
>
> Several files/tables are needed to be able to open a model. If some of these files/tables are missing a message will be displayed indicating the names of the missing files/tables.

To open a model:

1. On SAP Predictive Analytics main screen, click *Social* > *Load a Social Network Analysis Model* in the Social section.
2. In *Data Type*, select the storage format of the model you want to open.
3. Use the *Browse* button located on the right of the *Folder* field to select the folder or database that holds the model that you want to open.

   When displaying the content of a database, you may need to click *Refresh* to see all the models, especially those created recently.



The following table lists the information provided for each model.

Column	Description	Values
Name	Name under which the model has been saved.	Character string

Column	Description	Values
Class	Class of the model, that is the type of the model.	○ `Kxen.Classification`: Classification/Regression with nominal target ○ `Kxen.Regression`: Classification/Regression with continuous target ○ `Kxen.Segmentation`: Clustering with SQL Mode ○ `Kxen.Clustering` : Clustering without SQL Mode ○ `Kxen.TimeSeries`: Time Series ○ `Kxen.AssociationRules`: Association Rules ○ `Kxen.SimpleModel`: Classification/Regression and Clustering multi-target models, any other model ○ `Kxen.Social`
Version	Number of the model version when the model has been saved several times.	Interger starting at 1
Date	Date when the model has been saved.	Date and time in the `yyyy-mm-dd hh:mm:ss` format
Comment	Optional user defined comment that can be used to identify the model.	Character string

4. Select a model from the list
5. Click *Open*.
   The *Social Network Analysis Options* panel appears.

Related Information

Files Created When Saving a Model [page 149]

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering a SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Gender-Related Language

We try not to use gender-specific word forms and formulations. As appropriate for context and readability, SAP may use masculine word forms to refer to all genders.

www.sap.com/contactsap

THE BEST RUN SAP