Introduction 1
SAS Enterprise Miner Interface 4
Tutorial Script 5
  Starting SAS Enterprise Miner, Creating a Project, and Defining a Library 5
  Creating Data Sources 8
  Creating a Diagram 15
  Developing a Scorecard 17
    Grouping the Characteristic Variables into Attributes 18
    Regression and Scaling 27
    Reject Inference 36
    The Final Scorecard 40
References 41

Introduction

This tutorial covers how to use SAS Credit Scoring for Enterprise Miner to build a consumer credit scorecard. The tutorial assumes you are familiar with the process of credit scoring. It is focused on reviewing the features and functionality of the core set of credit scoring nodes, and should not be considered a complete review of SAS Enterprise Miner capabilities. The analysis typically would include other important steps, such as exploratory data analysis, variable selection, model comparison, and scoring.

Credit scoring, as defined by SAS, is

- applying a statistical model to assign a risk score to a credit application or an existing credit account
- building the statistical model
- monitoring the accuracy of one or more statistical models
- monitoring the effect that score-based decisions have on key business performance indicators

Credit scoring is the set of decision models and their underlying techniques that aid lenders in the granting of consumer credit. These techniques describe who will get credit, how much credit they should receive, and what operational strategies will enhance the profitability of the borrowers to the lenders (Thomas, Edelman, and Crook 2002).
Although credit scoring is not as glamorous as the pricing of exotic financial derivatives, it is one of the most successful applications of statistical and operations research techniques in finance and banking. Without an accurate and automated risk assessment tool, the phenomenal growth of consumer credit would not have been possible over the last 40 years (Thomas, Edelman, and Crook 2002).

In its simplest form, a scorecard is built from a number of characteristics (that is, input or predictor variables). Each characteristic is comprised of a number of attributes. For example, age is a characteristic, while “25-33” is an attribute. Each attribute is associated with a number of scorecard points. These scorecard points are statistically assigned to differentiate risk, based on the predictive power of the characteristic variables, correlation between the variables, and business considerations.

For example, using the Example Scorecard in Figure 1, an applicant who is 35, makes $38,000, and owns his own home would be accepted for credit by this financial institution’s scorecard.

The total score of an applicant is the sum of the scores for each attribute present in the scorecard. Smaller scores imply a higher risk of default, and vice versa.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Attribute</th>
<th>Scorecard Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGE</td>
<td>&lt;22</td>
<td>100</td>
</tr>
<tr>
<td>AGE</td>
<td>22&lt;=AGE&lt;26</td>
<td>120</td>
</tr>
<tr>
<td>AGE</td>
<td>26&lt;=AGE&lt;29</td>
<td>180</td>
</tr>
<tr>
<td>AGE</td>
<td>29&lt;=AGE&lt;32</td>
<td>200</td>
</tr>
<tr>
<td>AGE</td>
<td>32&lt;=AGE&lt;37</td>
<td>210</td>
</tr>
<tr>
<td>AGE</td>
<td>37&lt;=AGE&lt;42</td>
<td>220</td>
</tr>
<tr>
<td>AGE</td>
<td>&gt;=42</td>
<td>250</td>
</tr>
<tr>
<td>HOME</td>
<td>OWN</td>
<td>225</td>
</tr>
<tr>
<td>HOME</td>
<td>RENT</td>
<td>110</td>
</tr>
<tr>
<td>INCOME</td>
<td>&lt;10000</td>
<td>120</td>
</tr>
<tr>
<td>INCOME</td>
<td>10000&lt;=INCOME&lt;17000</td>
<td>140</td>
</tr>
<tr>
<td>INCOME</td>
<td>17000&lt;=INCOME&lt;28000</td>
<td>180</td>
</tr>
<tr>
<td>INCOME</td>
<td>28000&lt;=INCOME&lt;35000</td>
<td>200</td>
</tr>
<tr>
<td>INCOME</td>
<td>35000&lt;=INCOME&lt;42000</td>
<td>225</td>
</tr>
<tr>
<td>INCOME</td>
<td>42000&lt;=INCOME&lt;58000</td>
<td>230</td>
</tr>
<tr>
<td>INCOME</td>
<td>&gt;=58000</td>
<td>260</td>
</tr>
</tbody>
</table>

Let cutoff=600

So, a new customer applies for credit.....

AGE          35           210 points
INCOME        $38K       225 points
HOME         OWN        225 points

Total                          660 points

Decision:               GRANT CREDIT

Figure 1: Example Scorecard
Figure 2 displays a process flow diagram that can be used to develop a scorecard using SAS Credit Scoring for Enterprise Miner 5.3.

SAS Credit Scoring for Enterprise Miner adds these specific nodes to the tool set to support scorecard development:

- **Interactive Grouping** node (IGN) facilitates the grouping (that is, the classing of characteristics into attributes or bins) of input variables that is essential to building a credit scorecard. While performing an initial automatic grouping that can provide optimal splits, this node enables the user to re-group the variables through an interactive interface. Additionally it also has the capability of variable screening/selection.

- **Scorecard** node follows the Interactive Grouping node and uses the grouped variables as inputs in a logistic regression model. In addition, it scales the regression parameters to compute score points and the resulting scorecard. Finally, the Scorecard node performs score and characteristic (variable) analysis that helps in understanding the scorecard, and aids in crafting a score-based strategy.
- **Reject Inference** node offers three standard, industry accepted methods for inferring the performance of the rejected applicant data by the use of a model built on accepted applicants.
- **Credit Exchange** node enables the use of scorecards in the SAS Credit Risk Solution. Since it plays no part in the development of the scorecard, it is beyond the scope of this tutorial.

## SAS Enterprise Miner Interface

Figure 3 displays and describes the interface.

![SAS Enterprise Miner Interface](image)

**Figure 3: SAS Enterprise Miner Interface**
The SAS Enterprise Miner 5.3 interface simplifies many common tasks associated with the construction of predictive models. The interface is divided into six interface components:

- **Toolbar**—The toolbar in SAS Enterprise Miner is a graphical set of node icons and tools that you use to build process flow diagrams in the diagram workspace. To display the text name of any node or tool icon, position your mouse pointer over the icon. The Credit Scoring tab contains the SAS Credit Scoring for Enterprise Miner add-on tools.
- **Project panel**—Use the project panel to manage and view data sources, diagrams, results, and project users.
- **Properties panel**—Use the properties panel to view and edit the settings of data sources, diagrams, nodes, results, and users.
- **Diagram workspace**—Use the diagram workspace to build, edit, run, and save process flow diagrams. This is where you graphically build, order, and sequence the nodes that you use to mine your data and generate reports.
- **Help panel**—The help panel displays a short description of the property that you select in the properties panel. Extended help can be found in the Help Topics selection from the Help main menu.
- **Status bar**—The status bar is single pane at the bottom of the window that indicates the execution status of a SAS Enterprise Miner task.

The tutorial script that follows accompanies two associated SAS data sets, ACCEPTS and REJECTS, that contain, respectively, data regarding accepted applicants and data regarding rejected applicants. Both SAS data sets will be used during the tutorial, as well as the data set containing split points for a variable called LTV (loan to value) that is used to show the import feature of the Interactive Grouping node.

---

**Tutorial Script**

**Starting SAS Enterprise Miner, Creating a Project, and Defining a Library**

On your local machine, create a directory called C:\EM_DEMO (or use another directory that you have write access to) and load the ACCEPTS and REJECTS data sets in that directory.

1. Start SAS OnDemand for Academics: Enterprise Miner by double-clicking the link in your SAS OnDemand for Academics Control Center. The following screen appears.
2. Enter your user name and password, and then click **Log On** to invoke SAS Enterprise Miner.
3. Click **New Project**. The following screen appears. In the **Name** field, enter the name of the project: EM5_Credit_Scoring_Demo.

![Create New Project](image)

4. Click the **Start-Up Code** tab in the Create New Project window and enter the following code.

![Create New Project](image)

The first statement defines a SAS global macro variable called EM_IGN_ApplyLevelRule. The second statement turns this SAS global macro variable off. This code ensures that if an input interval variable has a number of unique levels less than the number specified in the Number of Bins property option in the Interactive Grouping node, then that input will be pre-binned. Pre-binning will be discussed later in the Interactive Grouping section of this tutorial.

5. Click **OK**. (Tip: You can view the SAS Log to ensure that your library was assigned correctly by selecting **View->Log**.)

You now have created a SAS Enterprise Miner project named EM5_Credit_Scoring_Demo and your screen should look like the following screen.
Creating Data Sources

In this section, you define the ACCEPTS SAS data set as a SAS Enterprise Miner data source. A SAS Enterprise Miner data source defines all the information about a SAS table or view to another file type that is needed for data mining, including the name and location of the data set, variable roles, measurement levels, and other attributes that guide the data mining process. Once defined, the data sources can be used in any diagram within a project and can be copied from one project to another.

It is important to note that data sources are not the actual training data, but instead are the metadata that defines the source data. The source data itself must reside in an allocated library. You have already allocated the libref DEMODATA as part of the start-up code for the project.

1. Right-click the Data Sources folder in the Project Navigator (or select File->New->Data Sources) to open the Data Source Wizard.
2. Because the input data is already uploaded to the SAS Server, select **Metadata Repository** from the drop-down menu, and then click **Next** in the Metadata Source window. Note that other products such as SAS Data Integration Studio and SAS Enterprise Guide can be used to build and, in turn, register tables to the SAS Metadata for retrieval in SAS Enterprise Miner.

3. Click **Browse** to open the Select a SAS Table window.

4. Click the SAS library that you assigned in the start-up code, and then click the **ACCEPTS** SAS table.

5. Click **OK**, and then click **Next**.
The Data Table Properties sheet opens. There are 26 variables and 5,837 observations.

Metadata is data about data sets. Some metadata, such as field name, are stored with the data. Other metadata, such as how a particular variable in a data set should be used in a predictive model, must be manually specified. Defining modeling metadata is the process of establishing relevant facts about the data set prior to model construction.

6. Click **Next** to apply advisor options. Two options are available:

   a. **Basic**—Use the Basic option when you already know the variable roles and measurement levels. The initial role and level are based on the variable type and format values.

   b. **Advanced**—Use the Advanced option when you want SAS Enterprise Miner to automatically set the variable roles and measurement levels. Automatic initial roles and level values are based on the variable type, the variable format, and the number of distinct values contained in the variable.

   Click **Advanced**.
7. Click **Customize** to view additional variables rules you can impose. Selecting each rule provides a description of the rule. For example, **Missing Percentage Threshold** specifies the percentage of missing values required for a variable’s modeling role to be set to **Rejected**.

**Advanced Advisor Options**

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Missing Percentage Threshold</td>
<td>50</td>
</tr>
<tr>
<td>Reject Vars with Excessive Missing Values</td>
<td>Yes</td>
</tr>
<tr>
<td>Class Levels Count Threshold</td>
<td>20</td>
</tr>
<tr>
<td>Detect Class Levels</td>
<td>Yes</td>
</tr>
<tr>
<td>Reject Levels Count Threshold</td>
<td>20</td>
</tr>
<tr>
<td>Reject Vars with Excessive Class Values</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Missing Percentage Threshold**

Specify a maximum percentage of missing values for variables to be rejected. The default value is 50.
8. Click **OK** to use the defaults for this example.

9. Click **Next** in the Apply Advisor window to generate the metadata and open the columns metadata.

10. Click the **Name** column header to sort the variables alphabetically. Redefine these variable roles.

![Data Source Wizard - Column Metadata](image)

<table>
<thead>
<tr>
<th>Name</th>
<th>Role</th>
<th>Level</th>
<th>Report</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>age_oldest_tr</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>app_id</td>
<td>ID</td>
<td>Nominal</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>Input</td>
<td>Binary</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>bankruptcy</td>
<td>Input</td>
<td>Binary</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>down_grt</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>fico_score</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>gen_amt</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>gen_term</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>tv</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>meps</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>num_borrowers</td>
<td>Input</td>
<td>Binary</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>purch_price</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>purpose</td>
<td>Input</td>
<td>Binary</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>rev_util</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>rt_denn</td>
<td>Input</td>
<td>Interval</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>

a. Set the role for **BAD** to **Target**. This is the binary GOOD/BAD response (target) variable.

b. Set the role for **WEIGHT** to **Frequency**. This contains the sampling weights that will be used for oversampling adjustment.

You can use the **Show code** button to write SAS code to conditionally assign variable attributes. This is especially useful when you want to apply a metadata rule to several variables.

For example: If role eq “INPUT” and level eq “NOMINAL” and type eq “N” then level =“INTERVAL”.

11. View the distribution of target variable **BAD**.

a. Select the **BAD** variable, and then click **Explore**.

b. Select the bar for the “bads” (1’s). The “bads” are highlighted in the data table.
To display a tool tip indicating the number of “bads,” place your cursor over this bar.

c. Click Close to close the Explore window.

12. Click Next to open the Decision Processing window. For now, select No to forego decision processing.

13. Click Next to open the Data Source Attributes window. You can use the Role drop-down menu to set other roles such as Train and Score, but for this example leave the role as Raw.
14. Click **Finish** to add the donor table to the **Data Sources** folder in the Project Navigator.

The data source can be used in other diagrams. You can also define global data sources that can be used across multiple projects. (See the online documentation for instructions on how to do this.)

Expand the **Data Sources** folder. Select the **ACCEPTS** data source, and notice that the Property sheet now shows properties for this data source.
Repeat the process to create the REJECTS data source. There will be no target variable or weight variable for the rejected applicant’s data set. Make sure you set the Role of the REJECTS data set to Score in Step 7 of the Data Source Wizard.

To build a model, you must first create an analysis diagram.

**Creating a Diagram**

1. Right-click the Diagrams folder in the project panel, and then select Create Diagram.

2. Enter a diagram name (in this example, Scorecard1) and click OK.

3. Expand the Diagrams folder to see the open diagram.

4. Click the diagram Scorecard1 icon. The Property sheet now shows properties for the diagram.

5. Drag and drop the ACCEPTS data source from the Data Sources folder onto the diagram workspace.
Although you will focus on developing one process flow diagram, an advantage of SAS Enterprise Miner is that you can open multiple diagrams at one time. You can also disconnect from and reconnect to a diagram as long as you have also configured the Analytics Platform 1.4 as a middle-tier component when using a SAS Enterprise Miner solution. Other users can also access the same project. However, only one user can open a diagram at a time. SAS Enterprise Miner for SAS 9.1.3 is well designed for multitasking and project sharing.

6. Next, you will partition the data to create training and validation data sets. Drag and drop a **Data Partition** node from the **Sample** tab on the toolbar. You can also add nodes by right-clicking in the background of the diagram workspace and selecting **Add node**. Connect the **ACCEPTS** Input Data Source node to the **Data Partition** node. For instructions on how to connect nodes, press F1 and read the **Connecting nodes** topic in the Getting Started with Enterprise Miner online help.
7. Next, you will change the default data set partition allocations. In the Property panel, change the Training data set percentage to 70% and change the Validation data set percentage to 30%. We will not be using a Test data set in this exercise. Tip: Make sure you select the Data Partition node to show the properties for this node.

Training data is data that contains input values and target values that are using for model fitting. Validation data is used to validate the suitability of a model developed using Training data. Test data is typically recommended to evaluate how well a model generalizes and to compare models.

---

**Developing a Scorecard**

Next you will focus on the following steps of the scorecard development process:

1. Grouping (Classing) the Characteristic Variables into Attributes
2. Regression and Scaling
3. Reject Inference
4. The Final Scorecard

Note that the above steps are only part of the scorecard development process. Other tasks (such as exploratory data analysis, variable selection, and model comparison), while important, have been eliminated from this tutorial for brevity.
1. **Grouping the Characteristic Variables into Attributes**
You will now use the Interactive Grouping node to perform variable grouping (or classing), which is a binning transformation performed on the input variables.

a. From the **Credit Scoring** tab, drag and drop an **Interactive Grouping** node onto the workspace diagram and connect it with the **Data Partition** node.

   The Interactive Grouping node performs the initial grouping (called auto-grouping) using the settings you specify in the Property sheet. You can then use these initial grouping as a starting point to modify the classes interactively.

   By default, the pre-binning interval variables are grouped based on 20 quantiles (the Number of Bins property). These quantiles (or bins) are then grouped based on a decision tree with user-specified properties in the Interactive Grouping node to produce the final groupings.

b. Change the following options from the Property sheet:
   - Interval Grouping Method: Change to Monotonic Event Rate
   - Ordinal Grouping Method: Change to Monotonic Event Rate
   - Maximum Branch: Change to 10

   Do not run the node.

c. **Click** the icon of the **Interactive** property.

   ![Image](https://example.com/image.png)

   This message reminds you that before any interactive session in SAS Enterprise Miner, you need to 1) run the predecessor nodes, and 2) update the node you’re interactively using. **Click OK**. The following screen appears.

   ![Image](https://example.com/image.png)

   d. **Click Yes**.

   e. **Click OK** when the Run Status window appears.
f. Click **Results** and the following window opens.

The Output Variables window displays each variable’s Gini statistic and Information Value (IV) statistic. Note that a variable receives an Exported Role of Rejected if the variable’s Information Value (IV) is less than 0.10.
Recall that IV is used to evaluate a characteristic’s overall predictive power, that is, the characteristic’s ability to separate between good and bad loans. IV is calculated as:

\[ \sum_{i=1}^{L} (\text{Distr Good}_i - \text{Distr Bad}_i) \ln \left( \frac{\text{Distr Good}_i}{\text{Distr Bad}_i} \right) \]

where \( L \) is the number of attribute (levels) of the characteristic. Some rules of IV are listed below:

- Less than 0.02: unpredictive
- 0.02 to 0.10: weak
- 0.10 to 0.30: medium
- 0.30+: strong

The window also displays the Gini statistic for each characteristic input. The Gini statistic can be used as an alternative to IV. The formula for Gini is somewhat more complicated than IV and can be found in the online help and documentation.

In the Properties panel of the Interactive Grouping node, you can specify the cutoff values for the Gini and IV statistics. For example, the default IV cutoff of 0.10 for rejecting a characteristic can be changed to another value using the Information Cutoff Value property.

The Statistics Plot window shows a bar chart of each variable along with its corresponding IV. You can move your cursor over a bar to display tool tip information, which includes the variable name, IV, and variable role. The variables fico_score, tot_rev_line, age_oldest_tr, tot_derog, ltv, rev_util, and tot_tr are the variables considered as candidate inputs to build the final scorecard in the regression step.

The IV and Gini statistics might change if the groupings of the attributes are changed.

The auto binning provides a good starting point to create the groupings for the scorecard, but the Coarse Detail and Fine Detail tabs allow you to fine-tune the groupings.

g. Close the Results window.

h. Click the icon of the Interactive property of the Interactive Grouping node to open the Interactive Grouping window.
i. Select the **tot_rev_line** variable from the **Select Variable** drop-down menu.

The **tot_rev_line** variable represents how much credit (in dollars) the applicant has in all of their lines of credit.

j. Click the **Coarse Detail** tab. The following window opens.
Recall that weight of evidence (WOE) measures the strength of an attribute (that is, a group) of a characteristic in separating between good and bad accounts. WOE is based on comparing the proportion of “goods” to “bads” at each group level. It is defined as $\ln\left(\frac{\text{Distr Good}_i}{\text{Distr Bad}_i}\right)$ for each group $i$ of a characteristic. Negative numbers imply that a particular group is isolating a higher proportion of “bads” than “goods.” That is, negative WOE values are worse in the sense that applicants in a group with a higher negative number are a worse credit risk. Missing values by default are assigned their own group.

As expected and as shown in the previous plot, as an applicant’s amount of credit increases, so does the WOE values. The shape of the WOE curve seen in the graph is representative of how the points in the scorecard will be assigned.

k. Click the **Fine Detail** tab. The following window opens.
The Fine Detail tab shows the details of each group for the variable \textbf{tot_rev_line}. The graph at the bottom is the distribution of the bad loans within each group. It is easy to add an additional cutoff value to one of the groups. Suppose you want to make 3,000 a cutoff value in the scorecard for this characteristic variable.

1. Select the row that contains 3,000 in the score range. Right-click and select \textbf{Split Bin}.

m. In the Split Bin window, type 3000 as the new cutoff value and click \textbf{OK}.

You can see that Group 2 contains another bin that has the groups for \textbf{tot_rev_line}. These are now updated with 3000 as a new cutoff value.
Another option included in the Interactive Grouping node is that you can combine multiple bins within a group.

For example, suppose you want the revenue lines between 18200 and 28400 to be a single score range within Group 5.

1. Highlight the rows that contain the revenue lines 18200 and 28400. Right-click and select **Merge Bins**.
2. The window is again updated to reflect a new bin that contains revenue lines between 18200 and 28400 in Group 5.

Another option in the Fine Detail tab enables you to create defined groups from the bins. Suppose you want the last two revenue line bins in Group 5 to be in separate group.
o. Highlight the last two score ranges of Group 5 (revenue lines from 47659 to greater than 62898). Right-click and select New Group. The following window shows the updated window that has a new group that contains the total revenue lines from 47659 to greater than 62898.

p. Toggle back to the Coarse Detail tab. Notice the WOE graph.

q. The interactive changes you made in the Fine Detail tab did not affect the WOE graph that much. Our overall trend still has a non-monotonic distribution based on Group 6. Notice observation in the missing category have a WOE value similar to those in Group 1 versus the nonmissing values in Group 6. Next move the missing values of Group 6 to Group 1. Notice there is indeed an improvement in the WOE trend.
When changes occur in the Fine Detail tab, this affects the WOE graph. For example, there might be a characteristic input that should have increasing monotonic WOE values for each group. If the auto-binning of the Interactive Grouping node does not find these groupings, then the Fine Detail tab can be very powerful in fine-tuning the groupings to achieve a monotonic WOE graph.

The Interactive Grouping node can create groups based on several binning techniques, including statistically optimal ones. The user has the ability to change these bins based on business knowledge and known bias in the data to make the WOE trends logical. The changes previously made are suggested only if the analyst has expert knowledge and has a specific reason for changing the bins of a characteristic variable.

Also, after changes are made in the Interactive Grouping node as shown above, the statistics for WOE, IV, and Gini can possibly change. Some of the variables that were examined in the Results window previously might now not be candidates for input into the scorecard based on the IV and Gini statistics.

r. Close the Interactive Grouping window and click **Yes** to save the changes.

s. Return to the Interactive Grouping node in the diagram workspace. Right-click and select **Results**. The following window opens.
Notice the original variables are still candidate inputs into the scorecard, but the statistics for Gini and IV changed due to the changes made in the Interactive Grouping option.

t. Close the Results window to get back to the diagram process flow.

2. Regression and Scaling
We are now ready to use the grouped variables in a logistic regression model to create our scorecard. For this purpose we will be using the Scorecard node.

a. Drag and drop a **Scorecard** node onto the diagram and connect it with the **Interactive Grouping** node.
The Scorecard node is used to develop a preliminary scorecard with logistic regression and scaling. In SAS Enterprise Miner, there are three types of logistic regression selection methods to choose: forwards, backwards, and stepwise. There is also a selection in the Properties panel of the Scorecard node for None, so that all variable inputs enter the model.

After the logistic regression method of selection is chosen, the regression coefficients are used to scale the scorecard. Scaling a scorecard refers to making the scorecard conform to a particular range of scores. Some reasons for scaling the scorecard are for ease of interpretation, legal requirements, and to have a transparent methodology widely understood among all users of the scorecard.

The two main elements of scaling a scorecard are as follows:

1) odds at a certain score

2) “points to double the odds” (pdo)

The scorecard points are associated with 1 and 2.

Figure 3 shows an example of scorecard scaling. The scorecard points are scaled so that a total score of 600 corresponds to a good/bad odds of 30:1 and so that an increase of 20 points corresponds to a doubling of the good/bad odds.
Scaling the Scorecard: Example

<table>
<thead>
<tr>
<th>Score</th>
<th>Odds</th>
</tr>
</thead>
<tbody>
<tr>
<td>600</td>
<td>30</td>
</tr>
<tr>
<td>601</td>
<td>31.1</td>
</tr>
<tr>
<td>602</td>
<td>34.5</td>
</tr>
<tr>
<td>620</td>
<td>60</td>
</tr>
</tbody>
</table>

Figure 3: Scaling the Scorecard Example

In the Properties panel of the Scorecard node, the three options Odds, Scorecard Points, and Points to Double the Odds control the way in which the scorecard is scaled. The details of how SAS Enterprise Miner scales the scorecard are beyond the scope of this tutorial. For more information, see Siddiqi (2006) and Anderson & Hardin (2009) for a more detailed discussion about scorecard scaling.

b. Click the icon of the Variables property of the Scorecard node to open the Variables window.
Note that for each original input variable there are now corresponding WOE_variables and GRP_ variables. These were created by the Interactive Grouping node. Only the variable exceeding Gini/IV cutoff of the Interactive Grouping node is set to Input. All original inputs are set to Rejected.

Within the Scorecard node, the user has the option of using either the WOE_ variables, which contains the weight of evidence of each binned variable, or the GRP_ variables, which contains the group ID. The Analysis Variables property of the Scorecard node is used to specify whether regression is using WOE_variables or GRP_ variables. The default is to use WOE_variables.

c. Select Detailed in the Scorecard Type field (under the Scaling Options in the Properties panel) for a more detailed scorecard output. Right-click and select Run to run the Scorecard node.

d. Select Results and the following Run Status window opens.

There are several output screens to view. The first one is the Fit Statistics table.

e. Maximize the Fit Statistics table as shown in the following window.
This table displays fit statistics such as the average square error (ASE), the area under the receiver operating characteristic curve (AUR), and the Kolmogorov-Smirnov (KS) statistic, among others. Notice the AUR is 0.766 for the Validation data set.

f. Restore the Fit Statistics table.

The scorecard is also produced in the Results window.

g. Maximize the Scorecard window as shown in the following window.
The detailed scorecard displays information such as the Scorecard Points for each attribute, Weight of Evidence, Event Rate (percentage of bad applicants in that score range), Percentage of Population, and the (regression) Coefficient for each attribute. The Percentage of Population is the percentage of bad applicants who have a score higher than the lower limit of the score range.

h. Restore the Scorecard window.

i. Maximize the Score Rankings Overlay plot as shown in the following window.
The graph shown above is the Cumulative Lift chart. Recall that lift is the ratio of the percent of targets (that is, bad loans) in each decile to the percent of targets in the entire data set. Cumulative lift is the cumulative ratio of the percent of targets up to the decile of interest to the percent of targets in the entire data set.

For lift and cumulative lift, the higher value in the lower deciles indicates a predictive scorecard model. Notice that both Lift and Cumulative Lift for this scorecard have high lift values in the lower deciles.

j. Restore the Score Rankings Overlay plot.

k. Maximize the Empirical Odds plot as shown in the following window.
An Empirical Odds plot is used to evaluate the calibration of the scorecard. The chart plots the observed odds in a score bucket against the average score value in each bucket. The plot can help determine where the scorecard is or is not sufficiently accurate. The odds are calculated as the logarithm of the number of bad loans divided by the number of good loans for each scorecard bucket range. Thus, a steep negative slope implies that the “goods” are tending to get higher scores than “bads.” As would be expected with the previous plot, as the scorecard points increase, so does the number of good loans in each score bucket.

In the Scorecard node Properties panel, the option Number of Buckets is the number of scorecard buckets. The default is 25.

1. Restore the Empirical Odds plot.

There are two other useful graphs from the Results window, the Kolmogorov-Smirnov (KS) plot and the ROC plot.

m. To view the KS plot, select View ->Strength Statistics->Kolmogorov-Smirnov Plot. The following plot appears.
The KS chart shows the KS statistics plotted against scorecard cutoff values. Recall that the KS statistic is the maximum distance between the empirical distribution functions for the goods and bads. The difference is plotted, for all cutoffs, in the KS plot. The weakness of reporting only the maxim difference between the curves is that it provides only a measure of vertical separation at one cutoff value, but not over all cutoff values. According to the plot above, the best cutoff is approximately 176 (where the KS is at a maximum). At a cutoff value of 176, the scorecard best distinguishes between good and bad loans.

To view the ROC plot, close the Kolmogorov-Smirnov Plot window and select View -> Strength Statistics -> ROC Plot from the Results window. The following plot appears.
The ROC plot is a graphical measure of sensitivity versus (1-specificity).

The AUR (which is close to 0.76 for the validation data from the previous Fit Statistics table) measures the area below each of the curves that you see drawn in the plot. The AUR is generally regarded as providing a much better measure of the scorecard strength than the KS statistic because the area being calculated encompasses all cutoff values. A scorecard that is no better than random selection has an AUR value larger than 0.50. The maximum value of the AUR is 1.0.

The preliminary scorecard that was built in the previous demonstration used only known good and bad loans from the accepted applicants only. The scorecard modeler needs to apply the scorecard to all applicants (accepted and rejected). The scorecard needs to generalize to the “through the door” population. The next section, Reject Inference, is about attempting to solve the sample bias issue so that the developmental sample will be similar to the population to which the scorecard will be applied.

3. **Reject Inference**
The next step of the scorecard development process is to use the rejected applicants and perform Reject Inference to create an augmented data set that is representative of the “through the door” applicant population.

a. Drag and drop a **Reject Inference** node onto the diagram and connect it with the **Scorecard** node.
b. Additionally, drag and drop the REJECTS data source onto the diagram and connect it with the Reject Inference node. Make sure that the REJECTS data source is defined as a SCORE data set. You can do this through the Role option in the Properties panel.

The diagram now looks as follows.

The Reject Inference node attempts to infer the good/bad behavior (performance) of the rejected applicants using three industry-accepted inference methods.

Following are the general properties for Reject Inference including a brief overview of each of the inference methods.

**General Properties**

**Inference Method**—Use the Inference Method property to specify the method that you want to use to classify REJECTS data set observations. The three choices are Fuzzy, Hard Cutoff, and Parceling.

- **Fuzzy**—Fuzzy classification uses partial classifications of “good” and “bad” to classify the rejects in the augmented data set. Instead of classifying observations as “good” and “bad,” fuzzy classification allocates weight to observations in the augmented data set. The weight reflects the observation’s tendency to be good or bad. The partial classification information is based on the p(good) and p(bad) from the model built on the Accepts for the REJECTS data set. Fuzzy classification multiplies the p(good) and p(bad) values that are calculated in the Accepts for the Rejects model by the user-specified Reject Rate parameter to form frequency variables. This results in two observations for each observation in the Rejects data. One observation has a frequency variable (Reject Rate * p(good)) and a target variable of 0, and the other has a frequency variable (Reject Rate * p(bad)) and a target value of 1. Fuzzy is the default inference method.

- **Hard Cutoff**—Hard Cutoff classification classifies observations as “good” or “bad” observations based on a cutoff score. If you choose Hard Cutoff as your inference method, you must specify a Cutoff Score in the Hard Cutoff properties. Any score below the hard cutoff value is allocated a status of “bad.” You must also specify the Rejection Rate in General properties. The Rejection Rate is applied to the REJECTS data set as a frequency variable.
- **Parceling**—Parceling distributes binned scored rejects into “good” and “bad” based on expected bad rates \( p(\text{bad}) \) that are calculated from the scores from the logistic regression model. The parameters that must be defined for parceling vary according to the Score Range method that you select in the Parceling Settings section. All parceling classifications require the Reject Rate setting as well as bucketing, score range, and event rate increase. See the Parceling Settings section for more information.

**Rejection Rate**—If you choose Hard Cutoff as your inference method, you must provide values for the Rejection Rate and for the Cutoff Score properties. If you choose Parceling as your inference method, you must provide values for the Rejection Rate. Rejection Rate is used as a frequency variable. The “bad” rate for rejects is calculated as \( \frac{\text{number of beds}}{\text{number of beds} + \text{number of goods}} \). Rejection rates are real numbers between 0.0001 and 1. The default rejection rate is 0.3.

- **Hard Cutoff Properties**
  The Reject Inference node uses the Cutoff Score property when the selected inference method is Hard Cutoff.

- **Cutoff Score**—The Cutoff Score that you specify is the threshold score that is used to classify “good” and “bad” observations in the Hard Cutoff method. Scores below the threshold value are assigned a status of “bad;” otherwise, the observation is classified as “good.”

**Parceling Properties**

You must configure Parceling properties when you choose Parceling as the selected inference method.

**Score Range Method**—Use the Score Range Method property to specify the way that you want to define the range of scores to be bucketed. The four method choices are Accepts, Rejects, Scorecard, and Manual.

- **Accepts**—The Accepts score range method distributes the rejects into equal-sized buckets based on the score range of the ACCEPTS data set.
- **Rejects**—The Rejects score range method distributes the rejects into equal-sized buckets based on the score range of the REJECTS data set.
- **Scorecard**—The Scorecard score range method distributes the rejects into equal-sized buckets based on the score range that is output by the AUGMENTED data set.
- **Manual**—The Manual score range method distributes the rejects into equal-sized buckets based on the range that you input.

**Score Buckets**—Use the Score Buckets property to specify the number of buckets that you want to use to parcel the data set into during attribute classification. Permissible Score Buckets property values are integers between 1 and 100. The default setting for the Score Buckets property is 25.

**Event Rate Increase**—The proportion of “bad” and “good” classified observations in the REJECTS data set is not expected to approximate the proportion of “bad” and “good” classified observations in the ACCEPTS data set. Logically, the “bad” rate of the REJECTS data set should be higher than that of the ACCEPTS data set. It is appropriate to use some coefficient to assign a higher proportion of Rejects as “bad.” You use the Event Rate Increase property setting to specify the adjustment value for the Reject “bad” rate. To configure a 20% increase, set the Event Rate Increase property to 1.2.

a. Keeping all default options (note that the default inference method is Fuzzy Augmentation), run the node by right-clicking the node and selecting **Run**.

b. When complete, open the results by right-clicking the node and selecting **Results**.
The results include distribution plots for score and predicted probabilities for accepts and inferred samples side-by-side. It also displays the printed output.

c. Expand the Output window and scroll down to find the event rates for various samples including that for the augmented sample.
The output from the Reject Inference node is the Augmented Data with both Accepts and Rejects appended together. The Training Data Event Rate and the Validation Data Event Rate is the event rate (that is, the bad rate) for the accepted applicant’s data set. The Required Inferred Event Rate is the event rate for the rejected applicant’s data set. The Augmented Data Event Rate is the event rate for the AUGMENTED (accepted and rejected applicants) data set. The Summary Statistics sections display basic summary statistics for both the accepted and rejected applicants’ data.

4. Final Scorecard
The next step of the scorecard development process is to repeat the steps of Interactive Grouping and Regression and Scaling on the augmented data.

a. Connect a Data Partition node to the Reject Inference node. In the Properties panel change the Training data set percentage to 70% and the Validation data set percentage to 30%.

b. Connect an Interactive Grouping node to the Data Partition node.

c. Connect a Scorecard node to the Interactive Grouping node.

Your diagram will now look like the following.
d. Right-click the Scorecard node and select Run.

You now have a scorecard analysis that includes both accepted and rejected applicants.

References

