Table of contents

|  |
| --- |
| 1. [Introduction to SQL Server Data Mining](https://www.sqlshack.com/introduction-to-sql-server-data-mining/)   <https://www.sqlshack.com/introduction-to-sql-server-data-mining/> |
| 1. [Naive Bayes Prediction in SQL Server](https://www.sqlshack.com/naive-bayes-prediction-in-sql-server/)   <https://www.sqlshack.com/naive-bayes-prediction-in-sql-server/> |
| 1. [Microsoft Decision Trees in SQL Server](https://www.sqlshack.com/microsoft-decision-trees-in-sql-server/)   <https://www.sqlshack.com/microsoft-decision-trees-in-sql-server/> |
| 1. [Microsoft Time Series in SQL Server](https://www.sqlshack.com/microsoft-time-series-in-sql-server/)   <https://www.sqlshack.com/microsoft-decision-trees-in-sql-server/> |
| 1. [Association Rule Mining in SQL Server](https://www.sqlshack.com/the-association-rule-mining-in-sql-server/)   <https://www.sqlshack.com/the-association-rule-mining-in-sql-server/> |
| 1. [Microsoft Clustering in SQL Server](https://www.sqlshack.com/microsoft-clustering-in-sql-server/)   <https://www.sqlshack.com/microsoft-clustering-in-sql-server/> |
| 1. Microsoft Linear Regression in SQL Server   <https://www.sqlshack.com/microsoft-linear-regression-in-sql-server/> |
| 1. [Implement Artificial Neural Networks (ANNs) in SQL Server](https://www.sqlshack.com/implement-artificial-neural-networks-anns-in-sql-server/)   <https://www.sqlshack.com/implement-artificial-neural-networks-anns-in-sql-server/> |
| 1. [Implementing Sequence Clustering in SQL Server](https://www.sqlshack.com/implementing-sequence-clustering-in-sql-server/)   <https://www.sqlshack.com/implementing-sequence-clustering-in-sql-server/> |
| 1. [Measuring the Accuracy in Data Mining in SQL Server](https://www.sqlshack.com/measuring-the-accuracy-in-data-mining-in-sql-server/) |

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/introduction-to-sql-server-data-mining/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Introduction to SQL Server Data Mining

July 23, 2019 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

Prediction, is it a new thing for you? You won’t believe you are predicting from the bed to the office and to back to the bed. Just imagine, you have a meeting at 9 AM at the office. If you are using public transport, you need to predict at what time you have to leave so that you can reach the office for the meeting on time. Time may vary by considering the time and the day of the week, and the traffic condition etc. Before you leave your home, you might predict whether it will rain today and you might want to take an umbrella or necessary clothes with you. If you are using your vehicle then the prediction time would be different. If so, you don’t need to worry about the rain but you need to consider the fuel level you need to have to reach to the office. By looking at this simple example, you will understand how critical it is to predict and you understand that all these predictions are done with your experience but not by any scientific method.

The next question would be how to implement any data mining solution in a real-world scenario. Well, you might have heard of the famous story of [Beer-Nappy](https://bigdatabigworld.wordpress.com/2014/11/25/beer-and-nappies/) at the popular supermarket chain. That is just a simple example of data mining implementation. So let’s see how we define data mining.

## What is Data Mining?

There are several definitions for data mining with respect to business as well as for academics. Data mining is a practice that will automatically search a large volume of data to discover behaviors, patterns, and trends that are not possible with the simple analysis. Data Mining should allow businesses to make proactive, knowledge-driven decisions that will make the place better ahead of their competitors.

Data warehouse, from its mandate to store a large volume of data including the last years of data. The data warehouse is used for descriptive analysis (What happened) and diagnostic analysis (Why it happened). However, business needs to do analysis beyond that. Data mining can be utilized for Predictive Analysis (What will happen) and Prescriptive Analysis (How can we make it happen).

## Data Mining in SQL Server

SQL Server is mainly used as a storage tool in many organizations. However, with the increase of many businesses’ needs people are looking to different features of SQL Server. People are looking at data warehousing with SQL Server. SQL Server is providing a Data Mining platform which can be utilized for the prediction of data.

There are a few tasks used to solve business problems. Those tasks are Classify, Estimate, Cluster, forecast, Sequence, and Associate. **SQL Server Data Mining has nine data mining algorithms** that can be used to solve the aforementioned business problems. **The following are the list of algorithms that are categorized into different problems.**

**Classify:** Categorized depending on the various attributes. For example, whether a customer is a prospect customer depending on other data such as Age, Gender, Marital Status, Occupation, Education Qualification, etc.

**Estimate:** Estimation will be done using the parameters. For example, house prices will be predicted depending on the house location, house size, etc.

**Cluster:** also named as segmentation. Depending on the various attribute natural grouping is done. Customer Segmentation is the classical business example for the clustering.

**Forecast:** Predict continuous variable for with the time. Predicting sales volume for the next couple of years is a very common scenario in the industry.

**Associate:** Finding common items or groups in one transaction. The transaction can be a supermarket sales, or medicine or online sales.

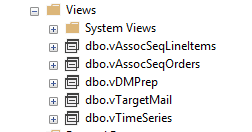
**Sequence:** Predicting the Sequence of events.

## Platform

SQL Server 2017 is used in this article but if you have SQL Server 2012 onwards you can still follow this.

In this article series, we will be using a sample data set which you can download and run through with the article. You can download [AdventureWorks database](https://docs.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-2017) and install it to your SQL Server instance.

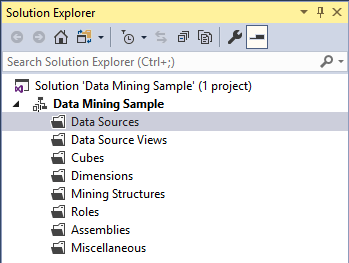
There are fact and dimension tables in the sample database. However, we will be using the below-listed views predominantly here.



During the article series, we will look at these views in detail.

## Data Mining Project

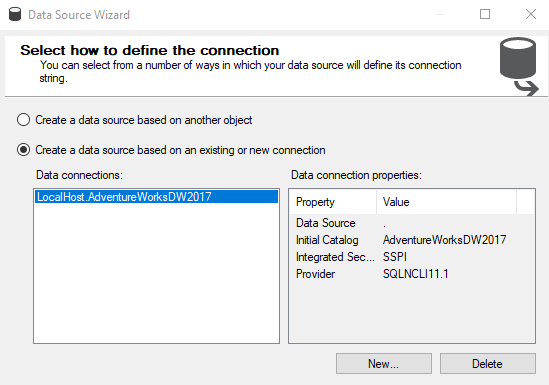
Let us create a data mining project. Open **Microsoft** **Visual Studio**and create a **Multidimensional**project under **Analysis Service** and select **Analysis Services Multidimensional and Data Mining**project. Following is the **Solution Explorer** for the created project.



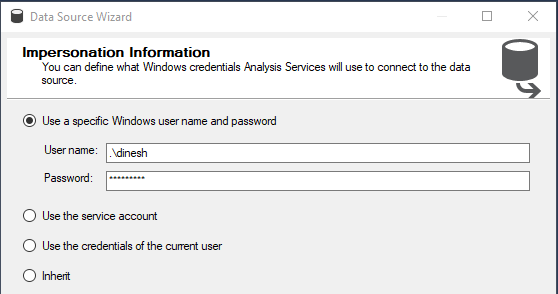
For data mining, we will be using three nodes, **Data Sources, Data Source Views,**and **Data Mining.**

## Data Sources

We need to configure the data source to the project as shown below. The data source makes a connection to the sample database, AdventureWorksDW2017.



After providing the credential to the source database, next is to provide the credentials to the Analysis service to connect to the database.



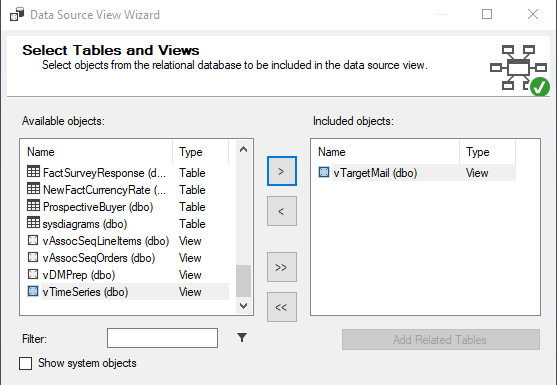
Analysis service will be used to store the data mining models and analysis service only use windows authentication. Any of the four options can be used to provide the necessary connection.

With this, you have configured the data source to the project and of course, you can modify them later. Also, you can create multiple sources for a project.

## Data Source View

Next step is to select a data source view. The data source view is a subset of the tables or views. Since you might not require all the tables and views for the project, from the data source view, you can choose the needed objects.

There should be one selected data source for a given data source view. Though you can create multiple data sources, you can attach only data source for one data source view. Also, if you haven’t created a data source before, from the **Data Source View** wizard, you can create the data source.



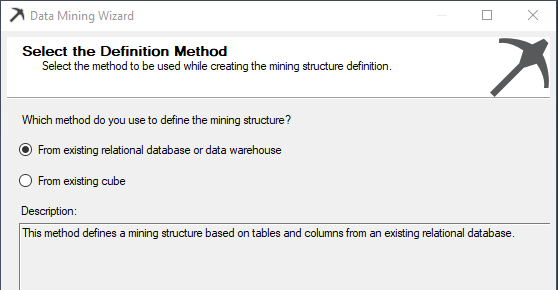
In the data source view, you can select the objects you need from the available objects. You can filter the objects. If you have selected tables that have foreign key constraints, you can automatically select the related tables by selecting **Add Related Table.**

Similar to the data sources, you can create multiple data source views. However, you can have only one data source for a given data source view.

## Data Mining

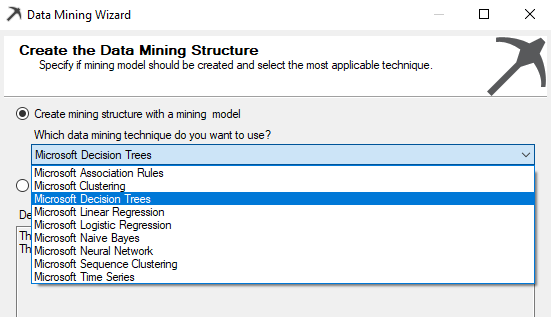
Now you have done the basic setup to start the data mining project. Next is to create a data mining project. Similar to other configurations, data mining structure creation will be done with the help of a wizard.

The following will be the wizard for the data mining model creation.



In the above dialog box, there are two types of sources, whether it is from a relational database or an OLAP cube.

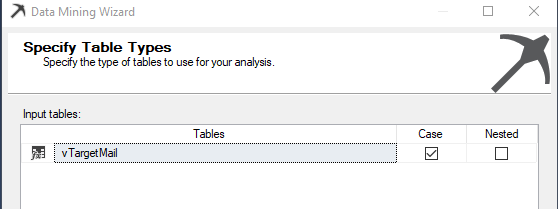
Next, the technique or algorithm is selected.



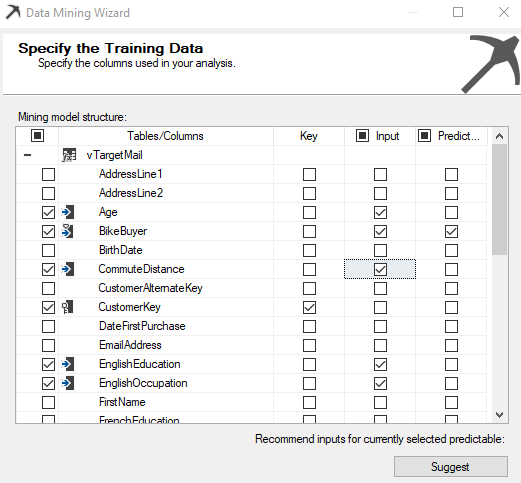
Nine data mining algorithms are supported in the SQL Server which is the most popular algorithm. However, you would have noticed that there is a **Microsoft** prefix for all the algorithms which means that there can be slight deviations or additions to the well-known algorithms.

The next correct data source view should be selected from which you have created before.

Next is to choose the Case and Nested options. The case table is the table that contains entities to analyze and the nested table is the table that contains additional information about the case.



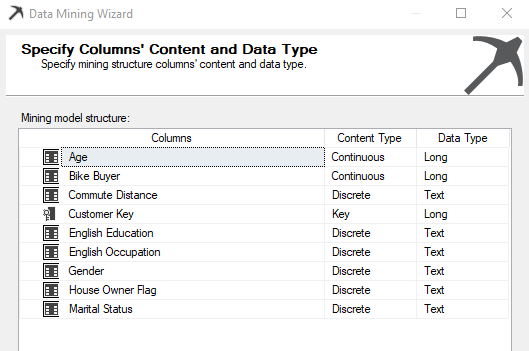
Sometimes, you do not need all the attributes for the purposes. For example, customer address attributes do not make any sense to appear as an impacted attribute for the final decision. From the following screen, you can select only the attributes that you would think will make an impact.



In the above screenshot, Customer Key is the key column while Age, Bike Buyer, Commute Distance, Education, and Occupation are the inputs to predict whether a Bike Buyer or not.

If you don’t have any clue about your data set, you can use the Suggest button and get some idea about the key impacted attributes.

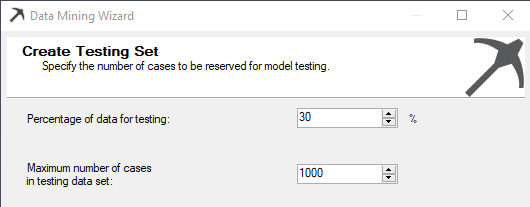
In the next screen, Content-Type and Data types are listed and users can modify them if needed.



Some algorithms will not be working with some content type. For example, Microsoft Naïve Bayes will not be possible if you have selected a Continuous content type.

There are five types of Content-Type such as Continuous, Cyclical, Discrete, Discretized and Ordered. Discrete data can take only integer values whereas continuous data can take any value. For instance, the number of patients treated by a hospital each year is discrete whereas hospital income is continuous. Discretization means transferring continuous variables into discrete variables.

Whenever the data mining model is created, it is always important to test your model with a valid data set. Train data set will be used to train the model while the test data set is used to test the built model. The following screen will show how to configure test and train data set.



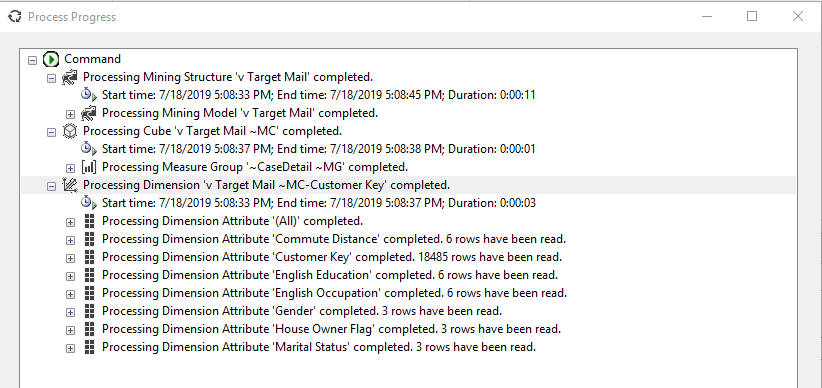
Typically 70/30 is the split for train/test data set. Input data will be randomly split into two sets, a training set and a testing set, based on the percentage of data for testing and a maximum number of cases in testing data set you to provide. The training set is used to create the mining model. The testing set is used to check model accuracy.

**The percentage of data for testing** specifies percentages of cases reserved for the testing set. The**Maximum number of cases in testing data set** limits the total number of cases in the testing set. If both values are specified, both limits are enforced.

## Model Processing

After the data Mining model is created, it has to be processed. We will discuss the processing option in a separate article. However, for the moment let us say, processing the data mining model will deploy the data mining model to the SQL Server Analysis Service so that end users can consume the data mining model.

Once the model is processed, it will be shown as follows.



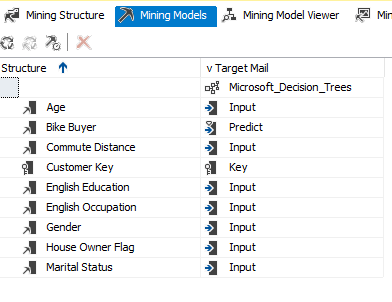
Once the model is processed it is ready, next is to consume the data mining model.

## Mining Model Views

After the model is created, the next is to visualize the model. There are five tabs to views these models. There are Mining Structure, Mining Models, Mining Model Views, Mining Accuracy chart, and Mining Model Prediction.

Most of the tabs are relevant to the Data Mining algorithms that were selected before. Therefore, that discussion will be saved for incoming articles. Mining Model tab is common for all the other mining algorithms.

There can be multiple mining models in this tab.



As we know, prediction can go wrong. However, we need to know what is the accuracy level of the data mining model that we provide. Accuracy chart will provide you multiple options to measure the accuracy of the model that you built, which will be discussed in a separate article.

## What’s Next

I hope this article helped you gain some basic understanding of data mining. If you are willing to join the journey to learn data mining with SQL Server, setup the environment and get your hand dirty with this, stay tuned to explore Microsoft Naive Bayes algorithm in my [next article](https://www.sqlshack.com/naive-bayes-prediction-in-sql-server/) on SQLShack.

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/naive-bayes-prediction-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Naive Bayes Prediction in SQL Server

August 2, 2019 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

In this article, we will walk through Microsoft Naive Bayes algorithm in SQL Server.

In my previous article, [SQL data mining](https://www.sqlshack.com/introduction-to-sql-server-data-mining/), we discussed what data mining is and how to set up the data mining environment in SQL Server. We briefly said that there are several algorithms which you can select during the setting up of the data mining environment. As mentioned before, we are going to discuss the first algorithm which is Microsoft Naive Bayes in this article.

## What is Naive Bayes

[Bayes theory](https://en.wikipedia.org/wiki/Bayes'_theorem) was discovered by Rev. Thomas Bayes in 1763 with the example of a newborn child who is witnessing a sunrise for the first time. Let us look at a more modern example. Let us look at an example from the medical arena. The hospital that sees patients for dengue has started to collect data. So they have found that there are several symptoms of dengue patients. High fever, body ache, vomiting, and headaches are few those symptoms and we will limit the symptoms for only those for the purpose of discussion. Now, if you are statistician, you would say, out of the dengue patients, 75% had a high fever and 80% had vomiting. Though these numbers give you some idea, what we need to know as a policy decision-maker, what is the percentage that the patient will have dengue given that he has a fever? Naive Bayes theory can convert to unknown from what you know.

Following is the equation for the famous Bayes theory.

P ( A / B) = P ( B / A) \* P (A) / P(B)

P ( A / B) = The likelihood of event A occurring, given that B is true.

P ( B / A ) = The likelihood of event B occurring given that A is true.

P ( A) and P(B) are the probabilities of events A and B independently of each other.

Let us look at this from the dengue and fever example.

The likelihood of having dengue given that the patient is having fever = the likelihood of the patient is having fever given that the patient is suffering from dengue \* Probability of patient having dengue / Probability of patient having a fever.

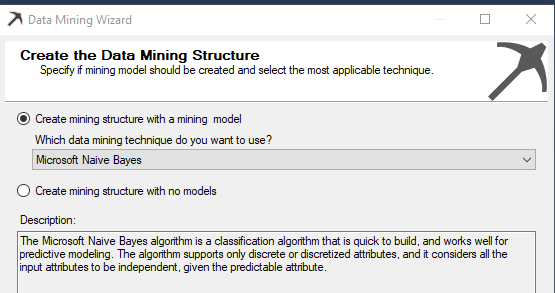
In the above example, to decide whether the patient is suffering from dengue, there are more than one parameter such as vomiting, body ache, headache etc. So you need all these parameters to be analyzed in order to predict whether the patient has dengue.

## Microsoft Naive Bayes

Microsoft Naive Bayes is a classification supervised learning. This data set can be bi-class which means it has only two classes. Whether the patient is suffering from dengue or not or whether your customers are bike buyers or not, are an example of the bi-class data set. There can be multi-class data set as well.

Let us take the example which we discussed in the previous article, AdventureWorks bike buyer example. In this example, we will use **vTargetMail** database view in the **AdventureWorksDW** database.

During the data mining algorithm wizard, the Microsoft Naive Bayes algorithm should be selected as shown in the below image.



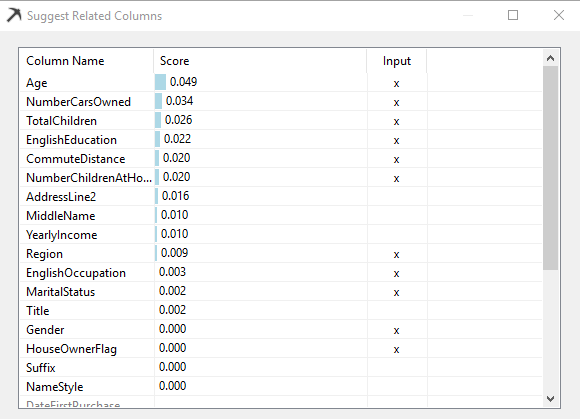
Next is to select relevant attributes which we will think will be impacted on bike buying. As we said last time, there are three types of attributes, Key, Predictable and Input.

So, we are going to predict, whether the customer will buy a bike or not. Therefore the predictable column is **Bike Buyer.**Each row should be uniquely identified using a Key and Customer Key is the key in this example.

Next is to select inputs which will impact customer buying.

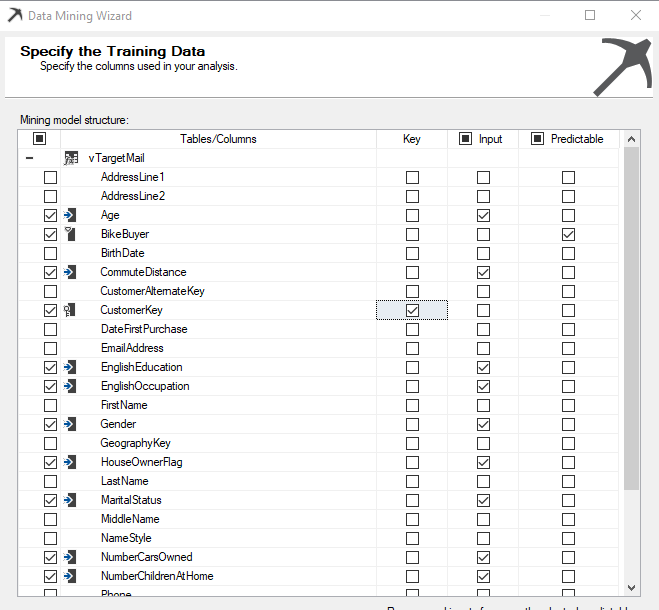
After choosing the Key and Predictable attributes and before the selection of input ttributes. 


In most cases, you can choose the necessary columns with your experience and sense. However, if you are unable to choose, there is a **Suggest** button which will give you suggestions as shown in the below image.

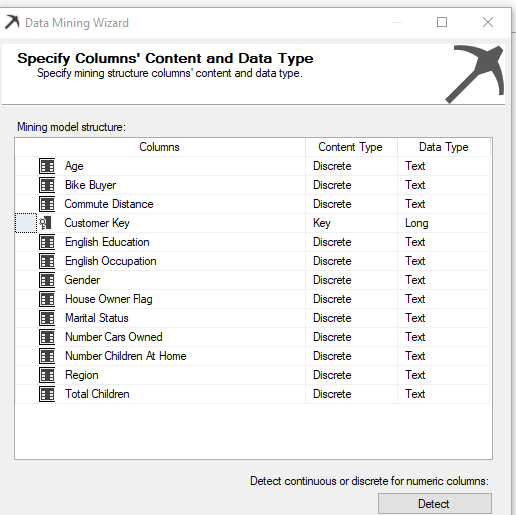


By analyzing data, a suggestion will provide you attributes which the system thinks will have a higher impact. You can choose the impact columns directly from this screen itself.

After completing the selection of input columns, the following screen will be displayed as shown in the below image.



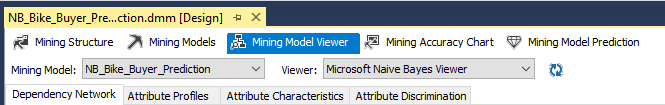
After the selection of attribute is completed, the next screen is to choose **Content-Type** and **Data Type**. You would notice that Age, Bike Buyer, Number Cars Owned, Number Children At Home and Total Children will have Discretized Content-Type and Data Type is **Long**. Discretized Content-Type is not compatible with Microsoft Naive Bayes. Therefore, those attributes have to be changed to Discrete, and Data type should be selected as text. The screen will appear like the following after you correctly configure the content type.



After this, next is to provide the test and train data set parameters and default values are used here. Next is to process the mining model so that it can be consumed by the end-users.

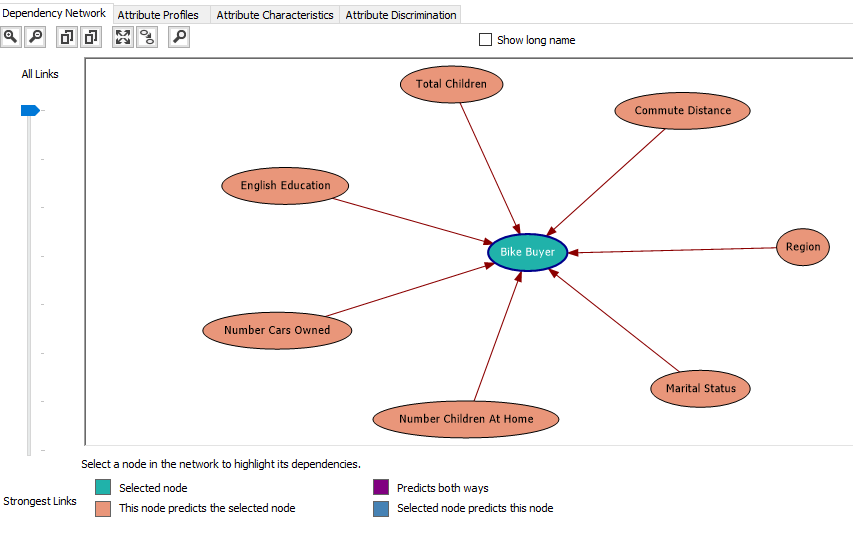
## Mining Model Viewer

After the mining model is built, next step is to analyze the model. The mining model Viewer is next to the model viewer. In the model viewer, there are four options namely, Dependency Network, Attribute Profiles, Attribute Characteristic and Attribute Discrimination as shown in the below image.



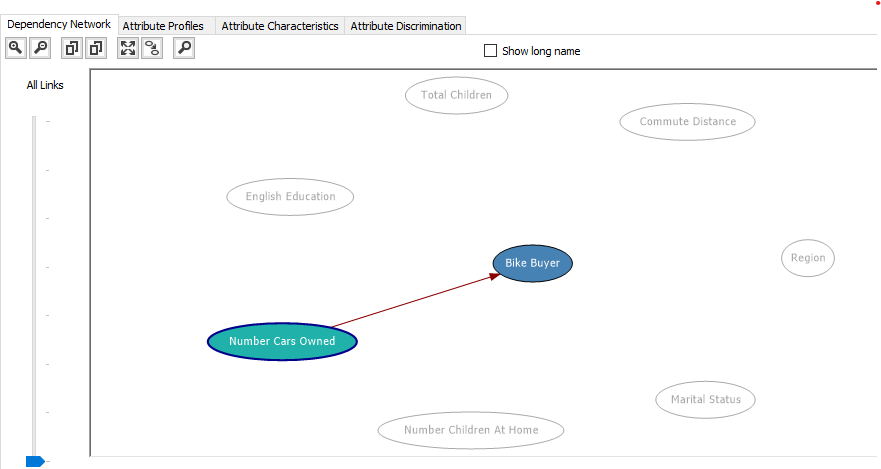
### Dependency Network

The dependency network is a clear indicator of which attributes make a high or low impact towards the predictable attribute. In the following image, it elaborates what are the attributes which have a higher impact to decide **Bike Buyer**.



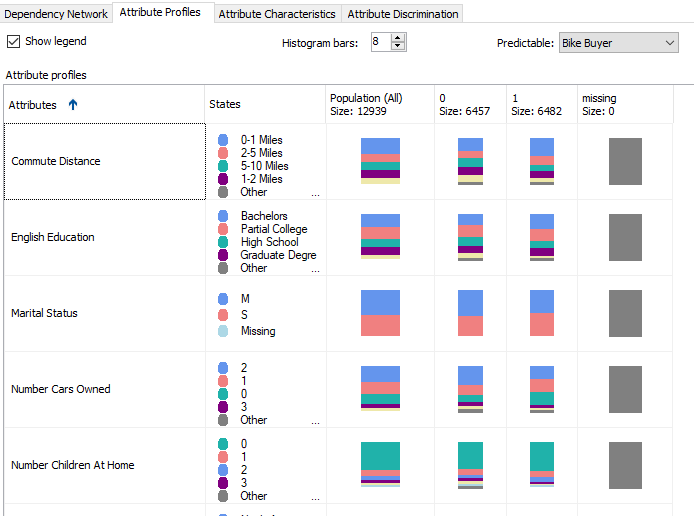
The above image indicates that Total Children, Commute Distance, Region, Marital Status, Number Children at Home, Number Cars Owned and English Education out of the other selected attributes. Out of these attributes, different attributes have different weightages of impacts. This can be identified by moving down the **All Links** to **Strongest Links** Slide bar which is at the left-hand side of the screen.

If the slide is at the very lowest place in the slide bar, you will see the strongest link as shown in the below image.

[](https://www.sqlshack.com/wp-content/uploads/2019/08/strongest-linked-attribute-for-bike-buyer-.png)

### Attribute Profile

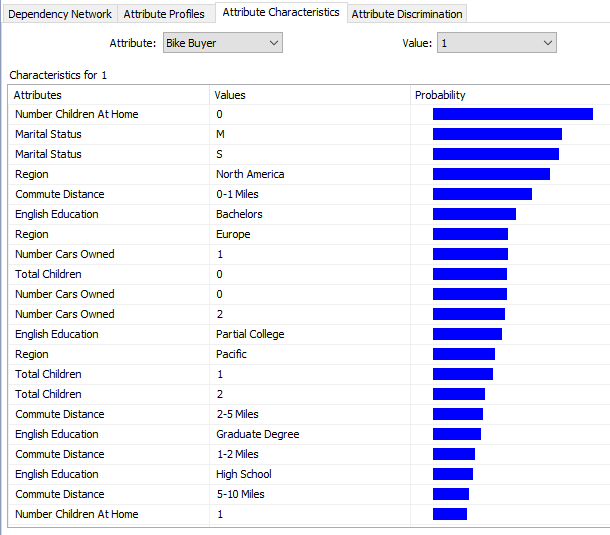
In this classification problem, we will have two classes. Buying Bike (1) or Not Buying Bikes (0). Attribute Profile shows how each category has different combinations.



This view can provide you a better view of understanding your data.

### Attribute Characteristics

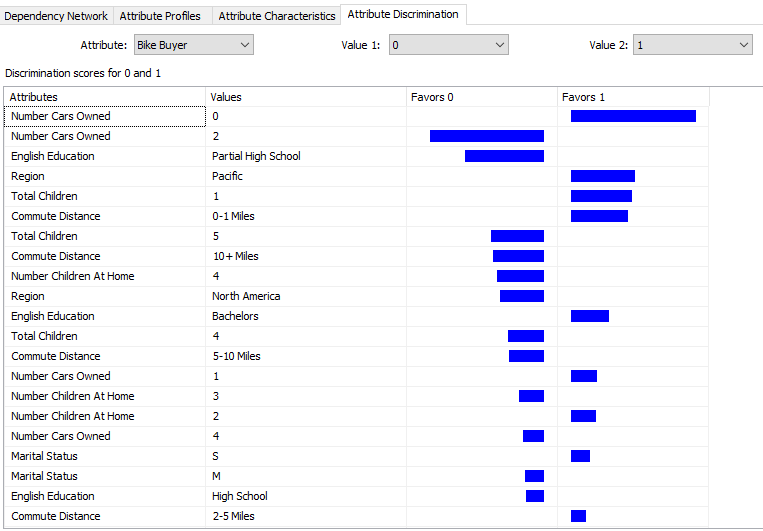
As a user, it would be nice to know what feature will make people buy a car so that they can target the customers who have those attributes.



The above image indicates that customers who do not have children at home tend to buy bikes. Also, North America Region and Customers who commute 0-1 Miles will tend to buy bikes. This information is handy for the marketing department so that they can specifically target customers with these parameters.

### Attribute Discrimination

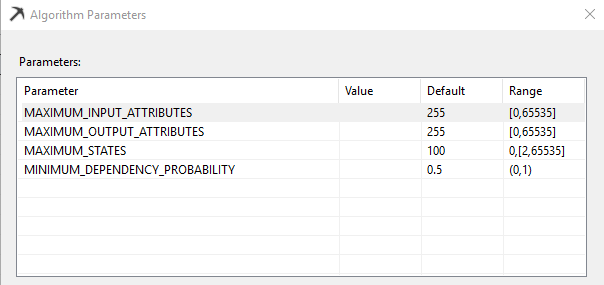
This tab gives you an answer to the question,” What is the difference between a bike buyer and not bike buyer”.



We will leave the prediction and accuracy calculation in a separate article as it is a discussion for all the algorithms.

## Model Parameters

Depending on the environment, there are model parameters for each algorithm. By using these parameters, you can fine-tune the model. In the case of Microsoft Naive Bayes, there are four parameters which can be modified from the **Mining Models** and by selecting **Set Algorithm Parameters** from the **Mining Model**option in the main menu.



### Maximum\_Input\_Attributes

Specifies the maximum number of input attributes that the algorithm can handle before invoking feature selection. Setting this value to 0 disables feature selection for input attributes. This is an Enterprise edition feature. The default value is 255.

### Maximum\_Output\_Attributes

Specifies the maximum number of output attributes that the algorithm can handle before invoking feature selection. Setting this value to 0 disables feature selection for output attributes. This is an Enterprise edition feature. The default value is 255.

### Maximum\_States

Specifies the maximum number of attribute states that the algorithm supports. If the number of states that an attribute has is greater than the maximum number of states, the algorithm uses the attribute’s most popular states and treats the remaining states as missing. This is an Enterprise edition feature. The default value is 100.

### Minimum\_Dependency\_Probability

Specifies the minimum dependency probability between input and output attributes. This value is used to limit the size of the content generated by the algorithm. This property can be set from 0 to 1. Increasing this value reduces the number of attributes in the model. If this has a higher values, you might not see some attributes in the Dependency Network. The default value is 0.5.

## Limitations of the Microsoft Naive Bayes Algorithm

The major issue is Microsoft Naive Bayes cannot handle continuous data. In the above example, we had to drop the yearly income though it is a very important attribute, simply because that attribute is a continuous variable.

Naive Bayes Algorithm has a basic assumption that input attributes are independent of each other. If you look at the example, we know that Occupation and Qualification have a link between them. However, this algorithm will not be able to understand the relation between inputs.

## Summary

Microsoft Naive Bayes algorithm is a basic algorithm when you start into a data mining project. It will provide you a basic understanding about your data.

In the next article, we will discuss Microsoft Decision Trees algorithms.

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/microsoft-decision-trees-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Microsoft Decision Trees in SQL Server

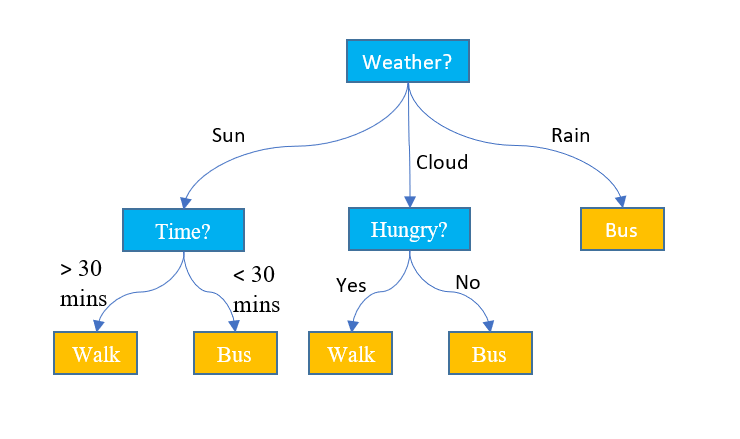
September 12, 2019 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

Decision trees, one of the very popular data mining algorithm which is the next topic in our Data Mining series. In the previous article [Introduction to SQL Server Data Mining](https://www.sqlshack.com/introduction-to-sql-server-data-mining/), we discussed what data mining is and how to set up the data mining environment in SQL Server. Then in the next article, Microsoft Naïve Bayes algorithm was discussed. In this Article, Microsoft Decision Trees are discussed with examples. The Microsoft Decision Trees algorithm is a classification and regression algorithm that works well for predictive modeling. The algorithm supports the prediction of both discrete and continuous attributes.

## What is Decision Trees

Decision Trees are one of the most common data mining algorithm. When you make a decision, you always tend to divide your problem. Let us say you want to go to one place from another place. To decide what time you should leave, you will have a lot of parameters in your mind. Depending on the day (weekend or weekday), type of mode of transport, time of traveling, and if there any special events, type of weather will decide the time. So when you decide on the time, there can combinations. For example, if it is raining, on a weekday, at a peak time, traveling time would be different for different combinations. All these combinations can be visualized into a tree format.

Following is an example of a Decision Tree, which discusses the mode of transport depending on another requirement.



Source: <https://www.displayr.com/what-is-a-decision-tree/>

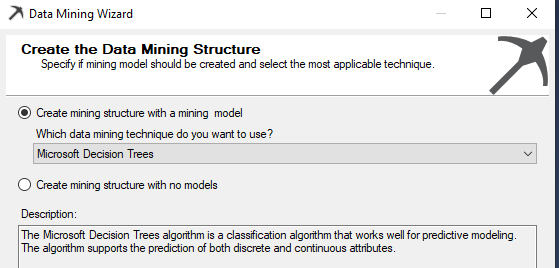
As you can see from the above figure, decision trees are extremely easy to understand. That is the most common reason why the decision trees are popular among most of the users.

## Microsoft Decision Trees in SSAS

In SQL Server, using data sets model can be built with Decision Tree algorithms and then predictions can be done from the built decision tree.

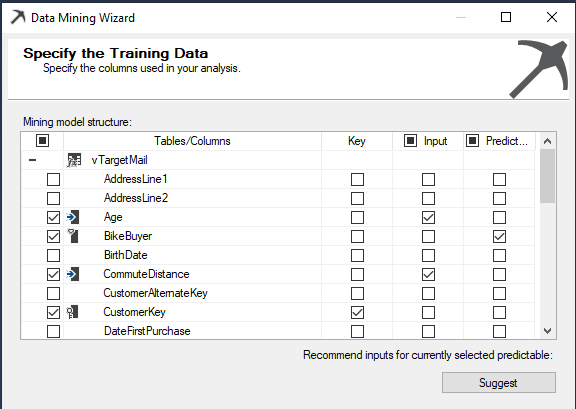
We will be using the same dataset vTargetMail view in the AdventureWorksDW database. As we discussed in the previous article, create a SSAS project in the Visual Studio. Then create a data source which will point to the AdventureworksDW database and DataSourceView in which vTargetMail is selected.

Next, create a mining structure and select the Microsoft Decision Trees as shown in the below figure.



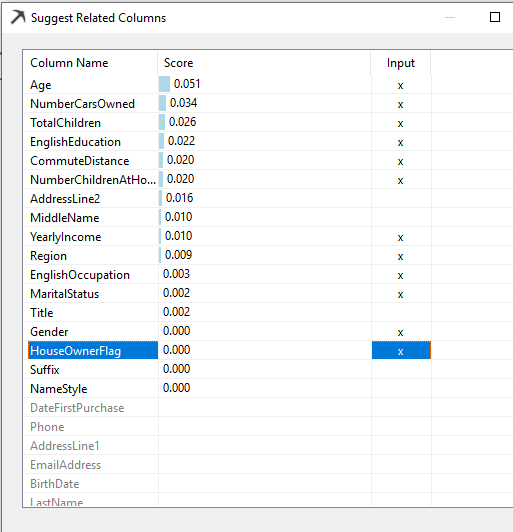
In the wizard, **vMailTarget** was selected as the case table. Next, is to select input and predicted columns. Since we are looking at predicting the bike buyer, **BikeBuyer** attribute is the predicted attribute.

**Age**, **Commute Distance**, **Education**, **Occupation**, **Gender**, **House Owner Flag**, **Marital Status**, **Number Cars Owned**, **Number** **Children At Home**, **Region**, **Total Children**, and **Yearly Income** are selected as input attributes as shown in the below figure.



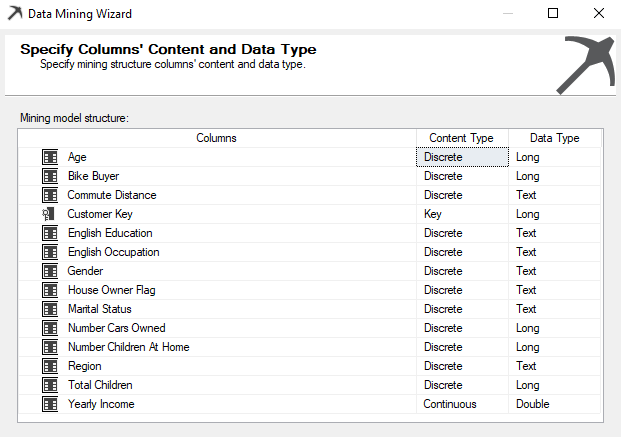
It should be noted that attributes such as **AddressLine1**, **AddressLine2**, **BirthDate**, **DateFirstPurchase** attribute are not selected as it is obvious that those attributes will not make any impact on a customer becoming a bike buyer.

If you are not sure about what input attributes to be selected, by clicking From the Suggest button, you can get what are most impacted attributes as shown in the below figure. This has to be done after choosing the predictable attribute. Also, you can choose those attributes from this screen itself so that it will be reflected in the previous screen.



You will observe that there are a few attributes that cannot be selected from the suggested dialog. Still, you can choose those from the previous screen.

Next is to select content type as shown in the following image.



Apart from the Yearly income, all the other continuous data types were changed to Discrete. Especially, the predictable attribute was changed to Discrete Content-Type. When this is Discrete, this will become a classification problem and when the prediction attribute is continued it will become a Regression problem.

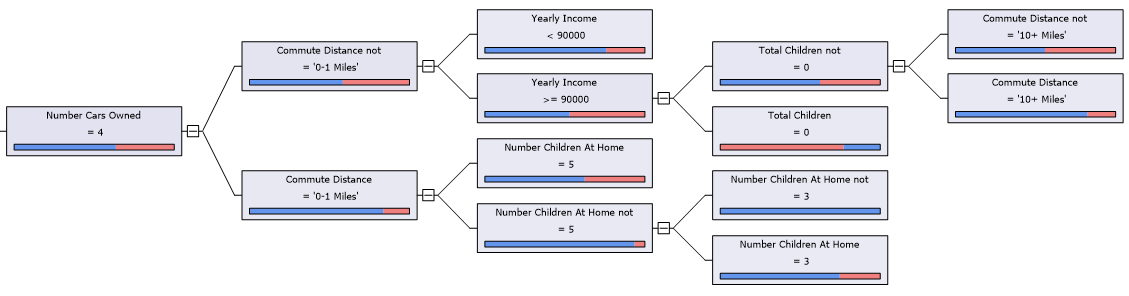
## Decision Trees for Classification Problem

After choosing the correct content types, next is to provide the train and test data set parameters. After naming the model, next is to process and view the results.

Following is the Full graph view of the decision. Though the graph is not fully readable in this article, this is to understand how different attributes are contributed to the tree view.

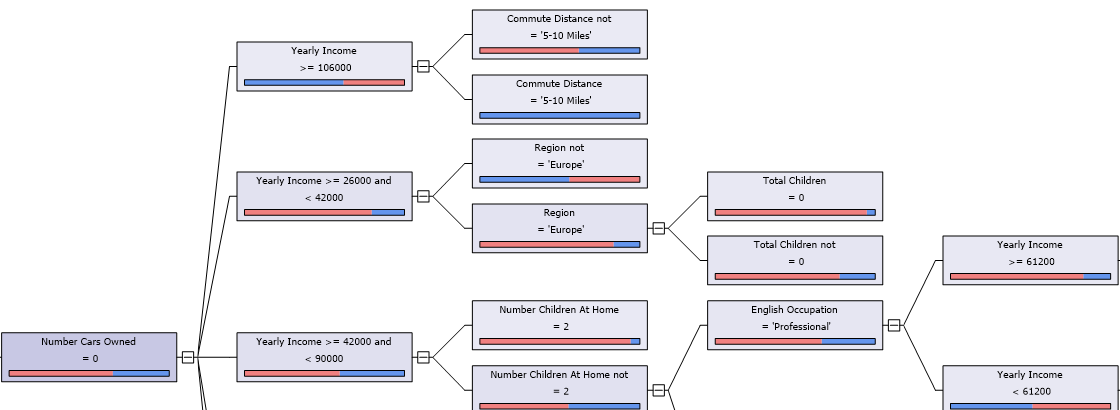


Let us analyze the graph in more detail.

[](https://www.sqlshack.com/wp-content/uploads/2019/09/one-tree-node-of-deicsion-tree-.png)

Data set is divided from the **Number Cars Owned** attribute. If a person owned 4 cars, the next parameter is Commute Distance, Yearly Income, and Total Children, etc.

Let us look at another tree node.

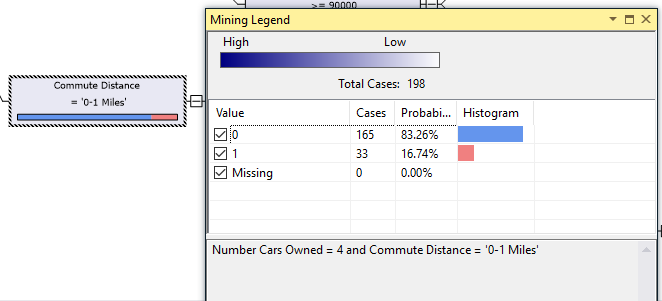
[](https://www.sqlshack.com/wp-content/uploads/2019/09/second-node-of-the-decision-tree-.png)

In this node, the second split is by the year income and split will be different depending on the year income.

Also, different nodes have different level. In the node of **Year Income > 106,000** and what only matters are the Commute Distance and other parameters can be ignored.

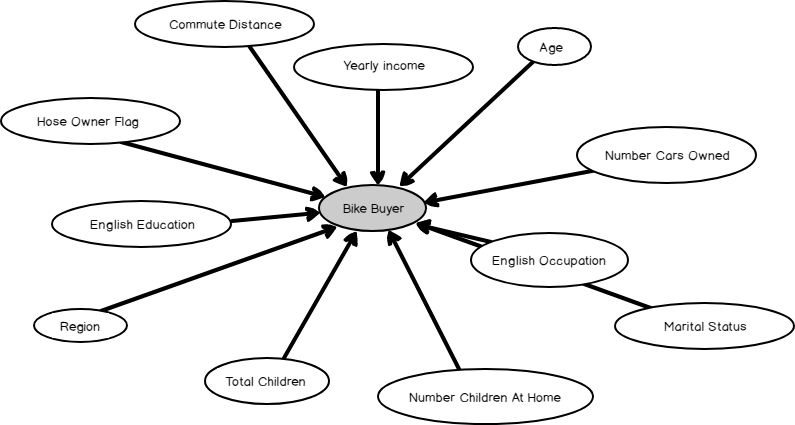
Next look at one node.

The selected node is **Number Cars Owned = 4 and Commute Distance = ‘0-1 Miles’**and the legend of the node is also shown below.



In this node, it is more probable that customers will not buy a bike as it has a probability of more than 83%. By looking at the node, from the colors you can get an idea of the distribution of the dataset with the predictable attribute.

Since there are many input attributes, it is important to understand the most weightage attributes. This can viewed from the **Dependency Network**.

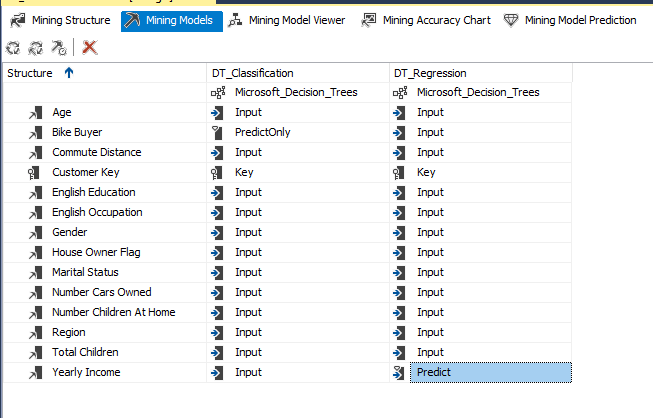


By adjusting the Strongest Links, you can find out what are the most dependent attribute. You will find that the **Number Cars Owned** is the most important attribute which is why it is used as the first split in the decision tree.

Advantage of decision trees over the Naïve Bayes for classification is that Decision Trees can incorporate continuous attributes such as yearly income.

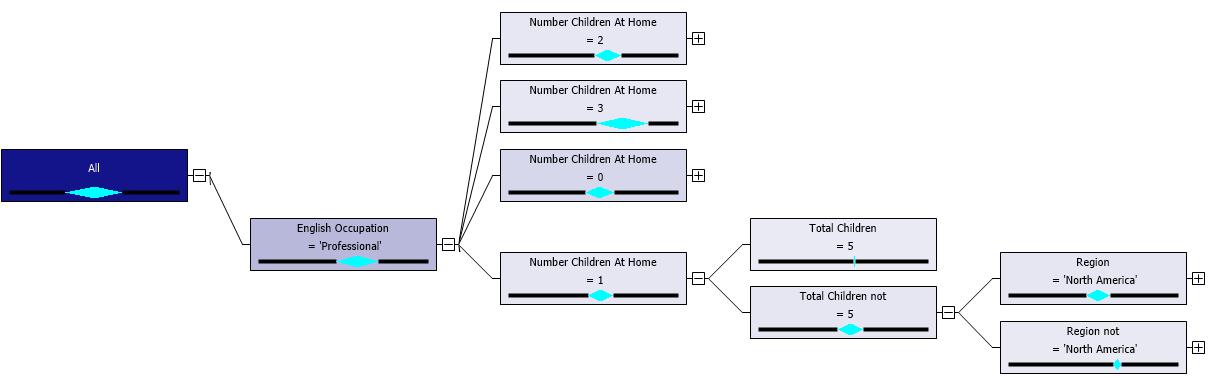
## Decision Trees for Regression Problem

Let us see how Decision Trees can be used as a solution to a regression problem. There is no need to add another data mining structure whereas you can add another mining model from the **Mining Model**tab and change the input and prediction attributes as shown in the below image.

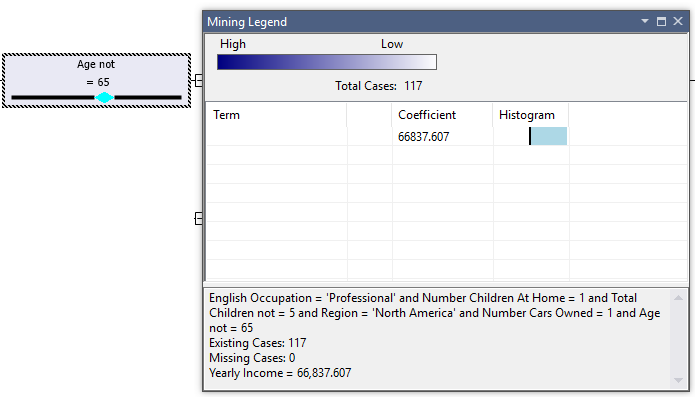


Next is to process the newly created model.

Following is the part of the decision tree for the regression problem.

[](https://www.sqlshack.com/wp-content/uploads/2019/09/solving-regression-problem-from-decision-tree-tech.png)

Let us look at one node and its legend.



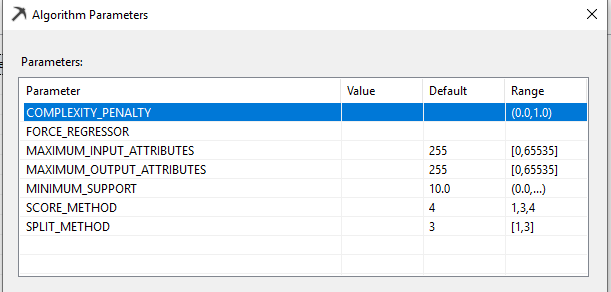
During the Classification problem, by looking at the node, you can get an idea about that node. In the regression problem also, by looking at the node, you can get an idea about the properties of that node. If the width of the diamond is high, this means that there is a high deviation in the predictable variable in the node.

If there are continuous attributes in the input, this yearly income will be a formula. Since in the selected example, all the attributes are discrete, the only value is assigned for the node. In this node the value is 66,837.607. This means a client with **English Occupation = ‘Professional’ and Number Children At Home = 1 and Total Children not = 5 and Region = ‘North America’ and Number Cars Owned = 1 and Age not = 65**will have an annual income of 66,837.607.

## Model Parameters

As discussed in the previous article, model parameters can be changed so that the model can be further improved to better results.

Model attributes, default value, range are shown in the Model Parameter dialog box.



### Complexity Penalty

Decreasing the value increases the likelihood of a split while increasing the value decreases the likelihood. The default value is based on the number of attributes for a given model. If there is 1 to 9 attributes default is 0.5 and it will be 0.9 if there are 10 to 99 attributes, and the default is 0.99 if there are 100 or more attributes.

### Force Regressor

This parameter forces the algorithm to use the indicated columns as regressors in the regression formula regardless of their importance as calculated by the algorithm. This parameter is only used for regression trees. This parameter is only available in the Enterprise Edition of the SQL Server.

### Maximum Input Attributes

This parameter specifies the maximum number of input attributes that the algorithm can handle before invoking feature selection. Setting this value to 0 disables feature selection for input attributes. This parameter is only available in the Enterprise Edition of the SQL Server.

### Maximum Output Attributes

This parameter the maximum number of output attributes that the algorithm can handle before invoking feature selection. Setting this value to 0 disables feature selection for output attributes. This parameter is only available in the Enterprise Edition of the SQL Server.

### Minimum Support

This parameter specifies the minimum number of cases that a leaf node must contain. Setting this value to less than 1 specifies the minimum number of cases as a percentage of the total cases. Setting this value to a whole number greater than 1 specifies the minimum number of cases as the absolute number of cases.

### Score Method

This Specifies the method used to calculate the split score. The available methods are Entropy (1), Bayesian with K2 Prior (3), or Bayesian Dirichlet Equivalent with Uniform prior (4).

### Split Method

Specifies the method used to split the node. The available methods are Binary (1), Complete (2), or Both (3).

## Tabl

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/microsoft-time-series-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Microsoft Time Series in SQL Server

December 12, 2019 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

The next topic in our Data Mining series is the popular algorithm, Time Series. Since business users want to forecast values for areas like production, sales, profit, etc., with a time parameter, Time Series has become an important data mining tool. It essentially allows analyzing the past behavior of a variable over time in order to predict its future behavior.

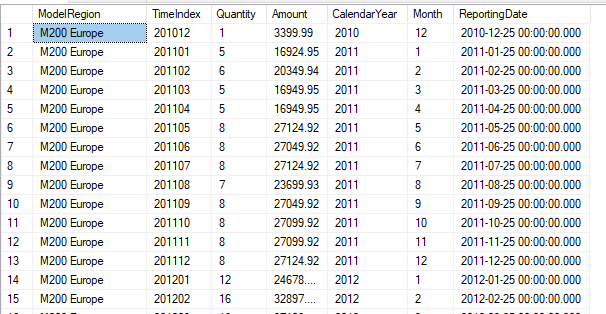
## Components in Time Series

A time series consists of five components:

* **Trend:** Trend is the movement of the values. Typically, a given series will have an upward or downward trend
* **Cyclical**: Upward or downward repetitive movement of the values over a longer period of time
* **Seasonal:** Similar to cyclical, but there can be multiple movements of the values over shorter periods of time, such as hourly, daily, weekly, monthly, etc.
* **Random:** There can be movements in the data values which are totally random but will have an impact on the time series trend. A time-series analysis should identify these exceptions, and account for them in predictions
* **Cross:** Other factors may affect the trend of a time series. For example, sales of item A may be dependent on seasonal factors, but may also be affected by the sales of item B. If we take the production of a crop as an example, it will be dependent on rainfall or temperature trends

## Times Series in SQL Server

To demonstrate time series analysis using SQL Server, we will use the **vTimeSeries** view in the **AdventureWorksDW2017** sample database. Here is the sample data set:



We will use only the first four columns, which are ModelRegion, TimeIndex, Quantity and Amount.

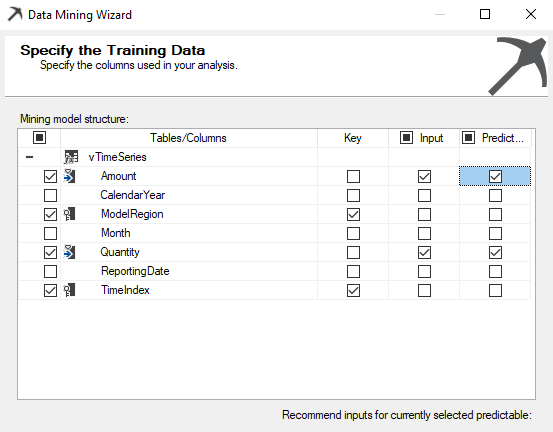
As discussed in the [first article](https://www.sqlshack.com/introduction-to-sql-server-data-mining/), create a sample analysis service project with Visual Studio or SQL Server Data Tools (SSDT). Then, create a Data Source connection to the **AdventureWorksDW2017** database and add the **vTimeSeries** view to the Data Source View.

Next, create a Mining Structure with the Microsoft Time Series data mining technique. Select it from the available list of data mining techniques.

Then, select the necessary attributes to create the Time Series model. The Time Series model needs two compulsory parameters and one optional parameter. It requires a single time column which will be the key for the model. This has to be a column with the same intervals. For example, if you have monthly data, the entire data set should be monthly and should not contain different intervals of data. Another compulsory column is the column that you want to predict from the Time Series technique. This should be a continuous and numerical variable such as sales, temperature, quantity, etc.

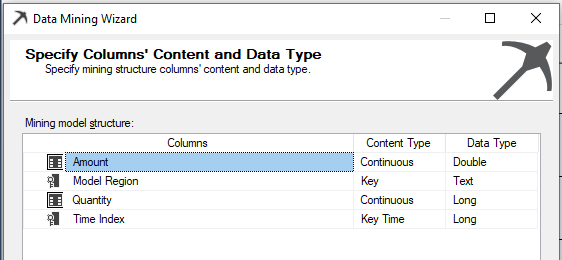
Optionally, you can define multiple series in a one-time series. For example, in a sales time series, you might want to analyze the trend by region. Therefore, the region will be an additional and optional key.

Here is the configuration for the parameters in the Time Series:



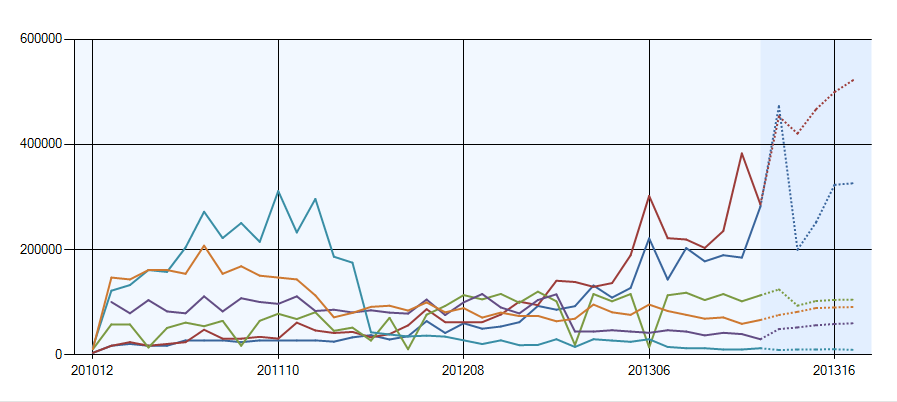
In the above configuration, both **Quantity** and **Amount** are configured as the input and prediction columns. As mentioned before, these two columns should be numerical and continuous variables. **TimeIndex** is the key column that is used to identify the time component of the data set. **ModelRegion** is the optional series column from which users can predict Region and Product Model quantities and sales amounts.

The following image shows data types, in case the user wants to change them:



However, in this example, you can leave the default data types as it is. With that, time series model creation is done, and the data mining model needs to be processed.

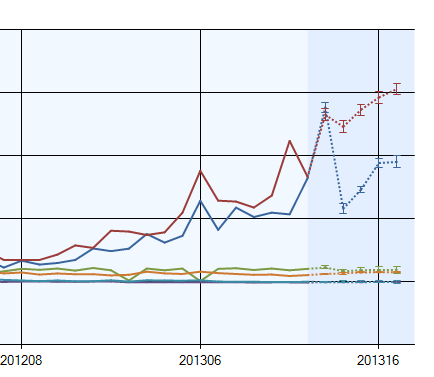
Let us view the time series trend with the predictions:



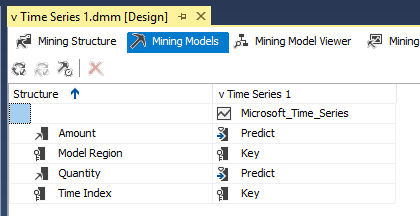
In the graph, predictions are shown by dotted lines. One important thing to remember is that the time series does not understand time index cycles. It just uses the number that follows the previous one. For example, after 201212, the next number will be 201213, not the calendar designation 201301.

From the configuration available at the top of the Mining Model Viewer, the user has the option to see more information.

By setting the **Show Deviation** to on, you can view the deviations for the predicted values to judge the accuracy of your model. A lower deviation means higher accuracy. Deviations are shown in the following image:



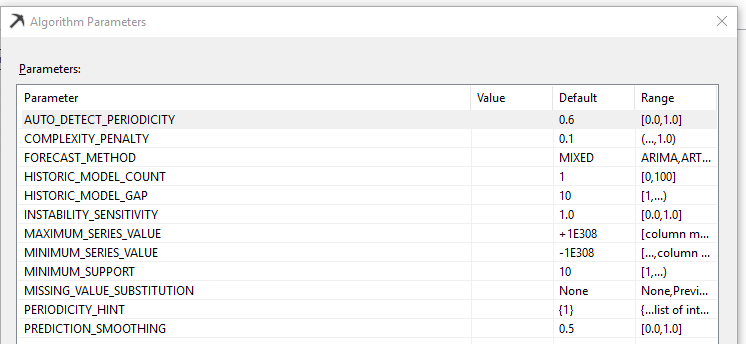
In the **Mining Models** tab, you can set the predictive attributes for **Predict** or **PredictOnly** as shown below:



**Predict** means the attribute is used to predict, and predicted value is used to predict the next values, whereas the **PredictOnly** parameter means the attribute is used only for prediction, and predicted value is not used for the next predictions.

## Model parameters

Model parameters are used to change the parameters to suit the data environment. Although the default parameters provide the best results, users have the option to change them accordingly:



### AUTO\_DETECT\_PERIODICTY

This parameter specifies a value between 0 and 1 used to detect periodicity for the time series. By setting it to 1, the time series algorithm will automatically detect the periodicity. However, this can cause a performance issue during model building. Setting the value to 0 indicates that the algorithm will detect only the strong periodic data.

### FORECAST\_METHOD

FORECAST\_METHOD specifies which forecasting algorithm is used. If the MIXED method is chosen, it creates models for both ARTXP and ARIMA time series algorithms, and their results will be combined during the prediction phase. In the standard edition of SQL Server, the models are combined using an automatic ratio that favors ARTXP for near-term and ARIMA for long-term prediction. In higher editions such as Enterprise edition, the models are combined and weighted according to the value set for PREDICTION\_SMOOTHING. When the FORECAST\_METHOD is set to either ARTXP or ARIMA, the value for the PREDICTION\_SMOOTHING parameter is ignored.

### PERIODICITY\_HINT

This parameter provides a hint to the algorithm as to the periodicity of the data so the Time Series model performs better. Although the Time Series has the option of identifying the periodicity, it is better to provide the periodicity to the model. For example, if you have data with a periodicity of monthly, weekly and daily, you can configure PERODICITY\_HINT such as {12,7,1}.

### MISSING\_VALUE\_SUBSTITUTION

For Time Series, there cannot be gaps in the data set. However, due to various practical reasons, there may be instances where all data cannot be captured. SQL Server Time Series provides a method to substitute missing values. The default is None, which is suited for a data set without missing values. Mean values set the mean of the existing values to the missing values, whereas the Previous option sets the missing value with the previous values. Also, the user can set a constant value; this is not recommended.

1. [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/the-association-rule-mining-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Association Rule Mining in SQL Server

January 28, 2020 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

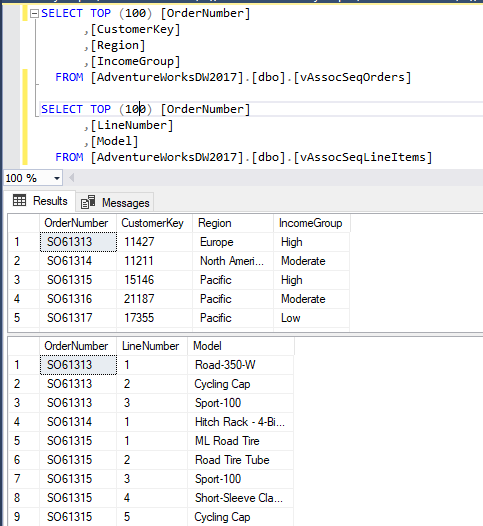
Association Rule Mining in SQL Server is the next article in our data mining article series in which we have discussed Naïve Bayes, Decision Trees, and Time Series until now. Association Rule Mining, also known as Market Basket Analysis, mainly because Association Mining is used to find out the items which are bought together by the customers during their shopping.

The most popular Association Rule Mining example that you will find is the story at the supermarket chain in the US. It is said that they have found out that the customers that are buying beer will buy nappies for their kids. After this finding, management has taken a decision to move the beer palette close to the nappy palette. By doing so, of course, they were able to increase sales. In addition to the money, they were able to make their customers happy. Also, customers buying time was reduced and so the congestion in the supermarket. This means that the Association Rule Mining is helpful to users in many ways.

Though Association Mining is always discussed with shopping, there are other possible areas of applications such as troubleshooting, medicine, and marketing, etc. In troubleshooting, by using Association Rule, you can diagnose what issues occur together. Also, in the domain of medicine, Association Rule will help to find out what types of the disease occur together. This means there are a lot of ways of utilizing the Association Rule in business.

## How to Use Association Rule Mining in SQL Server

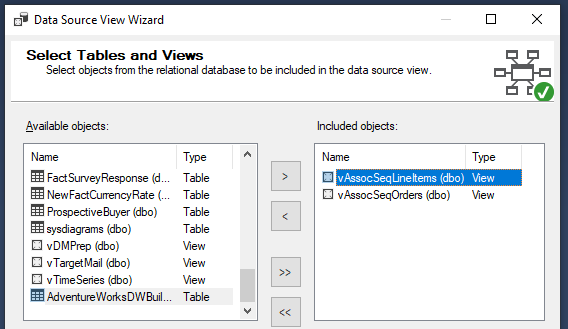
This time, there is a small change to the SSAS data mining project to what we have done before. This is due to the fact that we will be using a couple of views in the **AdventureWorksDW** database, whereas we were using only one view in all previous examples. Those two views are **vAssocSeqOrders** and **vAssocSeqLineItems**. **vAssocSeqOrders** view has orders while the **vAssocSeqLineItems** view has order lines for the orders. Following screenshot shows sample data set in those two views:



If you carefully look at the above screenshot, for the order number SO61313**,**there are three order lines in the second view.

Let us open SQL Server Data Tools (SSDT) and create an SSAS project to set up the Association Rule Mining. Then create a data source pointing to the **AdventureworksDW** database as we did in the previous articles.

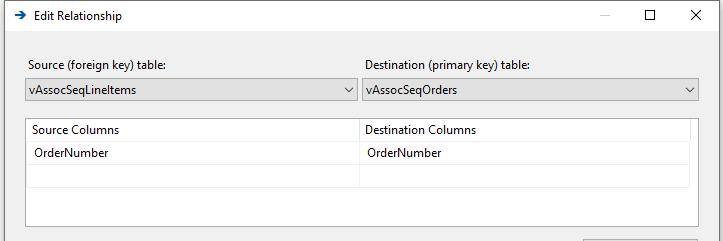
For the data source view, let us add the specified views as shown in the below screenshot:



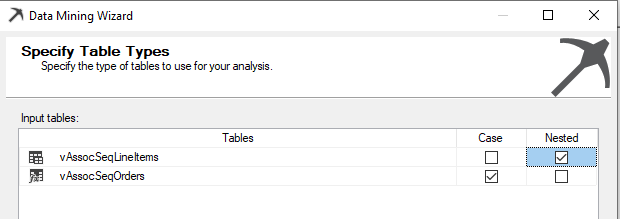
When these two views are added to the data source views, the relation between these two views is not added by default. This means you need to join those two views manually.



Verify the relationship by double-clicking the arrow sign. **Source** and **Destination** tables should be as shown along with the **OrderNumber** column. If it is in reverse, click the **Reverse** button to change it.

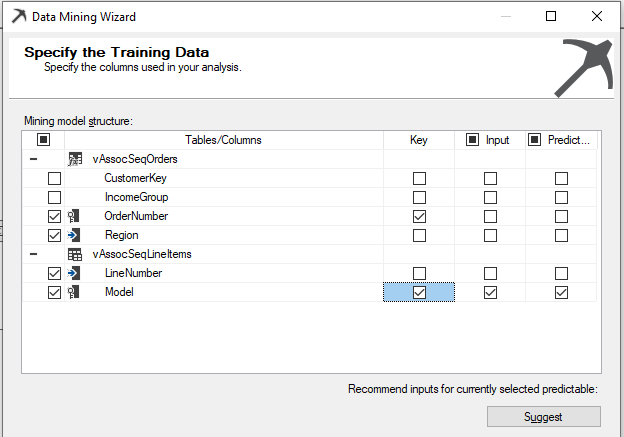


Next, we need to choose the case and the nested table. Up to now, we had to choose only the case table in our previous examples. However, in the Associating Rule Mining, since there are two views, we need to choose the case and the nested table, as shown in the below screenshot.



**vAssocSeqOrders** view was chosen as the Case table where the **vAssocSeqLineItems** is chosen as the Nested table.

The objective of the Association Rule Mining is to find out what models are selling together. Therefore, the product model will be the input as well as the predict attribute. **OrderNumber** and the **Model** are the keys. You can see those selections as shown in the below screenshot:

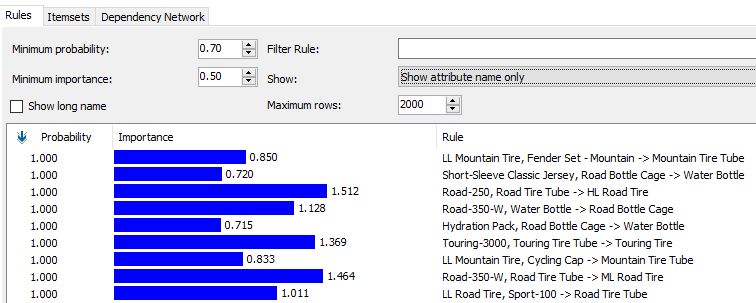


After the Association Rule configuration is completed, then the model can be processed. Then users can review the prediction model and perform the predictions.

## Mining Model Viewer

Let us view the data patterns from the Association Rule model, which was built before.

In the Mining Model viewer, there are three tabs to view the data patterns. In the **Rules** tab, it will show the rules that can be derived fro the Association Rule Mining model in the sample set.

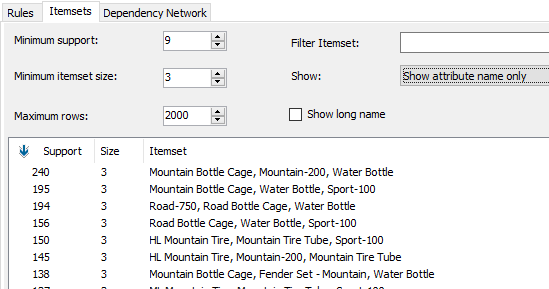


The main part of the Rule tab is the rule grid, which displays the all qualified Association Rule Minings along with their probabilities and their importance. The importance score will tell how useful the rule is. If the importance score is high, most likely greater than 1, the rule is of higher quality.

In the above screenshot, the customer who buys LL Mountain Tire and Fender Set-Mountain will buy Mountain Tire Tube. Probability 1 means that it will be true always. The importance of this rule is 0.850.

The Minimum importance level can be set for a model before processing so that the processing performance can be improved.

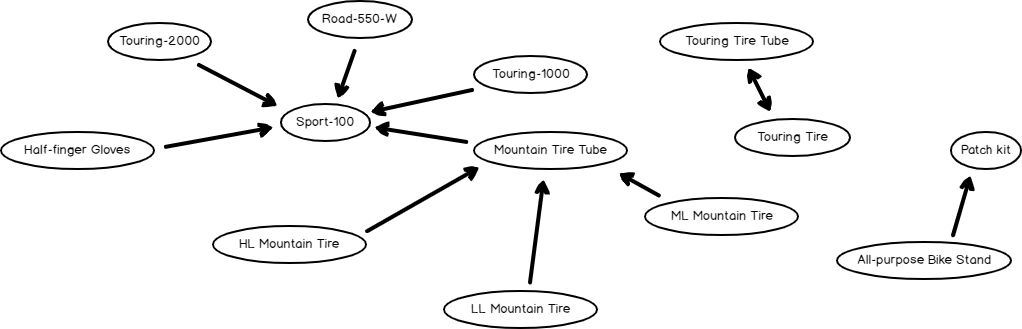
The next tab is **ItemSets** table, which will display the frequent itemsets discovered from the Association Rule algorithm.



Users can set the minimum support at this view as well as a model parameter so that the performance of the model process will be improved. Users can also select the minimum item set. In this example, minimum itemset size is set at 3, which means that three combinations of models are selected. Also, if needed, it is possible to put rules which consist of a specific item model by setting up filtering in **Filter Itemset**.

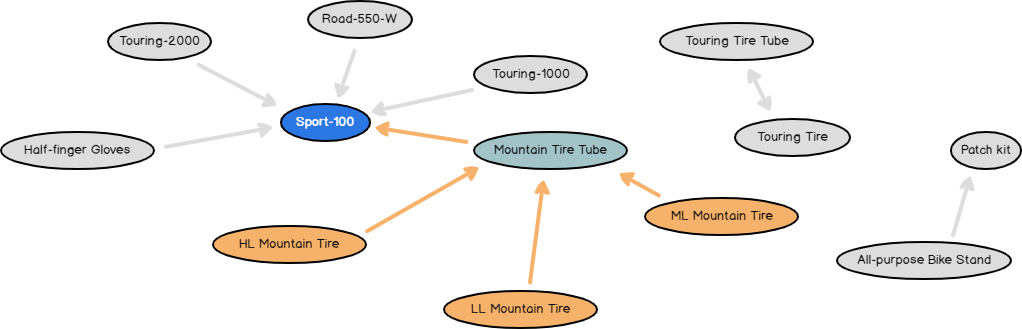
In the above data set, Mountain Bottle Cage, Mountain-200, and Water Bottle are in 240 orders so that you can identify the frequency of the itemset combinations.

The third tab in the model viewer, **Dependency Network,** graphically illustrates the relationship between the itemsets, as shown in the below screenshot.



If you analyze the above screenshot, you will see that Sport-100 will be bought by the customers who bought the Touring-1000, Touring-2000, Road-550-W and Half-Finger Gloves separately. There is another great finding that customers who buy Touring Tire will buy Touring Tire Tube and importantly, vice versa is also true. This is indicated by the arrow that points both ways.

If you click any node, those nodes will be highlighted with different colors, as shown in the below screenshot.

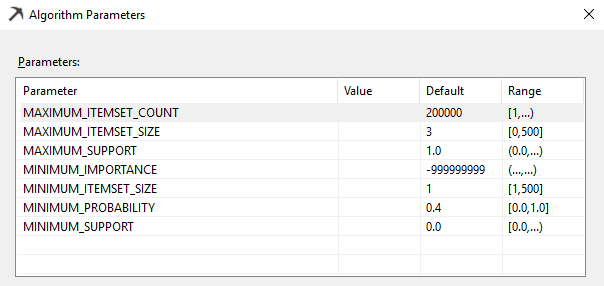


This will indicate different relations of the selected node with the other nodes in the relationship diagrams.

## Model Parameters

Model parameters can be set so that the Association Rule Mining can be configured to improve performance and accuracy.

This can be set from the following dialog box.



### MAXIMUM\_ITEMSET\_COUNT

The default value for this parameter is 200,000. This parameter defines how many predications will be generated.

### MAXIMUM\_ITEMSET\_SIZE

This parameter defines the maximum number of itemsets. The default value for this parameter is 3. Reducing this number will reduce the model processing time.

### MAXIMUM\_SUPPORT

This parameter defines the maximum support threshold of a frequent itemset. This parameter can be used to filter out those items that are too frequent, which is obvious. This parameter is available only in Enterprise edition.

### MINIMUM\_IMPORATANCE

This threshold will filter out rules which are less than the defined parameter value. This parameter is available only in Enterprise edition.

### MINIMUM\_ITEMSET\_SIZE

This parameter defines the minimum number of itemsets. The default value for this parameter is 1. Reducing this number will not reduce the model processing time. This parameter is available only in Enterprise edition.

### MINUMUM\_PROBABILITY

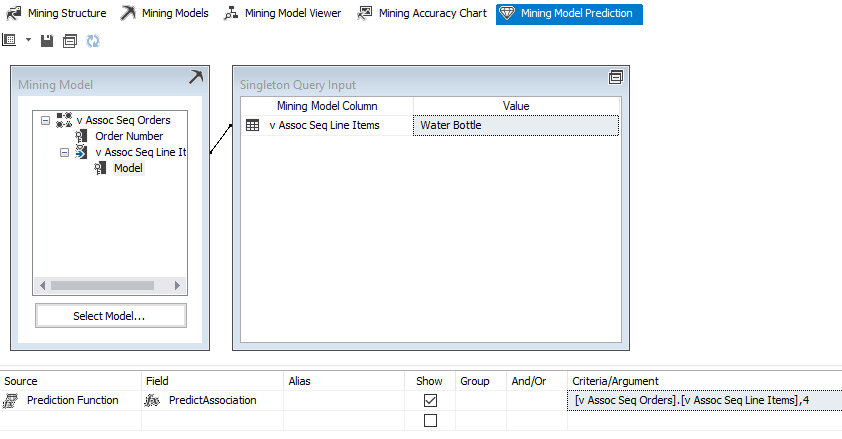
This parameter specifies the minimum probability that a rule is true. For example, setting this value to 0.5 specifies that no rule with less than 50% probability is generated.

### MINIMUM\_SUPPORT

This parameter specifies the minimum number of cases that must contain the itemset before generating a rule. Setting this value to less than 1 specifies the minimum number of cases as a percentage of the total cases. Setting this value to a whole number greater than 1 specifies the minimum number of cases as the absolute number of cases that must contain the itemset. The algorithm may increase the value of this parameter if memory is limited.

## Prediction

Predication is an important part of any data mining algorithm. Predication can be done from the following **Mining Model Prediction** tab, as shown below:



The above screenshot shows how to predict what are the items which will be bought by the customers who had bought Water Bottle.

## Conclusion

In conclusion, the Association Rule Mining method is an excellent way of finding associated items and optimizing article procurement and allocation in case of shopping applications.

1. [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/microsoft-clustering-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Microsoft Clustering in SQL Server

February 20, 2020 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

Microsoft Clustering is the next data mining topic we will be discussing in our SQL Server Data mining techniques series. Until now, we have discussed a few data mining techniques like: Naïve Bayes, Decision Trees, Time Series, and Association Rules.

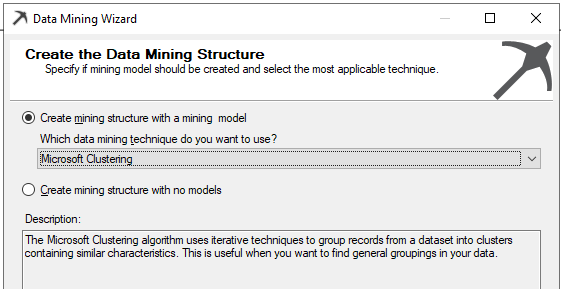
Microsoft Clustering is an unsupervised learning technique. In supervised training, there will be a variable that is already tagged to. In unsupervised training, there is no previously set variable as such.

Clustering is used to find out imperceptible natural grouping in a data set. This data set can be a large data set. Further, if there are a large number of attributes, you need a special technique to find natural grouping as the manual grouping is impossible.

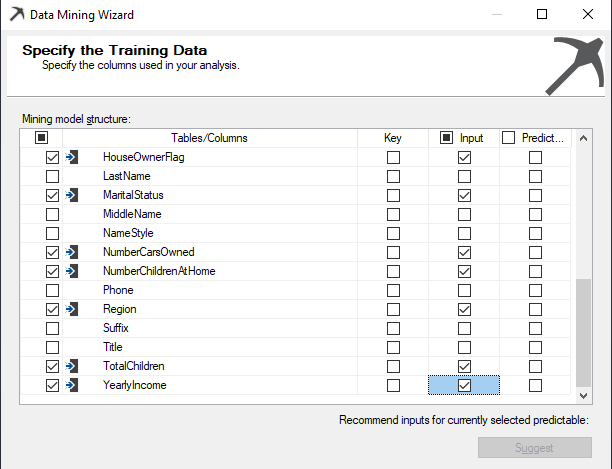
Let us see how we can perform clustering in the Microsoft SQL Server platform. In this example, we will be using the **vTargetMail** view in the **AdventureWorksDW** sample database, as we did for previous examples in the series.

Let us first create a data source and the Data Source View as we did in the other examples. In this, the data source would be **AdventureWorksDW,**while **vTargetMail** is the data source views.

In the wizard, the next is to choose the data mining technique:



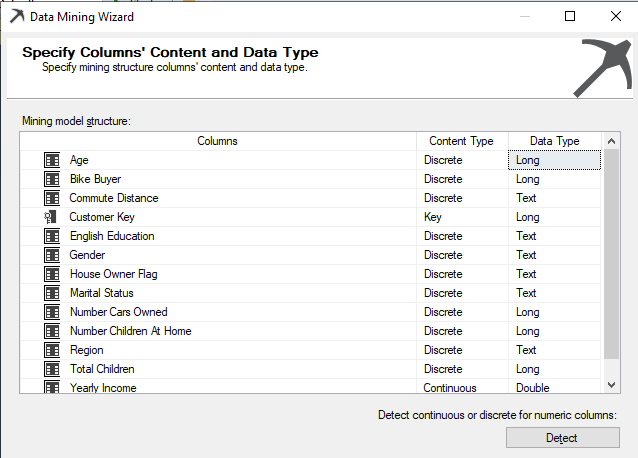
Since there is only one view in the Data Source View, **vTargetMail** will be the Case table and next is to choose relevant attributes as shown in the below screenshot:



In the above, the Customer Key is chosen as the Key from the algorithm. Since it is assumed that attributes such as Middle Name, Title will not make major contributions towards the natural grouping, input variables are chosen with sense. If not, there will be unnecessary processing time for the data mining structures. So in the above selection, **Age, BikeBuyer, CommuteDistance, EnglishEducation, EnglishOccupation, Gender, HouseOwnerFlag, MaritalStatus, NumberCarsOwned, NumberChildrenatHome, Region, TotalChildren**and**YearlyIncome** were chosen as relevant attributes.

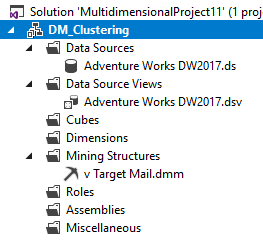
Since we are using the Microsoft Clustering algorithm, there is no need to choose Predict variable. This is why we said earlier that the Microsoft Clustering is an unsupervised learning technique.

Next is to select the correct Content types, though there are default Content types. Content types can be modified from the following screen:



In the above screenshot, for the numerical data type or long data type, the default content type will be Continuous. For example, columns like Age, Number Cars owned are numeric data types. Though they are numeric, we know that values are Discrete as Number Cars Owned contain values such as 0, 1, 2, 3, etc. Content-Type of **Age, Bike Buyer, Number Cars Owned, Number Children at Home and Total Children** attributes were changed to Discrete from Continuous. We will leave the Yearly Income content type as Continuous.

In the data mining wizard, default settings are used and now you are ready to process the data mining structure as shown in the below screenshot. This is the Solution Explorer for the Clustering data mining technique:

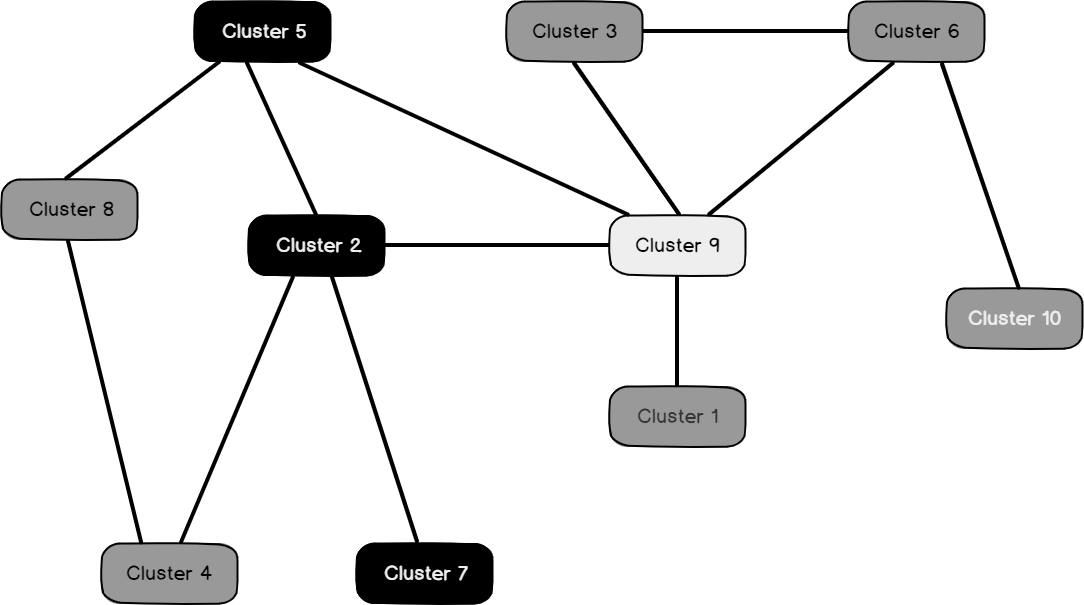


After processing the data mining structure, now let us view the results. There are four analysis graphs: Cluster Diagram, Cluster Profiles, Cluster Characteristics, and Cluster Discrimination.

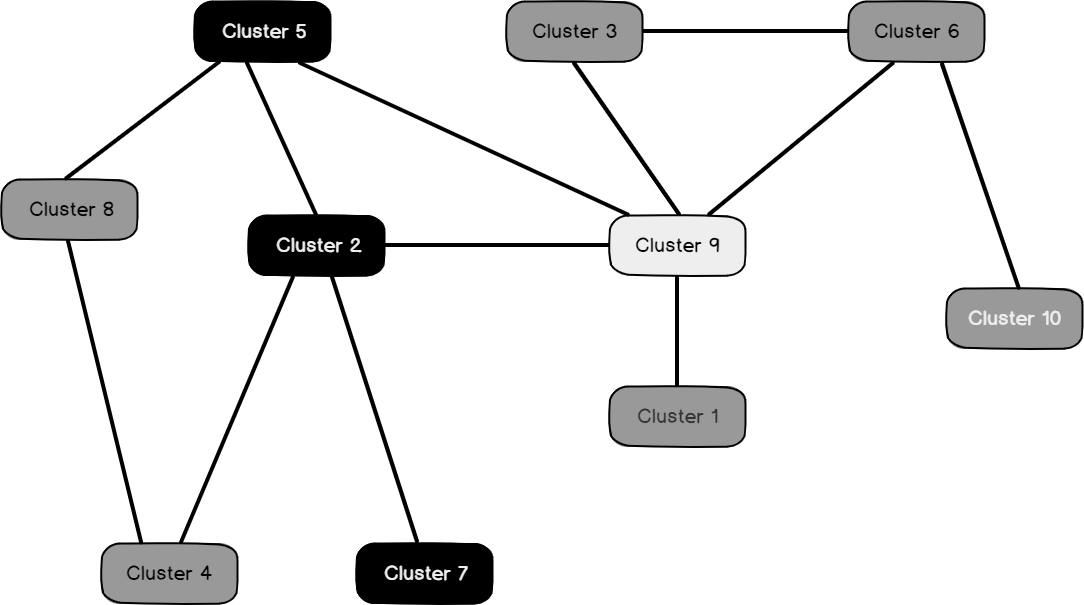
## Cluster Diagram

The cluster diagram has main two features. You can find the cluster distribution from the Cluster diagram. From the available drop-down, you can find the relevant cluster. When the relevant values are selected, the color will change accordingly.

Following is the Cluster diagram for Bike buyer =1. This means Cluster 5, Cluster 9 and Cluster 4 have the Bike buyer = 1:



Following is the Cluster diagram for Marital Status = S:



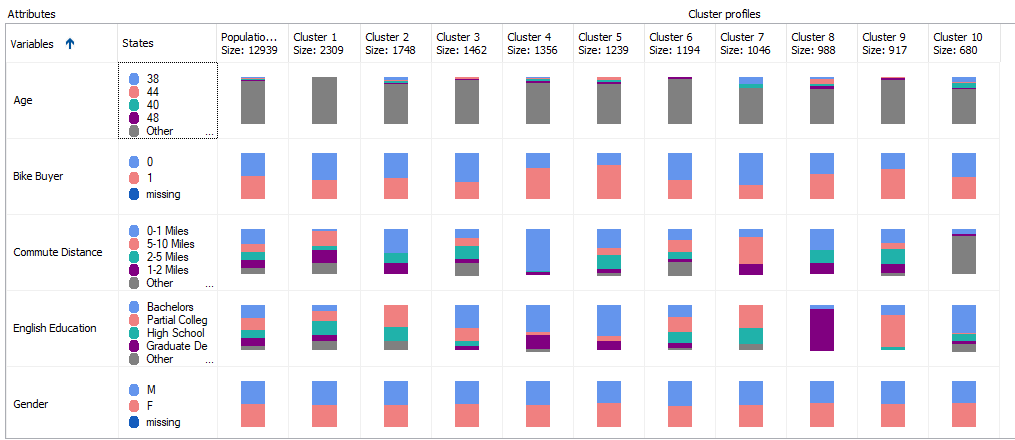
You can select the attribute and its values from the two available dropdowns.

In addition, a Cluster diagram can find out similar clusters and weak and strong links by moving the sliders up or down.

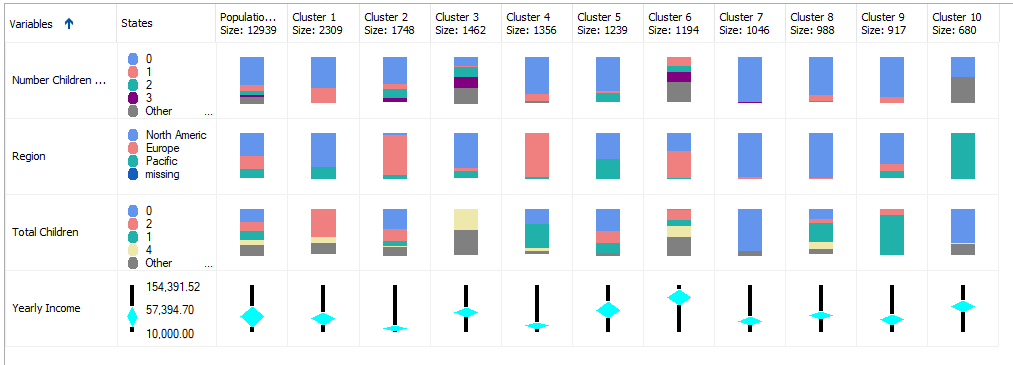
## Cluster Profiles

Each cluster has its own property. You can view the cluster profiles from this view. Since there are a few numbers of attributes, cluster profiles are shown in two images.

This is the first screenshot, which shows Age, Bike Buyer, Commute Distance, English Education, and Gender attributes:

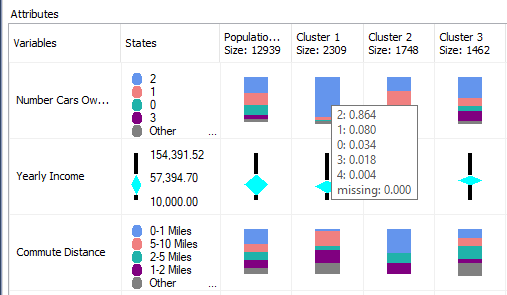
[](https://www.sqlshack.com/wp-content/uploads/2020/02/cluster-profles-for-microsoft-clustering-technique-1.png)

This is the second screenshot, which shows Number Children at Home, Region, Total Children, Yearly Income attributes:

[](https://www.sqlshack.com/wp-content/uploads/2020/02/cluster-profiles-for-microsoft-clustering-techniqu-1.png)

Except for Yearly Income, all the other attributes are dropdowns attributes that are shown in stack bars. Continuous attribute, Yearly Income is shown as five numbers (Minimum, Maximum, Mean, First Quartile, and Third Quartile) format.

If you make a move on top of each stack, you will see the contributions, as shown in the below screenshot:



If you closely look at these cluster profiles, you will see that Cluster 4, Cluster 5 and Cluster 9 have a higher percentage of Bike Buyers. Out of those three clusters, Cluster 4 has the highest percentage of customers who do not have a car. Further, you will see that Cluster 4 is a Low-Income as well. This means Cluster 4 can be labeled as **Low Income** **Bike Buyers without a Car**. You can rename the cluster with a business-friendly name accordingly that is shown in the below image:

[](https://www.sqlshack.com/wp-content/uploads/2020/02/after-chanigng-the-cluster-names-to-friendly-names-1.png)

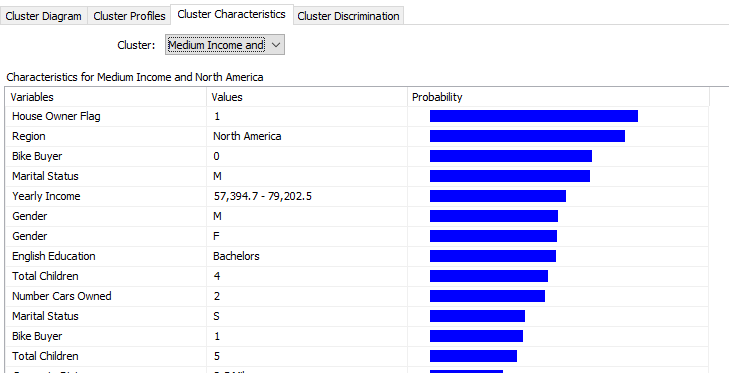
After naming these clusters, if you go back to the previous cluster diagram, you will see that new cluster names are updated, as shown in the below screenshot:



With this rename option, your cluster profiles are much readable than using the default cluster names.

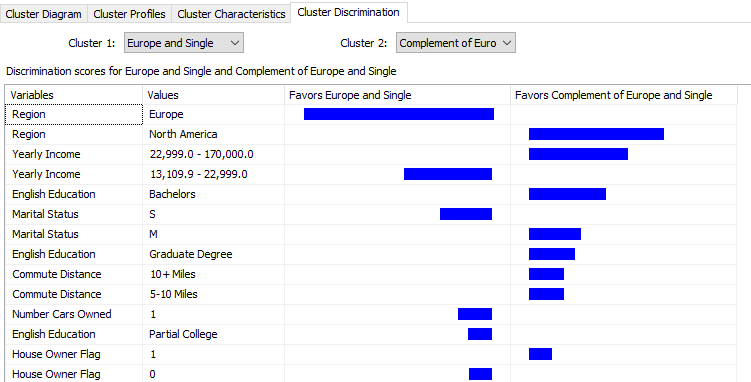
## Cluster Characteristics

You can view the characteristics of clusters using the Cluster Characteristics tab, as shown in the below screenshot:



## Cluster Discrimination

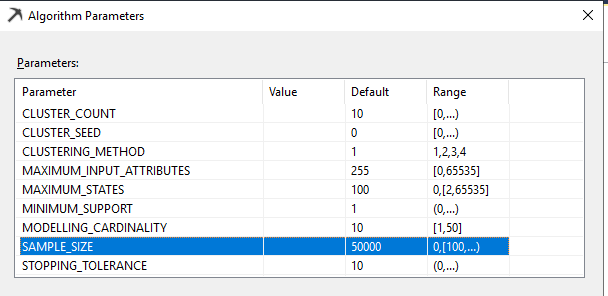
Since there are ten clusters in this example, sometimes you might be interested to know the difference between the two clusters. By choosing any two clusters, you will understand the difference between the selected clustered, as shown in the below screenshot:



In the above screenshot, the first cluster predominantly has customers in Europe while the other cluster contains customers from North America.

## Algorithm Parameters

As we discussed in the previous articles, we can gain better results by modifying the algorithm parameters from the following screenshot:



Let us look at important parameters for the Microsoft Clustering technique.

### Cluster Count

This parameter defines the number of clusters. As you can remember, we had 10 clusters in the example. There is no limit on the number of clusters. If this value is set to 0, a number of clusters will be decided by the algorithm. However, it is essential to limit the number of clusters to visualize better results and of course, the performance of the model. Typically, cluster count 5 is the optimal value.

### Cluster Method

The clustering method algorithm uses can be:

1. Scalable EM
2. Non-scalable EM
3. Scalable K-means
4. Non-scalable K-means

In Microsoft Clustering, there are two main methods for clustering: Expectation-Maximization (EM) and K-Means. EM cluster assignment method uses a probabilistic measure while K-Means uses Euclidean distance. EM method is also called as soft clustering as one object can be fallen into multiple clusters with different probabilities. Conversely, the K-mean clustering is called Hard Clustering.

Clustering is an iterative process. If there are a large number of data points, a large amount of memory will be consumed. With the scalable framework, if the algorithm finds that the data point is not going to change its cluster, those points are removed from the iteration. In this technique, data is loaded by chunk. The number of data points is defined by the SAMPLE SIZE parameter. By doing so, the clustering technique will be much scalable.

### Sample Size

If you have chosen any scalable clustering option, this parameter defines how many data points should be selected for each iteration. The default value is 50,000.

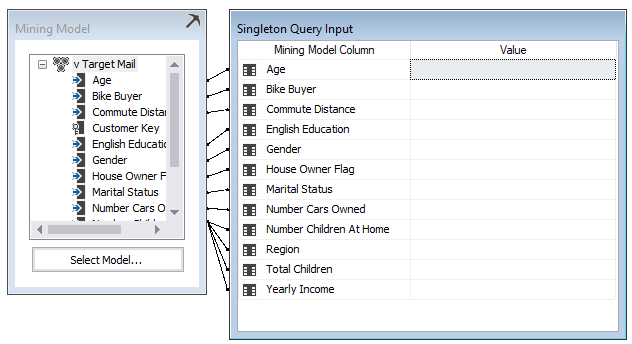
### Stopping Tolerance

This parameter determines the convergence point for the iteration process to stop. Increasing this number will cause the iteration to stop quickly. If you have a large data set, you can increase this value.

## Cluster Prediction

The next important aspect is predicting the cluster for a given data set. This can be done by using the Mining Model prediction tab.

First, you need to select the built data mining model. We will check the cluster for a one value set; hence we should select, Singleton Query option from the top:



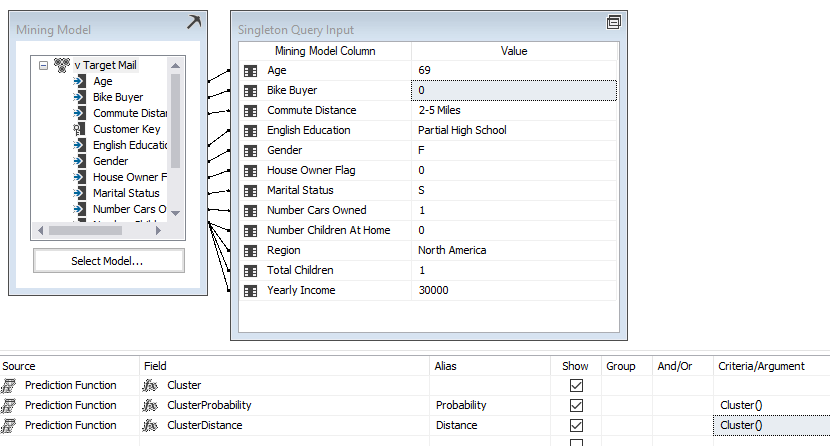
Now we need to enter values for each attribute. For the nominal variables, you need to select from the available values, whereas, for the continuous variables like Yearly Income, you need to type a value.

The following screenshot shows how values are entered for a variable:

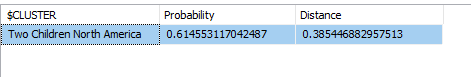


In case you don’t know values for any variables, you can tag that as **Missing**. However, it is important to catch as much as values possible.

Then you will choose Prediction function in the bottom half of the screen, as shown below:

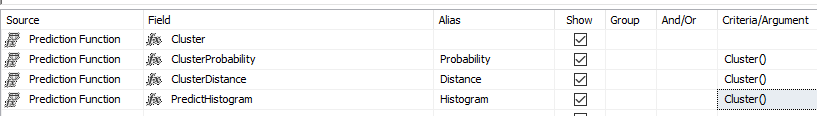


When the result option is clicked, you will get the following results, showing the relevant cluster that this data set belongs to:

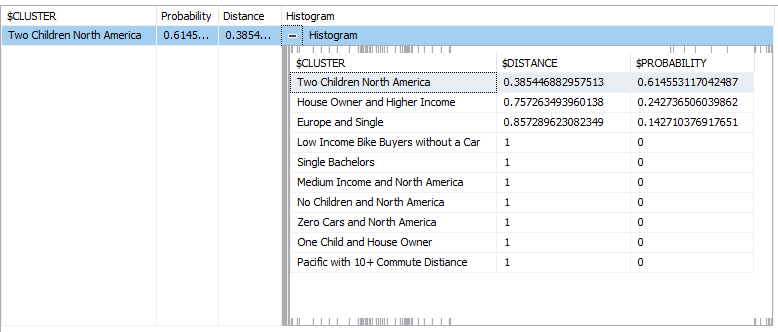


This result indicates that your data set belongs to the **Two Children North America**cluster with 0.61 probability. The Distance parameter indicates how far your data set is from this cluster. Basically, it is the inverse of the probability.

Then your next question would be what other probable clusters are. This can be obtained from the Predication function, Histogram, as shown in the below screenshot:



The following screenshot shows the results for the above prediction function:



The above result shows that this particular data set belongs to three different clusters with different probabilities.

## Summary

Clustering is an unsupervised technique that can be used to create natural grouping in a data set. There are two main techniques K-Means and EM. To facilitate large data volumes, the scalable option is available too. In Microsoft Clustering, there are multiple views to get more details into the clustering. Cluster Diagram provides you with the relationship of available clusters. The cluster Profile view provides the pictorial view of the cluster profiles. The Cluster Characteristics view will provide you with the details of a selected cluster, whereas Cluster Discrimination will provide you the option of comparison of two selected clusters.

From the built model, predications can be done where we can get the cluster it belongs to and its probability.

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/microsoft-linear-regression-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Microsoft Linear Regression in SQL Server

March 24, 2020 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

In this article, we will be discussing Microsoft Linear Regression in SQL Server. This is the next data mining topic in our SQL Server Data mining techniques series. Naïve Bayes, Decision Trees, Time Series, Association Rules, and Clustering are the other techniques that we discussed until today.

Microsoft Linear Regression is a forecasting technique. In this type of technique, there are multiple independent variables from which the dependent variable is predicted. For example, if you want to predict the house prices, you need to know the number of rooms, the area of the house, and other features of the house.

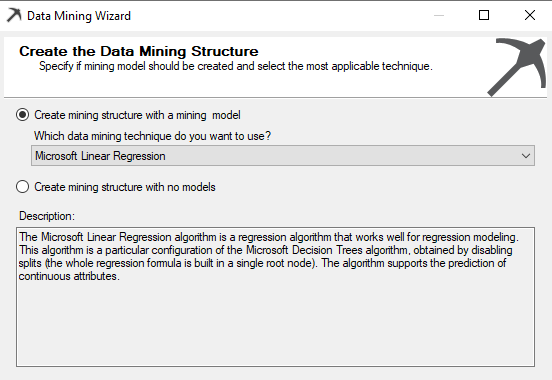
This means that the linear regression model can be represented as follows:

Y = a X1 + b X2 + . . . + z Xn + C

Let us see how we can use linear regression in the Microsoft SQL Server platform. As in the previous examples, today also, we will be using the **vTargetMail** view in the **AdventureWorksDW** sample database.

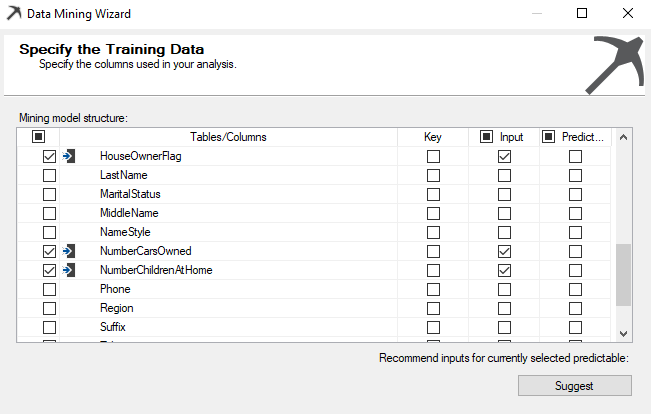
As we did for other data mining techniques, first, we need to create a data source and the Data Source View. The Data source is chosen as **AdventureWorksDW**and **vTargetMail** view is selected as the data source views.

We choose the Microsoft Linear Regression as the data mining technique, as shown in the below screenshot.



In this technique, the Microsoft decision trees algorithm is used. Unlike in the decision trees, linear regression will have only one node, and we will verify the results for linear regression with the decision trees at the end of the article.

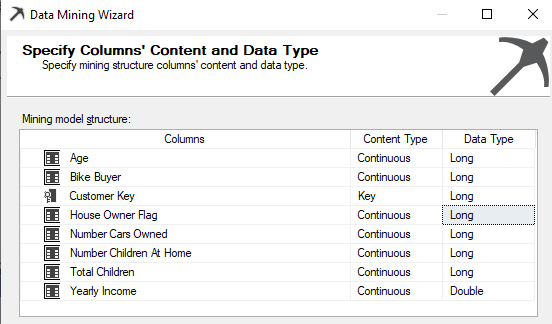
The **vTargetMail** will be the Case table and let us choose relevant attributes, as shown in the below screenshot.



The Customer Key is chosen as the Key from the algorithm from the above screen. In Microsoft Linear regression, all the inputs should be numerical; the text column should not be selected. Therefore, in the above selection, **Age, BikeBuyer, HouseOwnerFlag, NumberCarsOwned, NumberChildrenatHome, TotalChildren**are selected as input attributes. This is a major limitation in the Microsoft Linear Regression, which is not in the standard Linear Regression techniques.

In the previous examples, we have selected Bike Buyer as the predicted column. However, in the Microsoft Linear Regression, we are to predict **YearlyIncome.**

Though there are default Content types, there are instances where you need to change the content types. Content types can be modified from the following screenshot.



By default, **House Owner Flag** is selected Text data type, which has to be changed to the Long data type.

In the other screens in the data mining wizard, default settings are used. This is the Solution Explorer for the Microsoft Linear Regression data mining technique.

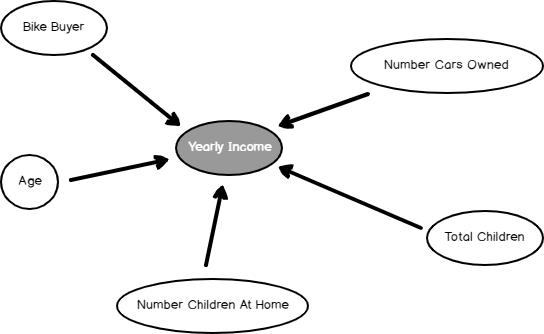


The next is to process the data mining structure. There will be a warning message saying that there is no split in decision trees. You can ignore this warning as for linear regression; there won’t be any split for the decision trees.

After processing the data mining structure, we are now ready to view the results.

## Viewing the Results

As we observed in many SQL Server algorithms, in linear regression, we can find the dependency network, as shown in the below screenshot.



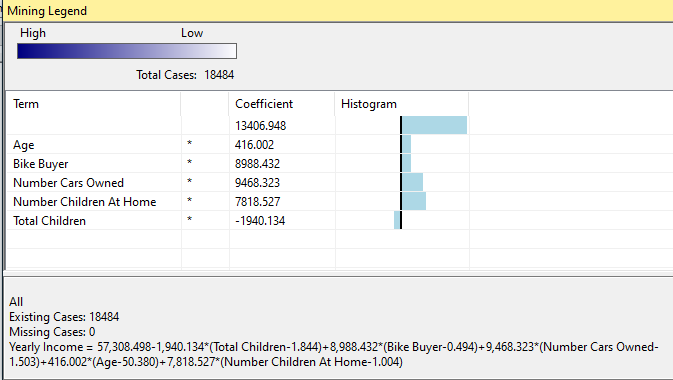
The Dependency network shows what the most dependent attributes to predict Yearly Income is. By sliding the slider down on the left-hand side, you can find out the significance of these attributes, as we observed in the Naïve base and Decision Trees.

In Microsoft Linear Regression, only another available view is Tree View. However, as indicated before, it is a one-node tree view.



From this view, you can get the linear regression equation, which is the final goal of this technique.

The following screenshot shows the linear regression equation.



This is the equation and you simply have to replace relevant values to predict the yearly income.

**Yearly Income = 57,308.498**

**– 1,940.134\*(Total Children-1.844)**

**+ 8,988.432\*(Bike Buyer-0.494)**

**+9,468.323\*(Number Cars Owned-1.503)**

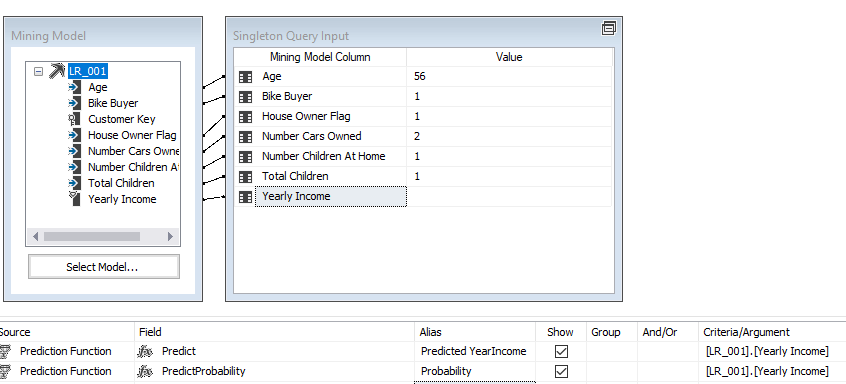
**+416.002\*(Age-50.380)**

**+7,818.527\*(Number Children At Home-1.004)**

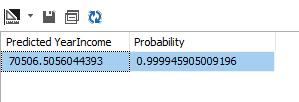
Let us see how we can predict from the built model using prediction feature.

## Prediction

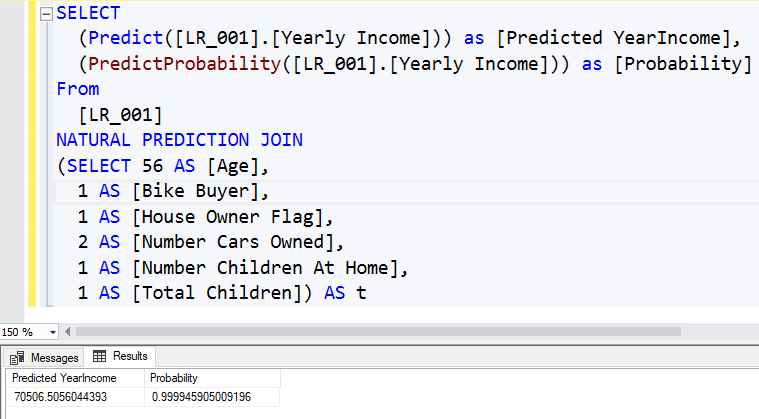
An important aspect of any data mining technique is to predict using the built model. Let us see how we can predict using the built Microsoft Linear Regression model. This can be done from the Mining Model Prediction tab, as shown in the below screenshot. In the following example, some values are provided for a given instance to predict the annual income.

[](https://www.sqlshack.com/wp-content/uploads/2020/03/prediction-using-built-linear-regression-model-.png)

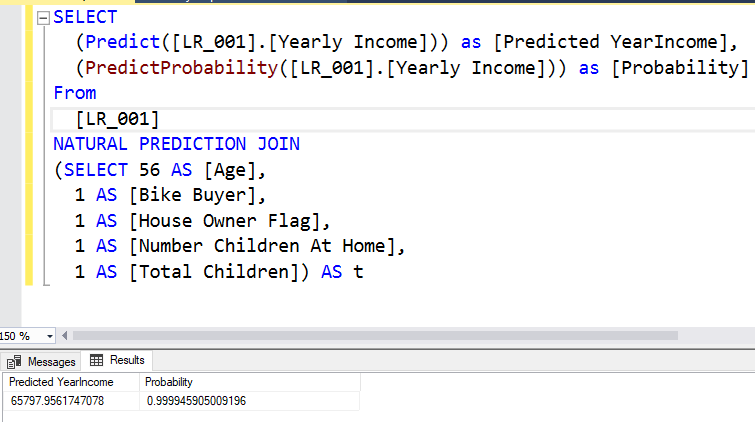
From the result tab, results can be views as shown in the below screenshot.



The same results can be obtained from SQL Server Management Studio by executing the DMX query. The following screenshot shows the query and its result.



It is important to note that if you do not have some attributes, you can still obtain the results. The following screenshot shows the prediction value from the Linear Regression model when the number of cars is unknown.

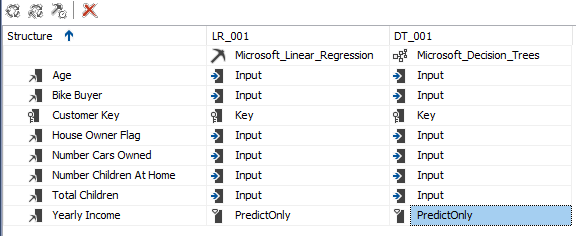


When an attribute is missing, that attribute’s part will be ignored from the entire equation.

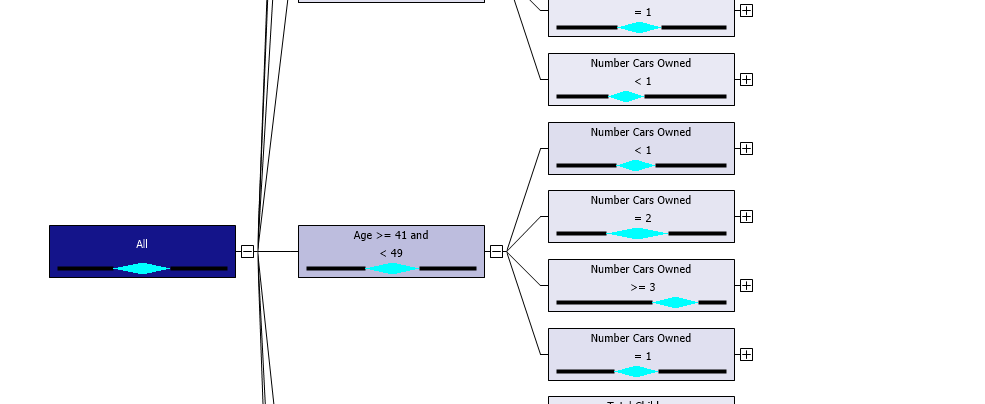
Let us validate the equation with the Decision Tree technique.

## Validation with Decision Tree

Adding another data mining technique in SQL Server is much simpler. You can add another mining model to the existing attributes in the Mining Model tab.



After processing the mining structure, you will observe the decision, as shown below.

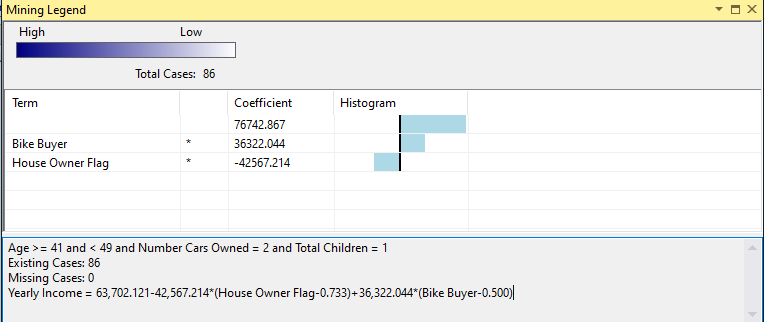
[](https://www.sqlshack.com/wp-content/uploads/2020/03/decision-trees-for-regression-.png)

Let us look at the equation at the main node.

**Yearly Income = 57,308.215 + 9,468.574\*(Number Cars Owned-1.503) + 415.816\*(Age-50.384) + 8,988.666\*(Bike Buyer-0.494) + 7,817.585\*(Number Children At Home-1.004) – 1,939.209\*(Total Children-1.844)**

You will find that it is the same equation that you got from the linear regression.

More than that equation, the decision tree has the additional advantage of having node wise equation. In the decision trees, if you click every node, you find an equation, as shown in the below screenshot.



This means that equation Yearly Income will be as 63,702.121-42,567.214\*(House Owner Flag-0.733)+36,322.044\*(Bike Buyer-0.500) is valid for Age >= 41 and < 49 and Number of cars owned = 2 and Total children = 1.

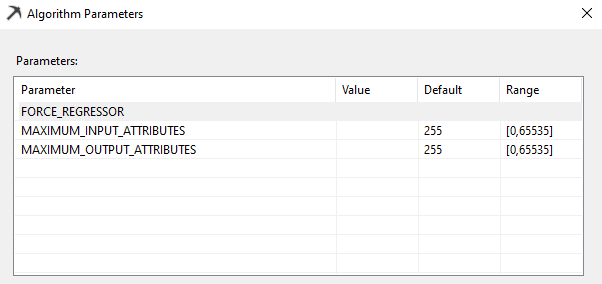
The following table shows the different equations at different nodes in the tree.

|  |  |
| --- | --- |
| **Data Set** | **Equation** |
| **Age >= 73 and < 81 and Total Children = 3** | **Yearly Income = 56,936.254-4,193.080\*(Bike Buyer-0.121)-20,137.503\*(Number Cars Owned-1.994)-1,936.065\*(Age-75.146)** |
| **Total Children = 3 and Age = 76** | **Yearly Income = 58,000.000-8,884.447\*(Bike Buyer-0.100)** |
| **Age = 73 and Total Children = 3** | **Yearly Income = 56,998.501+4,498.500\*(Bike Buyer-0.333)** |
| **Age >= 73 and < 81 and Total Children = 2 and Number Children At Home = 3** | **Yearly Income = 121,037.417+2,108.061\*(Age-75.667)+14,848.268\*(Bike Buyer-0.333)** |
| **Age >= 49 and < 51 and Total Children >= 4 and Number Cars Owned = 2 and Number Children At Home < 3** | **Yearly Income = 62,583.618-18,114.343\*(Number Children At Home-1.897)+5,525.516\*(Bike Buyer-0.793)+6,861.981\*(Age-49.759)-14,461.923\*(Total Children-4.017)** |

This means that decision trees are more accurate than Microsoft linear regression.

## Model Parameters

As we discussed, every data mining technique has its parameters to suit your data and environments.



### FORCE\_REGRESSOR

Microsoft Linear Regression algorithm detects best-fitted attributes automatically and generates the linear equation. In this attempt, it might drop some attributes. However, you can force any attribute that you wish to include in the equation by including in the FORCE\_REGRESSOR parameters. If there are multiple attributes, you can include all attributes, such as {Attribute 1}, {Attribute 2}.

## Summary

In this article, we discussed Linear regression as a forecasting technique. Microsoft Linear Regression technique was built on the Decision trees and we identified that the decision trees could be used as a regression technique as well.

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/implement-artificial-neural-networks-anns-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

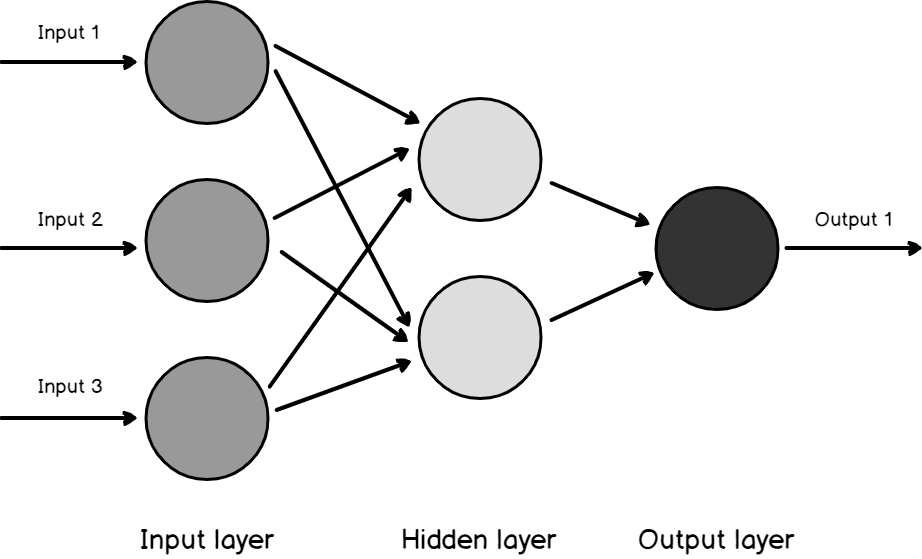
# Implement Artificial Neural Networks (ANNs) in SQL Server

April 14, 2020 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

In this article, we will be discussing Microsoft Neural Network in SQL Server. This is the seventh article of our SQL Server Data mining techniques series. Naïve Bayes, Decision Trees, Time Series, Association Rules, Clustering, and Linear Regression are the other techniques that we discussed until this article.

## What is an Artificial Neural Network?

An Artificial Neural Network (ANN) can be considered as a classification and as a forecasting technique. Microsoft Neural Network in SQL Server is typically a more sophisticated technique than Decision Trees and Naïve Bayes. This technique tries to simulate how the human brain works. In this technique, there are three layers, Input, Hidden, and Output, as shown in the below screenshot.



The input layer is mapped to the input attributes. If you remember the AdventureWorks example, we are looking at Age, Gender, Number of Children are the inputs to the Input layer.

The Hidden layer is an intermediate layer where every input with weightage is received to each node in the hidden layer.

The Output layer is mapped to the predicted attributes. In our AdventureWorks example, Bike Buyer will be mapped to the output layer.

A neuron is a basic unit that combines multiple inputs and a single output. Combinations of inputs are done with different techniques, and the Microsoft Neural Network uses Weighted Sum. Maximum, Average, logical AND, logical OR are the other techniques used by the different implementation.

After these inputs are calculated, then the activation function is used. In theory, sometimes, small input will have a large output, and on the other hand, large input might be insignificant to the output. Therefore, typically non-linear functions are used for activation. In Microsoft Neural Network uses tanh as the hidden layer activation function and sigmoid function for the output layer.

## Backpropagation

Backpropagation is the core part of the Artificial Neural Network. Unlike other techniques, this technique has the learning capability. The Learning capability is achieved via Backpropagation. In this technique, the error is calculated, and the weights points will be modified.

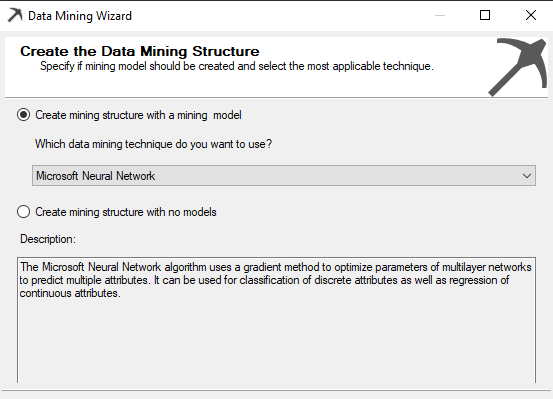
Let us see how Microsoft Neural Network in SQL Server works.

1. At the initial stage, random values between -1 to 1 are assigned as weightages
2. For the training set. The algorithm calculated the output and output error
3. The Backpropagation process calculates the error for each output and hidden neurons in the network
4. The weightages are modified
5. Repeat from step 2 until the condition is satisfied with minimum error

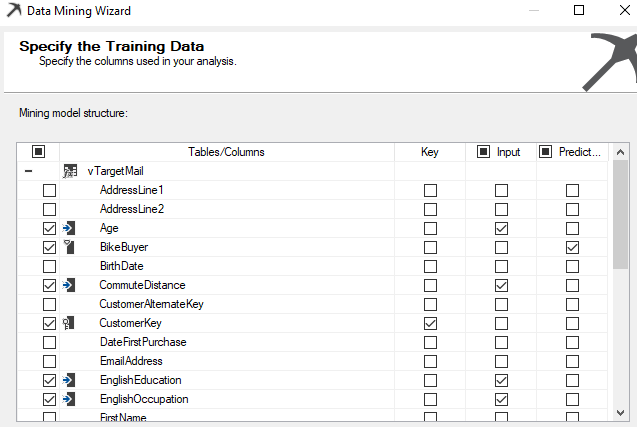
## Implementing an Artificial Neural Network in SQL Server

Let us do the same example of Bike Buyer that we did for Naïve Bayes and Decision Trees. Like we did for all the other examples, let us create the Data Source pointing to the AdventureWorksDW database and Data Source View with vTargetMail.

Then let us select the Microsoft Neural Network, as shown in the following screenshot.

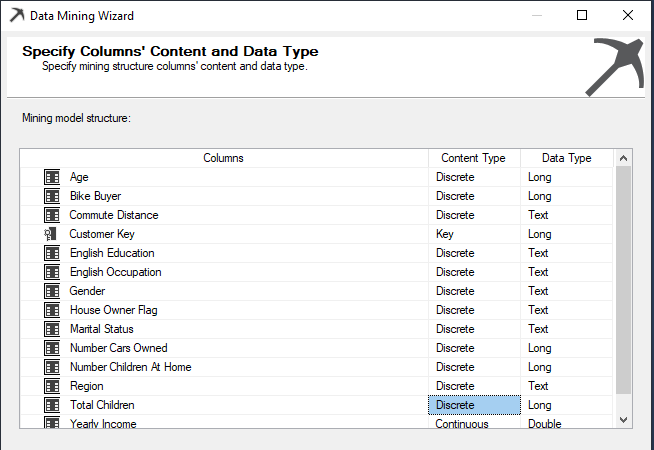


Then let us select the Input and Predict attribute, as shown in the below screenshot.



We have chosen input attributes that will make sense to predict the bike buyer. For example, we do not think that attributes such as Address, email address are important variables.

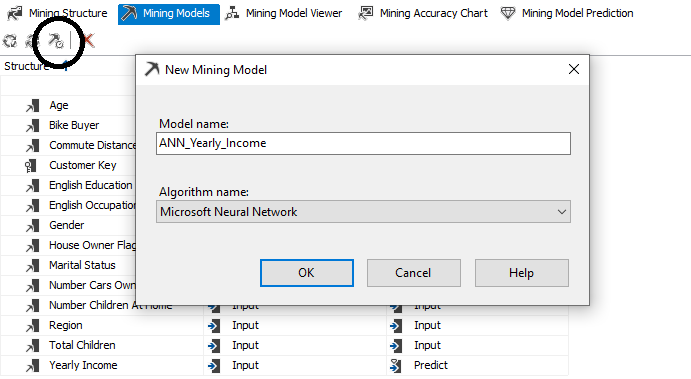
Next is to change the Content-Type. Though Neural Network supports continuous types, in this example, only Yearly Income should be continuous, and other content types should be changed to Discrete due to the nature of the data set. After those changes, the screen should be as follows.



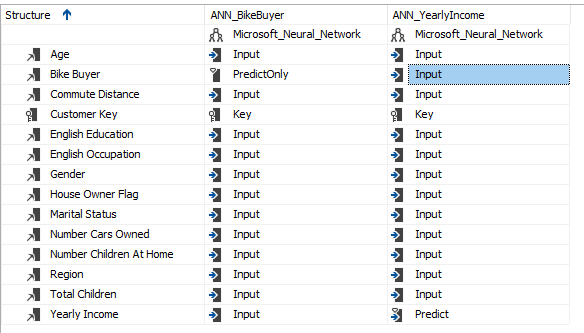
After this modification, the rest of the wizard can be configured with default values, as we discussed in our first article of the series.

The above model is to perform classification whether the customer is a bike buyer or a not. Now, let us add another neural network model to forecast yearly income, like the decision tree algorithm, Microsoft Neural Network in SQL Server can be used as a classification and forecasting technique.

You can add another mining model the same structure without creating another structure, as shown in the below screenshot.



After the model is created, you need to change the Yearly Income to predict, as shown in the below screenshot.

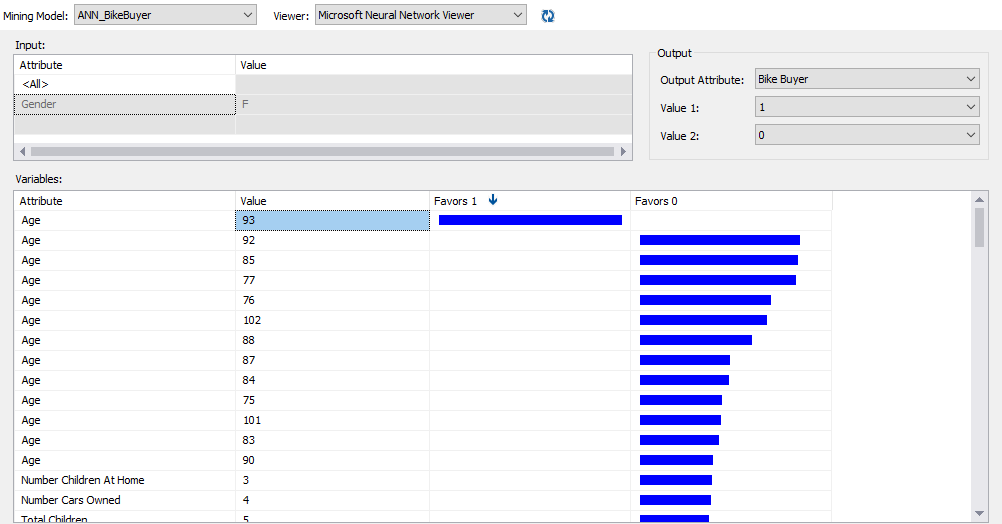


Now you have two mining models in the same data mining structure.

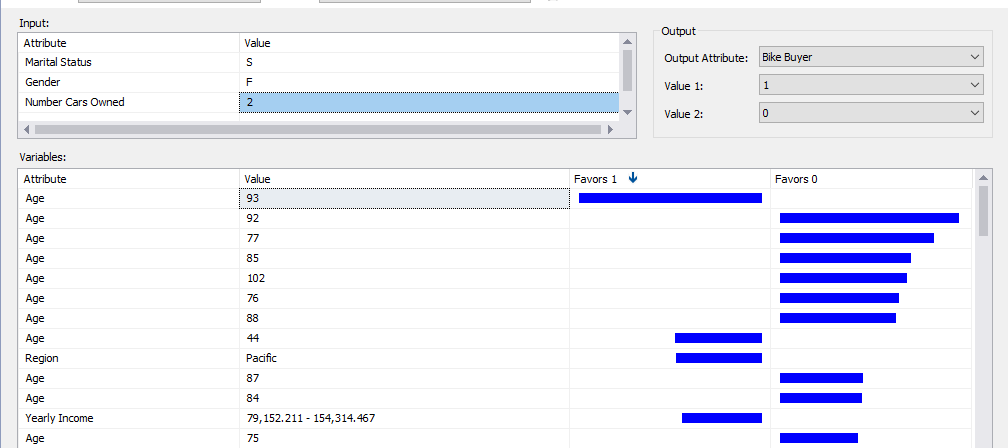
Then let us process both models together and view the results. Further, if you want, you can process the model by model.

## Model Viewer

Let us view the results for the Bike Buyer prediction model built using the Microsoft Artificial Neural Network algorithm, as shown in the below screenshot.

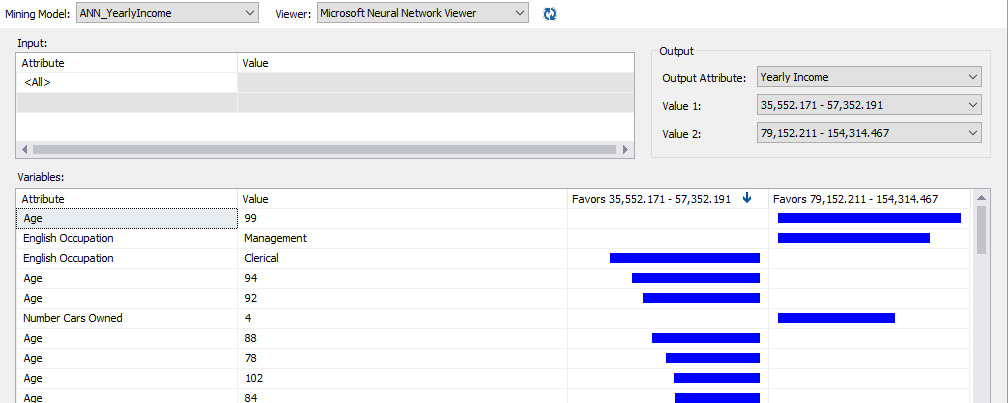
[](https://www.sqlshack.com/wp-content/uploads/2020/04/default-model-viewer-with-bike-buyer-prediction-mo.png)

Above screenshot indicates that Customers whose age is 93 are more favorable of buying a car. Further, we can filter the results. The following screenshot shows the results for Single, Female who has two cars.

[](https://www.sqlshack.com/wp-content/uploads/2020/04/mmodel-viewer-with-bike-buyer-prediction-model-wit.png)

By analyzing these views, the user can understand what are contributing attributes towards the classification of a Bike Buyer.

Let us look at the Year Income prediction model viewer, as shown in the below screenshot.

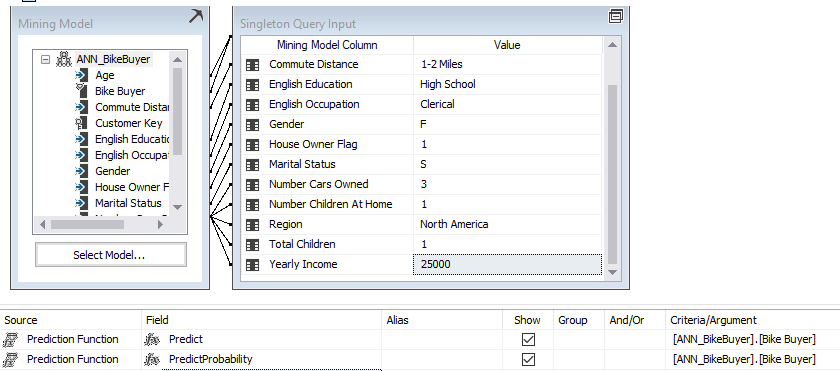
[](https://www.sqlshack.com/wp-content/uploads/2020/04/default-data-mining-model-viewer-for-yearly-income.png)

Since Yearly Income is a continuous attribute, you can choose them in ranges, as shown above. In this model, we can filter for several attributes. Similarly, you can get an understanding of what are the most significant factors for each range of Yearly Income attribute.

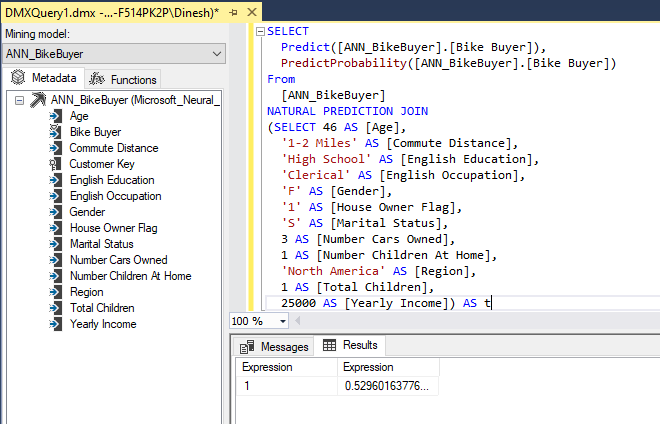
## Prediction

Let us see how we can use these models to predict. As we discussed in the previous article, Microsoft Neural Network in SQL Server can be used to predict from DMX queries and the provided user interface.

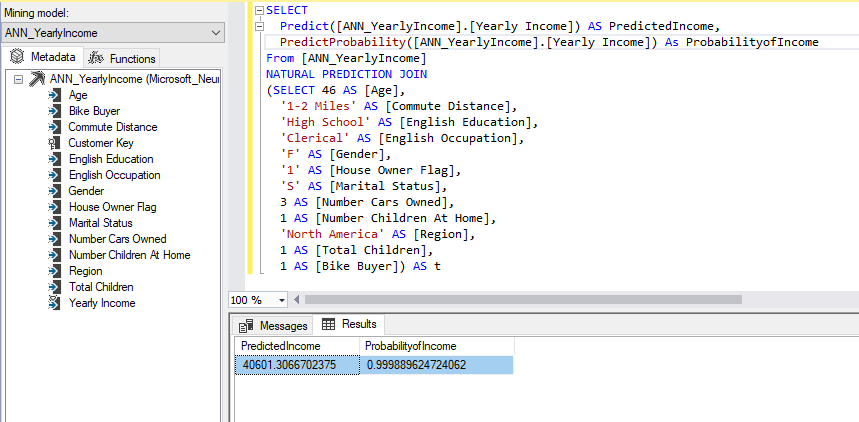
Following is the user interface for making predictions:



The same results can be achieved by using a DMX query from the SQL Server Management Studio (SSMS), as shown in the below screenshot.



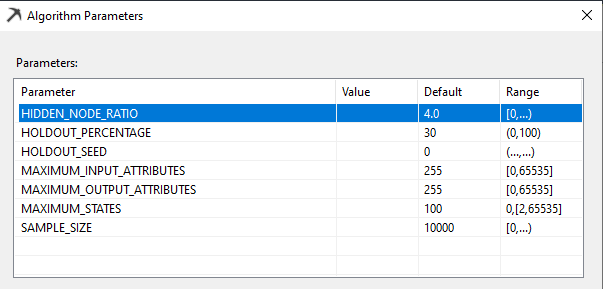
Let us forecast the Yearly Income using the ANN\_YearlyIncome model from the DMX, as shown below.



In case you are accessing data mining models from the application, DMX queries can be used.

## Model Parameters

There are Microsoft Artificial Neural Network related model parameters to achieve better results. As we discussed in the previous articles, by changing the parameter values, you will be able to achieve better results. All of these parameters are available only in Enterprise edition.



### HIDDEN\_NODE\_RATIO

This parameter specifies a number used in determining the number of nodes in the hidden layer. The algorithm calculates the number of nodes in the hidden layer as HIDDEN\_NODE\_RATIO \* sqrt({the number of input nodes} \* {the number of output nodes}).

### HOLDOUT\_PERCENTAGE

This parameter specifies the percentage of cases within the training data used to calculate the holdout error for this algorithm. HOLDOUT\_PERCENTAGE is used as part of the stopping criteria while training the mining model. The default value for this parameter is 30.

### HOLDOUT\_SEED

This parameter specifies a number to use to seed the pseudo-random generator when randomly determining the holdout data for this algorithm. This value is unique to this algorithm and is unrelated to any holdout parameters set in the mining structure. The default values for this parameter is 0

### SAMPLE\_SIZE

This parameter defines the number of cases that are used to train the model. The algorithm either uses the number specified by SAMPLE\_SIZE or total\_cases \* (1 – HOLDOUT\_PERCENTAGE/100), depending on which one is smaller.

## Conclusion

Microsoft Artificial Neural Network in SQL Server is one of the most sophisticated algorithms available in the SQL Server Data Mining family. This technique tries to simulate how the brain works with input and outputs. Further, this technique can be used to solve classification and regression problems like the Decision Tree algorithm. Both discrete and continuous input variables can be used for this technique.

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/implementing-sequence-clustering-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Implementing Sequence Clustering in SQL Server

April 24, 2020 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

In this article, we will be discussing Microsoft Sequence Clustering in SQL Server. This is the ninth article of our SQL Server Data mining techniques series. [Naïve Bayes](https://www.sqlshack.com/naive-bayes-prediction-in-sql-server/), [Decision Trees](https://www.sqlshack.com/microsoft-decision-trees-in-sql-server/), [Time Series](https://www.sqlshack.com/microsoft-time-series-in-sql-server/), [Association Rules](https://www.sqlshack.com/the-association-rule-mining-in-sql-server/), [Clustering](https://www.sqlshack.com/microsoft-clustering-in-sql-server/), [Linear Regression](https://www.sqlshack.com/microsoft-linear-regression-in-sql-server/), [Neural Network](https://www.sqlshack.com/implement-artificial-neural-networks-anns-in-sql-server/) are the other techniques that we discussed until this article.

## What is Sequence Clustering

The Microsoft Sequence Clustering algorithm is a combination of sequence analysis and clustering. This technique identifies natural groups (clusters) of similarly ordered events in a sequence.

In Sequence Clustering, we will be looking at the events which have occurred in sequence. Then the clustering is applied to that data set. The clusters can be used to predict the likely ordering of events in a sequence based on known characteristics.

In this technique, the Hidden Markov model will be used to generate the sequences while K-Means and Expectation-Maximization (EM) clustering techniques will be used for clustering. Since we have discussed clustering techniques in the Clustering article, let us discuss the Hidden Markov Process in short.

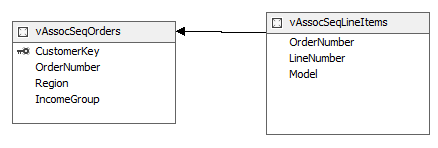
## Hidden Markov Process

If you wondering how Google ranks its pages, it is by using an algorithm called a PageRank. Page rank uses Markov Process. This technique was introduced by the mathematician named, Andrey Markov (1856 – 1922) in 1906. In Hidden Markov Process, we are looking at exploring the sequence of events.

## Implementation

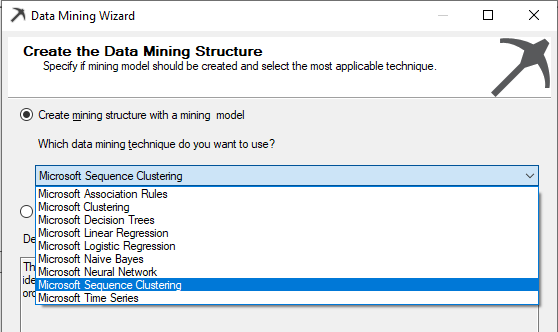
Let us see how we can implement, Microsoft Sequence Clustering in SQL Server. Like we did for all the other techniques, we need to create a SSAS project in this technique too. Similar to other projects, we need to create a data source to the **AdventureWorksDW** database.

In this technique, we have to use two views for the data source views similar to what we did in the [Association Rule technique](https://www.sqlshack.com/the-association-rule-mining-in-sql-server/). Those two views are **vAssocSeqOrders** and **vAssocSeqLineItems** added to the data source views as shown below.



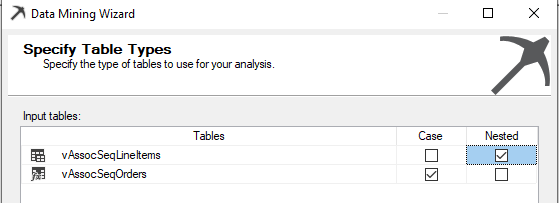
By default, there won’t be any relationships between these two views since there are no relationships built at the database level. Therefore, that relationship has to be built as shown in the above screenshot.

Next is to select the technique for the mining model. As shown in the below screenshot, we will be selecting the Microsoft Sequence Clustering from the available list.



After choosing the database source view, next is to choose the Case and Nested tables. Unlike most of the other techniques, there are two tables in this technique.

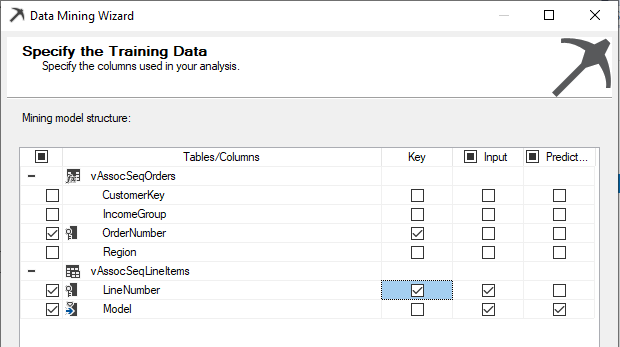
In this **vAssocSeqOrders** should be selected as the Case table where the vAssocSeqLineItems should be selected as the Nested as shown in the below screenshot.



Let us see what our objectives in this sample data are. If you recall what we did for the association rule, is that we are looking at what are the similar items selling together. In that example, we looked at only Order Number and the Model as inputs.

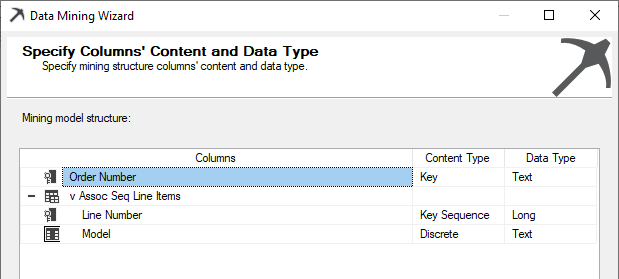
In the case of sequence clustering, we are looking at the sequences. Therefore, we will include the Line Number as the sequence parameter. Though this may not be a correct business case, however from the existing sample data, this is the only dataset that has a sequence attribute.

Therefore, we have selected Key, Input and Predict as shown in the following screenshot.



Ideally, we can choose CustomerKey as the key, but in this example, we do not have adequate data for the above case.

In the next couple of screens, default values are accepted.



From the algorithm, it is identified that the Line Number is the Key Sequence as shown in the above screenshot.

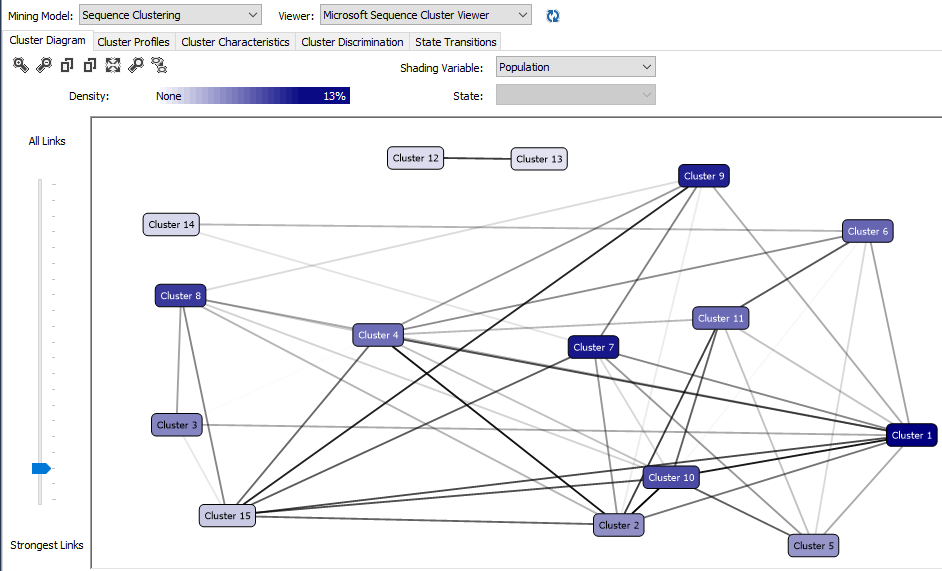
After the project is created, then the mining model has to be processed. After the model is processed successfully, the next is to view the results.

## Mining Model Viewer

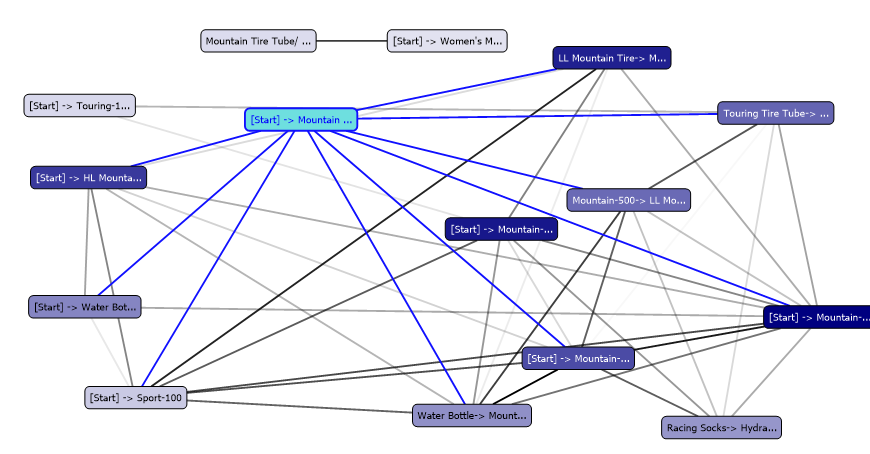
In the Microsoft Sequence Clustering, there is an additional view called State Transition than to the Microsoft Clustering Model viewers. Let us look at each viewer.

### Cluster Diagram

Since we have not limited the number of clusters, 15 clusters were defined. Clusters are named with default values that can be modified by looking at the cluster properties.

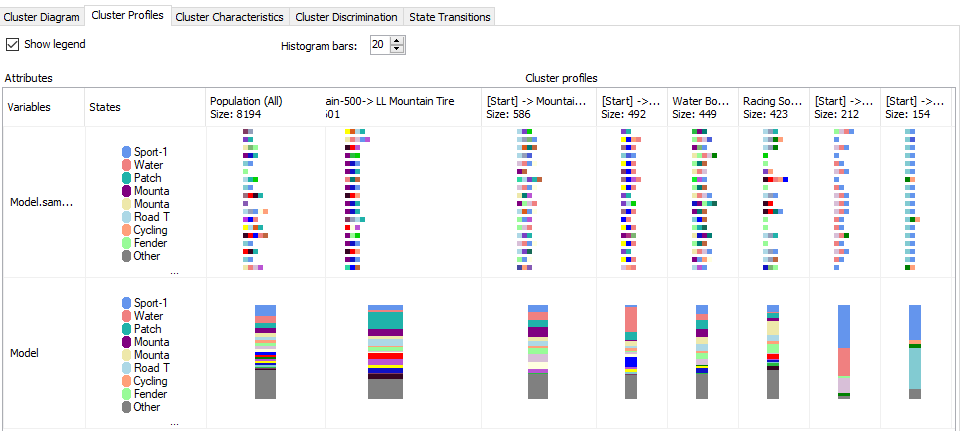
[](https://www.sqlshack.com/wp-content/uploads/2020/04/default-view-of-cluster-diagram.png)

After renaming the clustering to more understandable names, the cluster diagram is visualized like follows.

[](https://www.sqlshack.com/wp-content/uploads/2020/04/cluster-diagram-after-naming-them-to-a-meaningful.png)

### Cluster Profiles

In the cluster profile view, you can view all the cluster state and the transition in one view as shown below.

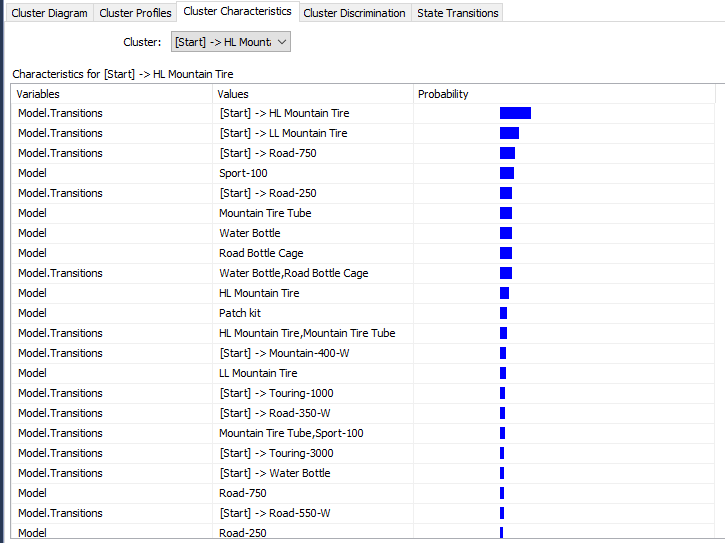
[](https://www.sqlshack.com/wp-content/uploads/2020/04/overview-of-cluster-profiles-in-microsoft-sequence.png)

Since we have renamed the cluster names, those names are reflected in this view as well as in other views.

### Cluster Characteristics

Cluster profiles view gives you a whole view of the clusters whereas the cluster characteristics view gives you details about each cluster. In the Cluster Characteristics view, you can view the details of the entire population as well as the selected cluster.

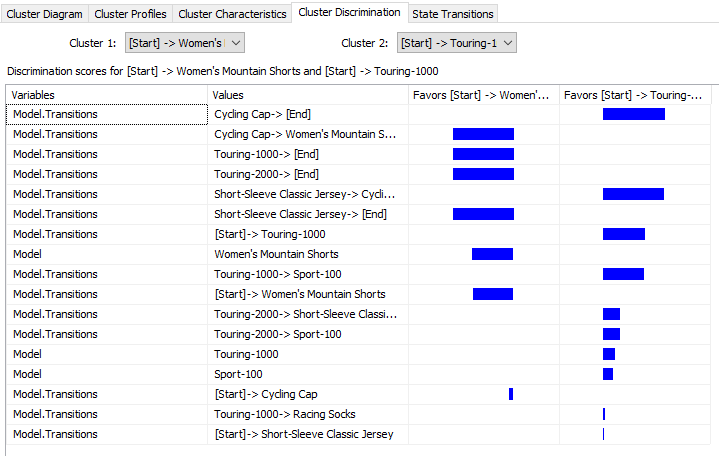
The following screenshot shows the cluster characteristics for a selected cluster.



In this cluster, HL Mountain Tire, LL Mountain Tire and Road-750 model are at the start of the sequence. Further, this cluster has a Sport-100 model predominantly.

### Cluster Discrimination

As always we need to compare between clusters to find out what are differences between the clusters. In the Cluster Discrimination, you can either view differences between two clusters or you can verify the differences between a cluster to all the other rest of the data.



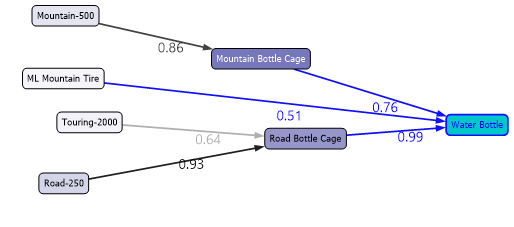
The above screenshot shows that in the first cluster more favors towards the Cycling Cap -> Women’s Mountains Shorts while the second cluster favors ending the sequence with Cycling Cap.

### State Transitions

State Transactions view is the only view that is new to the Microsoft Sequence Clustering technique. This view is created by using the Hidden Markov Model.

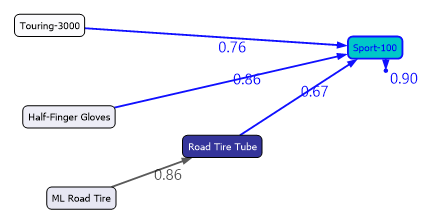
Let us analyze a few cases of state transactions from different clusters.

In the following state transition diagram, it indicates that the customer buys a water bottle, not as the first choice.

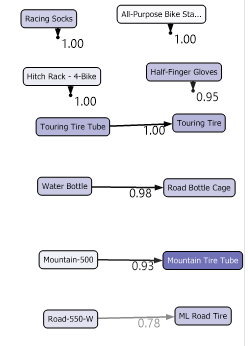


Above screenshot shows that people who buy Road Bottle Cage, there is 0.99 probability of buying a Water Bottle. Similarly, if you are buying Road-250 that there is a probability of 0.93 that buying Road Bottle Cage.

The following screenshot shows another case. In this case, people who are buying Sport-100, there is a chance that they will buy the same item again to a probability of 0.90.



The following screenshot has a few more scenarios.

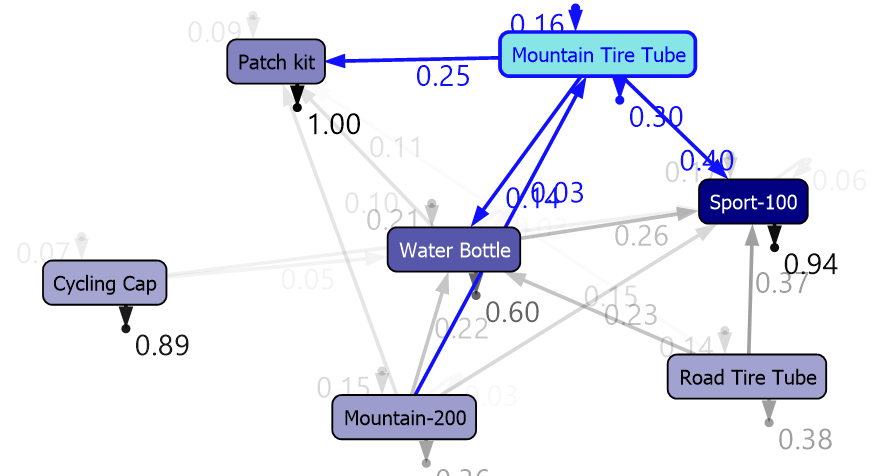


In this cluster, it is evident that customers are buying the same item again. Further, if your client is buying a Touring Tire Tube, there is a 100% probability that he will buy a Touring Tire.

## Model Parameters

In Microsoft Sequence Regression there are two important parameters. Those two parameters are CLUSTER\_COUNT and MAXIMUM\_SEQUENCE\_STATES. During the example, we got 15 clusters which are difficult to manage, so we can reduce the cluster count to a manageable number such as 5-8. MAXIMUM\_SEQUENCE\_STATES define the number of states. The number of states should be around 20-30 maximum. If you are setting this value to a number greater than 100, your model will be meaningless. The default value of this parameter is 64.

Let us set CLUSTER\_COUNT to 5 and MAXIMUM\_SEQUENCE\_STATES to 8 and reprocess the Microsoft Sequence Cluster Model. Following is the Markov Model for the entire population.



You can see that it is clearer than the previous model.

## Logistic Regression

We won’t be doing another article on Microsoft Logistic Regression as it is another variant for [Microsoft Neural Network](https://www.sqlshack.com/implement-artificial-neural-networks-anns-in-sql-server/). The only difference is that, in the Logistic Regression, there won’t be any hidden layer as we discussed in the last article.

## Conclusion

This is the last algorithm of the Microsoft SQL Server family. Microsoft Sequence Clustering in SQL Server is the combination of sequence and clustering techniques. In this technique, common sequences are clustered.

# [SQLShack](https://www.sqlshack.com/)

[Skip to content](https://www.sqlshack.com/measuring-the-accuracy-in-data-mining-in-sql-server/#content)

Top of Form

 [SQL Server training](https://www.sqlshack.com/sql-server-training/)

 [Español](https://www.sqlshack.com/es/)

Bottom of Form

# Measuring the Accuracy in Data Mining in SQL Server

April 29, 2020 by [Dinesh Asanka](https://www.sqlshack.com/author/dinesh-asanka/)

In this article, we will be discussing measuring Accuracy in Data Mining in SQL Server. We have discussed all the Data mining techniques that are available in SQL Server in a series of articles. The discussed techniques were [Naïve Bayes](https://www.sqlshack.com/naive-bayes-prediction-in-sql-server/), [Decision Trees](https://www.sqlshack.com/microsoft-decision-trees-in-sql-server/), [Time Series](https://www.sqlshack.com/microsoft-time-series-in-sql-server/), [Association Rules](https://www.sqlshack.com/the-association-rule-mining-in-sql-server/), [Clustering](https://www.sqlshack.com/microsoft-clustering-in-sql-server/), [Linear Regression](https://www.sqlshack.com/microsoft-linear-regression-in-sql-server/), [Neural Network](https://www.sqlshack.com/implement-artificial-neural-networks-anns-in-sql-server/), [Sequence Clustering](https://www.sqlshack.com/implementing-sequence-clustering-in-sql-server/). Data mining is a predicting technique using the existing pattern. It is obvious that we won’t be able to predict 100% accurately. However, since we are using data mining outcomes for better business decisions, the result should have better accuracy. If the accuracy is very low, we tend not to use those data mining models. Therefore, it is essential to find out how accurate your data mining models are.

## Accuracy in Classification Models

Out of the nine data mining models in SQL Server, three of them can be considered as classification models. The Classification models are Naïve Bayes Decision Trees, Neural Network. Though the logistic regression is a regression technique, that can be used for a classification problem as well. Since you have four models as a solution for the classification problem, we need to look at which algorithm should be selected to use. Obviously, you need to select the most accurate data mining model. To evaluate which algorithm to use, an accuracy test should be done.

Let us create simple four models using Naïve Bayes, Decision Trees, Logistic Regression, and Neural Network algorithms for measuring Accuracy in Data Mining.

## Setting up the Data Mining Project

First, as we did during the previous article, we need to create a connection to the database, AdventureWorksDW after creating an SSAS project using SQL Server Data Tools.

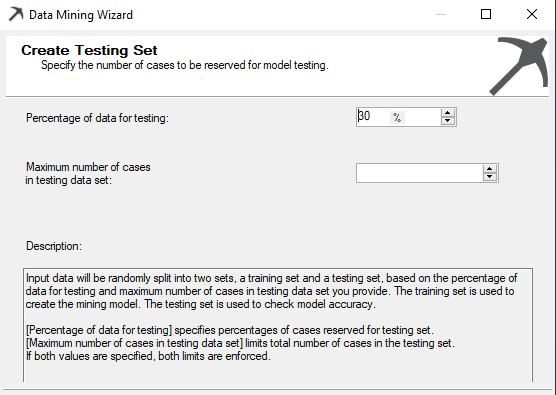
Next, we need to create the data source view and we need to add, **vTargetMail** view to the data source view.

Then, we will be creating a mining model choosing the Decision Tree algorithm and we will add the rest of the three algorithms later.

During the Data Mining project creation, Create a Testing Data Set is an important option for accuracy. This screen was ignored in the previous articles but it plays an important role during the Accuracy Measuring in data mining.

During the model building, we need two data sets. One is to train and the other is to test data set. The Train set will be used to build the model and the test data set will be used to evaluate the built model. This test data set will be used to measure the accuracy and other matrices.

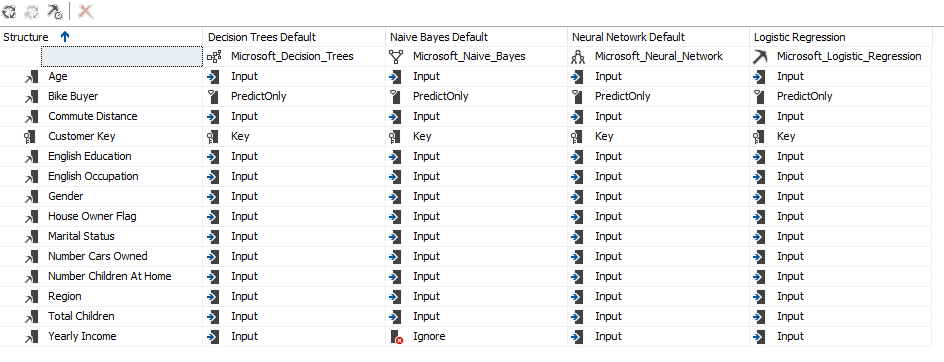
In the following screenshot, allow selecting data volume to the test data set.



There are two options to create a test data set. The first option is to define the percentage value for the test data set. Apart from the percentage setting, there is an option to set the number of cases for the test data set. If both values are specified in the above screen, both limits are enforced.

Typically, this is set to 30% meaning that 70% of data will be used to build the model while 30% is used to evaluate the model.

After one model is built, the rest of the techniques are added to the data mining model and the final model can be viewed as the following screenshot.

[](https://www.sqlshack.com/wp-content/uploads/2020/04/four-models-are-prepared-to-measure-the-accuracy-i.png)

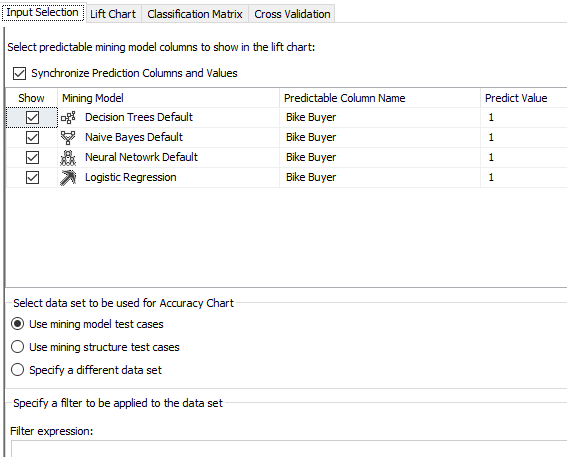
Since Yearly Income is a continuous variable, Microsoft Naïve Bayes has ignored that input variable as Microsoft naïve Bayes can have only discrete inputs. Apart from that ignored variable exception, everything else is the same across all the four algorithms.

Next, we will look at the Mining Accuracy Chart.

## Mining Accuracy Chart

In the previous articles, we looked at Mining Structure, Mining Models, Mining Model Viewer and Mining Model Predictions tabs. The only tab we have not discussed so far is the Mining Accuracy Chart tab. With this article, we will discuss the Mining Accuracy Chart tab in detail as accuracy Measuring in Data Mining.

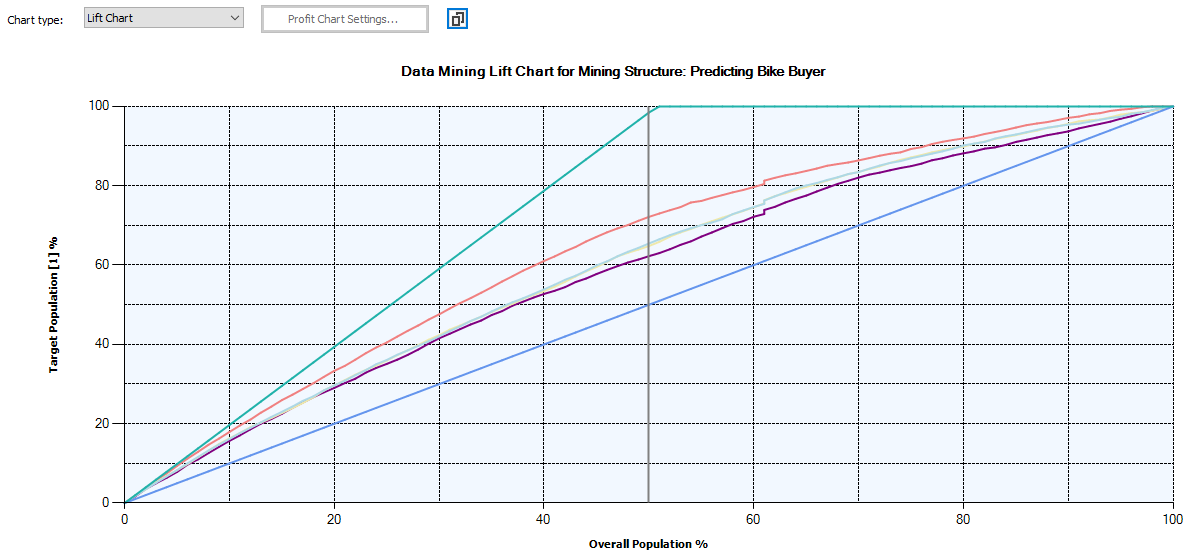
In the Input Selection, you can choose which models to evaluate. Since we used these models to predict Bike Buyer, Predictable Column Name is Bike Buyer and the Predict value is 1.



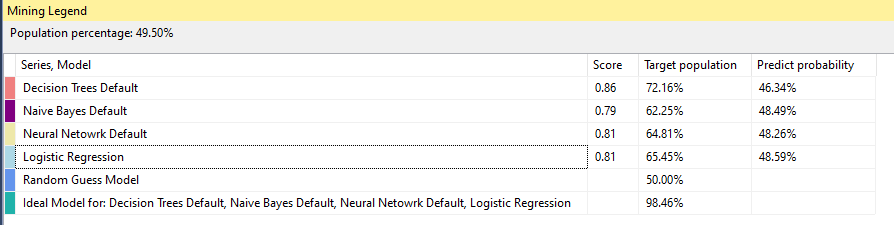
Next is to choose, test data for the Accuracy Chart. The first two options allow selecting the randomly selected test data set. However, you have the option of choosing a different data set for the evaluation purposes by using the Specify a different data set. In this option, you can create a filter expression as well. Let us assume that you need to test the built model for the customers with age over 40 years. You will not be able to do this by using any of the first two options. In the third option, you can select the data set and set the filter so that the evaluation is done only for the filtered data set.

### Lift Chart

A life chart will be used to evaluate the effectiveness of different data mining models. Following are the lift charts for different four models, random model, and the ideal model.

[](https://www.sqlshack.com/wp-content/uploads/2020/04/life-chart-in-accuracy-measuring-in-data-warehouse.png)

The following screenshot is the legend for the above chart.

[](https://www.sqlshack.com/wp-content/uploads/2020/04/mining-legend.png)

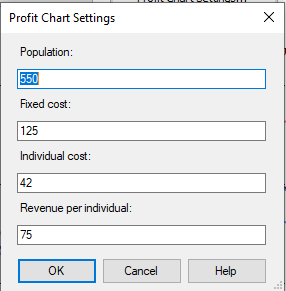
Since Neural Network and Logistic Regression have similar results, it is difficult to distinguish them in the chart.

The random model is 50% as we have two probable, buying a bike or not. If we do not use any data mining models, random is the model that will be automatically selected. Out of the four models, the best model is the model which is close to the Ideal model or the model which is away from the random models. This means that the best model for the above data set is Decision Trees. Logistic Regression, Neural Network and Naïve Bayes models are other models in the order of the effectiveness.

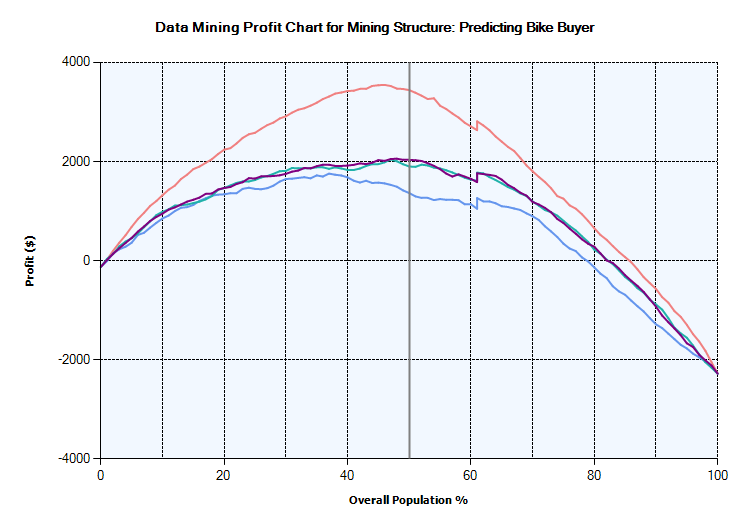
## Profit Chart

The Profit chart is somewhat unique in Microsoft tools. We use data mining to maximize profit.

Let us assume, we are looking at a promotion to improve the bike buyers. For a marketing campaign, there are four parameters, population. Fixed Cost, Individual cost and the expected revenue. Those parameters can be entered in the following screen. This is available after the selection of the Profit Chart.



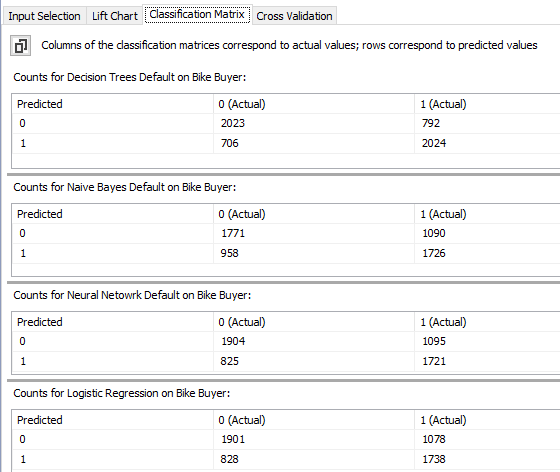
After the above data was entered, the following Profit chart can be observed.



Even the profit chart indicates that the Decision Tree algorithm is better than other techniques. In the decision tree, 45% of the population would make the maximum profit of 3,500 $. This means it is better to target only 45% of possible buyers who have the highest probability of buying. It is important to choose the correct parameters for the profit chart.

## Classification Matrix

Most of the time, the Classification Matrix is known as the Confusion Matrix. This is the most common matrix used to evaluate the effectiveness of the data mining models.



Let us look at the Decision tree classification matrix. In that model, there are 2023 cases where are actually not bike buyers and those are predicted as same. Similarly, there are 2024 actual bike buyers and which are predicted the same. Those are correct predictions. However, there are 706 cases which actually are not bike buyers but the decision tree model predicted them as possible bike buyers. 792 cases are another way around.

However, there are few other parameters that are derived from the above classification matrix. The above values are defined as follows.

|  |  |  |
| --- | --- | --- |
| **Predicted** | **0 (Actual)** | **1 (Actual)** |
| 0 | True Negative (TN) | False Negative (FN) |
| 1 | False Positive (FP) | True Positive (TP) |

The following are the basic measures that can be derived from the classification matrix.

|  |  |  |
| --- | --- | --- |
| **Measure** | **Formula** | **Description** |
| Accuracy | table formula 1 | What is the percentage of correct predictions? |
| Precision | table formula 2 | Percentage of the correct cases out of the selected cases |
| Recall | table formula 3 | Percentage of the correct cases out of the actual correct cases. |
| F1 Score | table formula 4 | Harmonic means of Precision and Recall |
| Matthews correlation coefficient (MCC) | [table formula 5](https://www.sqlshack.com/wp-content/uploads/2020/04/f5.png) | Balanced measure even if the class sizes are different. |

Let us look at different evaluation parameters for the different algorithms.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Accuracy | Precision | Recall | F1 Score | MCC |
| Decision Tree | 72.98 | 74.14 | 71.88 | 72.99 | 45.30 |
| Naïve Bayes | 63.07 | 64.31 | 61.29 | 62.76 | 25.58 |
| Neural Network | 65.37 | 67.60 | 61.12 | 64.19 | 29.46 |
| Logistic Regression | 65.63 | 67.73 | 61.72 | 64.59 | 30.03 |

In most of the tools such as Weka, Azure Machine learning has calculated most of these values but not in SQL Server.

After looking at the above values, from the classification matrices, the decision tree is the best algorithm from the available four. We discussed during our previous articles, that there are different mining model parameters relevant to different algorithms. To verify whether the parameters have improved the model, we can utilize the same techniques.

## Conclusion

Measuring Accuracy in data mining is an important aspect of data Mining. Since there are a few options to choose the necessary algorithms, it is essential to choose what is the best algorithms. In this, a Lift chart can be used as a visual tool to find a better model. Further, the Profit chart will be helpful to find out what is the optimum number of cases that can be chosen. Classification Matrix or the confusion matrix is used to derive various classification accuracy matrices.