Chapter 9: Predictive Analysis Using SAS Enterprise Miner

Introduction

In Chapters 5 through 8, you learned to view and analyze the DMR Publishing data in an effort to understand past and current customer characteristics and behavior. In this chapter, you will learn how to discover and measure patterns from past behavior and characteristics to predict future behavior using predictive modeling. This is one of most widely used techniques used in business today.

As detailed in Chapter 1, there are numerous uses for predictive modeling in marketing, risk, process improvement, customer retention, and more. In this chapter, you will build a model to predict the likelihood of loan default using credit risk data.

To achieve your goal, you will initiate a project in SAS Enterprise Miner and proceed through the model building process. You will explore and manipulate the data to prepare for modeling. You will build three common types of models: a decision tree, a neural network, and a regression model. You will compare the three models in an assessment node.

The steps for model development are as follows:

- *Select:* Initiate the project and bring the modeling data set into the project. Be sure that it correlates with the business purpose for building the model.
- *Explore:* Use visual tools to view the distribution and completeness of the data.
- *Modify:* Partition the data into training, test, and validation data sets. Impute missing values and filter outliers. Transform or segment interval variables if necessary. Create indicator variables for variables with character or non-sequential values.
- *Model:* Develop three different models: a decision tree, a neural network, and a regression model.
- *Assess*: Compare the three models by using the gains table and lift charts.

NOTE: For supporting documentation on every aspect of SAS Enterprise Miner, see the Contents section under Help in the main menu.

Select

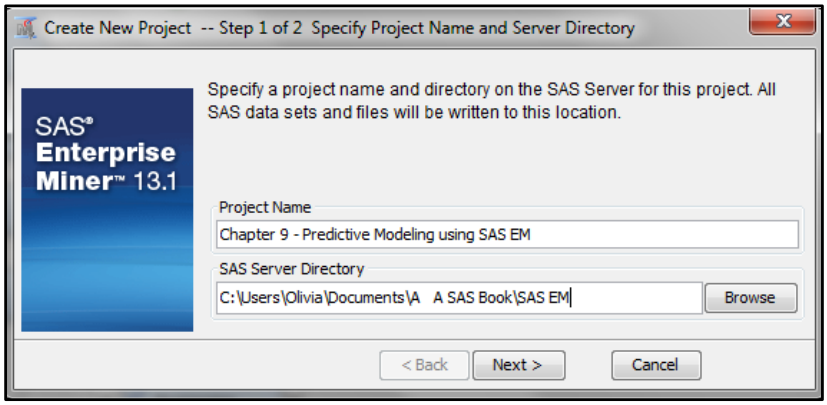
In this section, you initiate a project in SAS Enterprise Miner and select data for the model-building process.

Initiate the Project

ABC Bank provided a "snapshot" of customer data that includes customers who defaulted on their loans and customers who did not default on their loans. The customer payment, balance, and several demographic variables were appended from a credit bureau for a period six months earlier. The dependent variable is a field called LOAN_DEFAULT. The goal of the model is to learn whether you can predict who is most likely to default on a loan six months from now.

Open SAS Enterprise Miner and select **New Project**. Name your project and select a place to store the files as seen in Figure 9.1. Click **Next ▶ Finish**.

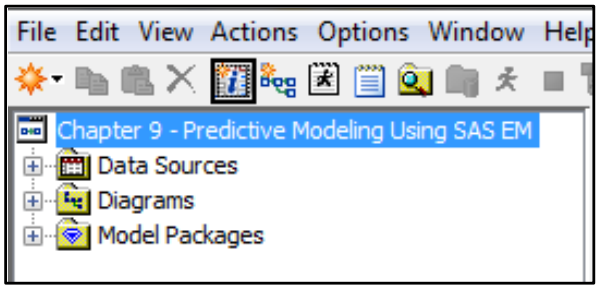
Figure 9.1: Initiate a New Project



Select the Data

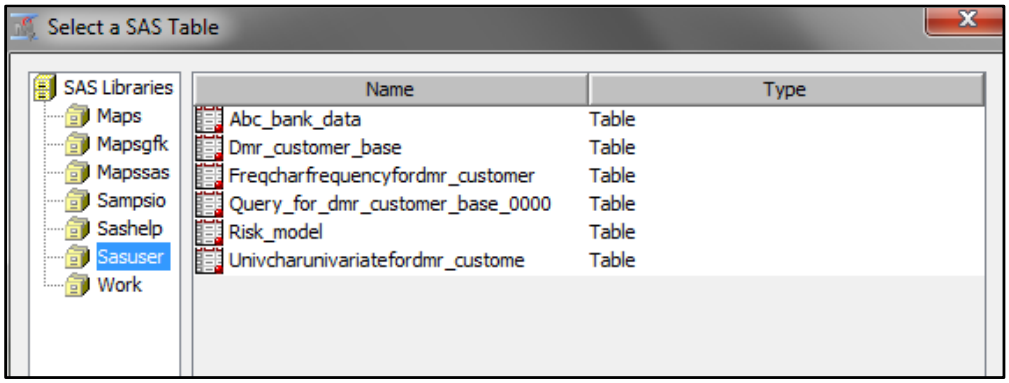
Open the Data Source Wizard by clicking the **Create Data Source** icon shown in Figure 9.2. We are searching for the ABC Bank data in the Sasuser folder.

Figure 9.2: Use the Data Source Wizard



When the window opens, click **Next** ▶ **Browse** and select Sasuser as shown in Figure 9.3.

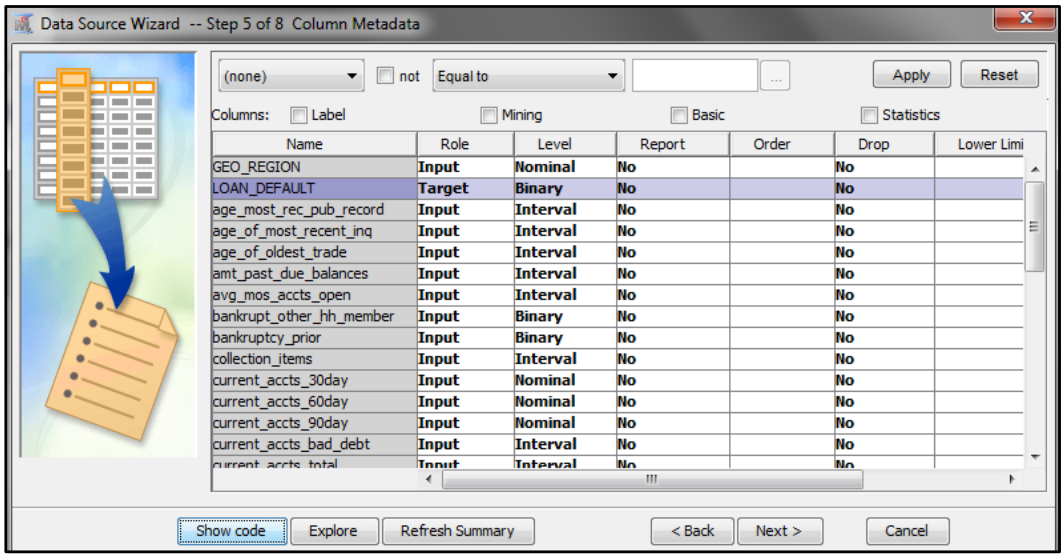
Figure 9.3: Select Sasuser Folder



Double-click the Sasuser folder and highlight ABC_Bank_Data. Click **OK** ▶ **Next** ▶ **Next** ▶ **Next** and you will be at **Step 5 of 8**. Click the button next to **Advanced** and then click **Next.** In this step, you assign the dependent variable.

The goal of the project is to build a model to predict loan default, so, the LOAN_DEFAULT variable is the dependent variable for the project. Next to the Name LOAN_DEFAULT, change the **Role** to **Target** and the **Level** to **Binary** (if it is not Binary by default) as shown in Figure 9.4.

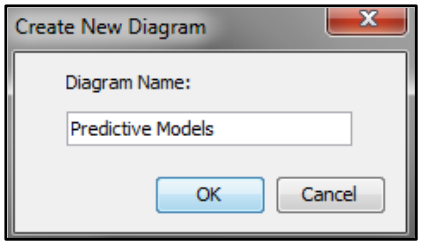
Figure 9.4: Assign Dependent Variable



Click **Next** ▶ **Next** ▶ **Next** ▶ **Finish**. The data set is now ready to begin the modeling process.

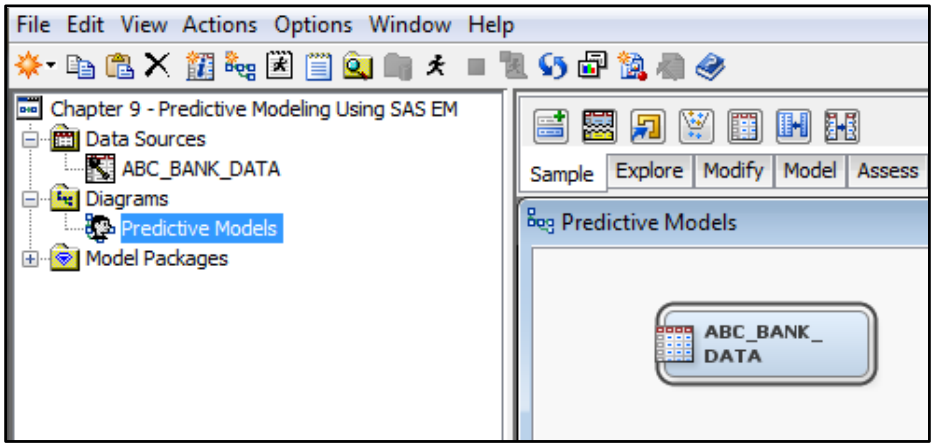
Next, create a new diagram. Click **File** ▶ **New** ▶ **Diagram** and enter a name as shown in Figure 9.5. Click **OK**.

Figure 9.5: Name a New Diagram



Once the diagram appears, drag the ABC_Bank_Data into the diagram as shown in Figure 9.6.

Figure 9.6: View the Project and the Data Source in the Modeling Diagram



Explore

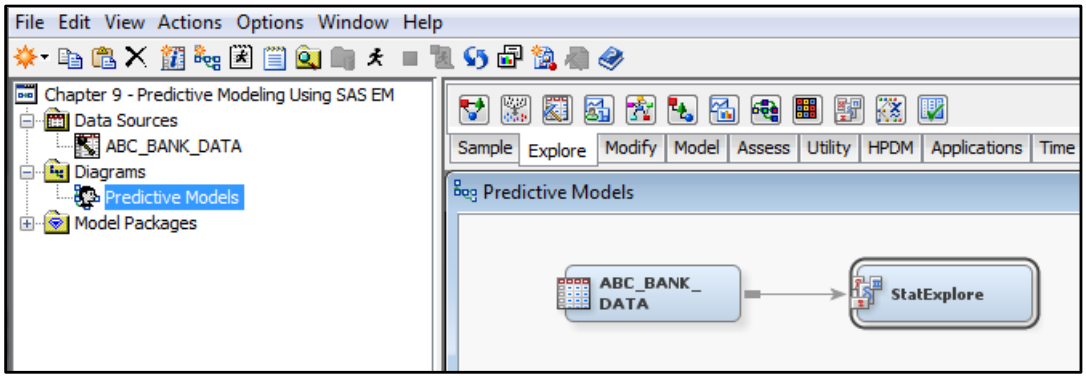
When developing a predictive model, you need to prepare the data for modeling, so you may need to take steps to handle missing values and outliers. You also want the relationship between the dependent variable and the independent variables to follow certain distributions.

To view the data, use the StatExplore and MultiPlot nodes. These nodes allow you to view the distributions of the variables, as well as their relationships to the target variable.

StatExplore

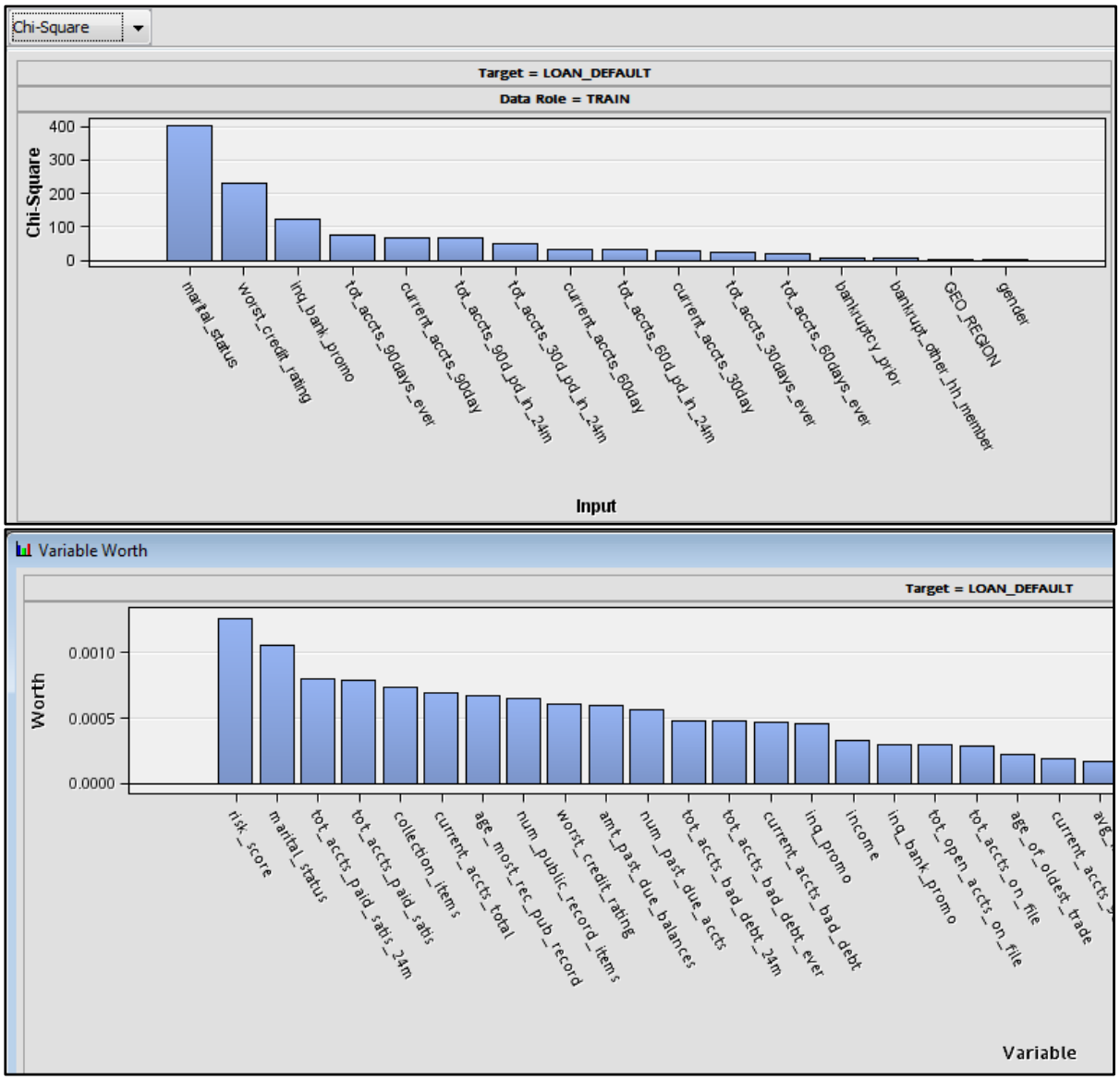
To initiate StatExplore, in the **Explore** tab look for the icon (third from the right), drag it onto the diagram, and connect it to ABC_Bank_Data with an arrow as seen in Figure 9.7.

Figure 9.7: Project View with the Data Source in the Modeling Diagram



Right-click the StatExplore icon; click **Run ▶ Yes**. When the run completes, select **Results**.

Output 9.1: StatExplore Results



The upper plot is the chi-square plot. It shows the strength of the relationship between the variables and the dependent variable. Here, marital_status has the strongest correlation with the target variable, LOAN_DEFAULT.

The Variable Worth plot measures and displays the independent variables according to their calculated worth. RISK_SCORE has the highest worth, with marital_status coming in second, tot_accts_paid_satis_last_24m coming in third, and so on. For more information on variable worth, see

the variable importance explanation in the decision tree section of the Contents within SAS Enterprise Miner. You can access Contents by clicking **Help ▶ Contents** on the upper menu.

The upper right window of the Results window offers a rich set of statistics on each variable. Enlarge the window and scroll down to see various features of the variables. In Output 9.2, you get a partial view that shows the class variables. Notice that marital_status has 924 missing values. These missing values will be imputed in the upcoming "Modify" section, using the impute process.

Output 9.2: StatExplore Statistics—Class Variables

Class Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Data Role	Variable Name	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
TRAIN	GEO_REGION	INPUT	4	0	West	36.95	Midwest	27.59
TRAIN	bankrupt_other_hh_member	INPUT	2	0	0	99.74	1	0.26
TRAIN	bankruptcy_prior	INPUT	2	0	0	91.89	1	8.11
TRAIN	current_accts_30day	INPUT	8	0	0	88.91	1	8.77
TRAIN	current_accts_60day	INPUT	7	0	0	94.31	1	4.79
TRAIN	current_accts_90day	INPUT	14	0	0	89.01	1	7.80
TRAIN	gender	INPUT	3	0	M	52.23	F	43.98
TRAIN	inq_bank_promo	INPUT	17	0	0	42.73	1	21.58
TRAIN	marital_status	INPUT	5	924	M	32.53	S	30.03
TRAIN	tot_accts_30d_pd_in_24m	INPUT	13	0	0	76.55	1	15.07
TRAIN	tot_accts_30days_ever	INPUT	15	0	0	66.21	1	18.36
TRAIN	tot_accts_60d_pd_in_24m	INPUT	10	0	0	88.95	1	8.37
TRAIN	tot_accts_60days_ever	INPUT	10	0	0	82.38	1	12.19
TRAIN	tot_accts_90d_pd_in_24m	INPUT	16	0	0	85.33	1	9.48
TRAIN	tot_accts_90days_ever	INPUT	16	0	0	78.58	1	12.52
TRAIN	worst_credit_rating	INPUT	7	0	1	64.41	6	22.83
TRAIN	LOAN_DEFAULT	TARGET	2	0	0	95.84	1	4.16

Output 9.3 shows that the independent variable "income" has 549 missing values. We will populate these in the upcoming "Modify" section using the impute process.

Output 9.3: StatExplore Statistics—Interval Variables

Interval Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

Variable	Role	Mean	Standard Deviation	Non Missing	Missing	Minimum	Median	Maximum	Skewness	Kurtosis
age_most_rec_pub_record	INPUT	5.762059	14.54792	28734	0	0	0	119	3.572219	14.41568
age_of_most_recent_inq	INPUT	4.551681	5.775763	28734	0	0	2	24	1.527642	1.530631
age_of_oldest_trade	INPUT	108.3721	90.67497	28734	0	0	86	829	1.486329	2.536665
amt_past_due_balances	INPUT	698.3108	7700.961	28734	0	0	0	1119197	113.1366	15791.41
avg_mos_accts_open	INPUT	52.02311	36.16629	28734	0	0	46	469	1.756677	6.893803
collection_items	INPUT	0.849655	2.411677	28734	0	0	0	50	6.109127	59.09067
current_accts_bad_debt	INPUT	0.844574	2.134596	28734	0	0	0	45	4.317058	29.69401
current_accts_total	INPUT	8.464711	7.932097	28734	0	0	6	75	1.532718	3.248087
income	INPUT	21034.03	23446.74	28185	549	0	14246	325183	2.414008	10.55158
inq_fin_last_6mos	INPUT	0.753706	1.722227	28734	0	0	0	38	5.214959	49.05728
inq_past_12mos	INPUT	1.124104	1.669604	28734	0	0	1	26	2.894731	15.42647
inq_promo	INPUT	3.361941	3.343572	28734	0	0	2	27	1.4799	2.96967
num_past_due_accts	INPUT	0.582098	1.392362	28734	0	0	0	41	5.148837	61.8276
num_public_record_items	INPUT	1.182432	3.00794	28734	0	0	0	65	5.463535	47.98795
risk_score	INPUT	808.7328	155.6376	28734	0	222	863	994	-1.07662	0.314598
tot_accts_bad_debt_24m	INPUT	0.611506	1.713559	28734	0	0	0	45	5.299255	49.85442
tot_accts_bad_debt_ever	INPUT	0.854528	2.154509	28734	0	0	0	45	4.293498	29.21094
tot_accts_on_file	INPUT	11.10138	8.838177	28734	0	1	9	81	1.390275	2.815527
tot_accts_open_last_24m	INPUT	2.772186	2.632364	28734	0	0	2	42	1.73403	6.044214
tot_accts_paid_satis	INPUT	7.539361	7.615784	28734	0	0	5	75	1.668434	3.840586
tot_accts_paid_satis_24m	INPUT	5.843844	5.720464	28734	0	0	4	64	1.65111	4.258955
tot_bal_open_accts	INPUT	10885.18	14350.11	28734	0	0	5988	251742	3.159175	20.69761
tot_open_accts_bal_gt_0	INPUT	3.580358	2.994385	28734	0	0	3	46	1.880609	7.488173
tot_open_accts_on_file	INPUT	5.394028	4.238754	28734	0	0	4	51	1.543187	4.295503

Close the **Results** window by clicking the X in the upper right corner.

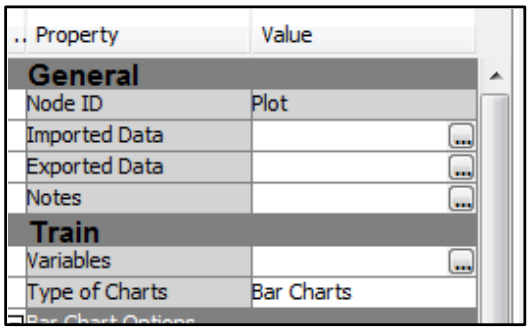
MultiPlot

The MultiPlot node allows you to examine the relationships between the variables, as well as their distributions. This ability is useful for determining the univariate power of an independent variable to predict the dependent variable. It can also help you determine the best transformation, if necessary.

To initiate MultiPlot, find the icon in the Explore tab (seventh from the left), drag it onto the diagram, and connect it to the ABC_Bank_Data with an arrow.

Check the Properties window on the left-hand side to ensure that the **Type of Charts** is set to **Bar Charts** as shown in Figure 9.8.

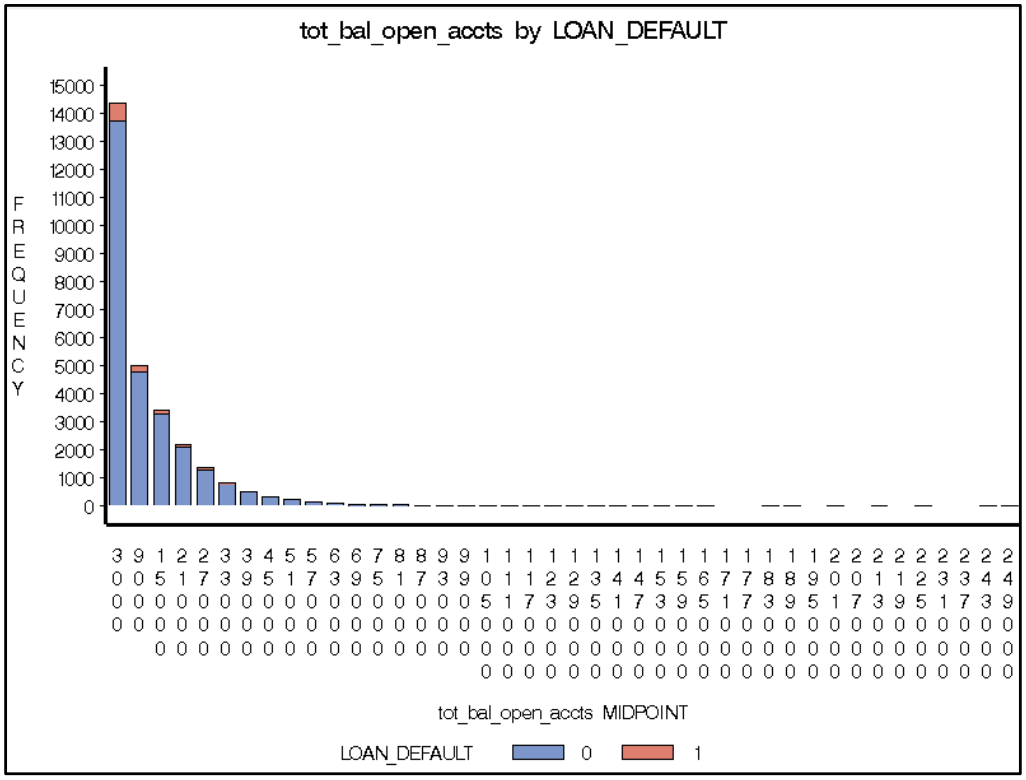
Figure 9.8: MultiPlot Properties Window



Right-click the icon and click **Run ▶ Yes**. When the run completes, click **Results**. The window opens with two sections. Expand the top panel to see the bar charts.

This process creates a frequency bar chart for each independent variable by LOAN_DEFAULT. The bar chart for tot_bal_open_Accts by LOAN_DEFAULT is shown in Output 9.4. The menu at the bottom of the chart enables you to look at the relationship between all the independent variables and LOAN_DEFAULT. Click the center arrow to start a slide show of the different frequencies. Or, use the right and left arrows to view at your own pace.

Output 9.4: MultiPlot Total Balance on Open Accounts by Loan Default—Bar Chart



These graphs offer a quick view of the distribution of each independent variable in relation to LOAN_DEFAULT. The variable is also highly skewed to the right. In subsequent steps, you will scrub the tot_bal_open_Accts variable for outliers and transform it to make it more normally distributed.

Modify

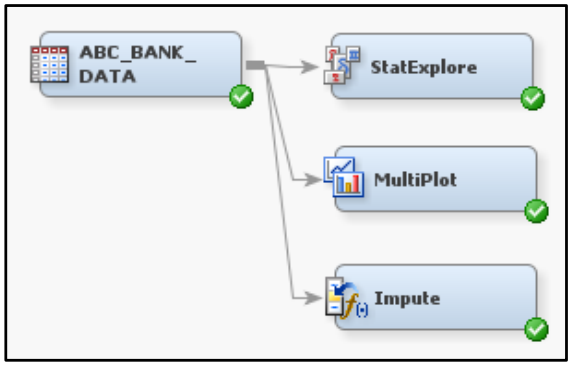
This section illustrates how to prepare the data for modeling. You will impute values to replace missing values, partition the data, filter outliers, and transform the independent variables.

Replace Missing Values via Imputation

The next step is to impute values for missing or incorrect observations. Recall that the StatExplore window revealed that the Marital Status and Income variables have missing values.

Begin by clicking **Modify** and dragging the **Impute** icon onto the diagram, as shown in Figure 9.9.

Figure 9.9: Impute Missing Values



The Property window is populated with default methods for imputing missing values. The class variables are replaced with the value that has the highest count. The highest count value is used as a replacement for Marital Status. Recall that Gender has a value = U for *unknown*. You can leave this value as is: Sometimes the fact that the Gender is unknown is predictive. If it were missing, it would be set to M because of a higher count of customers with Gender = M.

For interval variables, the default method is to replace missing values with the mean value. Because your number of missing values is small, this default is adequate. If it were a larger percentage, then using a more advanced method, such as tree, might be better. To learn more about the Tree Imputation Method, see Contents in the Help section of Enterprise Miner.

Right-click on the **Impute** icon and click **Run ▶ Yes**. When complete, click **Results**. Output 9.5 shows a partial view of the number of imputed values as well as the value used to replace the missing values.

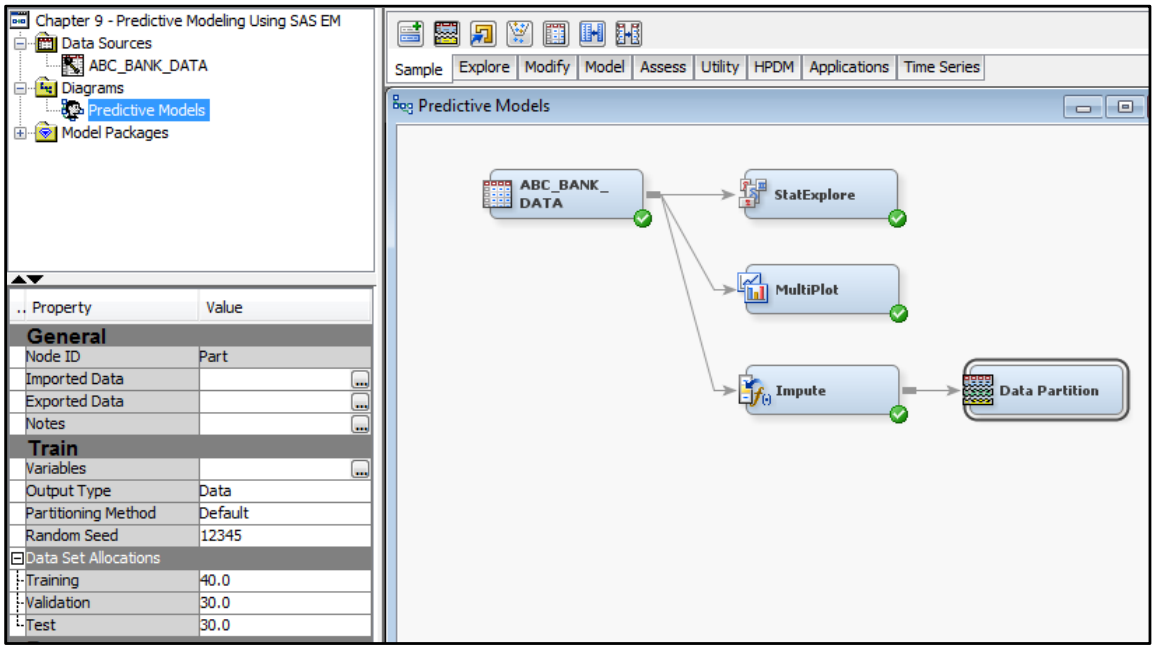
Output 9.5: Impute Missing Value Results

Variable Name	Impute Method	Imputed Variable	Impute Value
ncome	MEAN	IMP_income	21034.032358
marital_status	COUNT	IMP_marital_status	M

Partition Data into Subsamples

The next step is to partition the data into three subsamples: Train, Validate, and Test. In your diagram, click the **Sample** tab, drag the **Partition** icon onto the diagram, and connect it to the **Impute** icon as shown in Figure 9.10.

Figure 9.10: Partition Data



For every modeling technique, the **Train** data is the base data set used to build several interim models and a final model during the model building process. The **Validation** data is used to fine tune each model and avoid overfitting during the decision tree, neural network, and stepwise regression model building processes. The **Test** data is used to evaluate the final model on unbiased data not used in the model building process. The final model is the one with the best fit based on your preselected criteria.

You will build three models, one each of decision tree, neural network, and stepwise regression. These will be evaluated individually and also compared using the **Model Comparison** node.

Notice in the Property window at the left that the partitioning method, under **Train**, is set to **Default**. The choices are **Simple, Random**, **Cluster**, and **Stratified**. You want to use the **Stratified** method, and because you have a binary target variable, the default is **Stratified.** So, you don't need to make any change here. If you do change it to **Stratified**, the result will be the same.

Notice the default percentages in the **Property** window. Under **Data Set Allocations**, the default settings assign 40% to training, 30% to validation, and 30% to test. These defaults can be changed as long as the total equals 100%. Right-click on the Data Partition process and click **Run ▶ Yes**. You can view the results to see that the percentage of the dependent variable is close in each of the three sample data sets.

Manage Outliers

As described in Chapter 5, outliers are data points that are unusually large or small when compared to the rest of the values for a certain variable. While there is no precise rule to identify an outlier, you can look for values that are more than 3, 4, or 5 standard deviations from the mean.

Outliers may also represent data errors. Or, for some modeling applications, outliers may be very useful to your analysis. For example, when modeling a rare event such as fraud, you might find the outliers to be very predictive. The best approach to modeling rare events is using neural networks, which are less sensitive to outliers.

When building a regression model for a not-so-rare event, you will want to take steps to manage the outliers. Depending on the number of observations in your data set compared to the number of outliers, you can choose to handle them differently. The four most common methods for dealing with outliers are:

1. Cap the variable.
2. Drop the observations.
3. Treat them as errors or missing values.
4. Bucket the continuous variable into quartiles, deciles, or centiles.

Regression models are especially sensitive to outliers. If your data contains outliers, investigating the source of the data is good practice. Because your current data comes directly from a credit bureau and has not been altered to correct for outliers, it typically has some outliers. In this case study, you will filter the outliers. In practice, this is a simple first step using the Filter node. If you have the luxury of testing different methods for handling outliers, you can compare the model results and develop your own preference for managing outliers.

To filter outliers, click on the **Sample** tab and drag the **Filter** icon to your diagram, as shown in Figure 9.11. In the **Property** window, set the **Tables to Filter** to **All Data Sets.** The default setting for **Class Variables** is **Rare Values (Percentage).** Leave this setting as is. For **Interval Variables**, set the method to **Extreme Percentiles.** The **Extreme Percentiles** method filters out the top 5% of values. This setting can be changed by clicking the three dots to the right of **Tuning Parameters**.