



Data Mining Tutorial

Seth Paul

Jamie MacLennan

Zhaohui Tang

Scott Oveson

Microsoft Corporation

June 2005

Abstract: Microsoft® SQL Server™ 2005 provides an integrated environment for creating and working with data mining models. This tutorial uses four scenarios, targeted mailing, forecasting, market basket, and sequence clustering, to demonstrate how to use the mining model algorithms, mining model viewers, and data mining tools that are included in this release of SQL Server.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2003 Microsoft Corporation. All rights reserved.

Microsoft is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Contents

Data Mining Tutorial.....	1
Contents	i
Introduction	1
Mining Model Algorithms	6
Microsoft Decision Trees.....	6
Microsoft Clustering	6
Microsoft Naïve Bayes.....	7
Microsoft Time Series	7
Microsoft Association	7
Microsoft Sequence Clustering	8
Microsoft Neural Network	8
Microsoft Linear Regression	9
Microsoft Logistic Regression	9
Working Through the Tutorial	10
Preparing the SQL Server Database.....	10
Preparing the Analysis Services Database.....	11
Creating an Analysis Services Project.....	12
Creating a Data Source	12
Creating a Data Source View	13
Editing the Data Source View.....	15
Building and Working with the Mining Models.....	16
Targeted Mailing	17
Forecasting	47
Market Basket	56
Sequence Clustering	64

Introduction

The data mining tutorial is designed to walk you through the process of creating data mining models in Microsoft SQL Server 2005. The data mining algorithms and tools in SQL Server 2005 make it easy to build a comprehensive solution for a variety of projects, including market basket analysis, forecasting analysis, and targeted mailing analysis. The scenarios for these solutions are explained in greater detail later in the tutorial.

The most visible components in SQL Server 2005 are the workspaces that you use to create and work with data mining models. The online analytical processing (OLAP) and data mining tools are consolidated into two working environments: Business Intelligence Development Studio and SQL Server Management Studio. Using Business Intelligence Development Studio, you can develop an Analysis Services project disconnected from the server. When the project is ready, you can deploy it to the server. You can also work directly against the server. The main function of SQL Server Management Studio is to manage the server. Each environment is described in more detail later in this introduction. For more information on choosing between the two environments, see "Choosing Between SQL Server Management Studio and Business Intelligence Development Studio" in SQL Server Books Online.

All of the data mining tools exist in the data mining editor. Using the editor you can manage mining models, create new models, view models, compare models, and create predictions based on existing models.

After you build a mining model, you will want to explore it, looking for interesting patterns and rules. Each mining model viewer in the editor is customized to explore models built with a specific algorithm. For more information about the viewers, see "Viewing a Data Mining Model" in SQL Server Books Online.

Often your project will contain several mining models, so before you can use a model to create predictions, you need to be able to determine which model is the most accurate. For this reason, the editor contains a model comparison tool called the Mining Accuracy Chart tab. Using this tool you can compare the predictive accuracy of your models and determine the best model.

To create predictions, you will use the Data Mining Extensions (DMX) language. DMX extends SQL, containing commands to create, modify, and predict against mining models. For more information about DMX, see "Data Mining Extensions (DMX) Reference" in SQL Server Books Online. Because creating a prediction can be complicated, the data mining editor contains a tool called Prediction Query Builder, which allows you to build queries using a graphical interface. You can also view the DMX code that is generated by the query builder.

Just as important as the tools that you use to work with and create data mining models are the mechanics by which they are created. The key to creating a mining model is the data mining algorithm. The algorithm finds patterns in the data that you pass it, and it translates them into a mining model — it is the engine behind the process. SQL Server 2005 includes nine algorithms:

- Microsoft Decision Trees
- Microsoft Clustering
- Microsoft Naïve Bayes
- Microsoft Sequence Clustering
- Microsoft Time Series
- Microsoft Association
- Microsoft Neural Network
- Microsoft Linear Regression
- Microsoft Logistic Regression

Using a combination of these nine algorithms, you can create solutions to common business problems. These algorithms are described in more detail later in this tutorial.

Some of the most important steps in creating a data mining solution are consolidating, cleaning, and preparing the data to be used to create the mining models. SQL Server 2005 includes the Data Transformation Services (DTS) working environment, which contains tools that you can use to clean, validate, and prepare your data. For more information on using DTS in conjunction with a data mining solution, see "DTS Data Mining Tasks and Transformations" in SQL Server Books Online.

In order to demonstrate the SQL Server data mining features, this tutorial uses a new sample database called **AdventureWorksDW**. The database is included with SQL Server 2005, and it supports OLAP and data mining functionality. In order to make the sample database available, you need to select the sample database at the installation time in the "Advanced" dialog for component selection.

The audience for this tutorial is business analysts, developers, and database administrators who have used data mining tools before and are familiar with data mining concepts. If you are new to data mining, download "Preparing and Mining Data with Microsoft SQL Server 2000 and Analysis Services"

(msdn.microsoft.com/library/default.asp?url=/servers/books/sqlserver/mining.asp).

Adventure Works

AdventureWorksDW is based on a fictional bicycle manufacturing company named Adventure Works Cycles. Adventure Works produces and distributes metal and composite bicycles to North American, European, and Asian commercial markets. The base of operations is located in Bothell, Washington with 500 employees, and several regional sales teams are located throughout their market base.

Adventure Works sells products wholesale to specialty shops and to individuals through the Internet. For the data mining exercises, you will work with the

AdventureWorksDW Internet sales tables, which contain realistic patterns that work well for data mining exercises.

For more information on Adventure Works Cycles see "Sample Databases and Business Scenarios" in SQL Server Books Online.

Database Details

The Internet sales schema contains information about 9,242 customers. These customers live in six countries, which are combined into three regions:

- North America (83%)
- Europe (12%)
- Australia (7%)

The database contains data for three fiscal years: 2002, 2003, and 2004.

The products in the database are broken down by subcategory, model, and product.

Business Intelligence Development Studio

Business Intelligence Development Studio is a set of tools designed for creating business intelligence projects. Because Business Intelligence Development Studio was created as an IDE environment in which you can create a complete solution, you work disconnected from the server. You can change your data mining objects as much as you want, but the changes are not reflected on the server until after you deploy the project.

Working in an IDE is beneficial for the following reasons:

- You have powerful customization tools available to configure Business Intelligence Development Studio to suit your needs.
- You can integrate your Analysis Services project with a variety of other business intelligence projects encapsulating your entire solution into a single view.
- Full source control integration enables your entire team to collaborate in creating a complete business intelligence solution.

The Analysis Services project is the entry point for a business intelligence solution. An Analysis Services project encapsulates mining models and OLAP cubes, along with supplemental objects that make up the Analysis Services database. From Business Intelligence Development Studio, you can create and edit Analysis Services objects within a project and deploy the project to the appropriate Analysis Services server or servers.

If you are working with an existing Analysis Services project, you can also use Business Intelligence Development Studio to work connected the server. In this way, changes are reflected directly on the server without having to deploy the solution.

SQL Server Management Studio

SQL Server Management Studio is a collection of administrative and scripting tools for working with Microsoft SQL Server components. This workspace differs from Business Intelligence Development Studio in that you are working in a connected environment where actions are propagated to the server as soon as you save your work.

After the data has been cleaned and prepared for data mining, most of the tasks associated with creating a data mining solution are performed within Business Intelligence Development Studio. Using the Business Intelligence Development Studio tools, you develop and test the data mining solution, using an iterative process to determine which models work best for a given situation. When the developer is satisfied with the solution, it is deployed to an Analysis Services server. From this point, the focus shifts from development to maintenance and use, and thus SQL Server Management Studio. Using SQL Server Management Studio, you can administer your database and perform some of the same functions as in Business Intelligence Development Studio, such as viewing, and creating predictions from mining models.

Data Transformation Services

Data Transformation Services (DTS) comprises the Extract, Transform, and Load (ETL) tools in SQL Server 2005. These tools can be used to perform some of the most important tasks in data mining: cleaning and preparing the data for model creation. In data mining, you typically perform repetitive data transformations to clean the data before using the data to train a mining model. Using the tasks and transformations in DTS, you can combine data preparation and model creation into a single DTS package.

DTS also provides DTS Designer to help you easily build and run packages containing all of the tasks and transformations. Using DTS Designer, you can deploy the packages to a server and run them on a regularly scheduled basis. This is useful if, for example, you collect data weekly and want to perform the same cleaning transformations each time in an automated fashion.

You can work with a Data Transformation project and an Analysis Services project together as part of a business intelligence solution, by adding each project to a solution in Business Intelligence Development Studio.

Mining Model Algorithms

Data mining algorithms are the foundation from which mining models are created. The variety of algorithms included in SQL Server 2005 allows you to perform many types of analysis. For more specific information about the algorithms and how they can be adjusted using parameters, see "Data Mining Algorithms" in SQL Server Books Online.

Microsoft Decision Trees

The Microsoft Decision Trees algorithm supports both classification and regression and it works well for predictive modeling. Using the algorithm, you can predict both discrete and continuous attributes.

In building a model, the algorithm examines how each input attribute in the dataset affects the result of the predicted attribute, and then it uses the input attributes with the strongest relationship to create a series of splits, called nodes. As new nodes are added to the model, a tree structure begins to form. The top node of the tree describes the breakdown of the predicted attribute over the overall population. Each additional node is created based on the distribution of states of the predicted attribute as compared to the input attributes. If an input attribute is seen to cause the predicted attribute to favor one state over another, a new node is added to the model. The model continues to grow until none of the remaining attributes create a split that provides an improved prediction over the existing node. The model seeks to find a combination of attributes and their states that creates a disproportionate distribution of states in the predicted attribute, therefore allowing you to predict the outcome of the predicted attribute.

Microsoft Clustering

The Microsoft Clustering algorithm uses iterative techniques to group records from a dataset into clusters containing similar characteristics. Using these clusters, you can explore the data, learning more about the relationships that exist, which may not be easy to derive logically through casual observation. Additionally, you can create predictions from the clustering model created by the algorithm. For example, consider a group of people who live in the same neighborhood, drive the same kind of car, eat the same kind of food, and buy a similar version of a product. This is a cluster of data. Another cluster may include people who go to the same restaurants, have similar salaries, and vacation twice a year outside the country. Observing how these clusters are distributed, you can better understand how the records in a dataset interact, as well as how that interaction affects the outcome of a predicted attribute.

Microsoft Naïve Bayes

The Microsoft Naïve Bayes algorithm quickly builds mining models that can be used for classification and prediction. It calculates probabilities for each possible state of the input attribute, given each state of the predictable attribute, which can later be used to predict an outcome of the predicted attribute based on the known input attributes. The probabilities used to generate the model are calculated and stored during the processing of the cube. The algorithm supports only discrete or discretized attributes, and it considers all input attributes to be independent. The Microsoft Naïve Bayes algorithm produces a simple mining model that can be considered a starting point in the data mining process. Because most of the calculations used in creating the model are generated during cube processing, results are returned quickly. This makes the model a good option for exploring the data and for discovering how various input attributes are distributed in the different states of the predicted attribute.

Microsoft Time Series

The Microsoft Time Series algorithm creates models that can be used to predict continuous variables over time from both OLAP and relational data sources. For example, you can use the Microsoft Time Series algorithm to predict sales and profits based on the historical data in a cube.

Using the algorithm, you can choose one or more variables to predict, but they must be continuous. You can have only one case series for each model. The case series identifies the location in a series, such as the date when looking at sales over a length of several months or years.

A case may contain a set of variables (for example, sales at different stores). The Microsoft Time Series algorithm can use cross-variable correlations in its predictions. For example, prior sales at one store may be useful in predicting current sales at another store.

Microsoft Association

The Microsoft Association algorithm is specifically designed for use in market basket analyses. The algorithm considers each attribute/value pair (such as product/bicycle) as an item. An itemset is a combination of items in a single transaction. The algorithm scans through the dataset trying to find itemsets that tend to appear in many transactions. The SUPPORT parameter defines how many transactions the itemset must appear in before it is considered significant. For example, a frequent itemset may contain {Gender="Male", Marital Status = "Married", Age="30-35"}. Each itemset has a size, which is number of items it contains. In this case, the size is 3.

Often association models work against datasets containing nested tables, such as a customer list followed by a nested purchases table. If a nested table exists in the dataset, each nested key (such as a product in the purchases table) is considered an item.

The Microsoft Association algorithm also finds rules associated with itemsets. A rule in an association model looks like $A, B \Rightarrow C$ (associated with a probability of occurring), where A, B, C are all frequent itemsets. The ' \Rightarrow ' implies that C is predicted by A and B. The probability threshold is a parameter that determines the minimum probability before a rule can be considered. The probability is also called "confidence" in data mining literature.

Association models are also useful for cross sell or collaborative filtering. For example, you can use an association model to predict items a user may want to purchase based on other items in their basket.

Microsoft Sequence Clustering

The Microsoft Sequence Clustering algorithm analyzes sequence-oriented data that contains discrete-valued series. Usually the sequence attribute in the series holds a set of events with a specific order (such as a click path). By analyzing the transition between states of the sequence, the algorithm can predict future states in related sequences.

The Microsoft Sequence Clustering algorithm is a hybrid of sequence and clustering algorithms. The algorithm groups multiple cases with sequence attributes into segments based on similarities of these sequences. A typical usage scenario for this algorithm is Web customer analysis for a portal site. A portal Web site has a set of affiliated domains such as News, Weather, Money, Mail, and Sport. Each Web customer is associated with a sequence of Web clicks on these domains. The Microsoft Sequence Clustering algorithm can group these Web customers into more-or-less homogenous groups based on their navigations patterns. These groups can then be visualized, providing a detailed understanding of how customers are using the site.

Microsoft Neural Network

In Microsoft SQL Server 2005 Analysis Services, the Microsoft Neural Network algorithm creates classification and regression mining models by constructing a multilayer perceptron network of neurons. Similar to the Microsoft Decision Trees algorithm provider, given each state of the predictable attribute, the algorithm calculates probabilities for each possible state of the input attribute. The algorithm provider processes the entire set of cases, iteratively comparing the predicted classification of the cases with the known actual classification of the cases. The errors from the initial classification of the first iteration of the entire set of cases is fed back into the network, and used to modify the network's performance for the next iteration, and so on. You can later use these probabilities to predict an outcome of the predicted attribute, based on the input attributes. One of the primary differences between this algorithm and the Microsoft Decision Trees algorithm, however, is that its learning process is to optimize network parameters toward minimizing the error while the Microsoft Decision Trees algorithm splits rules in order to maximize information gain. The algorithm supports the prediction of both discrete and continuous attributes.

Microsoft Linear Regression

The Microsoft Linear Regression algorithm is a particular configuration of the Microsoft Decision Trees algorithm, obtained by disabling splits (the whole regression formula is built in a single root node). The algorithm supports the prediction of continuous attributes.

Microsoft Logistic Regression

The Microsoft Logistic Regression algorithm is a particular configuration of the Microsoft Neural Network algorithm, obtained by eliminating the hidden layer. The algorithm supports the prediction of both discrete and continuous attributes.

Working Through the Tutorial

Throughout this tutorial you will work in Business Intelligence Development Studio (as depicted in Figure 1). For more information about working in Business Intelligence Development Studio, see "Using SQL Server Management Studio" in SQL Server Books Online.

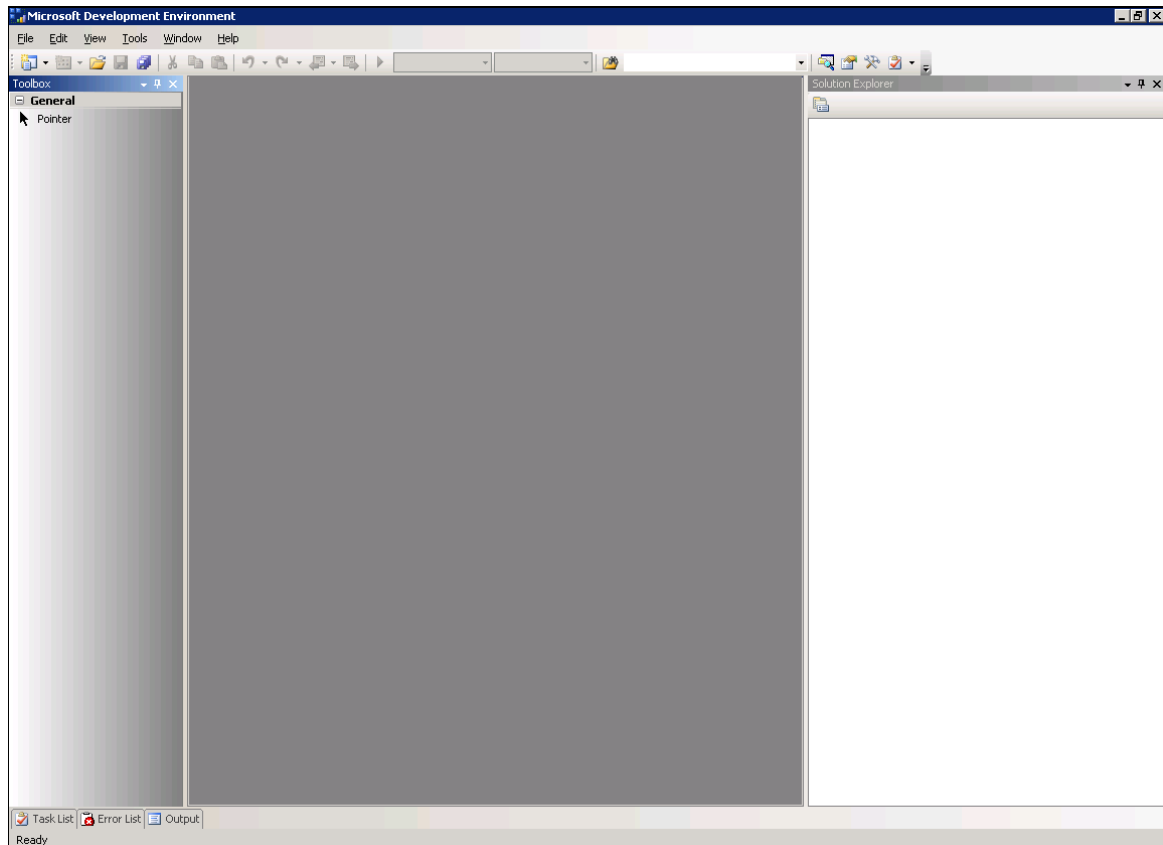


Figure 1 Business Intelligence Studio

The tutorial is broken up into three sections: Preparing the SQL Server Database, Preparing the Analysis Services Database, and Building and Working with the Mining Models.

Preparing the SQL Server Database

The **AdventureWorksDW** database, which is the basis for this tutorial, is installed with SQL Server (not by default, but as an option at installation time) and already contains views that will be used to create the mining models. If it was not installed at the installation time, you can add it by selecting **Change** button from **Control Panel → Add/Remove Programs → Microsoft SQL Server 2005**. Look for **AdventureWorksDW Sample Data Warehouse** under **Books online and Samples of Workstation Components**.

Preparing the Analysis Services Database

Before you begin to create and work with mining models, you must perform the following tasks:

1. Create a new Analysis Services project
2. Create a data source.
3. Create a data source view.

Creating an Analysis Services Project

Each Analysis Services project defines the schema for the objects in a single Analysis Services database. The Analysis Services database is defined by the mining models, OLAP cubes, and supplemental objects that it contains. For more information about Analysis Services projects, see "Creating an Analysis Services Project in Business Intelligence Development Studio" in SQL Server Books Online.

To create an Analysis Services project

1. Open Business Intelligence Development Studio.
2. Select **New** and **Project** from the **File** menu.
3. Select Analysis Services Project as the type for the new project and name it **AdventureWorks**.
4. Click **Ok**.

The new project opens in Business Intelligence Development Studio.

Creating a Data Source

A data source is a data connection that is saved and managed within your project and deployed to your Analysis Services database. It contains the server name and database where your source data resides, as well as any other required connection properties.

To create a data source

1. Right-click the **Data Source** project item in Solution Explorer and select **New Data Source**.
2. On the Welcome page, click **Next**.
3. Click **New** to add a connection to the **AdventureWorksDW** database.
4. The **Connection Manager** dialog box opens. In the **Server name** drop-down box, select the server where **AdventureWorksDW** is hosted (for example, localhost), enter your credentials, and then in the **Select the database on the server** drop-down box select the **AdventureWorksDW** database.
5. Click **OK** to close the **Connection Manager** dialog box.
6. Click **Next**.
7. By default the data source is named Adventure Works DW. Click **Finish**

The new data source, Adventure Works DW, appears in the Data Sources folder in Solution Explorer.

Creating a Data Source View

A data source view provides an abstraction of the data source, enabling you to modify the structure of the data to make it more relevant to your project. Using data source views, you can select only the tables that relate to your particular project, establish relationships between tables, and add calculated columns and named views without modifying the original data source.

For more information, see "Working with Data Source Views" in SQL Server Books Online.

To create a data source view

1. In Solution Explorer, right-click **Data Source View**, and then click **New Data Source View**.
2. On the Welcome page, click **Next**.
3. The **Adventure Works DW** data source you created in the last step is selected by default in the **Relational data sources** window. Click **Next**.
4. If you want to create a new data source, click **New Data Source** to launch the Data Source Wizard.
5. Select the tables in the following list and click the right arrow button to include them in the new data source view:
6. **vAssocSeqLineItems**
7. **vAssocSeqOrders**
8. **vTargetMail**
9. **vTimeSeries**
10. Click **Next**.
11. By default the data source view is named Adventure Works DW. Click **Finish**.

Data Source View Editor opens to display the Adventure Works DW data source view, as shown in Figure 2. Solution Explorer is also updated to include the new data source view. You can now modify the data source view to better work with the data.

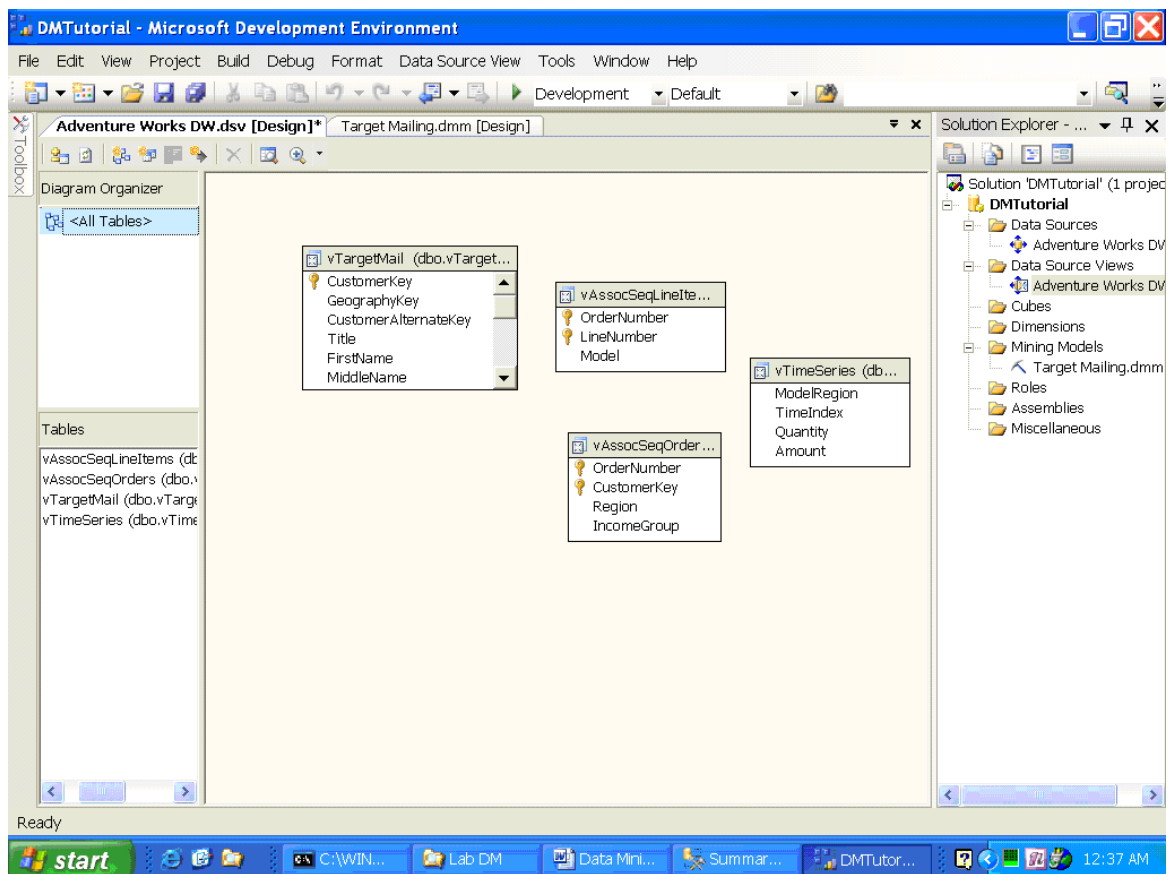


Figure 2 Adventure Works DW data source view

Editing the Data Source View

Using Data Source View Editor, you can make changes to the way you see the data in the data source. For example, you can change the name of any object to something that may be more relevant to the project. The name in the original data source is not modified, but you can refer to the object by this friendly name in your project.

In order to create the market basket and sequence clustering scenarios, you need to create a new many-to-one relationship between **vAssocSeqOrders** and **vAssocSeqLineItems**. Using this relationship you can make **vAssocSeqLineItems** a nested table of **vAssocSeqOrders** for creating the models.

To create a new relationship

1. In the data source view, select **OrderNumber** from the **vAssocSeqLineItems** table.
2. Drag the selected column into the **vAssocSeqOrders** table, and place it on the **OrderNumber** column.

A new many-to-one relationship exists between **vAssocSeqOrders** and **vAssocSeqLineItems**.

Building and Working with the Mining Models

The data mining editor (shown in Figure 4) contains all of the tools and viewers that you will use to build and work with the mining models. For more information about the data mining editor, see "Using the Data Mining Tools" in SQL Server Books Online.

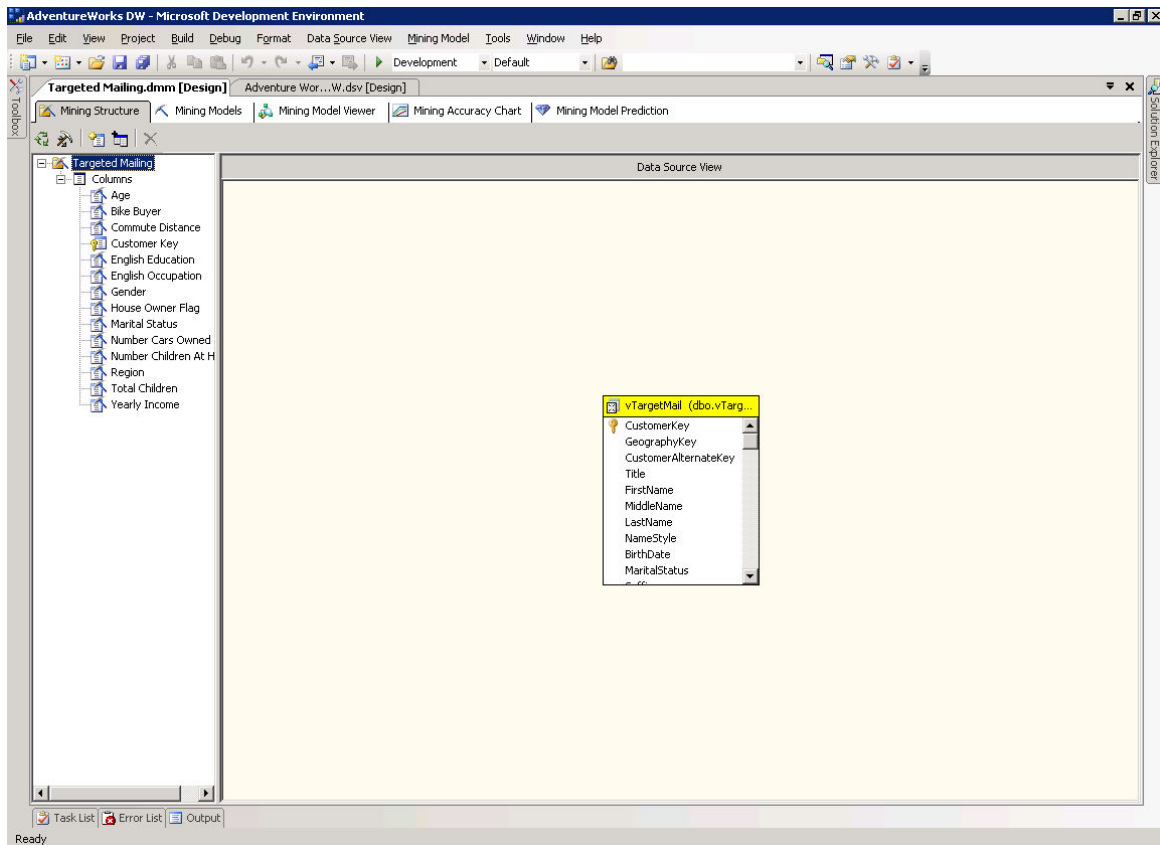


Figure 4 Data mining editor

Throughout this tutorial you will work through the following scenarios:

- Targeted mailing
- Forecasting
- Market basket
- Sequence clustering

In the targeted mailing scenario, you will build the models, compare their predictive capabilities using the Mining Accuracy Chart view, and create predictions using Prediction Query Builder. In the other scenarios, you will build and explore the models.

Targeted Mailing

The marketing department of Adventure Works is interested in increasing sales by targeting specific customers for a mailing campaign. By investigating the attributes of known customers, they want to discover some kind of pattern that can be applied to potential customers, which can then be used to predict who is more likely to purchase a product from Adventure Works.

Additionally, the marketing department wants to find any logical groupings of customers already in their database. For example, a grouping may contain customers with similar buying patterns and demographics.

Adventure Works contains a list of past customers and a list of potential customers.

Upon completion of this task, the marketing department will have the following:

- A set of mining models that will be able to suggest the most likely customers from a list of potential customers
- A clustering of their current customers

In order to complete the scenario, you will use the Microsoft Naïve Bayes, Microsoft Decision Trees, and Microsoft Clustering algorithms. The scenario consists of five tasks:

- Create the mining model structure.
- Create the mining models.
- Explore the mining models.
- Test the accuracy of the mining models.
- Create predictions from the mining models.

Create a Targeted Mailing Mining Model Structure Using the Wizard

The first step is to use the Mining Model Wizard to create a new mining structure. The Mining Model Wizard also creates an initial mining model based on the Microsoft Decision Trees algorithm.

To create the targeted mailing mining structure

3. In Solution Explorer, right-click **Mining Structures**, and then click **New Mining Structure**.

The Mining Model Wizard opens.

4. On the Welcome page, click **Next**.
5. Click **From existing relational database or data warehouse**, and then click **Next**.
6. Under **Which data mining technique do you want to use?**, click **Microsoft Decision Trees**.

You will create several models based on this initial structure, but the initial model is based on the Microsoft Decision Trees algorithm.

7. Click **Next**.

By default the Adventure Works DW is selected in the **Select Data Source View** window. You may click **Browse** to view the tables in the data source view inside of the wizard.

8. Click **Next**.
9. Select the **Case** check box next to the **vTargetMail** table, and then click **Next**.
10. Select the **Key** check box next to the **CustomerKey** column.

If the source table from the data source view indicates a key, the Mining Model Wizard automatically chooses that column as a key for the model.

11. Select the **Input** and **Predictable** check boxes next to the **BikeBuyer** column.

This action enables the column for prediction in new datasets. When you indicate that a column is predictable, the **Suggest** button is enabled. Clicking **Suggest** opens the **Suggest Related Column** dialog box, which lists the columns that are most closely related to the predictable column.

The **Suggest Related Columns** dialog box orders the attributes by their correlation with the predictable attribute. Columns with a value higher than 0.05 are automatically selected to be included in the model. If you agree with the suggestion, click **OK**, which marks the selected columns as inputs in the wizard. If you don't agree, you can either modify the suggestion or click **Cancel**.

12. Select the **Input** check boxes next to the columns listed in the following table.

Age	YearlyIncome	Region
CommuteDistance	HouseOwnerFlag	TotalChildren
EnglishEducation	LastName	
EnglishOccupation	MaritalStatus	
FirstName	NumberCarsOwned	
Gender	NumberChildrenAtHome	

You can select multiple columns by using the SHIFT key. Selecting a check box within the selected area specifies the same selection for each column.

13. Click **Next**.

14. In **Specify Columns' Content and Data Type**, click **Detect**.

An algorithm runs that samples numeric data and determines whether the numeric columns contain continuous or discrete values. For example, a column can contain salary information as the actual salary values, which is continuous, or it can contain integers representing encoded salary ranges (1 = < \$25,000; 2 = from \$25,000 to \$50,000, and so on) which is discrete.

15. Click **Next**.

16. In both **Mining Structure Name** and **Mining Model Name**, type *Targeted Mailing*.

17. Click **Finish**.

The data mining editor opens, displaying the mining structure named Targeted Mailing that you just created, as shown in Figure 5.

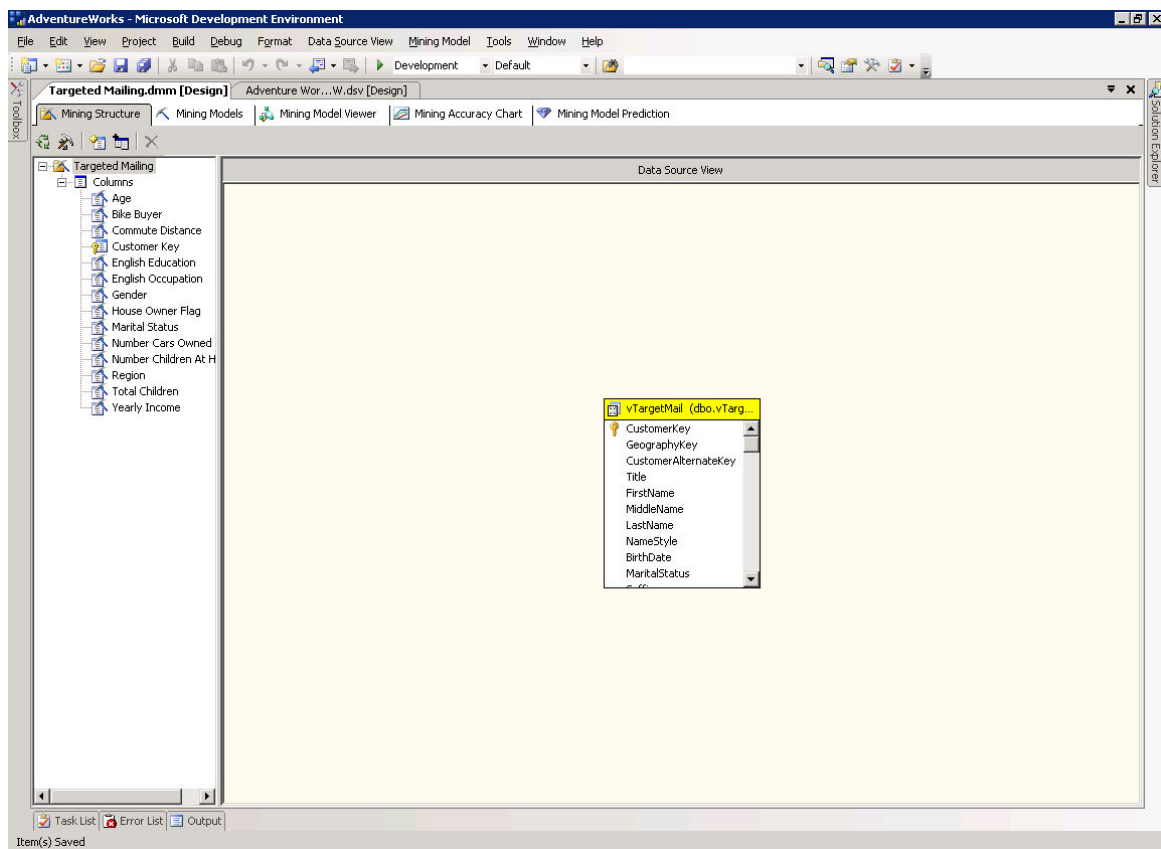


Figure 5 Targeted Mailing mining structure tab

Edit the Mining Models

The initial mining structure only contains a model based on the Microsoft Decision Trees model. In this section, you will define two additional models using the Mining Models tab of the data mining editor: a Microsoft Naïve Bayes model and a Microsoft Clustering model.

To create a Microsoft Clustering model

1. Click the **Mining Models** tab.
2. Right-click **Targeted Mailing** and then click **New Mining Model**.
3. In **Model Name**, type *TM_Clustering*.
4. In **Algorithm Name**, select **Microsoft Clustering**.
5. Click **OK**.

A new model appears in the Mining Models tab. A Microsoft Clustering model can cluster and predict continuous and discrete attributes. You can modify the column usage and properties for the new model.

Setting a column to **Predict** has no effect on the model training; it allows you to select that column in a PREDICTION JOIN query. However, the algorithm ignores columns set to **PredictOnly** when it creates clusters. The statistics for **PredictOnly** columns in a clustering model are determined as a final pass after the clustering operation is complete. This is beneficial if you want to see how an attribute is distributed across clusters created from other attributes, and it can expose deeper correlations.

To create a Microsoft Naïve Bayes model

1. Right-click **Targeted Mailing**, and then click **New Mining Model**.
2. In **Model Name**, type *TM_NaiveBayes*.
3. In **Algorithm Name**, click **Microsoft Naïve Bayes**.
4. Click **OK**.

A dialog box appears with the text explaining that the Microsoft Naïve Bayes algorithm does not support working with the Age, Yearly Income columns, which are continuous, and will be ignored

5. Click **Yes**.

A new model appears in the Mining Models tab. Although you can modify the column usage and properties for all of the models in this tab, in this case you can leave them as they are.

Process the Mining Models

Now that the structure and parameters for the mining models are complete, you can deploy and process the models.

To deploy the project and process the mining models

- Click **F5**.

Depending on what account Analysis Services is running under, you may need to change impersonation information of the data source. To change it, open **Adventure Works DW.ds** from your solution explorer and go to the **Impersonation Information** tab.

The Analysis Services database is deployed to the server and the mining models are processed. If the database has already been deployed to the server, you can just process the mining models using the following process.

To process the mining models

1. On the **Mining Model** menu, select **Process**.

The **Process Mining Structure** dialog box opens showing **Targeted Mailing** in Object list.

2. Click **Run**.

The **Process Progress** dialog box opens, displaying information about model processing. This may take some time, depending on your computer.

3. After processing is complete, click **Close** in both dialog boxes.

Note that processing the mining models may take several minutes depending on your computer configuration.

Exploring the Mining Models

After the models are processed, you can view them using the Mining Model Viewer tab in the data mining editor. Using the **Mining Model** combo box at the top of the tab, you can examine the models in the mining structure.

Microsoft Decision Trees Model

The Mining Model Viewer tab defaults to opening the Targeted Mailing mining model, the first model in the structure. The Tree viewer contains two tabs, **Decision Tree** and **Dependency Network**.

Decision Tree

On the **Decision Tree** tab, you can examine all of the tree models that make up the Targeted_Mailing model. There is one tree model for each predictable attribute in the model, unless feature selection is invoked. Because your model only contains a single predictable attribute, Bike Buyer, there is only one tree to view. If there were more trees, you could use the **Tree** box to choose another tree.

The Tree viewer defaults to only showing the first three levels of the tree. If the tree contains less than three levels, the Tree viewer only shows the existing levels. You can view more levels using the **Show Level** slider or the **Default Expansion** box. For more information about configuring the Tree viewer, see "Viewing with Tree Viewer" in SQL Server Books Online.

To create the tree shown in Figure 6

1. Slide **Show Level** to 5.
2. In the **Background** list, click **1**.

By changing this setting, you can quickly tell the number of cases for bike buyer equal to 1 that exist in each node. The darker the node, the more cases that exist.

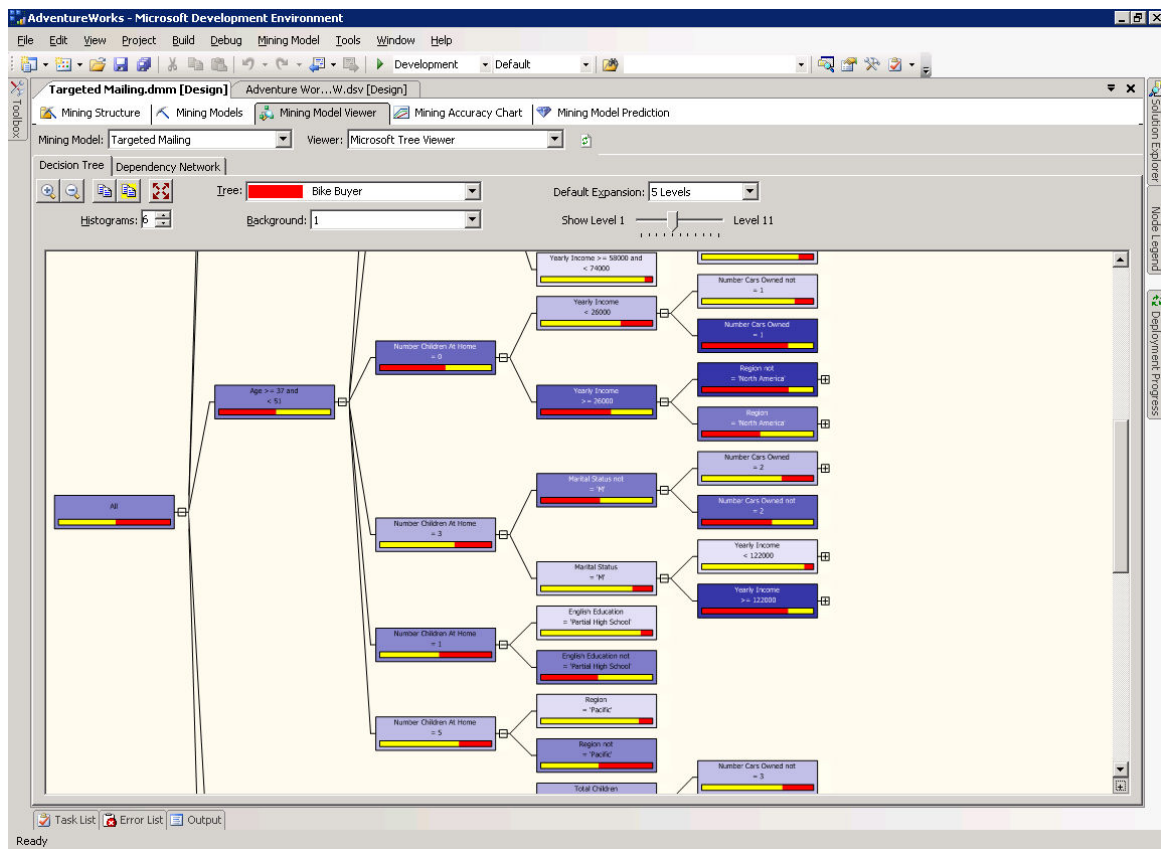


Figure 6 Decision Tree tab of the Targeted Mailing model

Each node in the decision tree displays three pieces of information:

- The condition required to reach that node from the node preceding it. You can see the full node path in either the legend or a ToolTip.
- A histogram that describes the distribution of states for the predictable column in order of popularity. You can control how many states appear in the histogram using the **Histogram** control.
- The concentration of cases, if the state of the predictable attribute specified in the **Background** control.

If drillthrough is enabled, you can see the training cases each node supports by right-clicking the node and then clicking **Drillthrough**.

Dependency Network

The **Dependency Network** tab displays the relationships between the attributes that contribute to the predictive ability of the mining model. The dependency network for the Targeted Mailing model is displayed in Figure 7.

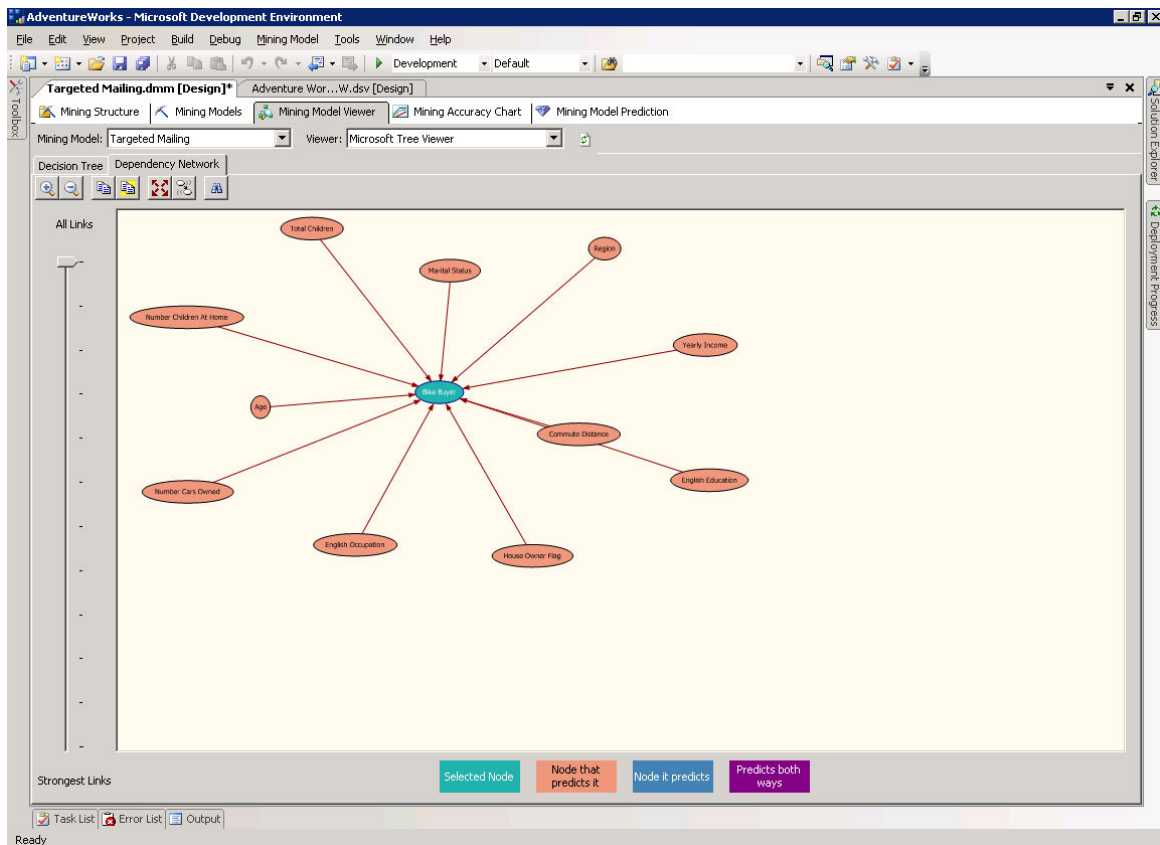


Figure 7 Dependency Network tab of the Targeted Mailing model

The center node in Figure 7, Bike Buyer, represents the predictable attribute in the mining model. Each surrounding node represents an attribute that affects the outcome of the predictable attribute. You can use the slider on the left side of the tab to control the strength of the links that are shown. Moving the slider down means that only the strongest links are shown.

Using the color legend at the bottom of the chart, you can see the nodes that a selected node predicts, or the nodes that the selected node is predicted by.

Microsoft Clustering Model

Use the **Mining Model** combo box at the top of the node to switch to the TM_Clustering model. The viewer for this model, the Cluster viewer, contains four tabs: **Cluster Diagram**, **Cluster Profiles**, **Cluster Characteristics**, and **Cluster Discrimination**. By default, the viewer displays the **Cluster Diagram** tab when it first opens.

For more information about using the Cluster viewer, see "Viewing with Cluster Viewer" in SQL Server Books Online.

Cluster Diagram

Using the **Cluster Diagram** tab, you can explore the relationships between the clusters discovered by the algorithm. The lines between the clusters represent "closeness" and are shaded based on how similar the clusters are. The actual color of the cluster represents the frequency of the variable and state (selected in the **Shading Variable** and **State** boxes at the top of the node) in each cluster. The default variable is **population**, but you can change this to any attribute in the model to find which clusters contain members with the attributes you want quickly. Using the slider on the left, you can filter out the weaker links and find the clusters that are the closest.

For example, set **Shading Variable** to **Bike Buyer** and **State** to **1**. As shown in Figure 8, Cluster 5 contains the highest density of bike buyers. The strongest relationship exists between Cluster 4 and Cluster 7.

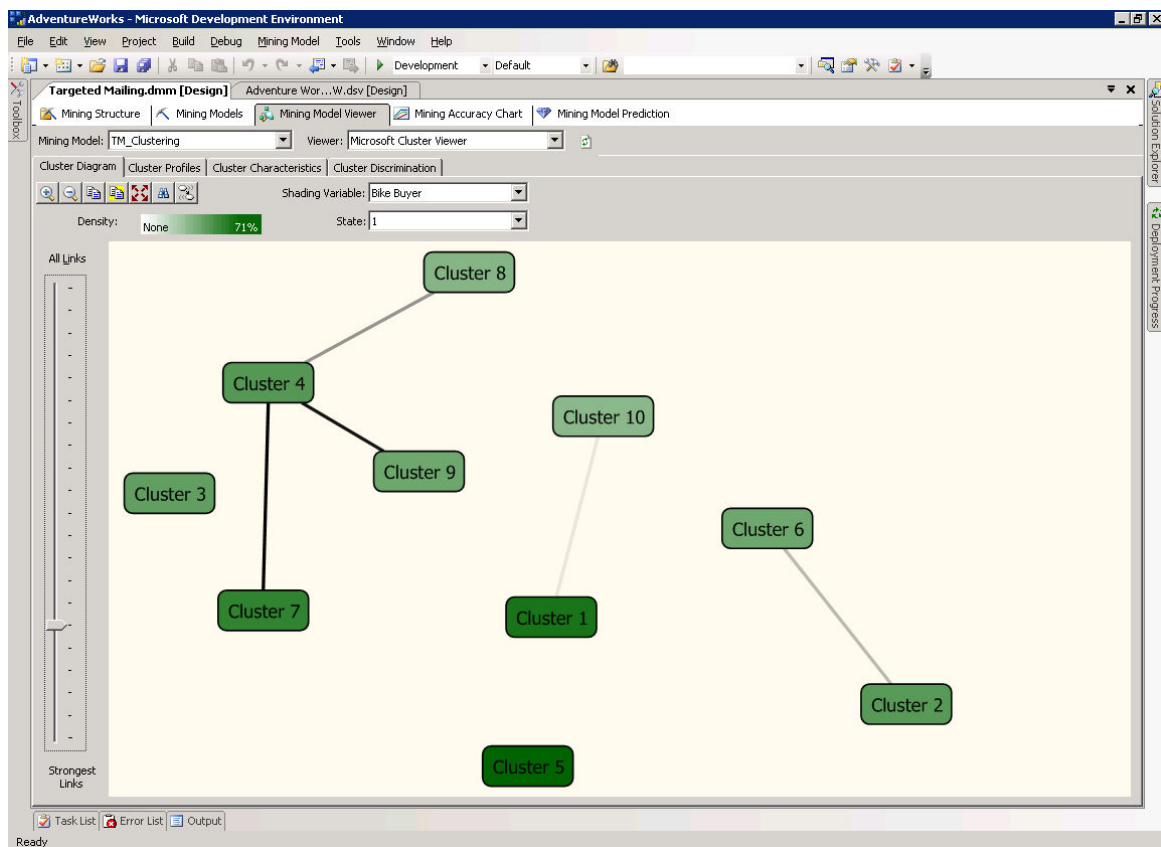


Figure 8 Cluster Diagram tab of the TM_Clustering model

Cluster Profiles

The **Cluster Profiles** tab provides an overall view of the TM_Clustering model. As Figure 9 shows, the **Cluster Profiles** tab contains a column for each cluster in the model. The first column lists the attributes that are associated with at least one cluster. The distribution of an attribute's states for each cluster fills out the rest of the viewer. The distribution of a discrete variable is shown as a colored bar with the maximum number of bars displayed in the **Bars per histogram** box. Continuous attributes are displayed using a diamond chart, which represents the mean and standard deviation in each cluster.

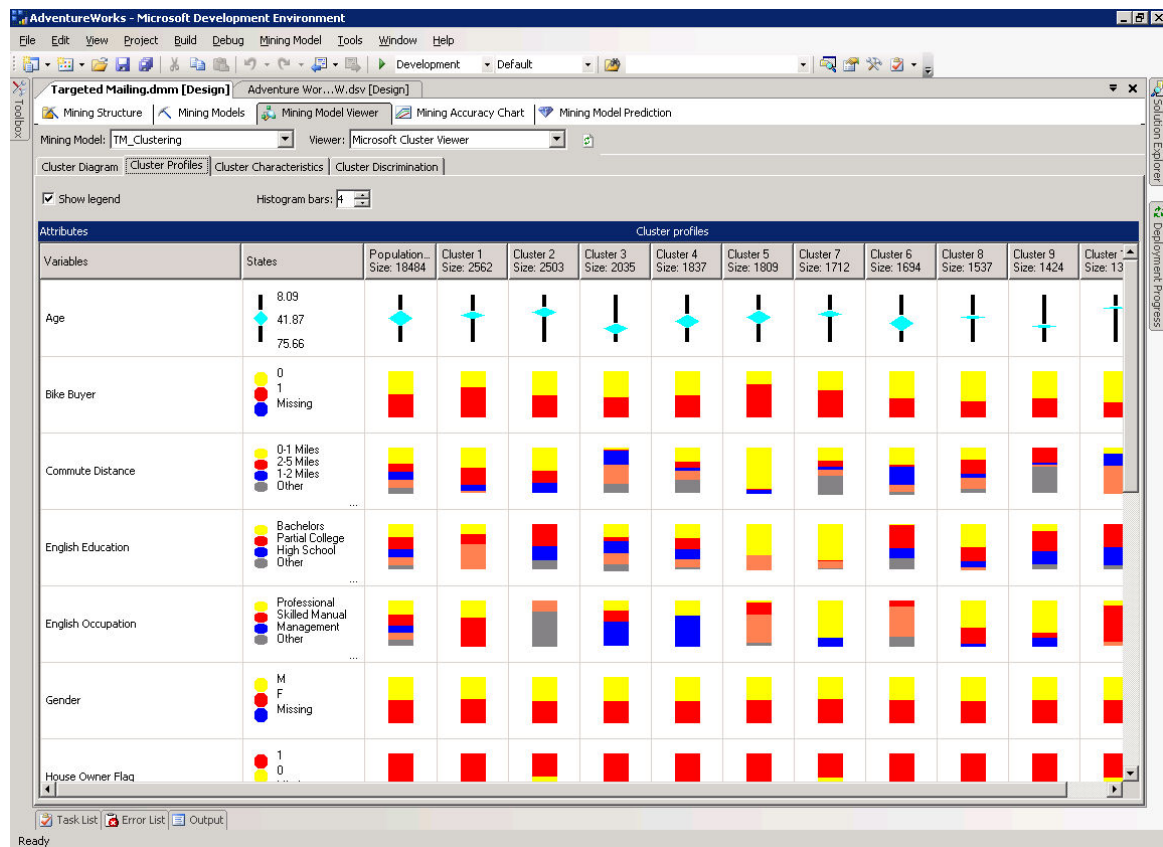


Figure 9 Cluster Profiles tab of the TM_Clustering model

Cluster Characteristics

Using the **Cluster Characteristics** tab, you can examine the characteristics that make up a cluster in more detail. For example, in Figure 10, you can see that people in Cluster 5 (the bike buyers) tend to have such characteristics as: they commute short distances (0-1 mile), they do not own a car, and they are married.

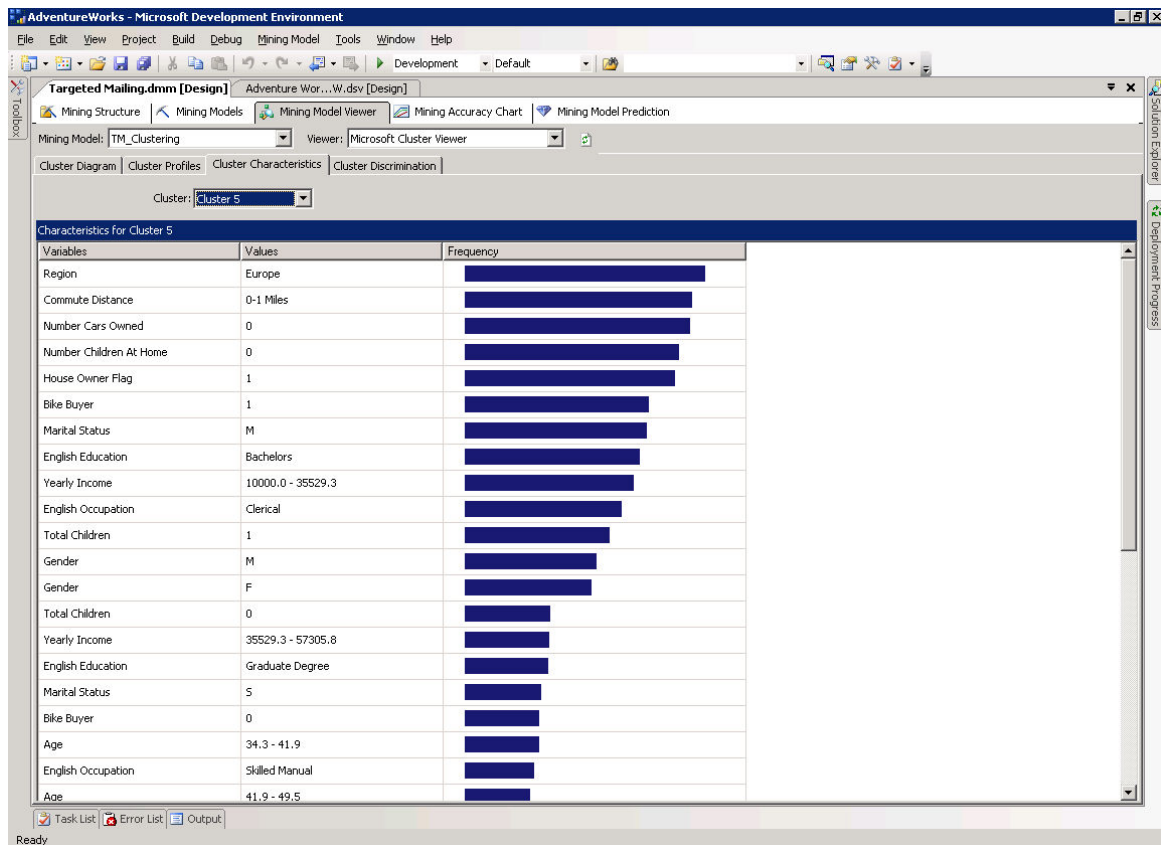


Figure 10 Cluster Characteristics tab of the TM_Clustering model

Cluster Discrimination

Using the **Cluster Discrimination** tab, you can explore the characteristics that distinguish one cluster from another. After you select two clusters from the **First cluster** and **Second cluster** boxes, the viewer determines the differences and displays them ordered by the attributes that distinguish the clusters the most.

Figure 11 compares Cluster 5 and Cluster 10 from the TM_Clustering model. Cluster 5 contains the highest density of bike buyers, and Cluster 10 contains the lowest density of bike buyers. For example, people in Cluster 10 tend to be from North America and younger (23-31) and people in Cluster 5 tend to be from Europe with a short commute distance (0-1 mile).

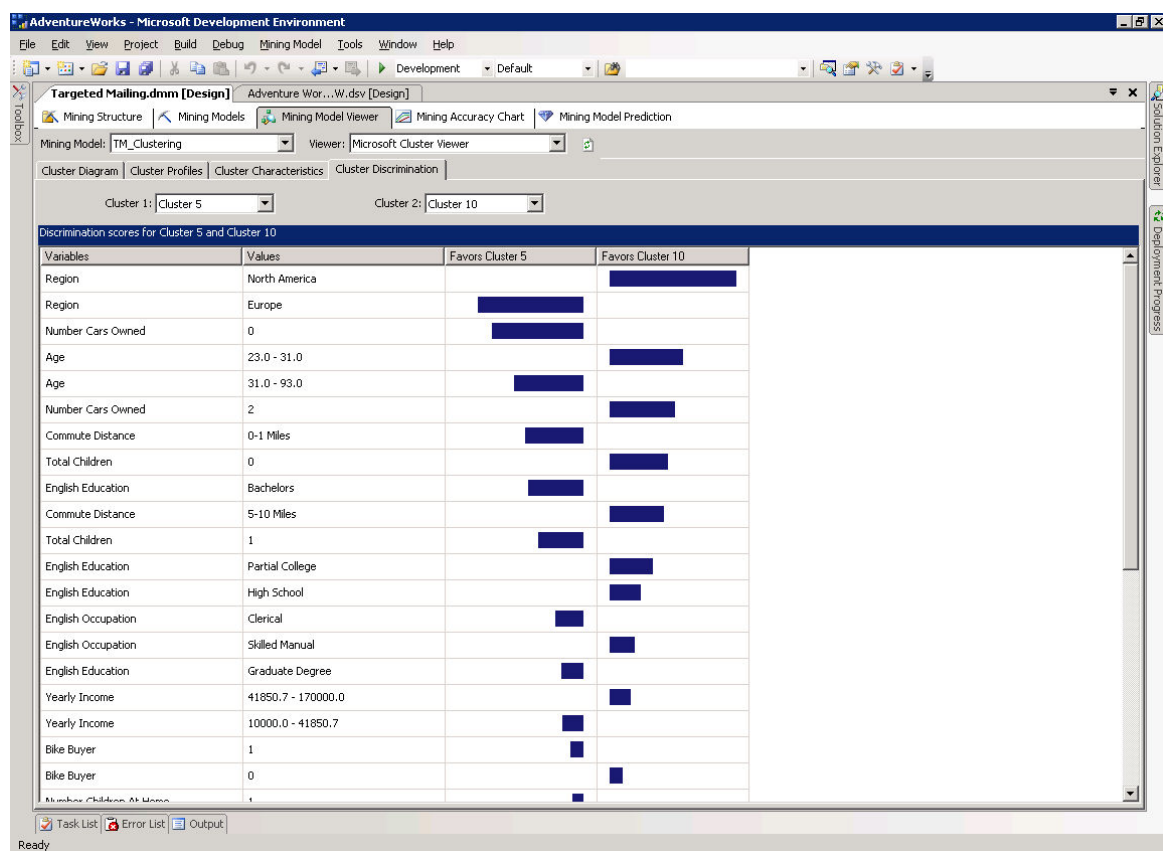


Figure 11 Cluster Discrimination tab of the TM_Clustering model

Microsoft Naïve Bayes Model

Use the **Mining Model** combo box to switch to the TM_NaiveBayes model. The viewer for this model, the Naïve Bayes viewer, contains four tabs: **Dependency Network**, **Attribute Profiles**, **Attribute Characteristics**, and **Attribute Discrimination**.

For more information about using the Naïve Bayes viewer, see "Viewing with Naïve Bayes Viewer" in SQL Server Books Online.

Dependency Network

The **Dependency Network** tab works the same way as the **Dependency Network** tab for the Tree viewer. Each node in the viewer represents an attribute, and the lines between nodes represent relationships. Figure 12 displays all of the attributes that affect the state of the predictable attribute, Bike Buyer.

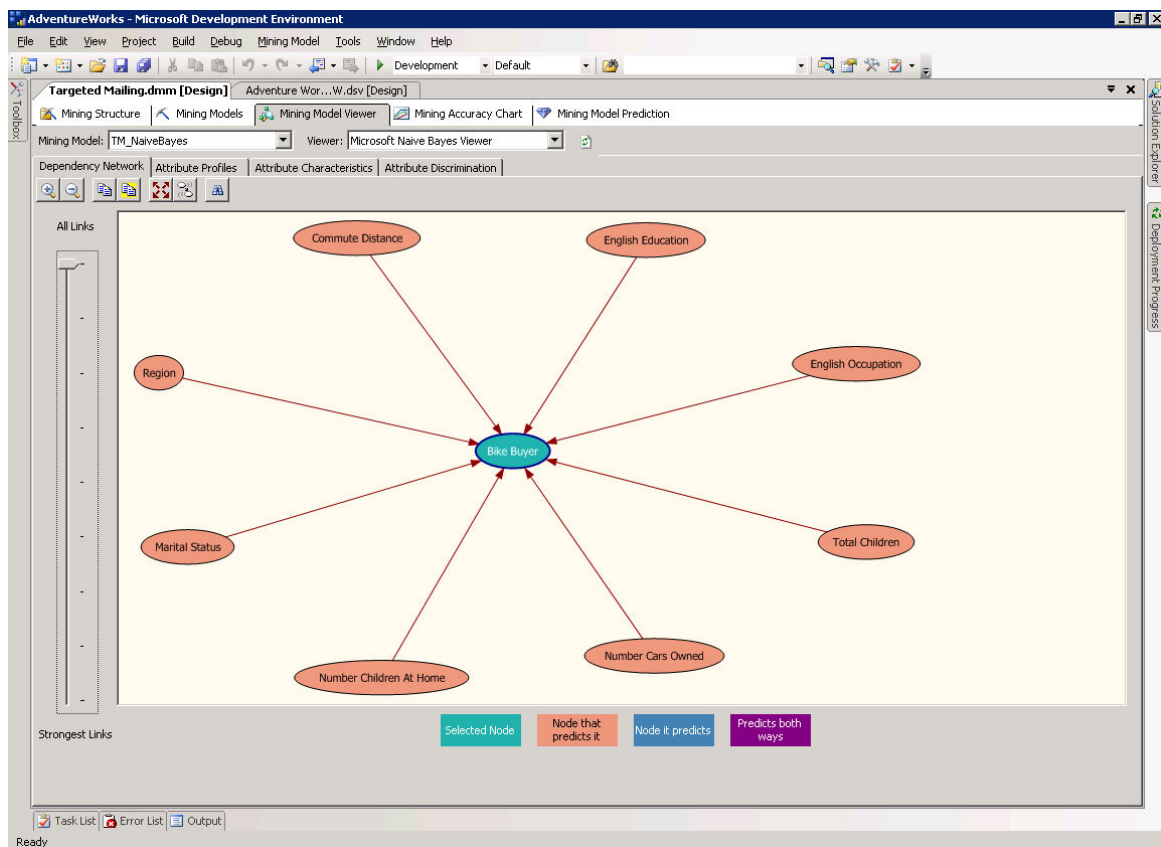


Figure 12 Dependency Network tab for the TM_NaiveBayes model

As you lower the slider, only the attributes that have the greatest effect on the **Bike Buyer** column remain. Using this technique, you find that the number of cars owned is the greatest factor in determining whether someone is a bike buyer.

Attribute Profiles

The **Attribute Profiles** tab describes how different states of the input attributes affect the outcome of the predictable attribute.

In the **Predictable** box, click **Bike Buyer**. The attributes that affect the state of the predictable attribute are listed along with values of each state of the input attributes, and their distributions in each state of the predictable attribute.

Figure 13 shows the **Attribute Profiles** tab for the Bike Buyer column in the TM_NaiveBayes model.

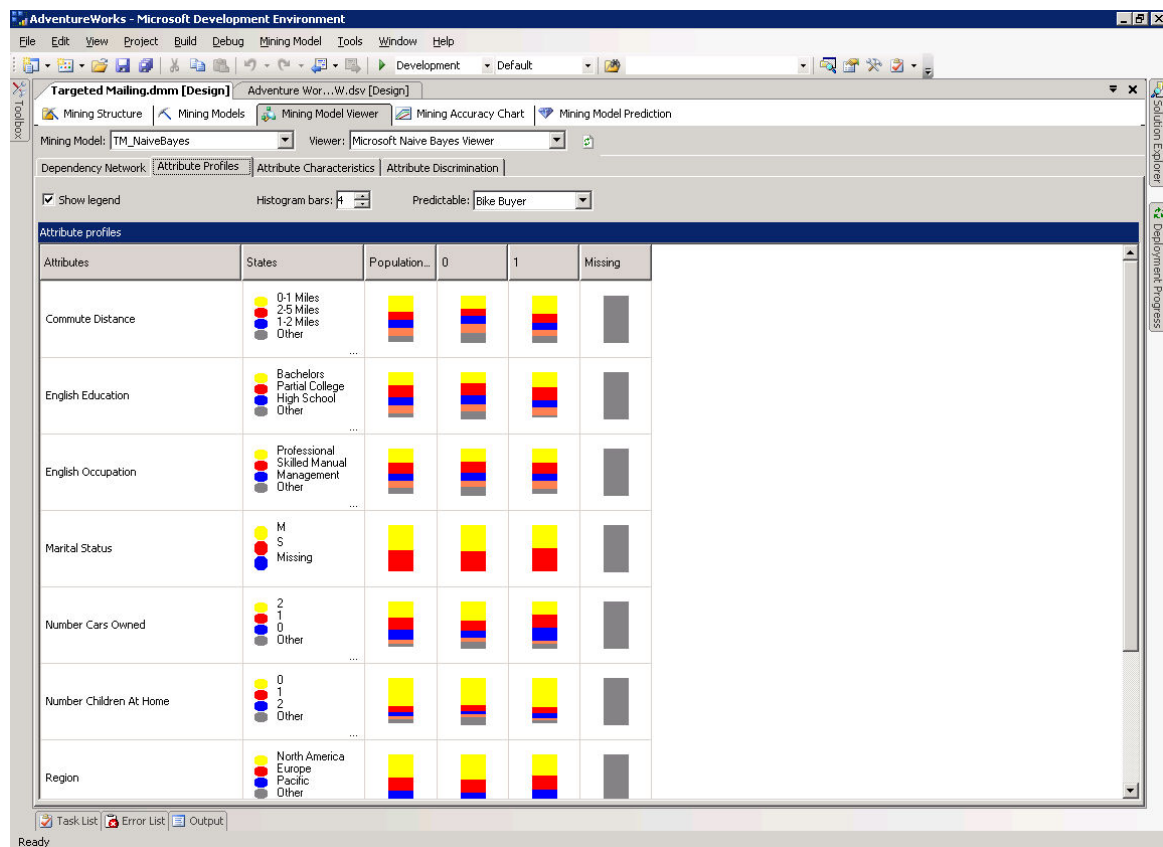


Figure 13 Attribute Profiles tab for an attribute in the TM_NaiveBayes model

Attribute Characteristics

Using the **Attribute Characteristics** tab, you can select an attribute and value to see how frequently values for other attributes appear in the selected value cases.

In **Attribute**, click **Bike Buyer**, and in **Value**, click **1**.

For example, Figure 14 shows that people who have no children at home tend to buy the most bicycles.

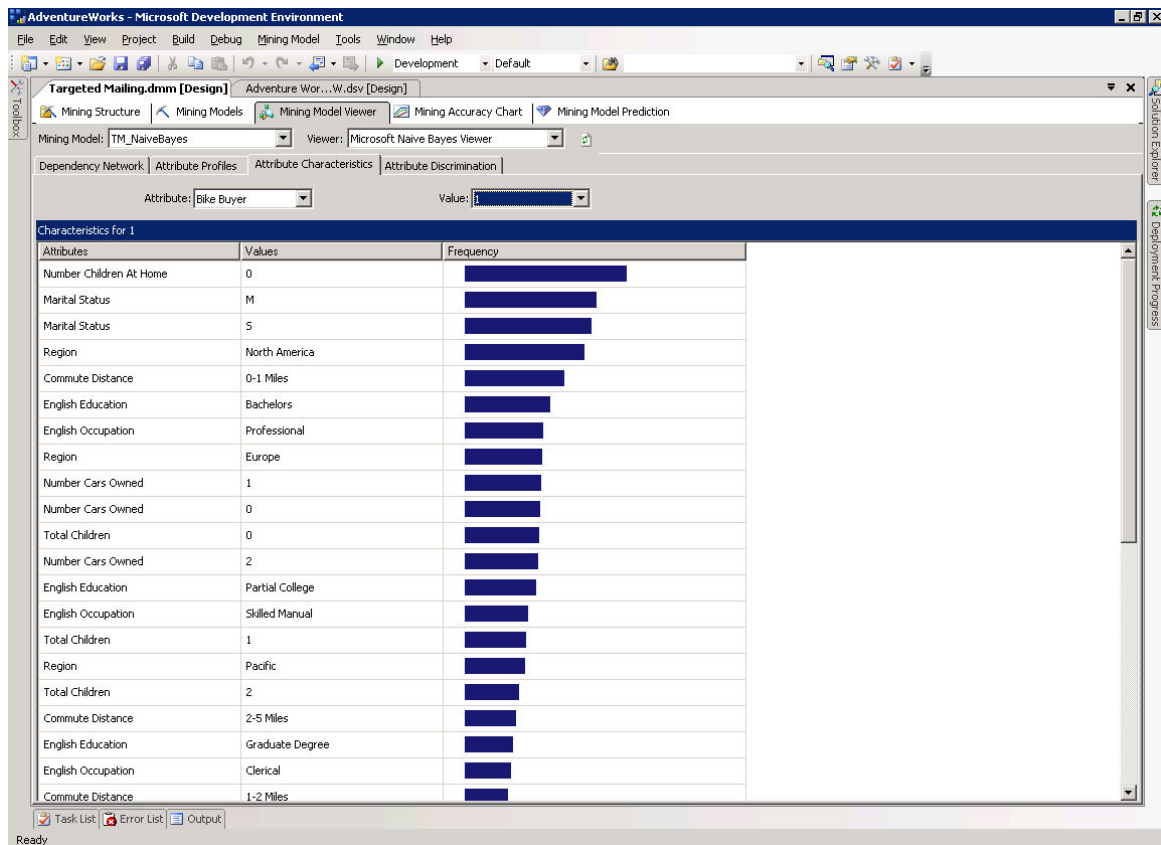


Figure 14 Attribute Characteristics tab for an attribute in the TM_NaiveBayes model

Attribute Discrimination

Using the **Attribute Discrimination** tab, you can investigate the relationship between two discrete values of the selected predictable attribute and other attribute values. Because the TM_NaiveBayes model only has two states, 1 and 0, you do not have to make any changes to the viewer.

For example, Figure 15 shows that people who do not own cars tend to buy bicycles, and people who own two cars tend not to buy bicycles.

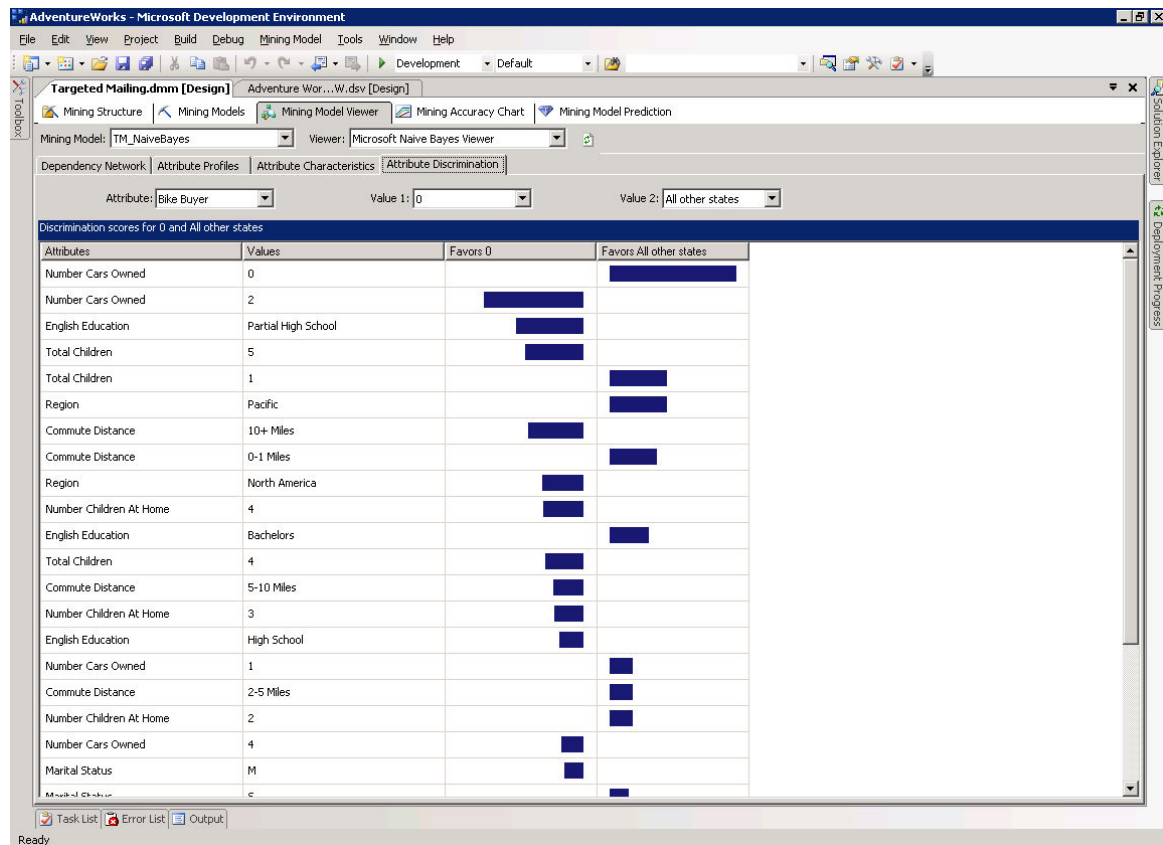


Figure 15 Attribute Characteristics tab for an attribute value in the TM_NaiveBayes model

Testing the Accuracy of the Mining Models

You have now processed and explored the mining models. But how well do they perform predictions? Does one of the targeted mailing models perform better than the others?

Using the Mining Accuracy Chart tab, you can calculate how well each of the models predicts and compare their results directly against each other. This method of comparison is also sometimes called a lift chart. The Mining Accuracy Chart tab uses test data, which is data separated from the original training dataset, to compare predictions against a known result. The results are then sorted and plotted on a graph, along with an ideal model to show how well the model performs at predictions. An ideal model represents a plot for a theoretical model that predicts the result correctly 100 percent of the time.

The lift chart is important because it helps to distinguish between nearly identical models in a structure, determining which provides the best predictions. Similarly, it shows which algorithm types perform the best predictions for a given situation. For more information about using the Mining Accuracy Chart tab, see "Comparing Data Mining Models with the Lift Chart" in SQL Server Books Online.

Open the tab, which is shown in Figure 16, by clicking **Mining Accuracy Chart**.

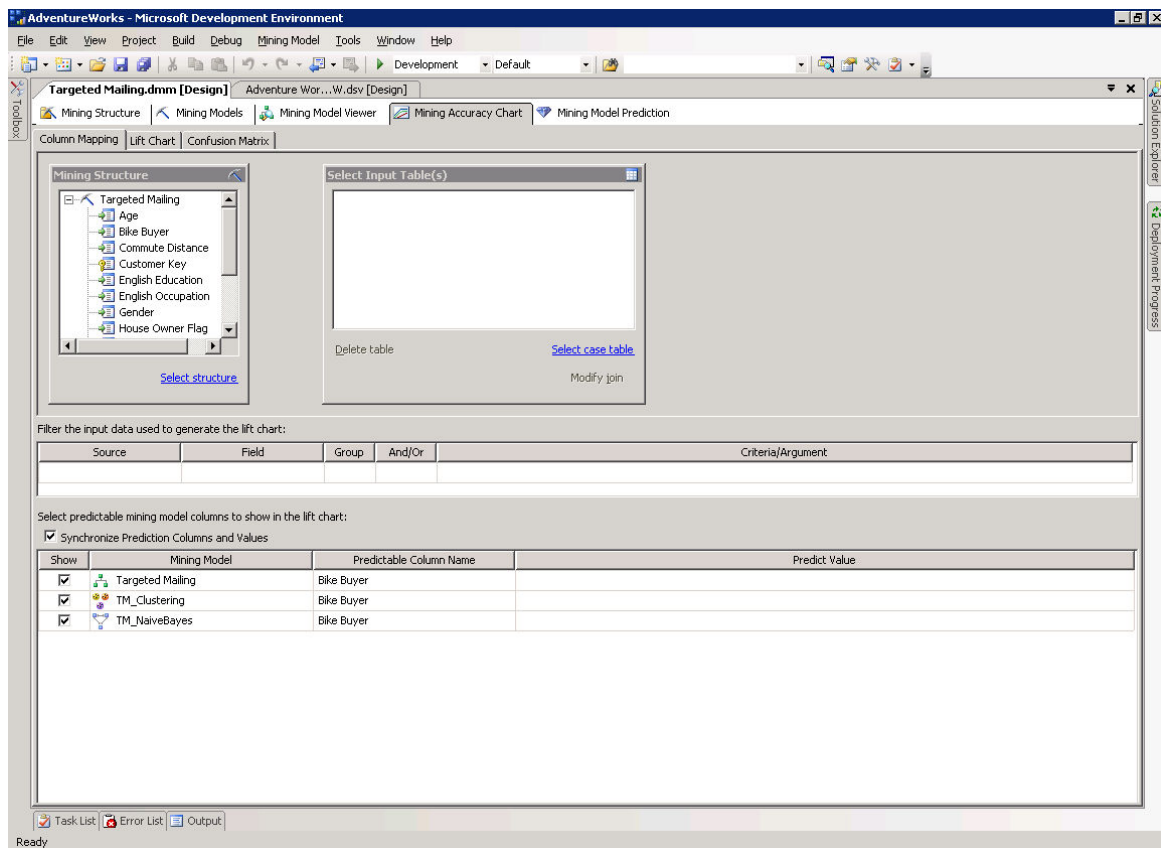


Figure 16 Mining Accuracy Chart tab

To create a new mining accuracy chart, you must perform the following tasks:

1. Map the columns of the model to the columns in the input dataset.
2. Filter the input data.
3. Select the models to compare and the predictable columns and values.

Note Before you can use the mining accuracy chart, you must deploy and process the mining models.

Mapping the Input Columns

The first step is to map the columns in the model to the columns in the test data. If the column names map directly, the tool automatically creates relationships.

To map the input columns to the mining structure

1. In the **Select Input Table(s)** box, click **Select case table**.

The **Select Table** dialog box opens, where you choose an input table that contains the test data that you want to use in the prediction queries to determine the accuracy of the models. For example, if you held out some TargetMail rows independent from vTargetMail that was used to process the models, you might select that table. However, in this tutorial we use the same data used to process the models as the input table.

2. In the **Select Table** dialog box, select **Adventure Works DW** from data source list.
3. Select **vTargetMail**.from Table/View list and click OK.

The columns the mining structure are automatically mapped to the columns with the same name in the input table, as shown in Figure 17.

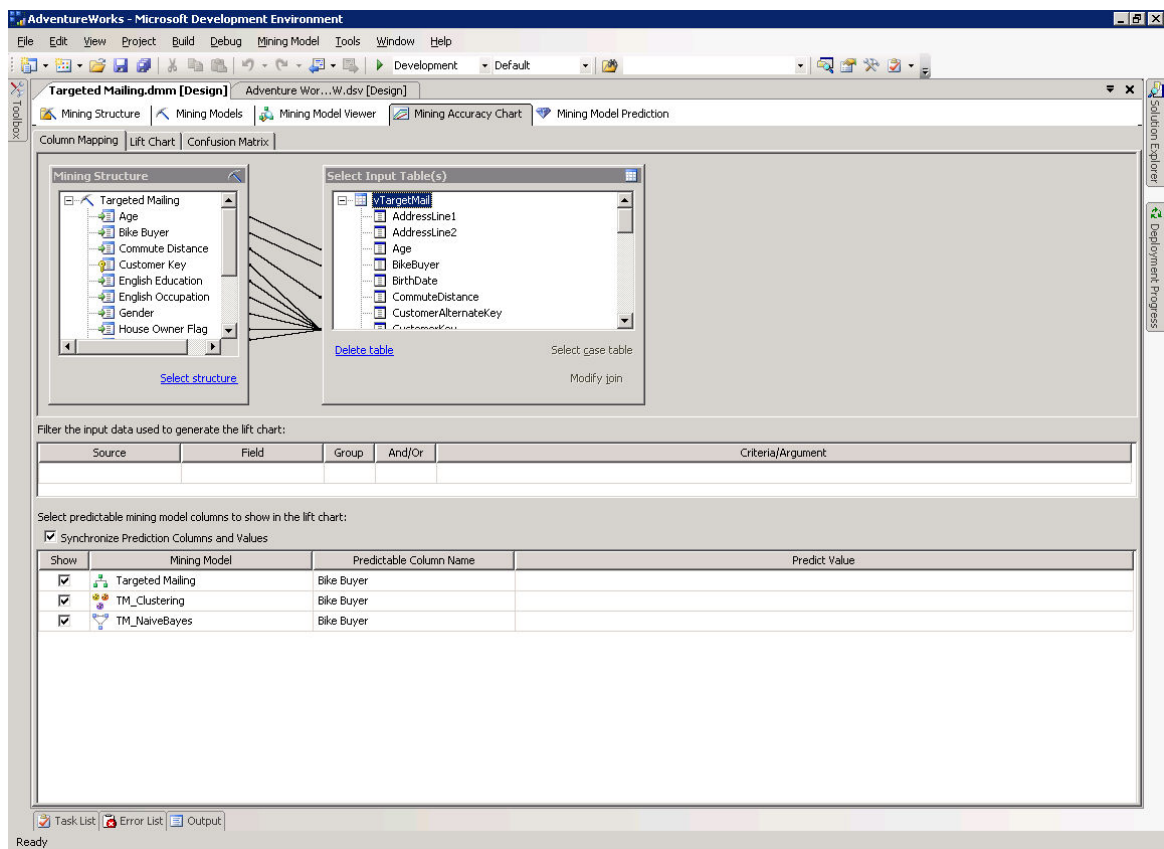


Figure 17 Mapped columns in the Mining Accuracy Chart tab

A prediction query is generated for each model in the structure based on the column mappings. You can delete a mapping by selecting the line that links the columns in **Mining Structure** and **Select Input Table(s)** and then pressing DELETE. You can also manually create mappings by clicking a column in **Select Input Table(s)** and dragging it onto the corresponding column in **Mining Structure**.

Filtering Input Rows

You can use the grid under **Filter the input data used to generate the lift chart** to filter the input data. You can drag columns from **Select Input Table(s)** to the grid, or you can select the values from combo boxes. For example, if you want to limit the input rows to those where the **YearlyIncome** column is greater than x, in the **Field** column, select **YearlyIncome**, and then in the **Criteria/Argument** column, type >x.

You will not filter the data in this tutorial.

Selecting the Models, Predictable Columns, and Values

The next step is to select the models that you want to include in the lift chart and the predictable column that they will be compared against. By default, all of the models in the mining structure are selected. You can choose not to include a model, but for this tutorial, leave them as they are.

You can create two types of accuracy charts. If you select a predictable value, you will see a chart like the one in Figure 18, which shows how much lift the model provides. If you do not include a Predict Value, as shown in Figure 19, the chart will show how accurate the model is.

To show the lift of the models

1. For each remaining model, in **Predictable Column Name**, click **Bike Buyer**.
2. For each remaining model, in the **Predict Value** column, click **1**.

To show the accuracy of the models

- In **Predictable Column Name**, click **Bike Buyer**.

Leave the **Predict Value** column empty.

If the **Synchronize Prediction Columns and Values** check box is selected, the predictable column is synchronized for each mining model in the mining structure.

Note The mining model columns listed in the **Predictable Column Name** box are restricted to columns that have the usage type set to **Predict** or **Predict Only**, and where the mining structure column on which this mining column is based has a content type of **Discrete** or **Discretized**.

In some advanced scenarios, you may want to generate a lift chart that includes a predictable column in two mining models that are not based on the same structure column but contain the same data. If you clear the **Synchronize Prediction Columns and Values** check box, you can select any valid predictable column and value. The results are plotted together, regardless of whether they make sense.

Viewing the Lift Chart

Click the **Lift Chart** tab to view the lift chart. When you click the tab, a prediction query runs against the server and database for the mining structure and input table. The predicted results are compared to the actual values that is known and sorted by probability, and the results are plotted on the graph. For more information about using the chart, see "Lift Chart" in SQL Server Books Online.

If you specified a predictable value, the lift chart is plotted as shown in Figure 18.

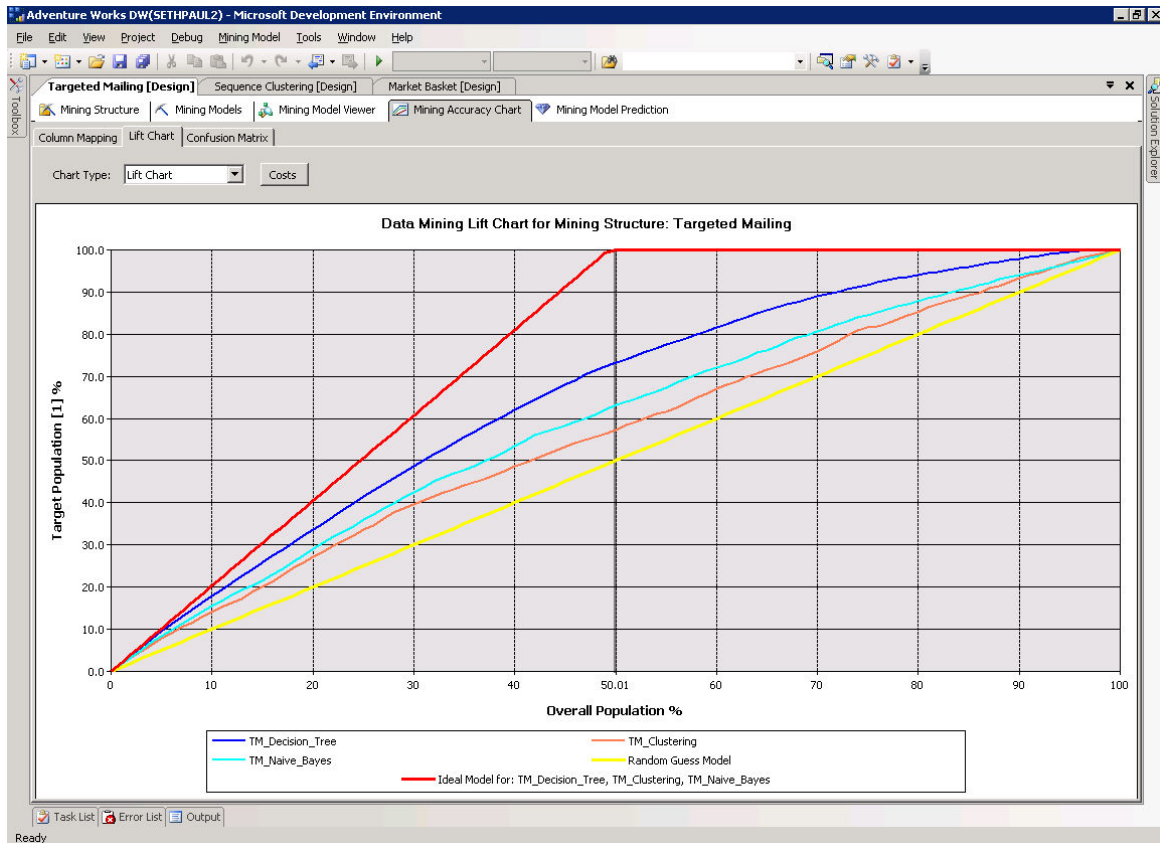


Figure 18 Lift provided by each model plotted against the ideal model

If you did not specify a predictable value, the lift chart shows the accuracy of the mining model predictions as shown in Figure 19.

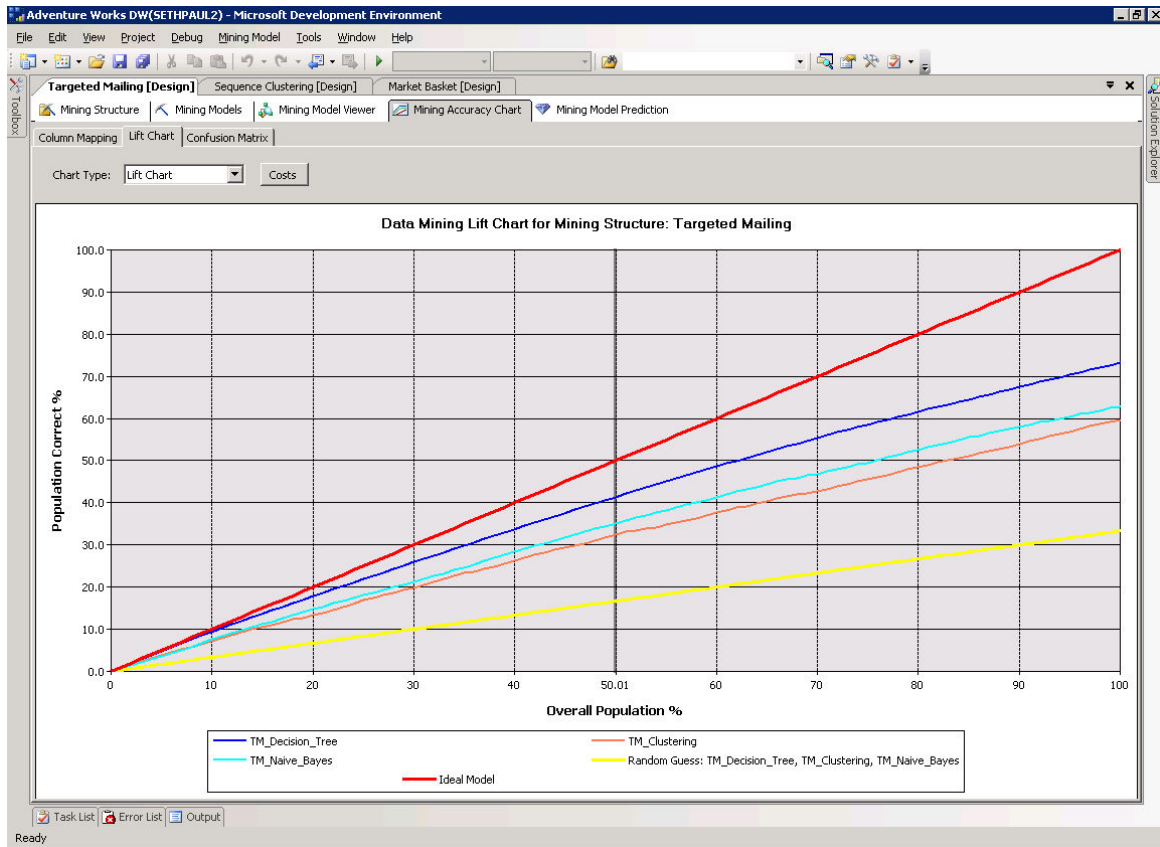


Figure 19 Accuracy of each model plotted against the ideal model

Creating Predictions

Now that you are satisfied with the mining models, you can begin to create DMX prediction queries using Prediction Query Builder. Prediction Query Builder is similar to Access Query Builder, in which you use drag-and-drop operations to build the queries. The tool contains three different views:

- Design
- Query
- Result

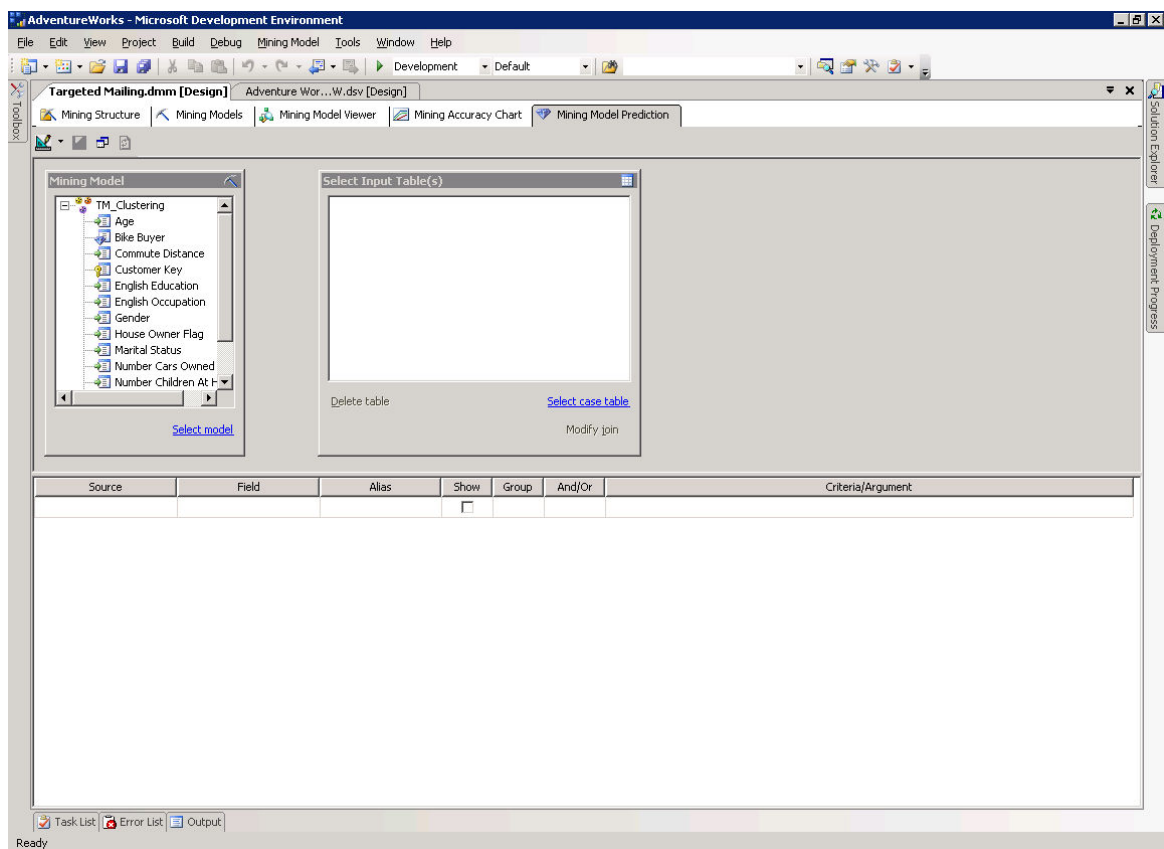


Figure 20 Default view of Prediction Query Builder

Using the Design and Query views, you can build and look at your query. You can then execute and view the results of the query in the Result view.

For more information about using Prediction Query Builder, see "Creating Prediction Queries Against Data mining Models" in SQL Server Books Online.

Creating the Query

The first step in creating the query is to select a mining model and input table.

To select a model and input table

1. In **Mining Model**, click **Select model**.

The **Select Mining Model** dialog box opens. By default, the first mining model in the mining structure is selected.

2. Navigate through the tree to **Targeted Mailing**, and then click **Targeted Mailing**.
3. In the **Select Input Table(s)** box, click **Select case table**.

The **Select Table** dialog box opens.

4. Navigate through the tree and select the **vTargetMail** table in the AdventureWorksDW data source view.

Note that typically you would have a separate table that contains your prospect customers and you would want to predict whether each customer would buy a bike or not (i.e., Bike Buyer column) based on other known information (i.e., other columns). However, for the sake of simplicity of the tutorial, we are using the same training data, **vTargetMail** as the prospect customers.

After you select the input table, Prediction Query Builder creates a default mapping between the mining model and input table based on the names of the columns, as shown in Figure 21.

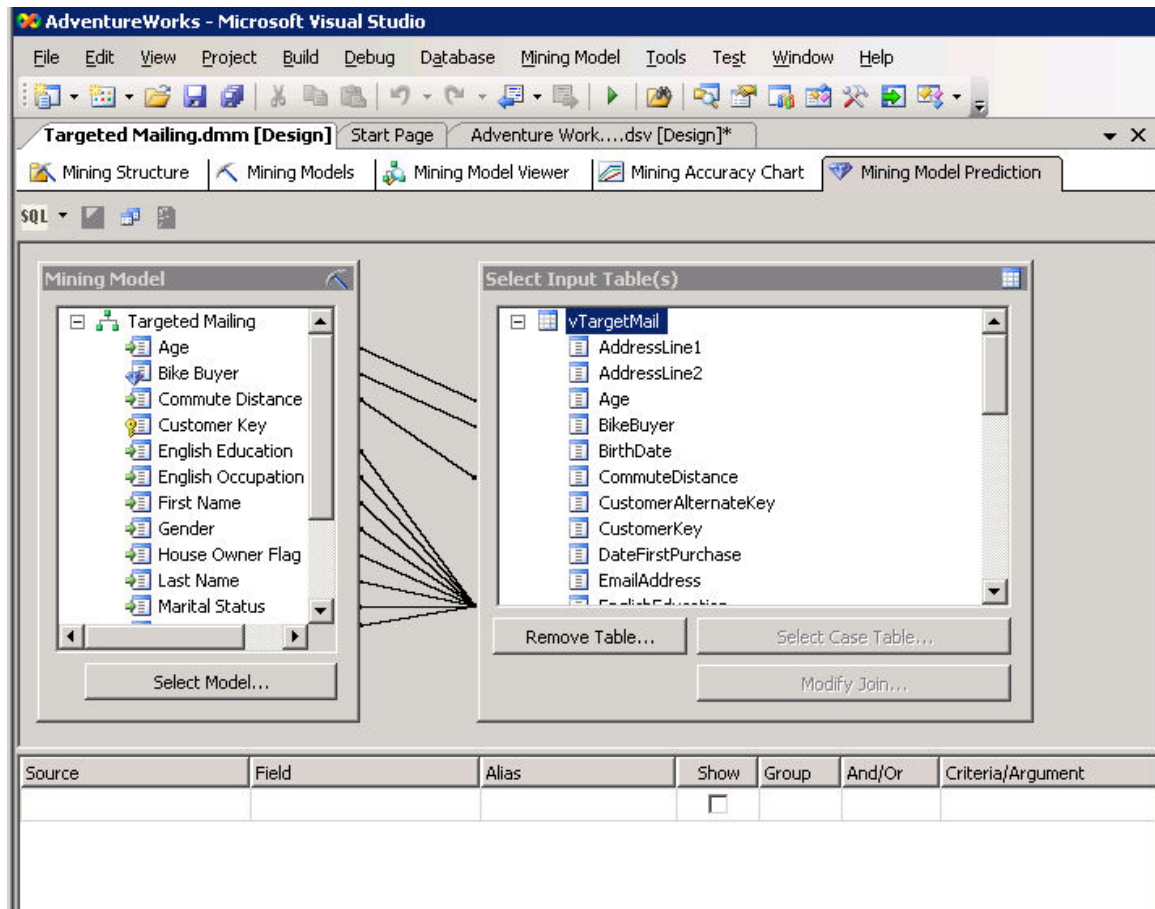


Figure 21 Mapped columns in the Mining Model Prediction tab

To build the prediction query

1. In the **Source** column, click the cell in the first empty row, and then click **vTargetMail table**.
2. In the **Field** column, next to the entry you created in step 1, click **CustomerKey**.
This adds the unique identifier to the prediction query so that you can identify who is and who is not likely to buy a bicycle.
3. Click the next cell in the **Source** column, and then click **Targeted Mailing** mining model.
4. In the **Field** cell, click **Bike Buyer**.
This specifies that the Microsoft Clustering model in the Targeted Mailing structure will be used to create the predictions.
5. Click the next cell under the **Source** column, and then click **Prediction Function**.
6. Next to **Prediction Function**, in the **Field** column, click **PredictProbability**.

Prediction functions provide information about how the model predicts. The **PredictProbability** function provides information about the probability of the prediction being correct. You can specify parameters for the prediction function in the **Criteria/Argument** column.

7. In the **Criteria/Argument** column, type [Targeted Mailing].[Bike Buyer].

This specifies the target column for the **PredictProbability** function. For more information on functions, see "DMX Function Reference" in SQL Server Books Online.

Your screen should now look like Figure 22.

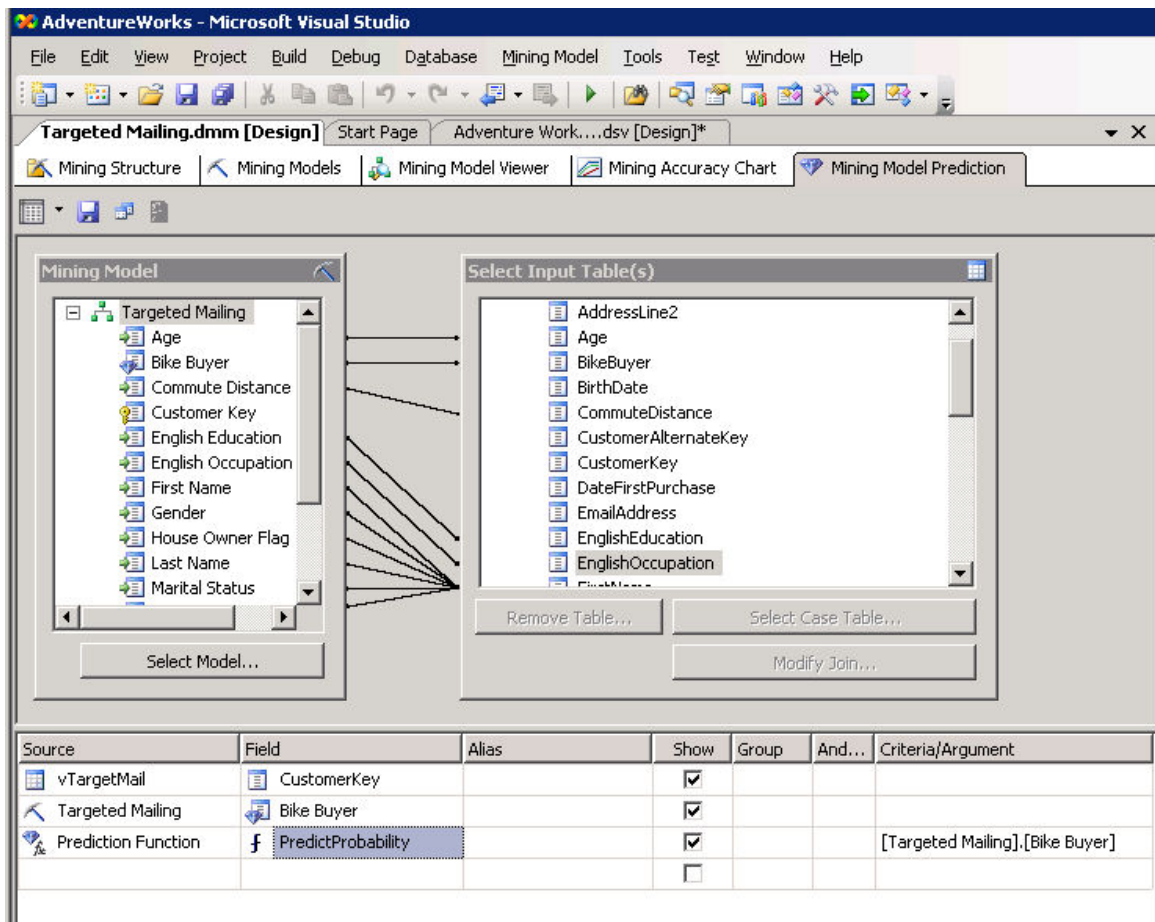
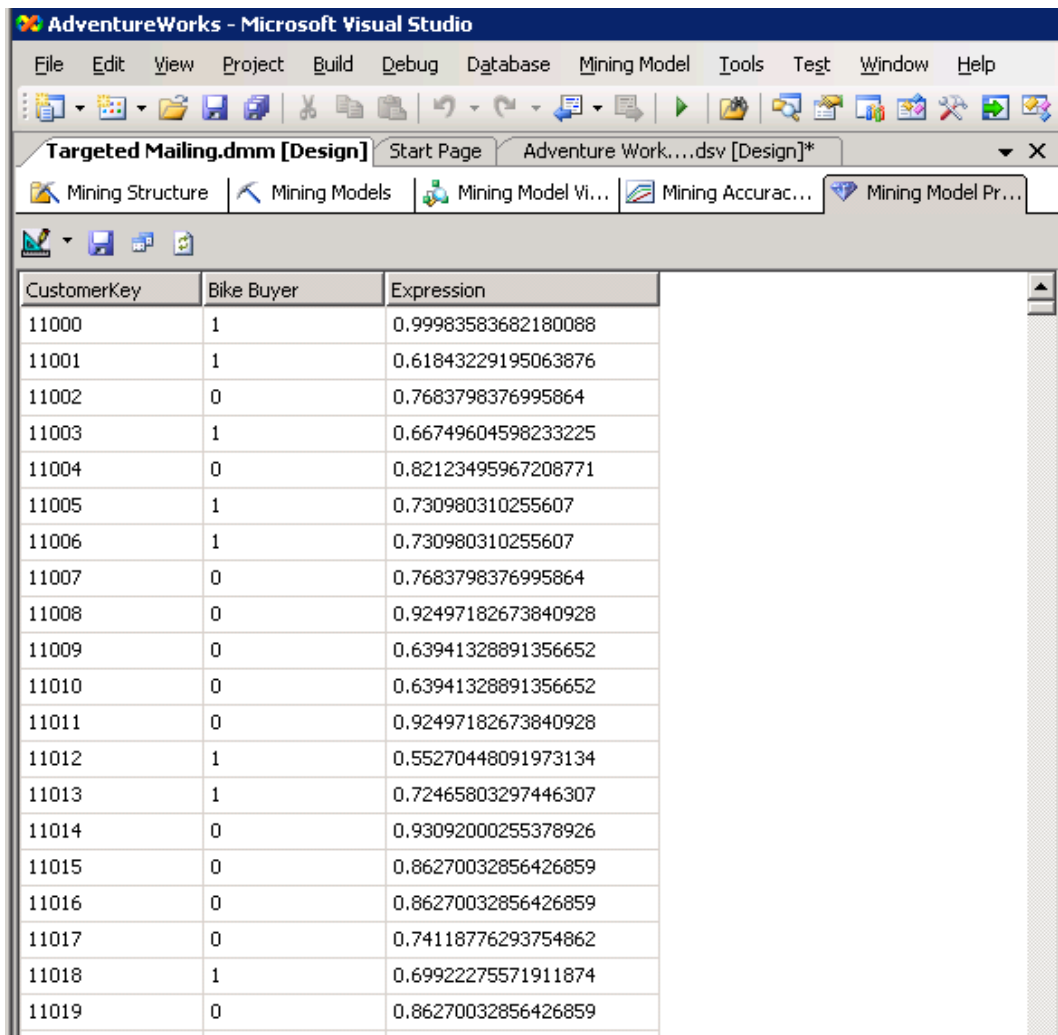


Figure 22 Prediction Query Builder in the Mining Model Prediction tab

By clicking the icon in the upper-left corner of the view, you can switch to the Query view and look at the DMX code that Prediction Query Builder created. You can also run the query, modify the query, and run the modified query, but the modified query is not persisted if you switch back to the Design view.

Viewing the Results

You can run the query by clicking the arrow next to the icon in the top left corner of the tab, and then clicking **Result**. Figure 23 displays the query results.



The screenshot shows the Microsoft Visual Studio interface with the 'AdventureWorks - Microsoft Visual Studio' title bar. The menu bar includes File, Edit, View, Project, Build, Debug, Database, Mining Model, Tools, Test, Window, and Help. The toolbar contains various icons for file operations, development, and testing. The 'Targeted Mailing.dmm [Design]' tab is active, showing a 'Mining Model Prediction Result' view. The table below displays the results of the prediction query.

CustomerKey	Bike Buyer	Expression
11000	1	0.99983583682180088
11001	1	0.61843229195063876
11002	0	0.7683798376995864
11003	1	0.66749604598233225
11004	0	0.82123495967208771
11005	1	0.730980310255607
11006	1	0.730980310255607
11007	0	0.7683798376995864
11008	0	0.92497182673840928
11009	0	0.63941328891356652
11010	0	0.63941328891356652
11011	0	0.92497182673840928
11012	1	0.55270448091973134
11013	1	0.72465803297446307
11014	0	0.93092000255378926
11015	0	0.86270032856426859
11016	0	0.86270032856426859
11017	0	0.74118776293754862
11018	1	0.69922275571911874
11019	0	0.86270032856426859

Figure 23 Mining Model Prediction Result tab

The **CustomerKey**, **BikeBuyer**, and **Expression** columns identify potential customers, whether they are bike buyers, and the probability of the prediction being correct. You can use these results to determine who should be sent an advertisement.

Forecasting

A sales analyst for Adventure Works has been asked to forecast the sale of bike models for the next year. In particular, he has been asked to find peak times for bike sales and how sales lead or lag with respect to region. Additionally, he will look to see whether sales of different models vary depending on the time of the year.

The analyst will investigate the data at the monthly level. Sales will be divided into three regions: Europe, North America, and Australia.

Upon completion of this task, the analyst will be able to answer the following questions:

- What time of year do sales peak?
- How do the different bike models interact over time?
- Is there a pattern to sales with respect to the three regions?

In order to complete the task, the analyst will use the Microsoft Time Series algorithm. The scenario consists of three tasks:

- Create the mining model structure.
- Edit the mining model.
- Explore the mining model.

Create a Forecasting Mining Model Structure Using the Wizard

The first step is to use Mining Model Wizard to create a new mining structure. The Mining Model Wizard also creates an initial mining model based on the Microsoft Time Series algorithm.

To create the forecasting mining structure

1. In Solution Explorer, right-click **Mining Structures**, and then click **New Mining Structure**.
The Mining Model Wizard opens.
2. On the Welcome page, click **Next**.
3. Click **From existing relational database or data warehouse**, and then click **Next**.
4. Under **What data mining technique do you want to use?**, click **Microsoft Time Series**.
5. Click **Next**.
By default Adventure Works DW is selected in the **Select data source view** window.
6. Select the **Case** check box next to the **vTimeSeries** table.
7. Select the **Key** check boxes next to the **TimeIndex** and **ModelRegion** columns.
8. Select the **Input** and **Predictable** check boxes next to the **Quantity** columns.
This indicates that you want to forecast these columns.
9. Click **Next**.
10. Select **Key Time** in the drop box of **TimeIndex** column
The **TimeIndex** column is designated as a key time column and the **ModelRegion** column is designated as a key column. This means that a separate time series will be built for each unique entry in the **ModelRegion** column. The values in **TimeIndex** must be unique only across individual values in **ModelRegion**.
11. Click **Next**.
12. In both **Mining structure name** and **Model Name** box, type **Forecasting**, and then click **Finish**.

The data mining editor opens, displaying the Forecasting mining structure you created.

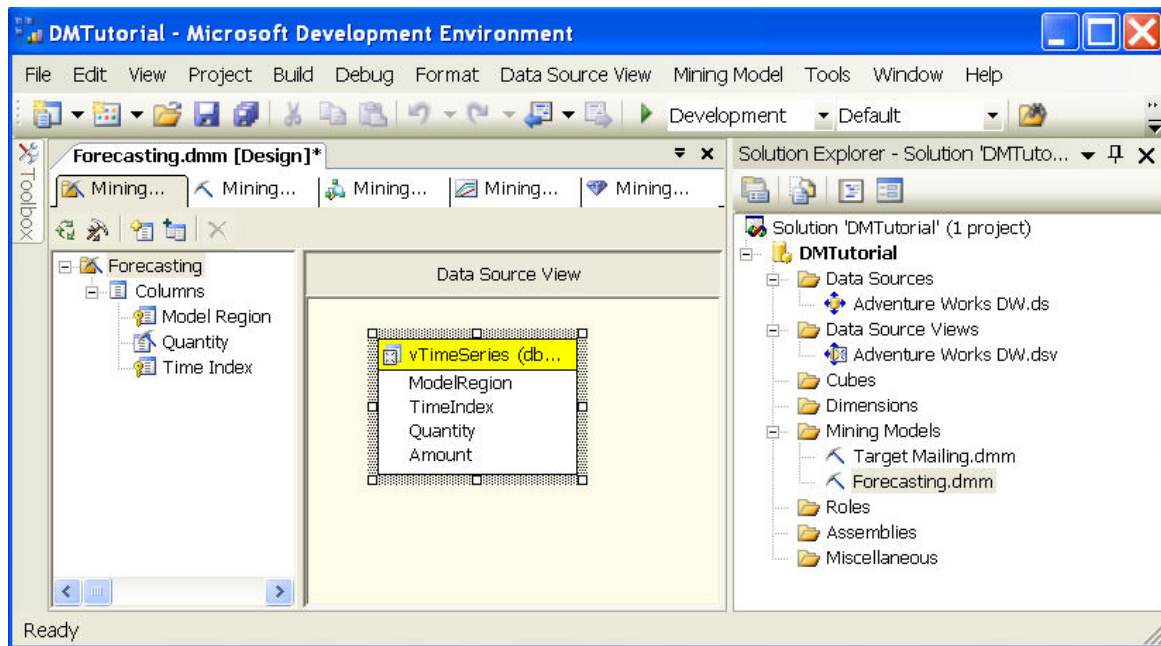


Figure 24 Mining model structure for the forecasting scenario

Edit the Mining Model

The mining structure shown in Figure 24 contains a single forecasting model that you defined in the Mining Model Wizard. Before you process and explore the model, you need to change its structure slightly and modify one property.

Modify the Mining Structure

You can change the mining structure using the Mining Structure tab of the data mining editor. You only used three columns to create the model: **TimeIndex**, **ModelRegion**, and **Quantity**. The Forecasting table also contains an **Amount** column that you can use to forecast the amount of sales. Using the Mining Structure tab, you can add this column from the data source view to the mining structure.

To add the Amount column to the Forecasting mining structure

1. Select the **Amount** column in the **vTimeSeries** table in the **Data Source View** window.
2. Drag the **Amount** column from the **Data Source View** window into the list of columns for the Forecasting structure.

The **Amount** column now exists as part of the Forecasting mining structure.

Modify the Mining Model

Because you added a new column to the structure, you must define how it will be used in the Mining Model tab. Figure 24 shows what the tab for this mining structure looks like.

The Mining Models tab lists the columns contained in the mining structure, under **Structure**, and those contained in the model, under the name of the model (in Figure 25, **Forecasting**). Click the names of the columns or the name of the model to make modifications.

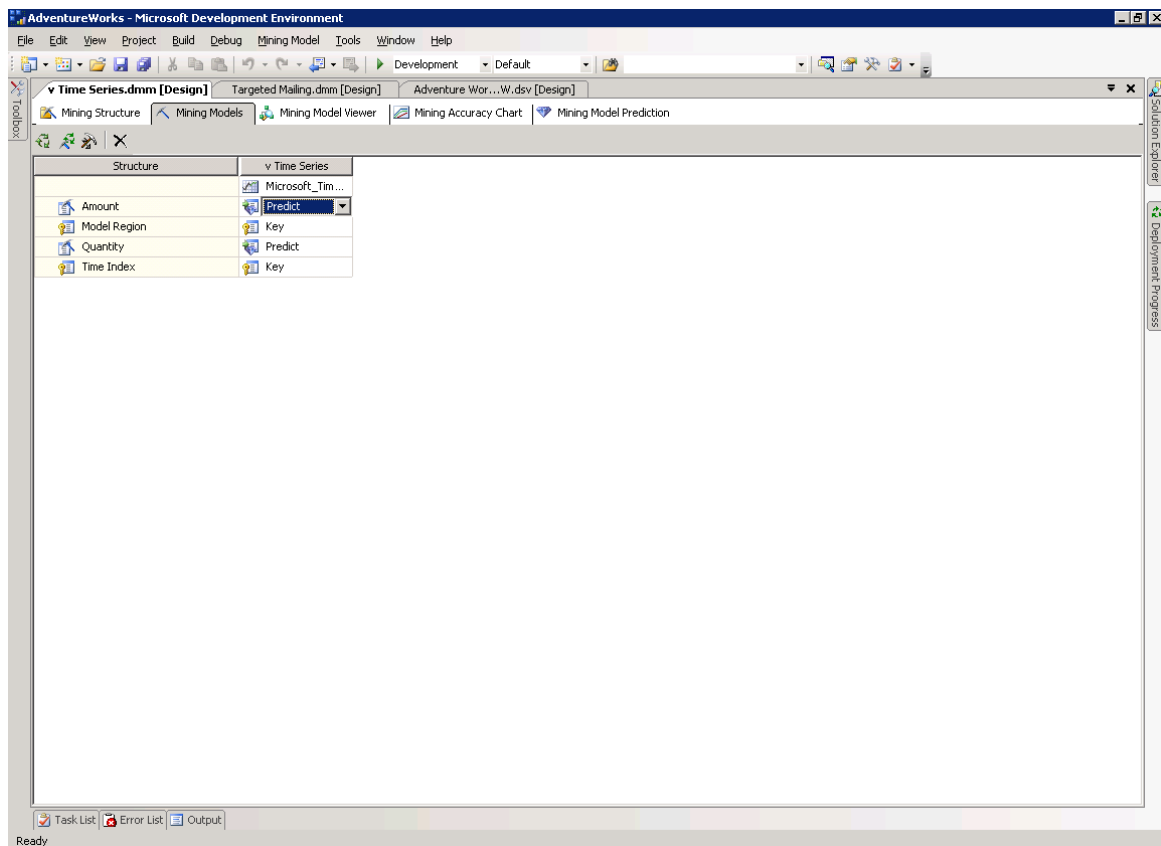


Figure 25 Mining Models tab of the Forecasting structure

Note In this tab you can also create new models based on the same structure, and you can adjust the algorithm and column properties for each model. You must process the model before these changes take effect.

In the Forecasting mining model, the **Amount** column is used as an input and to forecast future sales. Therefore, you must set the properties of the column so that it can be used as both an input column and a predictable column.

To define how the Amount column will be used

1. Under the **Mining Models** tab, click the **Amount** row of **Forecasting** model.
A list box containing **Ignore**, **Input**, **Predict**, and **PredictOnly** appears.
2. Click **Predict**.

The **Amount** column is now both an input and a predictable column.

You can also change the properties of individual columns by selecting the column and opening the properties window (right-click the column name, and then click **Properties**). By selecting the column under the model (in this case, under **Forecasting**), you can change the properties just for that model. By selecting the column under **Structure**, you can change properties of the column for the structure, which affects every model associated with the structure.

If you select **Forecasting**, you can change properties and parameters associated with the model. The Microsoft Time Series algorithm contains several parameters that affect how a model is created. For more information about these parameters, see "Time Series Algorithm Parameters" in SQL Server Books Online.

For this model you will adjust the value of the PERIODICITY_HINT parameter, which gives the algorithm information about often the data is repeated. The data in **AdventureWorksDW** is patterned on a monthly basis, and the periodicity is at the yearly level. Therefore, you will set the PERIODICITY_HINT parameter to **12**, indicating that a pattern repeats itself every year.

To change the PERIODICITY_HINT parameter

1. Right-click **Forecasting**, and select **Set Algorithm Parameters**.
The **Algorithm Parameters** dialog box opens.
2. Set **PERIODICITY_HINT** to {12}.

Process the Mining Model

Now that the structure and parameters for the mining model are set, you can process the model. The method for processing the **Forecasting** mining model is the same as that for the **Targeted Mailing** models. For more information, see "Targeted Mailing" earlier in this document.

Exploring the Mining Model

Now that the model is built and processed, you can explore the results using the Time Series viewer in the Mining Model Viewer tab. The Time Series viewer contains two tabs: **Decision Tree** and **Charts**. For more information about these tabs, see "Viewing with Time Series Viewer" in SQL Server Books Online.

The Microsoft Time Series algorithm builds a model for each distinct series that exists in the dataset. For example, each model for each region in the dataset contains information about sales over the time period. Therefore, a separate time series exists for each model in each region for both quantity and amount.

In this section, you will explore the time series for the amount of sales for Europe, North America, and the Pacific.

Decision Tree Tab

The **Decision Tree** tab (Figure 26) enables you to look at the decision tree that was created when the model was processed. In **Tree**, select the **M200 Pacific: Amount** model.

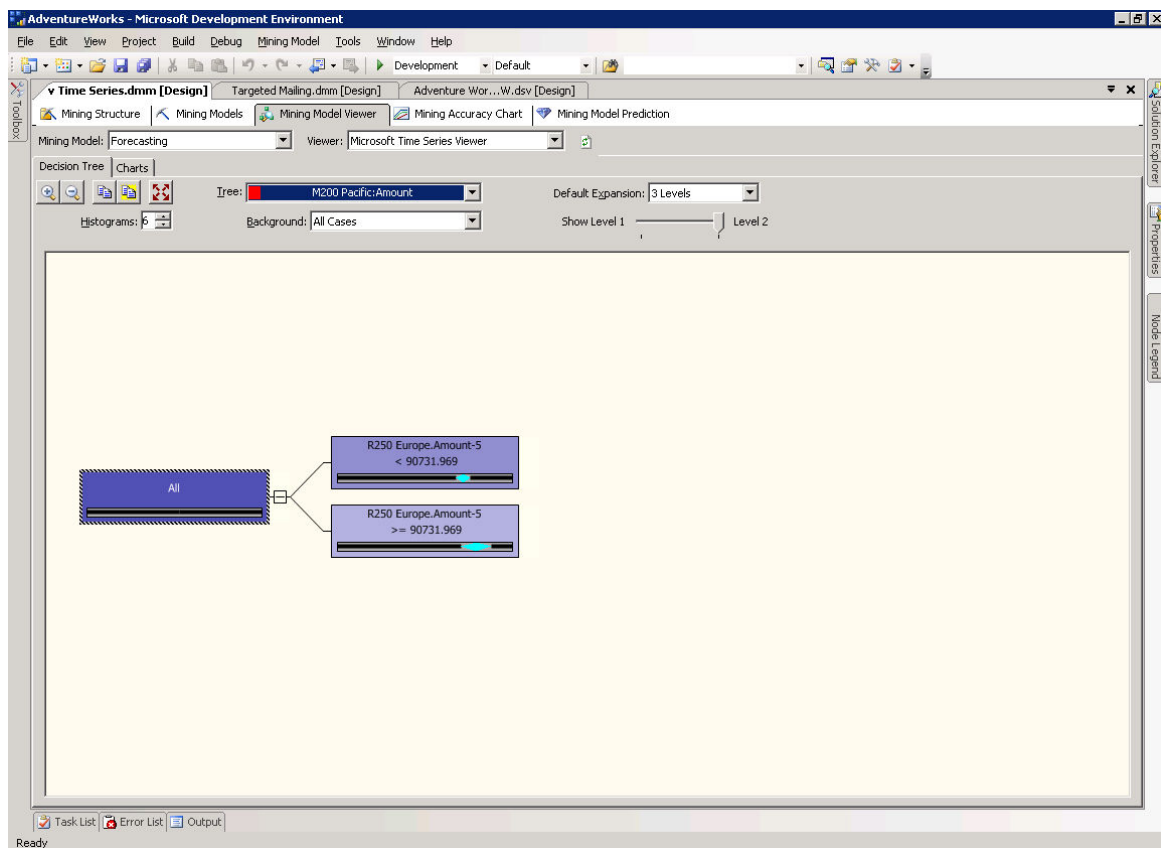


Figure 26 Decision Tree tab of the Forecasting model

Each node in the decision tree displays three pieces of information:

- The concentration of cases for the state of the predictable attribute specified in the **Background** control. The **Node Legend** window and ToolTip give the exact number of cases.
- The regression formula for the node.
- A diamond chart that represents the range of the range of the attribute. The diamond is located at the mean for the node, and the width of the diamond represents the variance of the attribute at that node. A thinner diamond indicates that the node can create a higher quality prediction.

Charts Tab

Using the **Charts** tab, you can investigate the time series that are created by the algorithm.

To select a time series

1. Select the following time series from the drop-down list.
 - **R750 Europe:Amount**
 - **R750 North America:Amount**
 - **R750 Pacific:Amount**

2. Click **OK**.

The legend on the right side of the viewer lists the series selected from the drop-down list along with check boxes. By selecting and clearing check boxes in the legend, you can control which time series are displayed in the viewer.

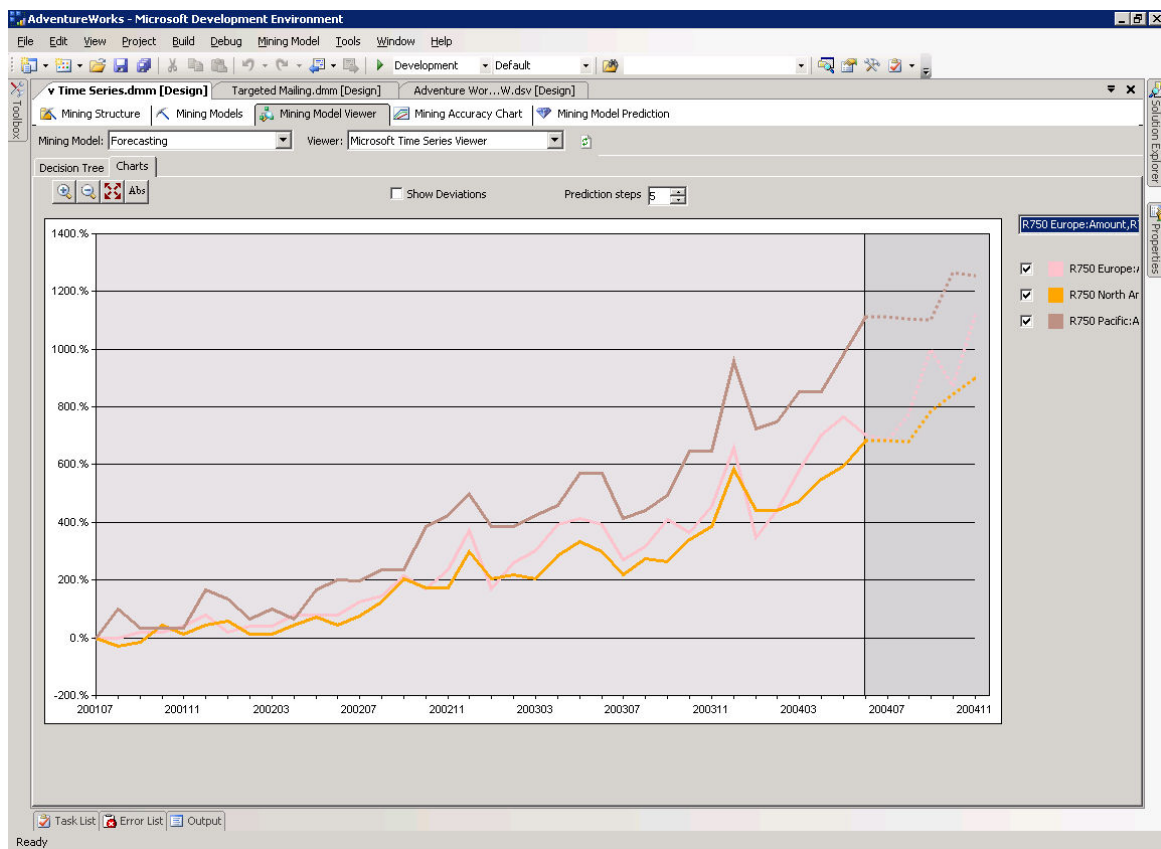


Figure 27 Charts tab of the Forecasting model

The chart displays both historical and future data. Future data is shaded to differentiate it from historical data. Use **Prediction Steps** to control how many future steps of data are displayed. Use **Show Deviations** to add error bars to the predictions.

As you can see in Figure 27, sales are generally increasing with a peak every 12 months (in December). The predictions continue this trend.

Market Basket

The marketing department of Adventure Works is interested in improving their Web site so that they can promote cross-selling.

Before they can update the site, they need to create a data mining model that can predict which products customers might want based on what customers already have in their baskets. These predictions will also assist the marketing department in placing items that tend to be bought together in close proximity on the Web site.

Upon completion of the task, the marketing department will have a model that can predict additional items that can appear in a shopping basket, or that a customer would like to add to a basket. In addition, they will have a complete mining model that shows groups of items from historical customer transactions.

In order to complete the task, the analyst will use the Microsoft Association algorithm. The scenario consists of three tasks:

- Create the mining model structure.
- Edit the mining model.
- Explore the mining model.

Create a Market Basket Mining Model Structure Using the Wizard

The first step is to use Mining Model Wizard to create a new mining structure. The Mining Model Wizard also creates an initial mining model. For the market basket scenario, you will use the wizard to create a mining structure and an associated mining model based on the Microsoft Association algorithm.

To create the Association mining structure

1. In Solution Explorer, right-click **Mining Structures**, and then click **New Mining Structure**.

The Mining Model Wizard opens.

2. On the Welcome page, click **Next**
3. Click **From existing relational database or data warehouse**, and then click **Next**.
4. Under **What data mining technique do you want to use?**, click **Microsoft Association Rules**.
5. Click **Next**.

By default, Adventure Works DW is selected in the **Select data source view** window.

6. Select the **Case** check box next to the **vAssocSeqOrders** table and the **Nested** check box next to the **vAssocSeqLineItems** table, and then click **Next**.
7. Clear the **Key** check box next to **CustomerKey** and the **Key** and **Input** check boxes next to **LineNumber**.

By default, **CustomerKey**, **OrderNumber**, and **LineNumber** are listed as **Key** types. **OrderNumber** will only be used as a key for Microsoft Association Rules model, so you must change the default setting.

8. Select the **Key**, **Input** and **Predictable** check boxes next to the **Model** column. Make sure that the final selection is the same as shown in figure 28a.

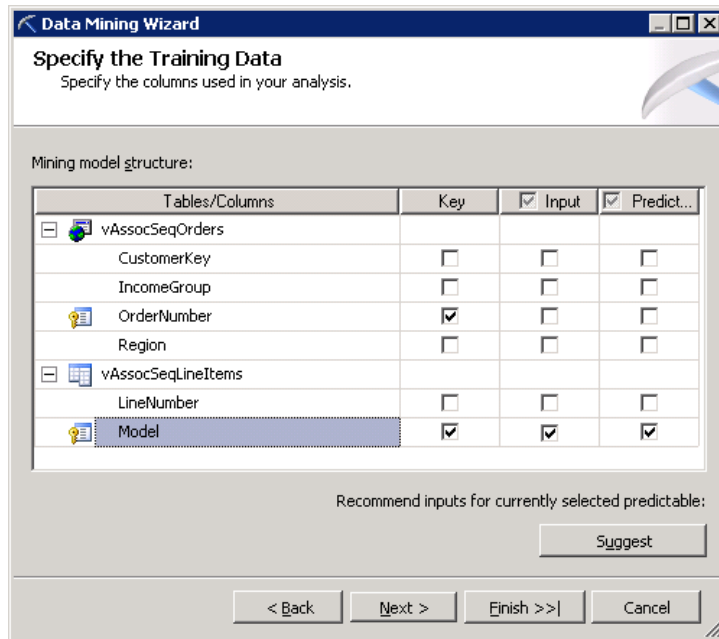


Figure 28a Attribute specification for the Association mining structure.

9. Click **Next**.
10. Click **Next**.
11. In Both Mining Structure Name and Mining **Model Name** box, type **Association**, and then click **Finish**.

The data mining editor opens, displaying the mining structure you created. Figure 28b shows the mining structure and mining model you created.

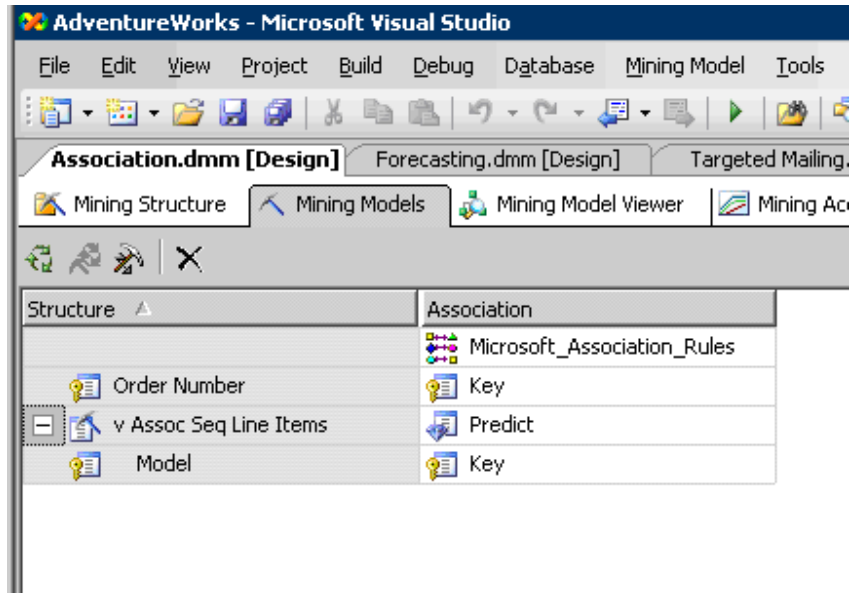


Figure 28b Mining Structure tab of the Association mining structure

Edit the Mining Model

Before you process the model, you must change the default values of two of the parameters: Support and Probability. Support defines the percentage of cases in which a rule must exist before it is considered to be a valid rule. Probability defines how likely an association must be before it can be considered valid.

To adjust Association model parameters

1. Select the **Mining Models** tab.
2. Right-click **Association**, and select **Set Algorithm Parameters**.

The **Algorithm Parameters** dialog box opens.

3. Set the following parameters:

Parameter	Value
MINIMUM_PROBABILITY	0.1
MINIMUM_SUPPORT	0.01

Process the Mining Model

Now that the structure and parameters for the mining model are set, you can process the model. The method for processing the Market Basket mining model is the same as with the Targeted Mailing models. For more information, see "Targeted Mailing" earlier in this document.

Explore Mining Models in the Editor

To open the Association viewer, select the **Mining Model Viewer** tab. The Association viewer contains three tabs: **Itemsets**, **Rules**, and **Dependency Net**. For more information about the Association viewer, see "Viewing with Association Viewer" in SQL Server Books Online.

Itemsets

The **Itemsets** tab displays three important pieces of information that relate to the itemsets found by the Microsoft Association algorithm: the support (the number of transactions that the itemset shows up in), the size (how many items are in the itemset), and the actual makeup of the itemset. Depending on how the algorithm parameters are set, the algorithm can generate a large number of itemsets. Using the controls at the top of the tab, you can filter the viewer to show only itemsets containing a specified minimum support and itemset size.

You can also use the **Filter itemset** box to filter the itemsets shown in the viewer. For example, to see only the itemsets that contain information about the Mountain-200 bicycle, type *Mountain-200*.

AdventureWorks - Microsoft Development Environment

Market Basket.dnm [Design] | Time Series.dnm [Design] | Targeted Mailing.dnm [Design] | Adventure Wor...W.dsv [Design]

Mining Structure | Mining Models | Mining Model Viewer | Mining Accuracy Chart | Mining Model Prediction

Mining Model: Association | Viewer: Microsoft Association Rules Viewer

Itemsets | Rules | Dependency Network

Minimum support: 213 | Filter Itemset: Mountain-200

Minimum itemset size: 0 | Show: Show attribute name and value

Maximum rows: 2000

Support	Size	Itemset
2477	1	Mountain-200 = Existing
730	2	Fender Set - Mountain = Existing, Mountain-200 = Existing
725	2	Mountain Bottle Cage = Existing, Mountain-200 = Existing
710	2	Mountain-200 = Existing, Sport-100 = Existing
589	2	Mountain-200 = Existing, Water Bottle = Existing
589	3	Mountain Bottle Cage = Existing, Mountain-200 = Existing, Water Bottle = Existing
500	2	HL Mountain Tire = Existing, Mountain-200 = Existing
331	2	Mountain-200 = Existing, Mountain Tire Tube = Existing
331	3	HL Mountain Tire = Existing, Mountain-200 = Existing, Mountain Tire Tube = Existing
327	2	Mountain-200 = Existing, Patch kit = Existing
220	3	Mountain Bottle Cage = Existing, Mountain-200 = Existing, Sport-100 = Existing

Task List | Error List | Output

Ready

Figure 29 Itemsets tab of the Microsoft Association algorithm

As you can see in Figure 29, only itemsets that contain the words "Mountain-200" are displayed. Each itemset returned contains information about transactions in which a Mountain-200 bicycle was sold. For example, the itemset that contains the value 710 under **Support** tells you that out of all of the transactions, 710 people who bought the Mountain-200 bicycle also bought the Sport-100 bicycle.

Rules

The **Rules** tab displays three pieces of information relating to the rules that the algorithm found.

Probability

The likelihood of a rule occurring.

Importance

A measure of the usefulness of a rule, with a higher value meaning a better rule. Simply looking at the probability can be misleading. For example, if every transaction contains an item *x*, the rule *y* predicts *x* has a probability of 1 — it will always occur. Even though the accuracy of the rule is very good, it does not relay very much information, because every transaction contains *x* regardless of *y*.

Rule

The definition of the rule.

As with the **Itemsets** tab, the rules can be filtered so that only the most interesting rules are shown. For example, suppose you only want to see the rules that include the Mountain-200 bicycle. If you type *Mountain-200* in the **Filter Rule** box, you receive the results shown in Figure 30.

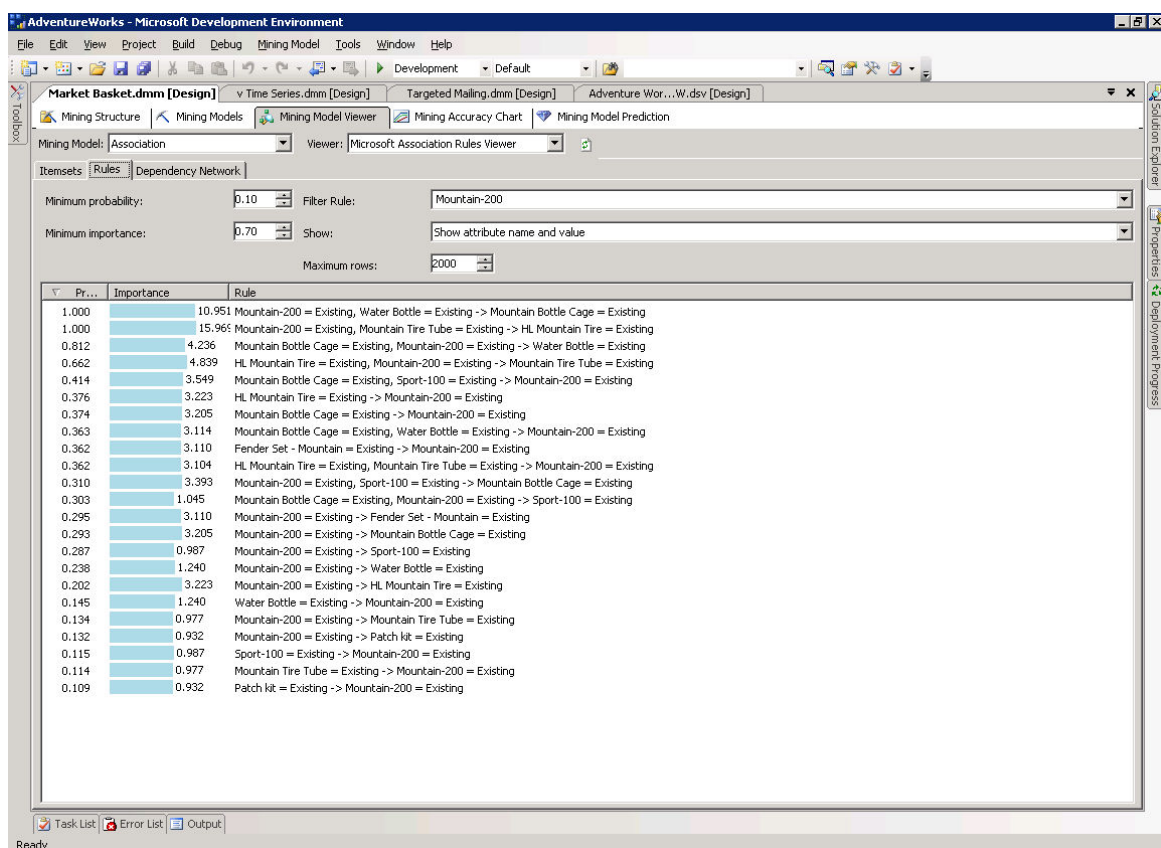


Figure 30 Rules tab of the Microsoft Association algorithm

As you can see in Figure 30, only rules that contain the words "Mountain-200" are displayed. Each rule can be used to predict the presence of an item in a transaction based on the presence of other items. For example, the first rule tells you that if someone buys a Mountain-200 bicycle and a 30-ounce water bottle, there is a probability of 1 that the person will also buy a Mountain bottle cage.

Dependency Net

Using the **Dependency Net** tab, you can investigate the interaction of the different items in the model. Each node in the viewer represents an item; for example, Mountain-200 = Existing (meaning that Mountain 200 exists in a transaction). By selecting a node, you can use the color legend at the bottom of the tab to determine which other items either determine, or are determined by, other items in the model.

The slider is associated with the probability of a rule. By sliding up or down, you can filter out weak associations.

For example, in the **Show** box, click **Show attribute name only**, and then click **Mountain bottle cage**. Zooming in, you see the Association viewer pictured in Figure 30. The viewer shows that Mountain bottle cage both predicts and is predicted by 30-ounce water bottle and Mountain-200, meaning that these items are likely to show up in a transaction together. This makes sense — if someone buys a bike, he or she is likely to buy a water bottle holder and water bottle.

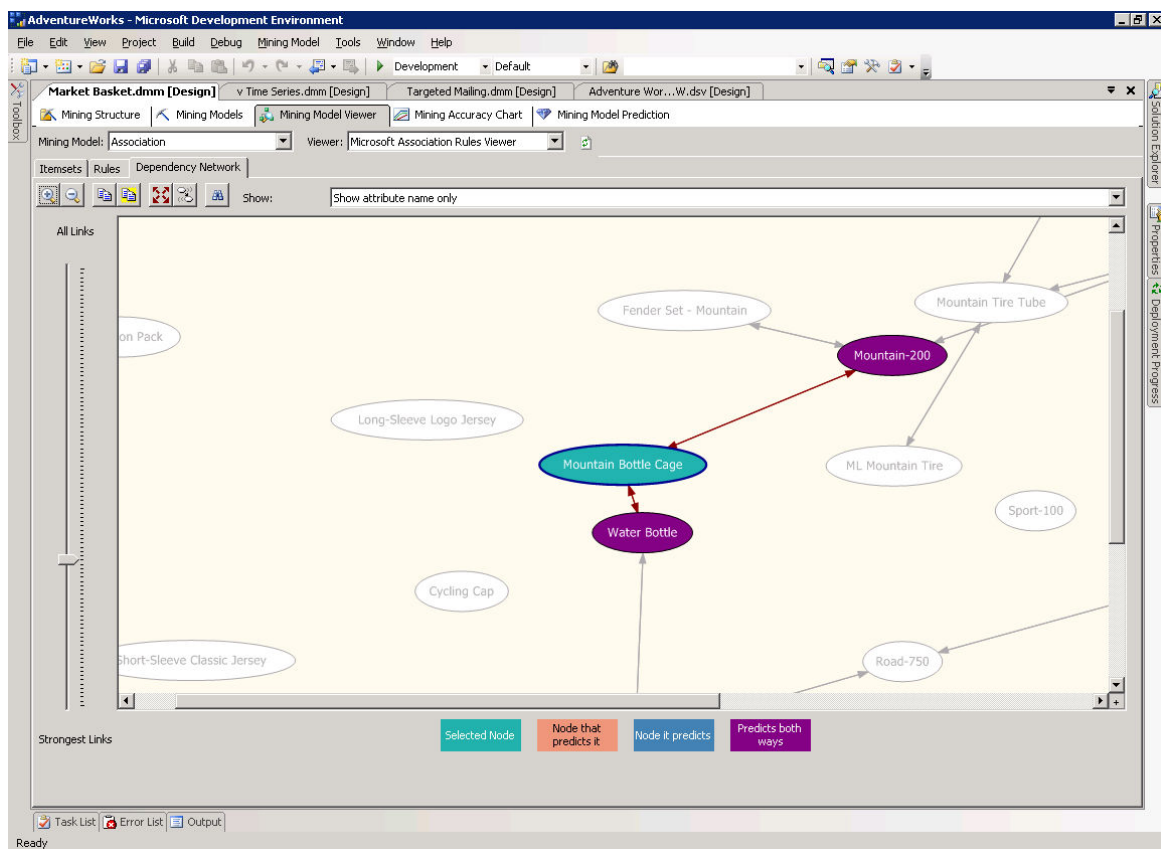


Figure 31 Dependency Net tab of the Microsoft Association algorithm

Sequence Clustering

The marketing department of Adventure Works is interested seeing how customers move through the Adventure Works Web site. They suspect that there is a pattern to the order in which customers put products into their shopping cart. Using the Microsoft Sequence Clustering algorithm, they can find the sequences by which customers add items to their carts. They can then use this information to streamline the flow of the Web site so that it leads customers to purchase additional products.

Upon completion of this task, the marketing department will have a model capable of predicting what a customer will put in his or her shopping basket next.

In order to complete the task, the analyst will use the Microsoft Sequence Clustering algorithm. The scenario consists of two tasks:

- Create the mining model structure.
- Explore the mining model.

Create a Sequence Clustering Mining Model Structure Using the Wizard

The first step is to use Mining Model Wizard to create a new mining structure. The Mining Model Wizard also creates an initial mining model based on the Microsoft Sequence Clustering algorithm.

To create the Sequence Clustering mining structure

1. In Solution Explorer, right-click **Mining Structures**, and then click **New Mining Structure**.

The Mining Model Wizard opens.

2. On the Welcome page, click **Next**.
3. Click **From existing relational database or data warehouse**, and then click **Next**.
4. Under **What data mining technique do you want to use?**, click **Microsoft Sequence Clustering**.
5. Click **Next**.

By default, Adventure Works DW is selected in the **Select data source view** window. Click **Browse** to view the tables in the data source view inside of the wizard.

6. Select the **Case** check box next to the **vAssocSeqOrders** table and the **Nested** check box next to the **vAssocSeqLineItems** table.
7. Click **Next**.
8. Clear the **Key** check box next to **CustomerKey**.

By default, **OrderNumber** and **LineNumber** are listed as **Key** types, which is correct.

9. Select the **Input** and **Predictable** check boxes next to the **Model** columns. Make sure that the final selection is the same as shown in figure 32a.

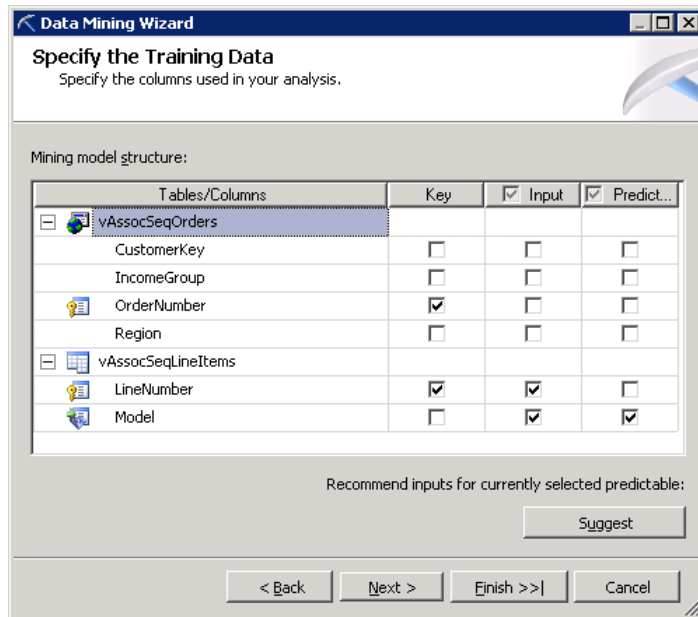


Figure 32a Attribute specification for the sequence clustering mining structure.

10. Click **Next**.
11. In both **Mining Structure Name** and **Mining Model Name** box, type *Sequence Clustering*, and then click Finish.

The data mining editor opens, displaying the new Sequence Clustering mining structure.

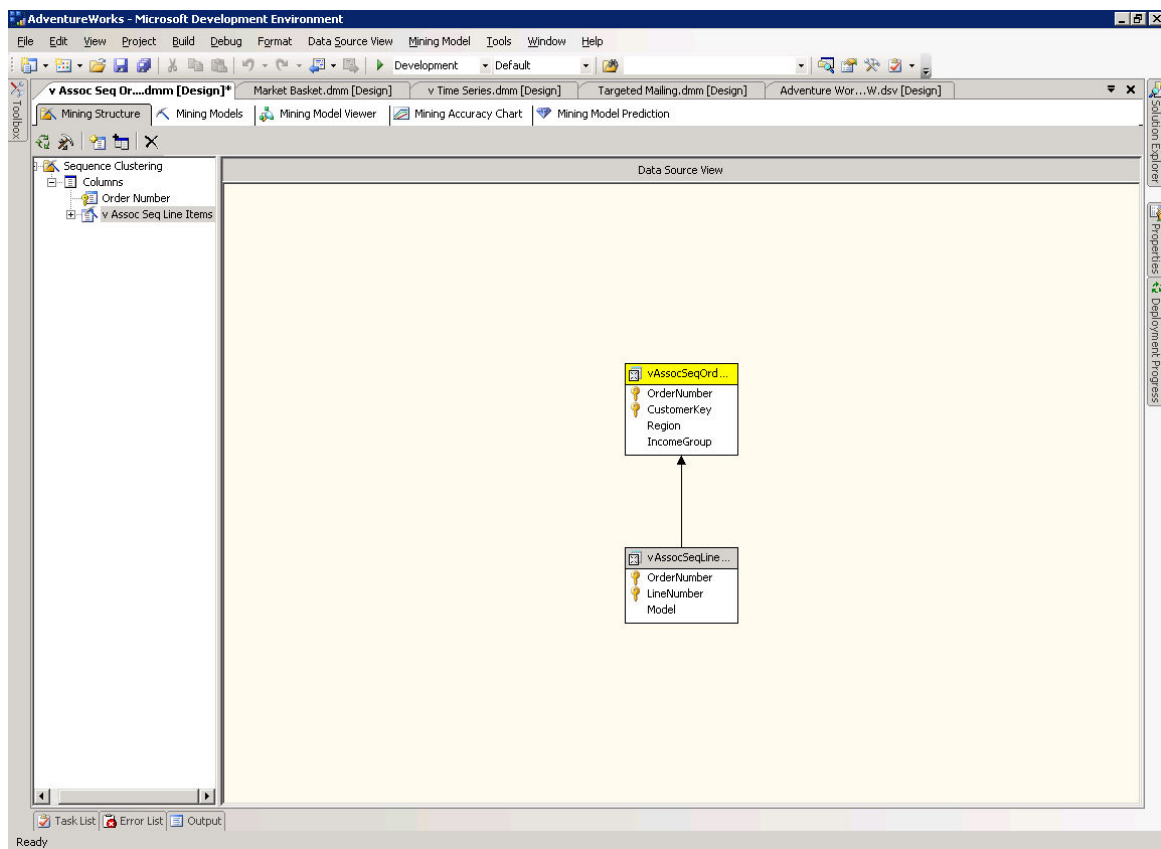


Figure 32b Sequence Clustering mining structure

You do not need to make any changes to the mining model structure or mining model in the data mining editor, so you can just process the model.

Process the Mining Model

The method for processing the Sequence Clustering mining model is the same as that for the Targeted Mailing models. For more information, see "Targeted Mailing" earlier in this document.

Explore Mining Models in the Editor

Use the Sequence Clustering viewer to explore the mining model you created for the sequence clustering scenario. To open the Sequence Clustering viewer, click **Mining Model Viewer**. Like the Cluster viewer, the Sequence Clustering viewer contains five tabs: **Cluster Diagram**, **Cluster Profiles**, **Cluster Characteristics**, **Cluster Discrimination**, and **State Transitions**. For more information about using the Sequence Clustering viewer, see "Viewing with Sequence Clustering Series Viewer" in SQL Server Books Online.

Cluster Diagram

The **Cluster Diagram** tab graphically displays the clusters the algorithm discovered in the database. The layout represents the cluster relationships where similar clusters are grouped close together. By default, the node color represents the density of all cases in the cluster — the darker the node, the more cases it contains. You can also change the meaning of node color-coding so that it represents an attribute and state. For example, to generate the diagram shown in Figure 33, in the **Shading Variable** list, click **Model**, and in the **State** list, click **Cycling Cap**.

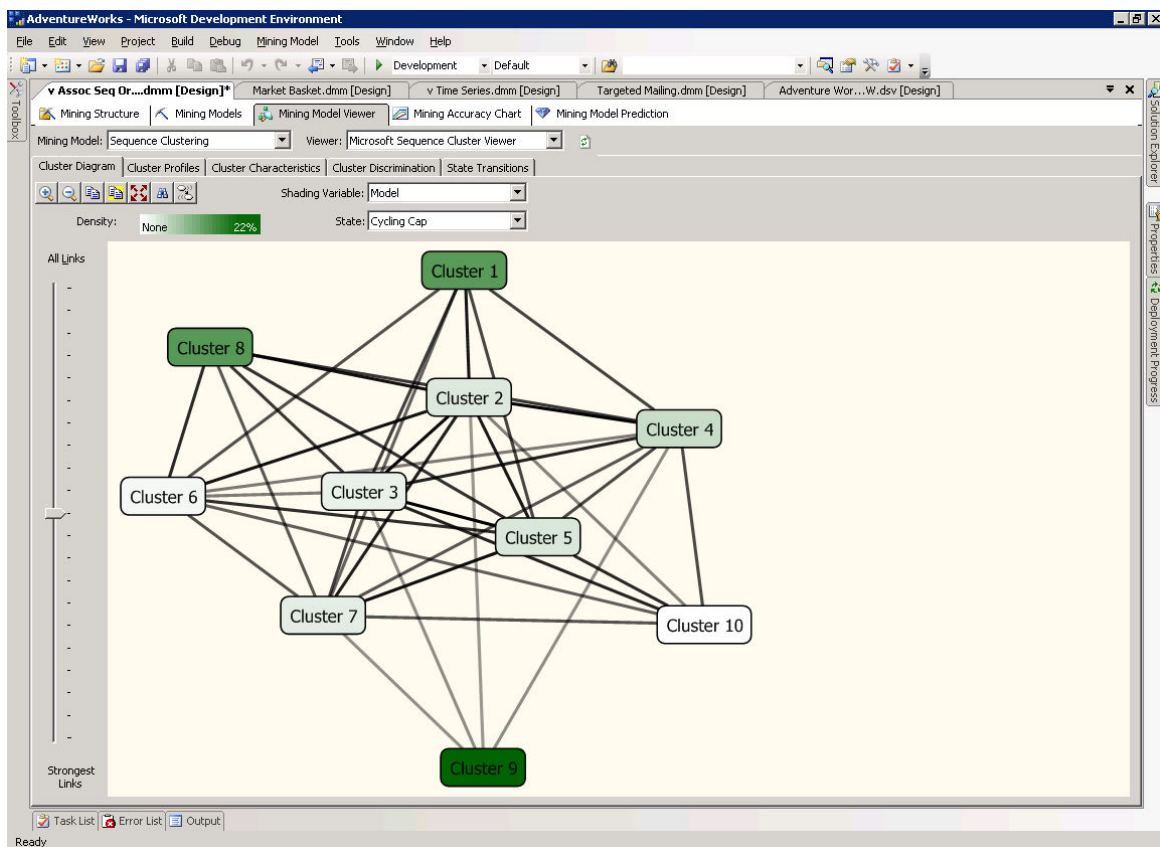


Figure 33 Cluster Diagram tab of the Microsoft Sequence Clustering model

You can see that Cluster 9 contains the highest density of cycling caps.

Cluster Profiles

The **Cluster Profiles** tab displays the sequences that exist in each cluster. The clusters are listed in individual columns to the right of the **States** column, and the rows listed in the **Variables** column show the variable distributions for a cluster.

In Figure 34, the **Model.samples** row represents sequence data, and the **Model** row describes the overall distribution of items in a cluster. Each line of the color sequences displayed in each cell of the **Model.samples** row represents the behavior of a randomly selected user in the cluster. Each color in the sequence histogram represents a product model.

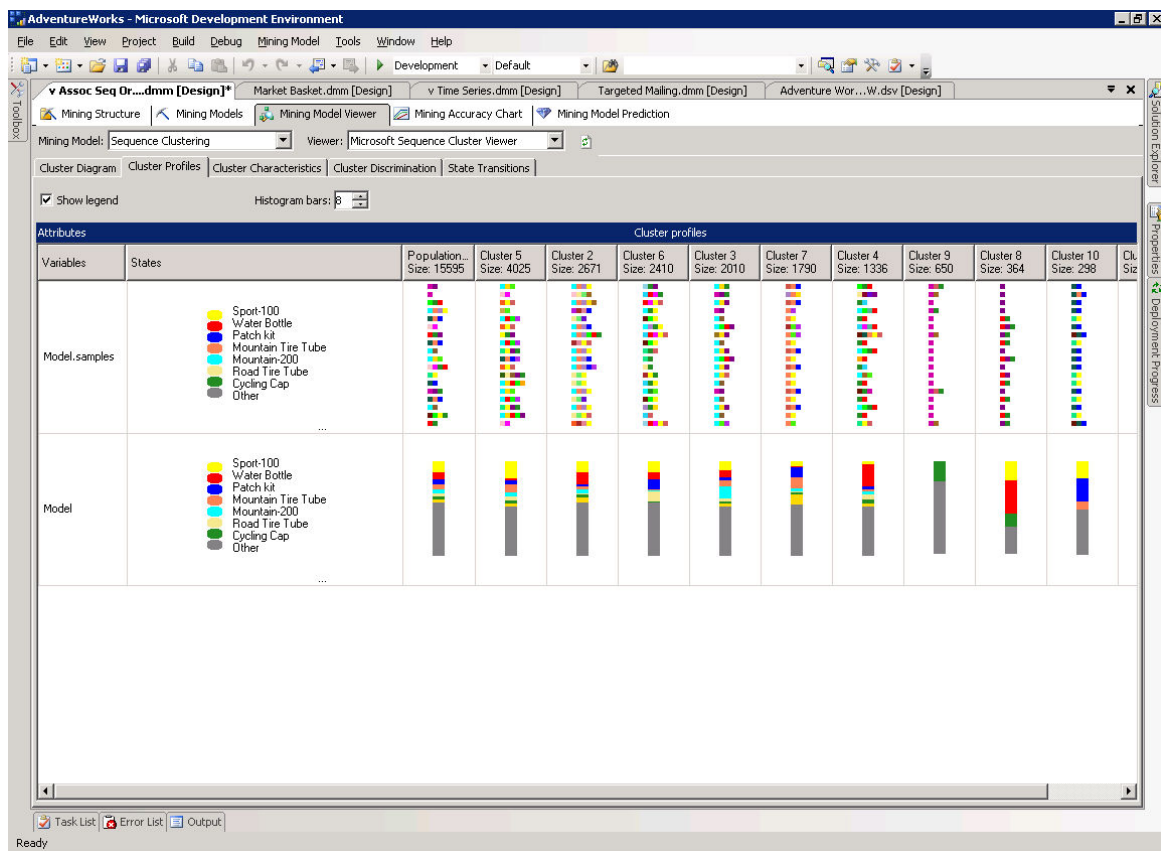


Figure 34 Cluster Profiles tab of the Microsoft Sequence Clustering model

For example, the aqua color in cluster 3 represents the Mountain-200 bicycle. Its presence as the first color in most of the sequences means that a customer is very likely to place the Mountain-200 bike in the shopping basket first.

Cluster Characteristics

The **Cluster Characteristics** tab summarizes the transitions between states in a cluster, with bars describing the importance of the attribute value for the selected cluster. For example, in Cluster 10, one of the most important profiles is that customers tend to place a ML Mountain tire and in the shopping cart first.

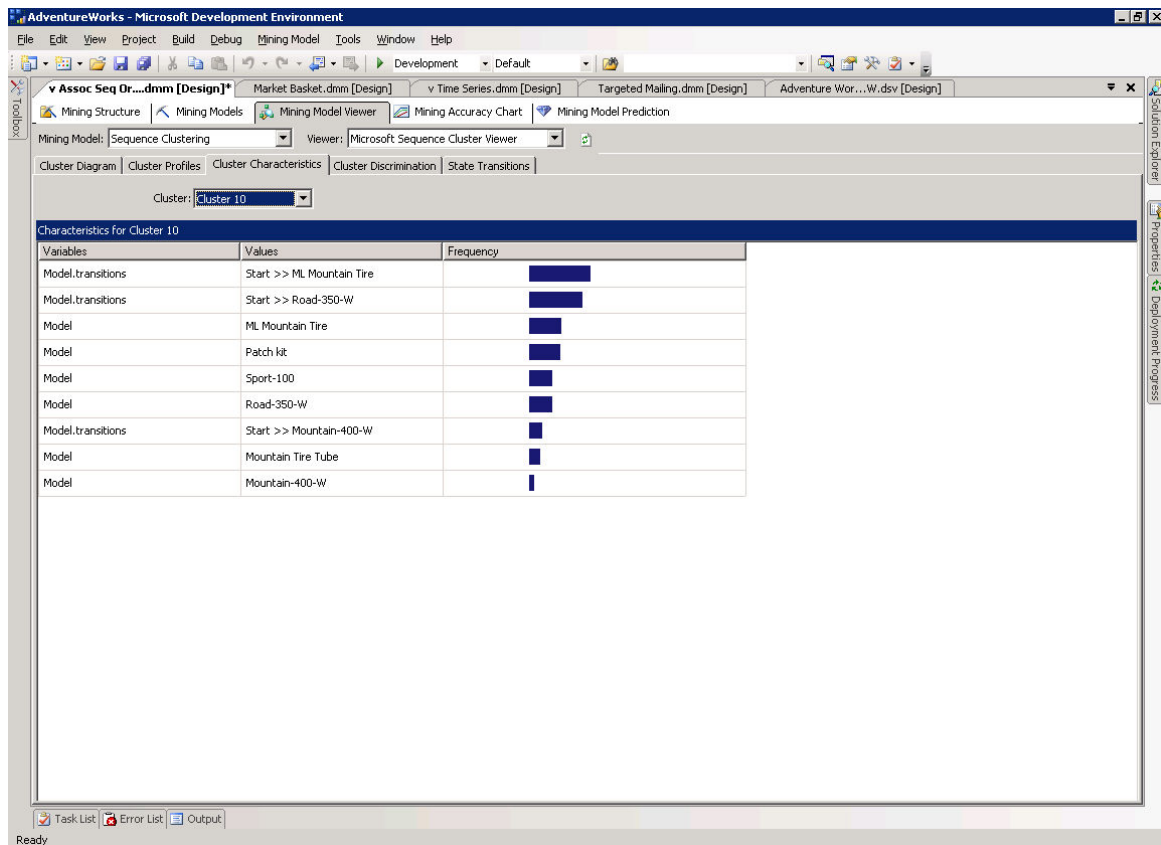


Figure 35 Cluster Characteristics tab of the Microsoft Sequence Clustering model

Cluster Discrimination

Using the **Cluster Discrimination** tab, you can compare two clusters, determining which models favor which clusters. The tab contains four columns: **Variables**, **Values**, **Favors Cluster (i)**, **Favors Cluster (i)**. If the cluster favors a specific model, a blue bar appears in one of the **Favors Cluster(i)** columns, in the row of the model listed in the **Variables** column. The longer the blue bar, the more the model favors the cluster.

For example, Figure 36 compares Cluster 2 with Cluster 5. A customer who purchases a bottle cage for a mountain bike (indicated by **Mountain Bottle Cage** in the **Values** column) is more likely to be in Cluster 5, and a customer who purchases a Touring tire (indicated by **Touring Tire** in the **Values** column) is more likely to be grouped into Cluster 2.

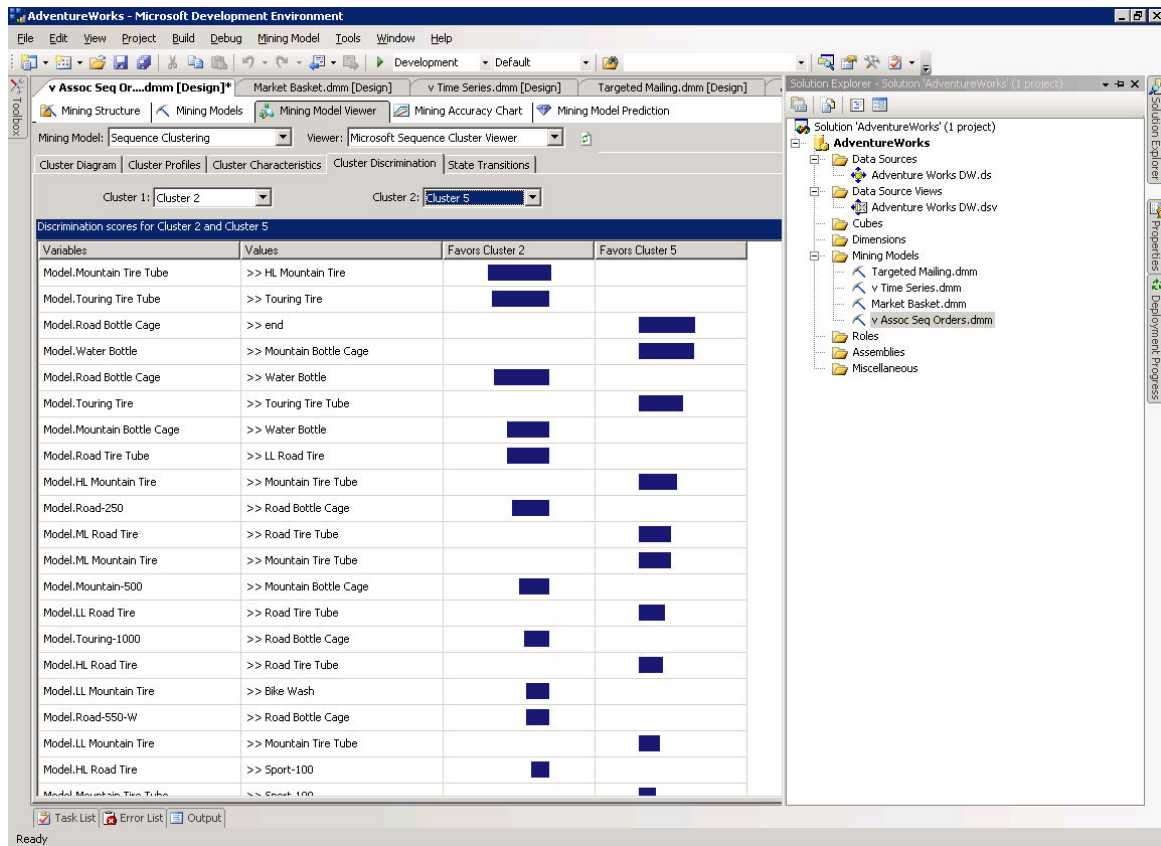


Figure 36 Cluster Discrimination tab of the Microsoft Sequence Clustering model

State Transitions

On the **State Transitions** tab, you can select a cluster and browse through its state transitions. Each node represents a state of the model (such as Mountain-200). A line represents the transition between states, and each node is based on the probability of a transition. The background color represents the frequency of the node in the cluster.

For example, select **Cluster 3** from **Cluster**, select the **Touring-3000** node, and lower the **All Links** slider a couple of notches. As you can see in Figure 37, if users put a Touring Tire into his shopping cart, there is a probability of 0.63 (indicated by the blue arrow) that he will next put a Touring Tire Tube into the cart, and a probability of 0.26 that he will end his shopping by placing a Sport 100 bike into his shopping cart.

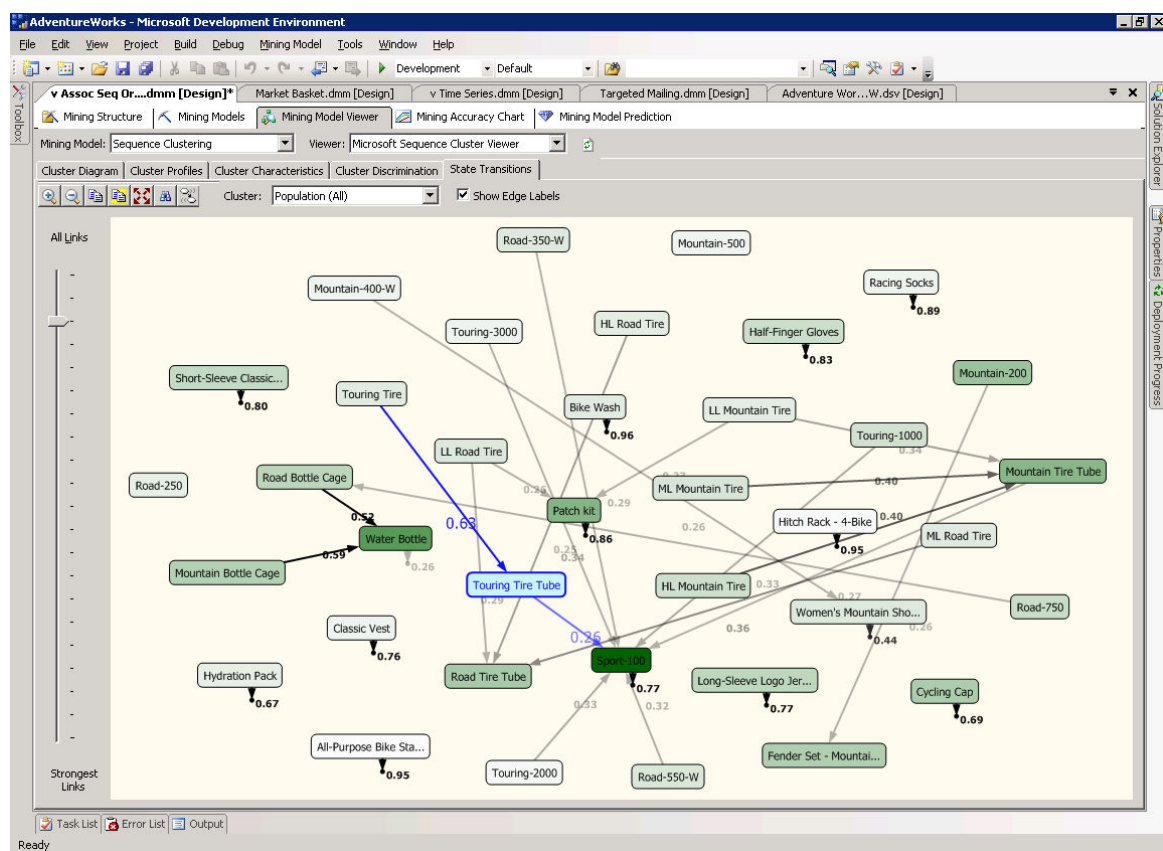


Figure 37 Cluster Transitions tab of the Microsoft Sequence Clustering model