Paper AA-02-2015

A Property & Casualty Insurance Predictive Modeling Process in SAS

Mei Najim, Sedgwick Claim Management Services, Chicago, Illinois

1.0 ABSTRACT

Predictive analytics has been developing in property & casualty insurance companies in the past two decades. Although the statistical foundations of predictive analytics have large overlaps, the business objectives, data availability, and regulations are different across property & casualty insurance, life insurance, banking, pharmaceutical, and genetics industries, etc. A property & casualty insurance predictive modeling process with large data sets will be introduced including data acquisition, data preparation, variable creation, variable selection, model building (a.k.a.: model fitting), model validation, and model testing. Variable selection and model validation stages will be introduced in more detail. Some successful models in the insurance companies will be introduced. Base SAS, SAS Enterprise Guide, and SAS Enterprise Miner are presented as the main tools for this process.

2.0 INTRODUCTION

This paper begins with a full life cycle of the modeling process from a business goal to model implementation. Each stage on the modeling flow chart will be introduced in one separate sub-session. The scope of this paper means to provide readers some understanding about the overall modeling process and gain some general ideas on building models in Base SAS, SAS Enterprise Guide, and SAS Enterprise Miner. This paper doesn't mean to be thorough with great details. Due to data proprietary, some simplified examples with Census data have been utilized to demonstrate the methodologies and techniques which would serve well on large data sets in real business world.

3.0 A PRPERTY & CASUALTY INSURANCE PREDICTIVE MODELING PROCESS

Any predictive modeling project process starts from a business goal. To attain that goal, data is acquired and prepared, variables are created and selected, and the model is built, validated, and tested. The model finally is evaluated to see if it addresses the business goal and should be implemented. If there is an existing model, we would also like to conduct a model champion challenge to understand the benefit of implementing the new model over the old one. In the flow chart (Figure 1), there are nine stages in the life cycle of the modeling process. There will be one dedicated section for each stage. The bold arrows in the chart describe the direction of the process. The light arrows show that at any stage, steps may need to be re-performed, resulting in an iterative process.

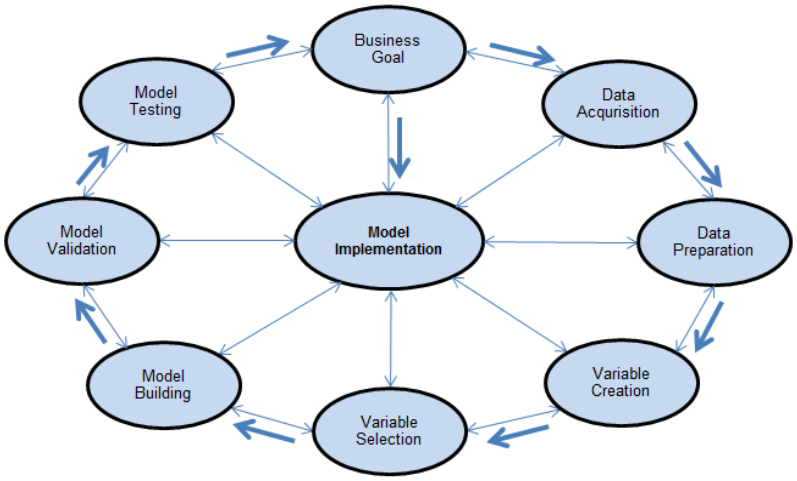


Figure 1. A Property & Casualty Insurance Predictive Modeling Process Flow Chart

3.1 BUSINESS GOALS AND MODEL DESIGN

Specific business goals/problems decide the model design - what type(s) of model(s) to build. Sometimes when business goals are not clear, we can start to look into the available data sources to find more information to help to (re)define those goals. In the insurance companies, pricing and underwriting properly, reserving adequately, and controlling costs to handle and settle claims are some of the major business goals for predictive modeling projects.

3.2 DATA SCOPE AND ACQUISITION

Based on the specific business goals and the designed model, data scope is defined and the specific data including internal and external data is acquired. Most middle to large size insurance organizations have sophisticated internal data systems to capture their exposures, premiums, and/or claims data. Also, there are some variables based on some external data sources that have been proved to be very predictive. Some external data sources are readily available such as insurance industry data from statistical agencies (ISO, AAIS, and NCCI), open data sources (demographics data from Census), and other data vendors.

3.3 DATA PREPARATION

3.3.1 Data Review, Cleansing, and Transformation

Understanding every data field and its definition correctly is the foundation to make the best use of the data towards building good models. Data review is to ensure data integrity including data accuracy, consistency, and that basic data requirements are satisfied and common data quality issues are identified and addressed properly. If the part of the data isn't reflected the trend into future, it should be excluded.

For example, initial data review is to see if each field has decent data volume to be credible to use. Obvious data issues, such as blanks and duplicates, are identified, and either removed or imputed based on reasonable assumptions and appropriate methodologies. Missing value imputation is a big topic with multiple methods so we are not going to go into detail in this paper.

Here are some common SAS procedures in both Base SAS and SAS Enterprise Guide:

```
PROC CONTENTS/PROC FREQ/PROC UNIVARIATE/PROC SUMMARY
```

The Data Explore Feature in SAS Enterprise Miner is also a very powerful way to quickly get a feel for how data is distributed across multiple variables. The example below with some data fields is from Census data source.

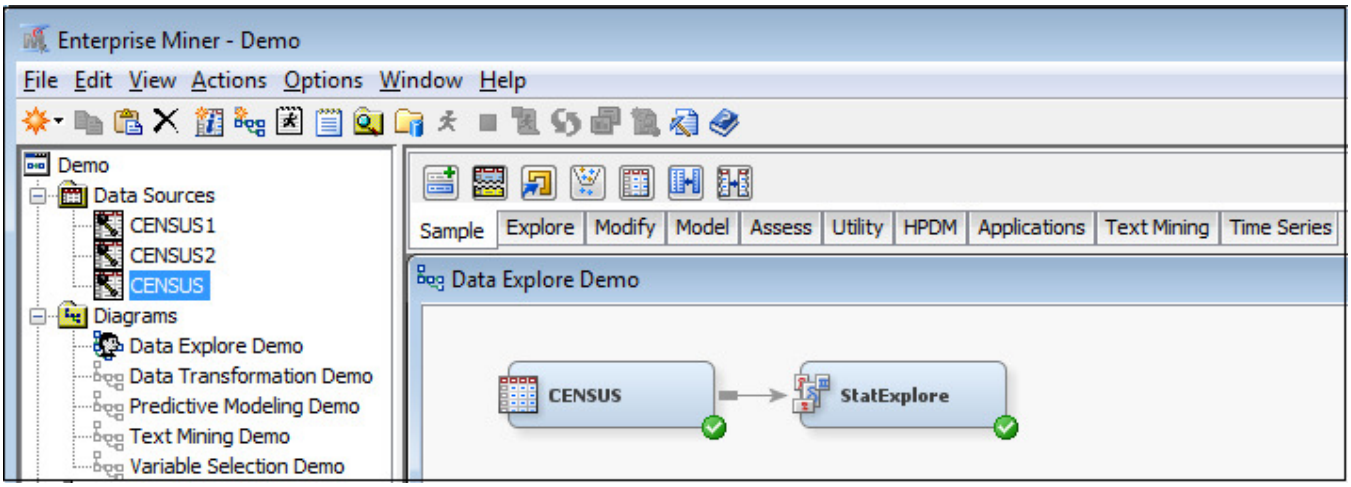


Figure 2. A Data Review Example using ***CENSUS*** Node (Data Source) and ***StatExplore*** Node in SAS Enterprise Miner

The diagram (Figure 3) below is the Data Explore feature from ***CENSUS*** node in the diagram (Figure 2) above. It shows how a distribution across one variable can be drilled into to examine other variables. In this example, the shaded area of the bottom graph represents records with median household income between $60,000 and $80,000. The top two graphs show how these records are distributed across levels of two other variables.

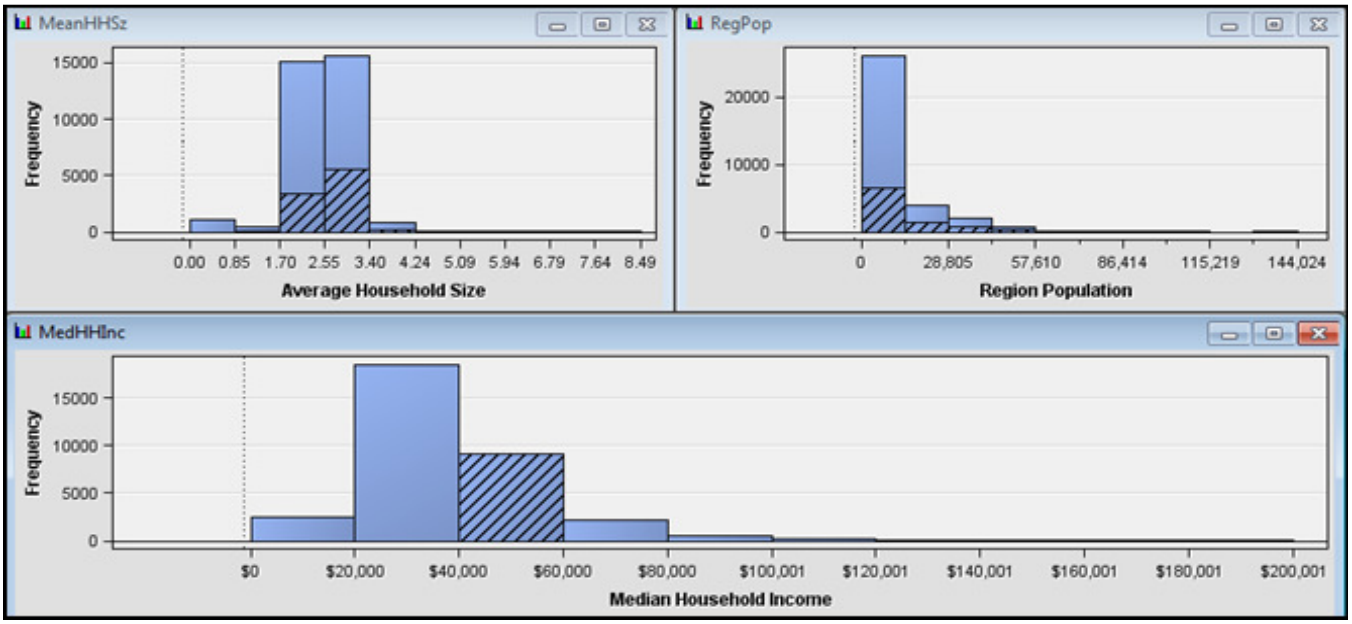
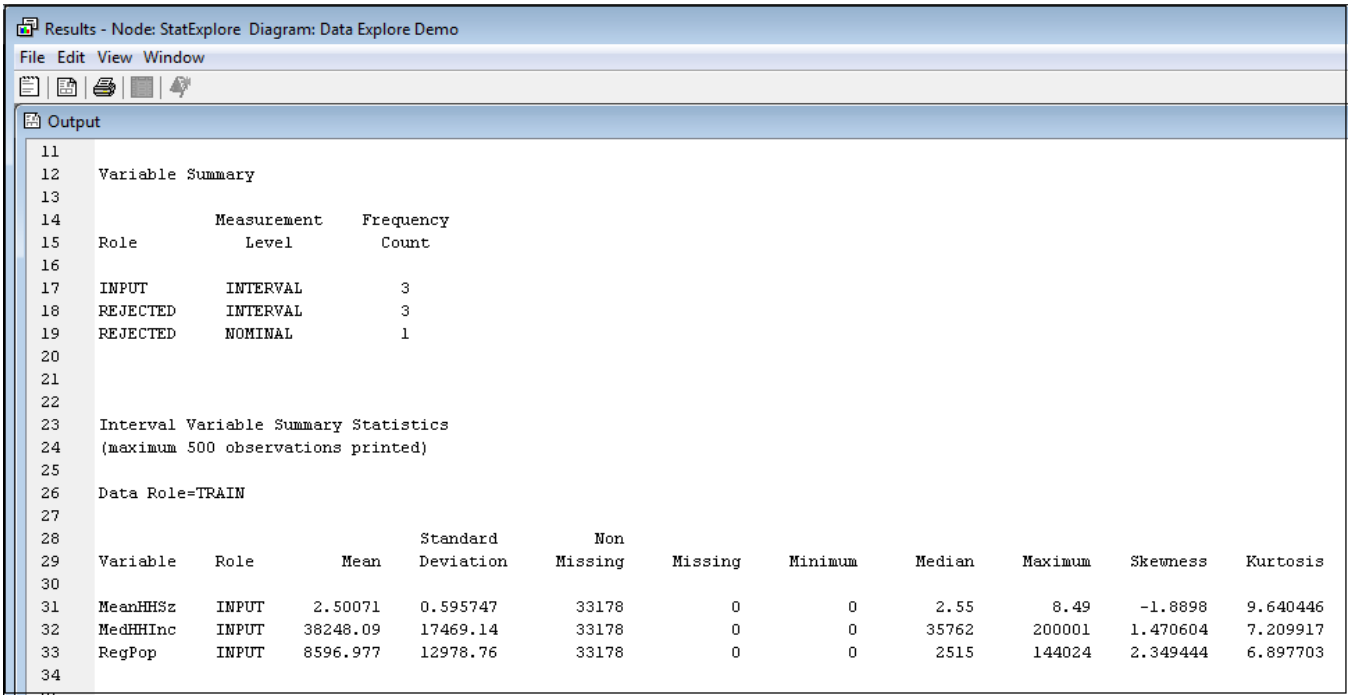


Figure 3. Data Explore Graphs from ***CENSUS*** Node (Data Source) in SAS Enterprise Miner

The result report (Figure 4) with thorough statistics for each variable is provided after running ***StatExplore*** node in the diagram (Figure 2). Please see the following diagram (Figure 4) of a result report with three data fields from CENSUS data. When the data source contains hundreds or thousands fields, this node is a very efficient way to conduct a quick data explore.




```
Variable Summary

             Measurement    Frequency
Role            Level         Count

INPUT         INTERVAL           3
REJECTED      INTERVAL           3
REJECTED      NOMINAL            1



Interval Variable Summary Statistics
(maximum 500 observations printed)

Data Role=TRAIN

                                  Standard          Non
Variable    Role         Mean     Deviation     Missing     Missing     Minimum      Median     Maximum     Skewness     Kurtosis

MeanHHSz    INPUT     2.50071      0.595747       33178           0           0        2.55        8.49      -1.8898     9.640446
MedHHInc    INPUT    38248.09      17469.14       33178           0           0       35762      200001     1.470604     7.209917
RegPop      INPUT    8596.977      12978.76       33178           0           0        2515      144024     2.349444     6.897703
```

Figure 4. Results Report Using ***StatExplore*** Node in SAS Enterprise Miner

When the variables exhibit asymmetry and non-linearity, data transformation is necessary. In SAS Enterprise Miner, ***Transform Variables*** node is a great tool to handle data transformation. The diagram below (Figure 5) is an example of data transformation procedure in SAS Enterprise Miner.

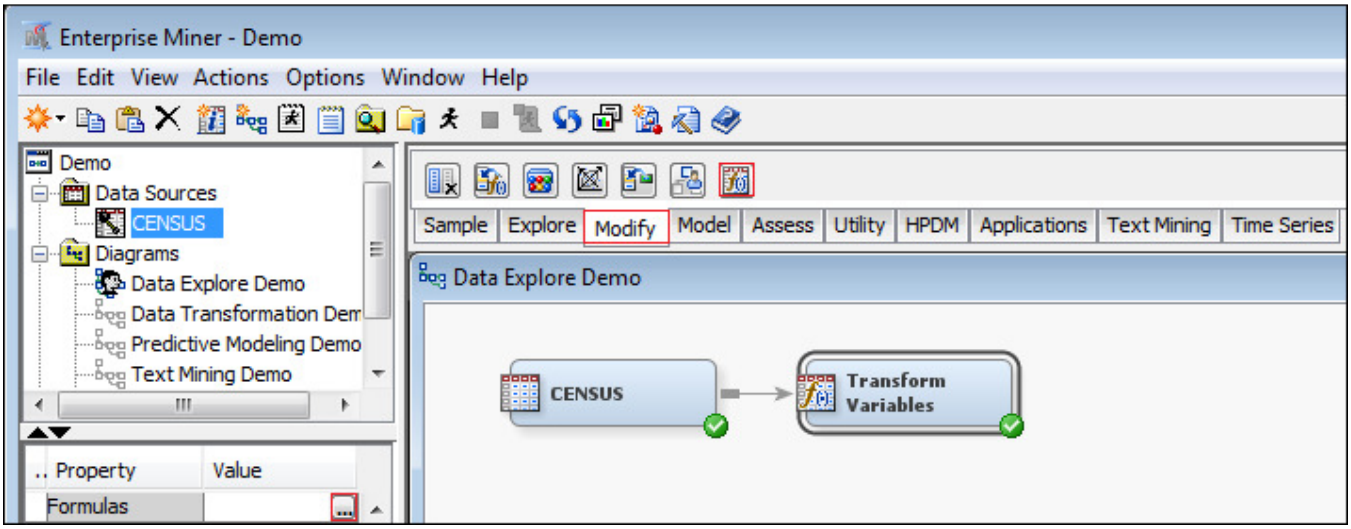


Figure 5. A Data Transformation Procedure Using ***Transform Variables*** Node in SAS Enterprise Miner

3.3.2 Data Partition for Training, Validation, and Testing

If data volume allows, data could be partitioned into training, validation, and holdout testing data sets. ***The training data set*** is used for preliminary model fitting. ***The validation data set*** is used to monitor and tune the model during estimation and is also used for model assessment. The tuning process usually involves selecting among models of different types and complexities with the goal of selecting the best model balancing between model accuracy and stability. ***The holdout testing data set*** is used to give a final honest model assessment. In reality, different break-down percentages across training, validation, and holdout testing data could be used depending on the data volume and the type of model to build, *etc.* It is not rare to only partition data into training and testing data sets, especially when data volume is concerned.

The diagram (Figure 6) below shows a data partition example. In this example, 80% of the data is for training, 10% for validation, and 10% for holdout testing.

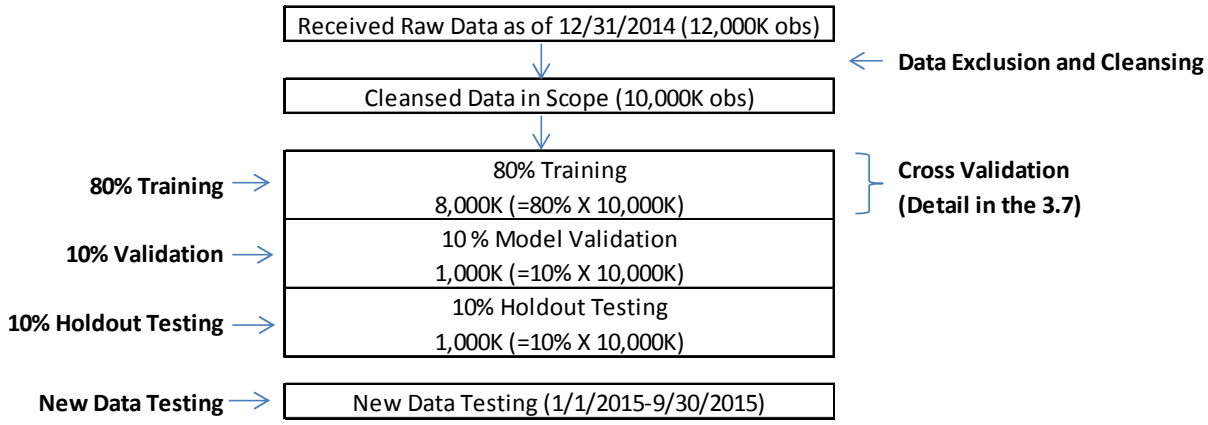


Figure 6. A Data Partition Flow Chart

The diagram below (Figure 7) is the data partition example in SAS Enterprise Miner. This node uses simple random sampling, stratified random sampling, or cluster sampling to create partitioned data sets.

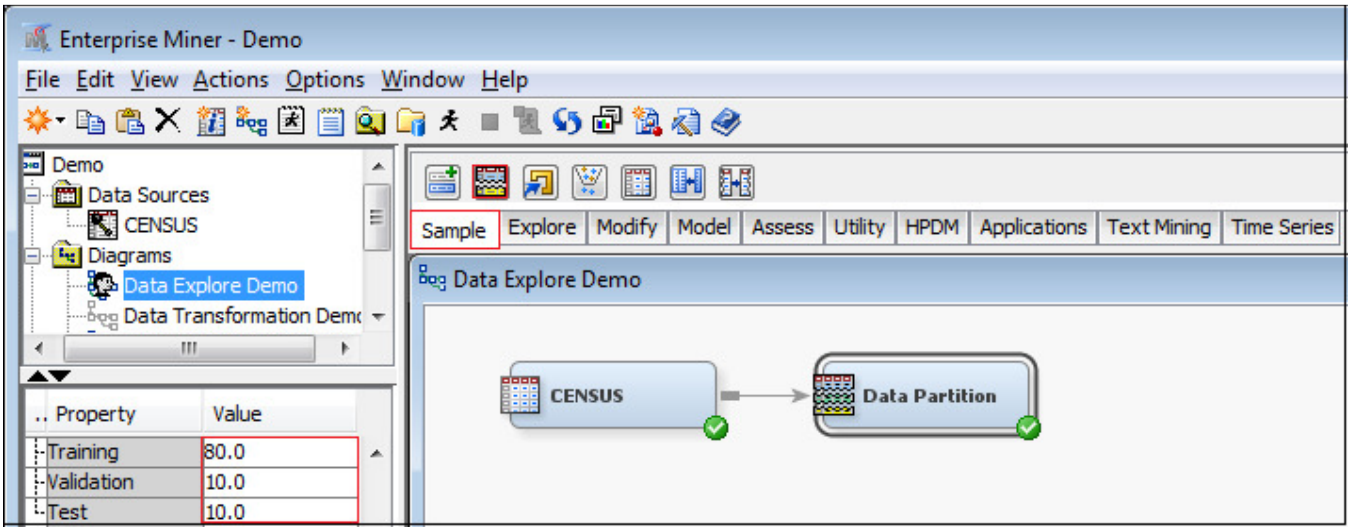


Figure 7. A Data Partition Procedure Example Using ***Data Partition*** Node in SAS Enterprise Miner

3.4 VARIABLE CREATION

3.4.1 Target Variable Creation (a.k.a.: Dependent Variable or Responsible Variable)

Every data mining project begins with a business goal which defines the target variable from a modeling perspective. The target variable summarizes the outcome we would like to predict from the perspective of the algorithms we use to build the predictive models. The target variable could be created based on either a single variable or a combination of multiple variables.

For example, we can create the ratio of total incurred loss to premium as the target variable for a loss ratio model.

Another example involves a large loss model. If the business problem is to identify the claims with total incurred loss greater than $250,000 and claim duration more than 2 years, then we can create a target variable - "1" when both total incurred loss exceeding $250,000 and claim duration more than 2 years, else "0".

3.4.2 Other Predictive Variables Creation

Many variables can be created directly from the raw data fields they represent. Other additional variables can be created based on the raw data fields and our understanding of the business. For example: loss month can be a variable created based on the loss date field to capture potential loss seasonality. It could be a potentially predictive variable to an automobile collision model since automobile collision losses are highly dependent on what season it is. When the claim has a prior claim, we can create a prior claim indicator which potentially could be predictive variable to a large loss model.

Another example involves external Census data, where a median household income field could be used to create a median household income ranking variable by ZIP code which could be predictive to workers' compensation model.

3.4.3 Text Mining (a.k.a.: Text Analytics) to create variables based on unstructured data

Text Analytics uses algorithms to derive patterns and trends from unstructured (free-form text) data through statistical and machine learning methods (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database), as well as natural language processing techniques. The diagram (Figure 8) below shows a text mining process example in SAS Enterprise Miner.

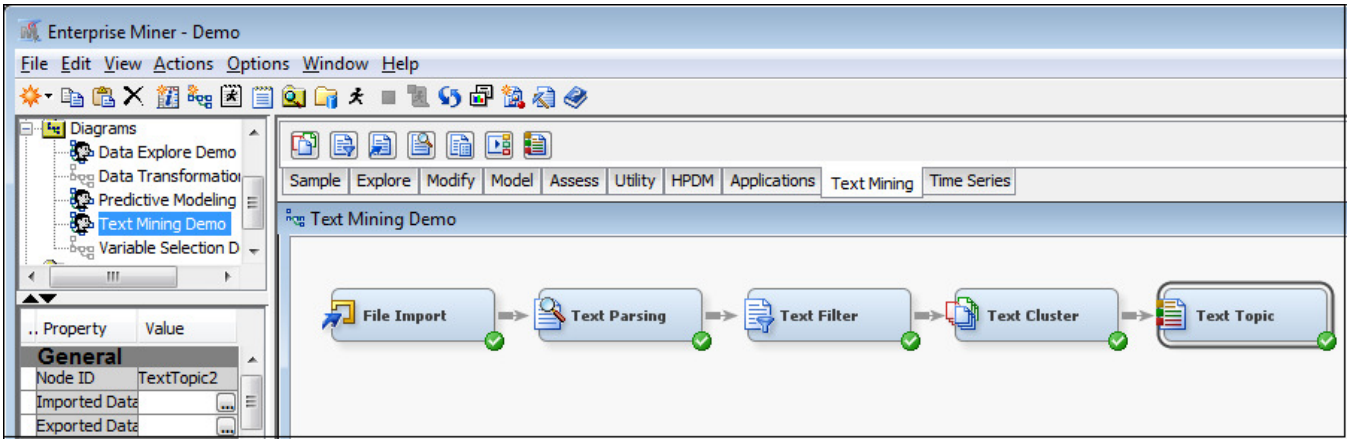


Figure 8. A Text Mining Procedure Example in SAS Enterprise Miner

3.4.4 Univariate Analysis

After creating target variable and other variables, univariate analysis usually has been performed. In the univariate analysis, one-way relationships of each potential predictive variable with the target variable are examined. Data volume and distribution are further reviewed to decide if the variable is credible and meaningful in both a business and a statistical sense. A high-level reasonability check is conducted. In this univariate analysis, our goal is to identify and select the most significant variables based on statistical and business reasons and determine the appropriate methods to group (bin), cap, or transform variables.

The SAS procedure `PROC UNIVARIATE` could be utilized in Base SAS and SAS Enterprise Guide. Some of the data review methods and techniques in data preparation could be utilized as well.

In addition to the previously introduced procedures, the diagram below (Figure 9) shows how Graph Explore procedure can also be used to conduct univariate analyses in SAS Enterprise Miner.

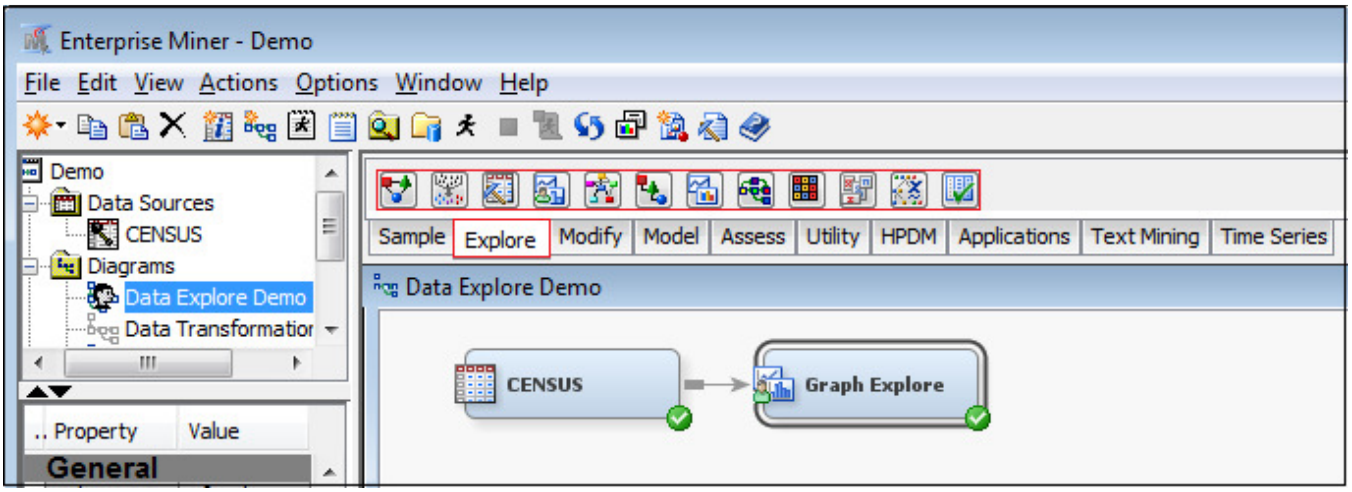


Figure 9. Univariate Analyses Using ***Graph Explore*** Node in SAS Enterprise Miner

3.5 VARIABLE SELECTION (a.k.a.: VARIABLE REDUCTION)

When there are over hundreds or even thousands of variables after including various internal and external data sources, the variable selection process becomes critical. Redundancy and irrelevancy are the two keys to reduce variables in the variable selection process. Redundancy means the variable doesn't provide any additional new information that other variables have already provided. Irrelevancy means that the variable doesn't provide any information about the target. See some common variable selection techniques below:

1) Correlation Analysis: Identify variables which are correlated to each other to avoid multicollinearity to build a more stable model

2) Multivariate Analyses: Cluster Analysis, Principle Component Analysis, and Factor Analysis. Cluster Analysis is popularly used to create clusters when there are hundreds or thousands of variables.

Some common SAS procedures as follows:

`PROC CORR/PROC VARCLUS/PROC FACTOR`

3) Stepwise Selection Procedure: *Stepwise selection* is a method that allows moves in either direction, dropping or adding variables at the various steps.

Backward stepwise selection starts with all the predictors to remove the least significant variable, and then potentially add back variables if they later appear to be significant. The process is one of alternation between choosing the least significant variable to drop and then re-considering all dropped variables (except the most recently dropped) for re-introduction into the model. This means that two separate significance levels must be chosen for deletion from the model and for adding to the model. The second significance must be more stringent than the first.

Forward stepwise selection is also a possibility, though not as common. In the forward approach, variables once entered may be dropped if they are no longer significant as other variables are added.

Stepwise Regression
This is a combination of backward elimination and forward selection. This addresses the situation where variables are added or removed early in the process and we want to change our mind about them later. At each stage a variable may be added or removed and there are several variations on exactly how this is done.

The simplified SAS code below shows how a stepwise logistic regression procedure to select variables for a logistic regression model (binary target variable) can be used in Base SAS/SAS EG.

```
proc logistic data = datasample;
   model target_fraud (event = '1')= var1 var2 var3
   /selection=stepwise  slentry=0.05  slstay=0.06;
output out=datapred1 p=phat lower=lcl upper=ucl;
run;
```

In SAS Enterprise Miner, procedures for selecting variables use the ***Variable Selection*** Node. This procedure provides a tool to reduce the number of input variables using R-square and Chi-square selection criteria. The procedure identifies input variables which are useful for predicting the target variable and ranks the importance of these variables. The diagram below (Figure 10) contains an example.

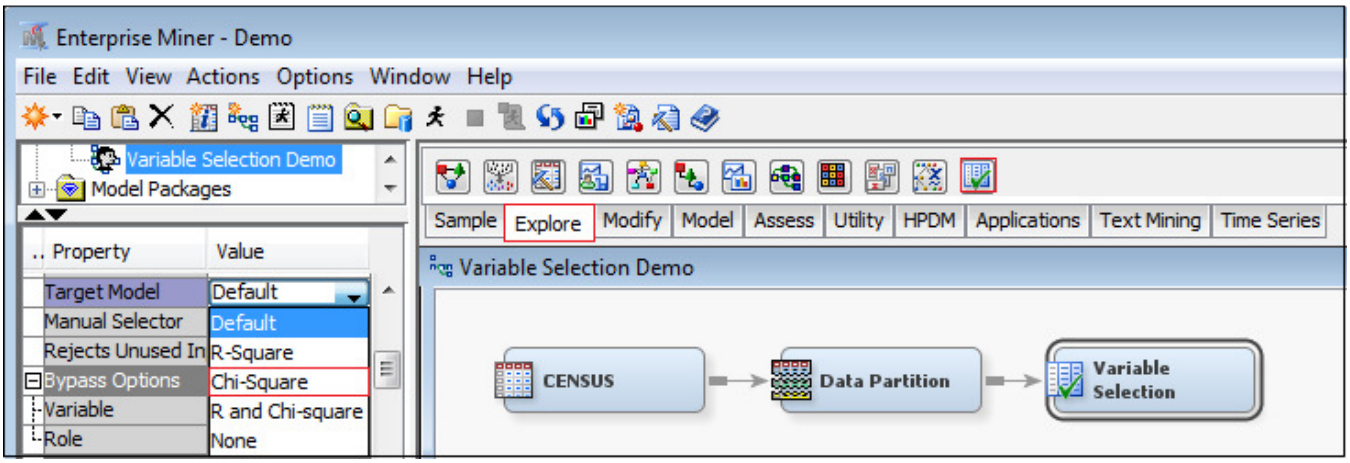


Figure 10. A Variable Selection Example Using ***Variable Selection*** node in SAS Enterprise Miner

In SAS Enterprise Miner, procedures for selecting variables involve using the ***Regression*** Node to specify a model selection method. If ***Backward*** is selected, training begins with all candidate effects in the model and removes effects until the Stay significance level or the stop criterion is met. If ***Forward*** is selected, training begins with no candidate effects in the model and adds effects until the Entry significance level or the stop criterion is

met. If ***Stepwise*** is selected, training begins as in the Forward model but may remove effects already in the model. This continues until the Stay significance level or the stop criterion is met. If ***None*** is selected, all inputs are used to fit the model. The diagram below (Figure 11) contains an example.

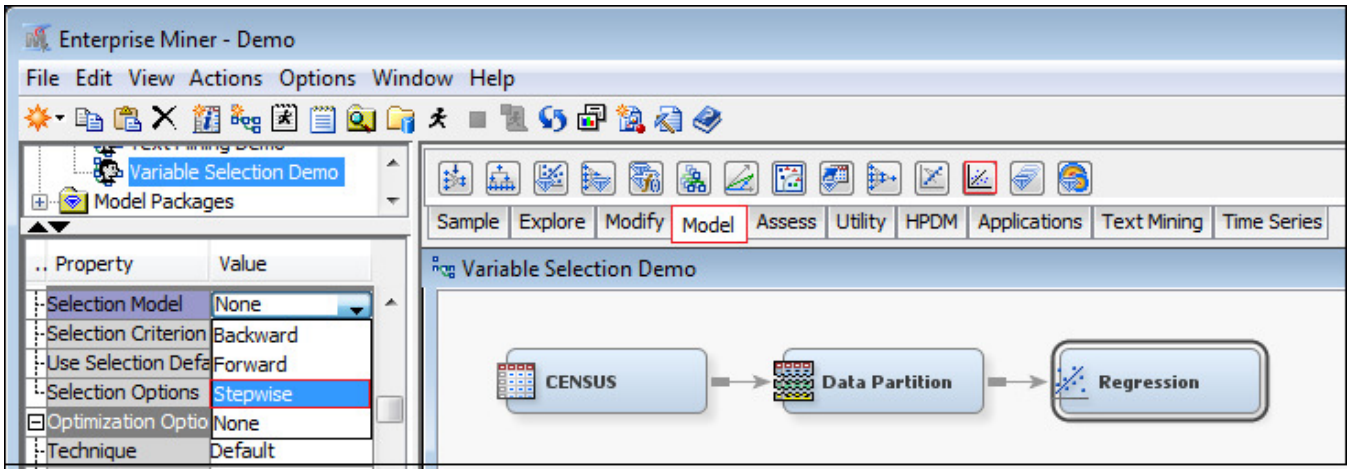


Figure 11. A Variable Selection Example Using ***Regression*** Node in SAS Enterprise Miner

In SAS Enterprise Miner, procedures for selecting variables can use the ***Decision Tree*** Node. This procedure provides a tool to reduce the number of input variables by specifying whether variable selection should be performed based on importance values. If this is set to ***Yes***, all variables that have an importance value greater than or equal to 0.05 will be set to Input. All other variables will be set to Rejected. The diagram below (Figure 12) contains an example.

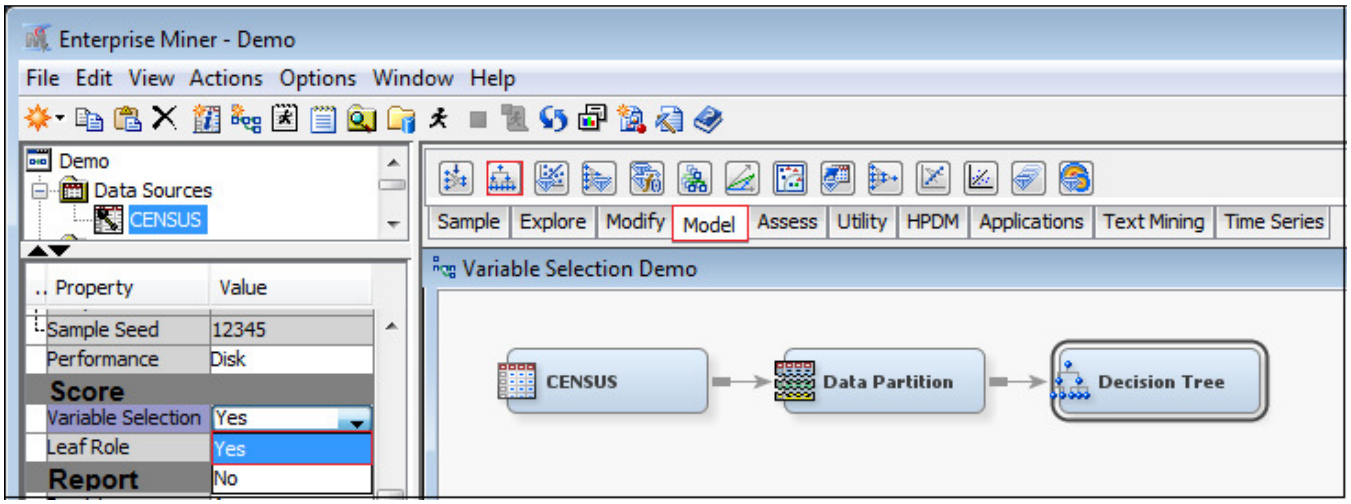


Figure 12. A Variable Selection Example Using ***Decision Tree*** Node in SAS Enterprise Miner

3.6 MODEL BUILDING (a.k.a.: MODEL FITTING)

An insufficient model might systematically miss the signal(s), which could lead to high bias and underfitting. The overly complicated model might perform well on a specific data set but mistakes the random noise in the data set for signal; such a model may lead to misleading results if it is used to predict on other data sets. There are usually many iterations to fit models until the final model which is based on both desired statistics (relatively simple, high

accuracy, and high stability) and business application. The final model includes the target variable, independent variables, and multivariate equations with weights and coefficients for the variables used. The Generalized Linear Modeling (GLM) technique has been popular in the property and casualty insurance industry for building statistical models.

Below is a simplified SAS code of a logistic regression fit using `PROC GENMOD` (GLM) in Base SAS/SAS Enterprise Guide:

```
proc genmod data=lib.sample;
  class var1 var2 var3 var4;
  model retention = var1 var2 var3 var4 var5
  /dist = bin
  link=logit lrci;
  output out=lib.sample p=pred;
run;
```

The same GLM logistic regression procedure can be done using ***Regression*** node with specifying model selection as ***GLM*** in SAS Enterprise Miner. The diagram below (Figure 13) contains an example.

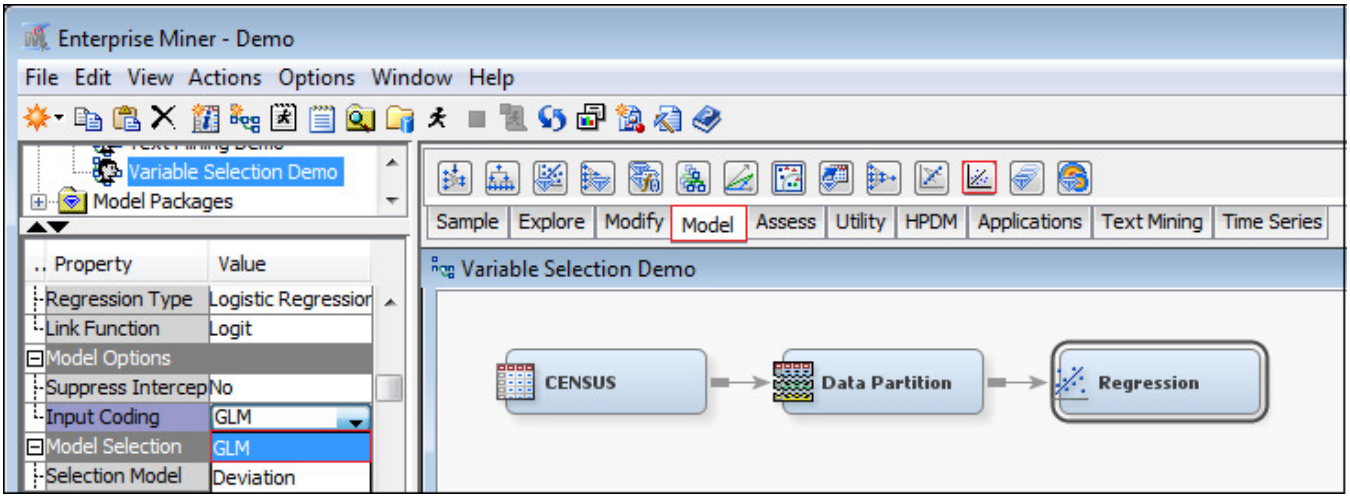


Figure 13. A GLM Logistic Regression Example Using ***Regression*** node in SAS Enterprise Miner

Interaction and correlation usually should be examined before finalizing the models if possible.

Other model building/fitting methodologies could be utilized to build models in SAS Enterprise Miner including the following three types of models (The descriptions below are attributable to SAS Product Documentation):

Decision Tree Model: Decision Tree is a predictive modeling approach which maps observations about an item to conclusions about the item's target value. A decision tree divides data into groups by applying a series of simple rules. The rules are organized hierarchically in a tree-like structure with nodes connected by lines. The first rule at the top of the tree is called the *root node*. Each rule assigns an observation to a group based on the value of one input. One rule is applied after another, resulting in a hierarchy of groups. The hierarchy is called a tree, and each group is called a node. The original group contains the entire data set and is called the root node of the tree. A node with all its successors forms a branch of the node that created it. The final nodes are called leaves. For each leaf, a decision is made and applied to all observations in the leaf. The paths from root to leaf represent classification rules.

Neural Network Model: Organic neural networks are composed of billions of interconnected neurons that send and receive signals to and from one another. Artificial neural networks are a class of flexible nonlinear models used for supervised prediction problems. The most widely used type of neural network in data analysis is the multilayer perceptron (MLP). MLP models were originally inspired by neurophysiology and the interconnections between neurons, and they are often represented by a network diagram instead of an equation. The basic building blocks of multilayer perceptrons are called **hidden units**. Hidden units are modeled after the neuron. Each hidden unit receives a linear combination of input variables. The coefficients are called the (synaptic) weights. An activation function transforms the linear combinations and then outputs them to another unit that can then use them as inputs.

Rule Induction Model: This model combines decision tree and neural network models to predict nominal targets. It is intended to be used when one of the nominal target levels is rare. New cases are predicted using a combination of prediction rules (from decision trees) and a prediction formula (from a neural network, by default).

The following diagram (Figure 14) shows a simplified example of model building procedure with four models in the SAS Enterprise Miner.

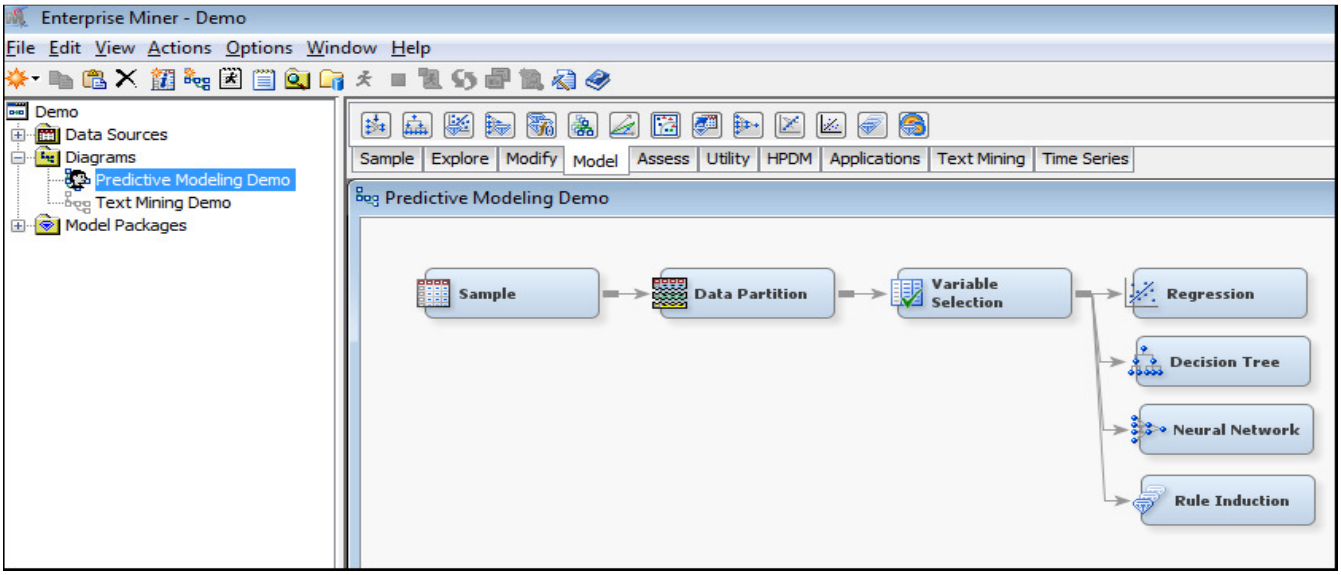


Figure 14. A Simplified Example of Model Building Procedure with Four Models

3.7 MODEL VALIDATION

Model validation is a process to apply the candidate models on the validation data set to select a best model with a good balance of model accuracy and stability. Common model validation methods include Lift Charts, Confusion Matrices, Receiver Operating Characteristic (ROC), Bootstrap Sampling, and Cross Validation, etc. to compare actual values (results) versus predicted values from the model. Bootstrap Sampling and Cross Validation methods are especially useful when data volume is not high.

Cross Validation is introduced through the example (Figure 15) is below. Four cross fold subset data sets are created for validating stability of parameter estimates and measuring lift. The diagram shows one of the cross folds, but the other three would be created by taking other combinations of the cross validation data sets.

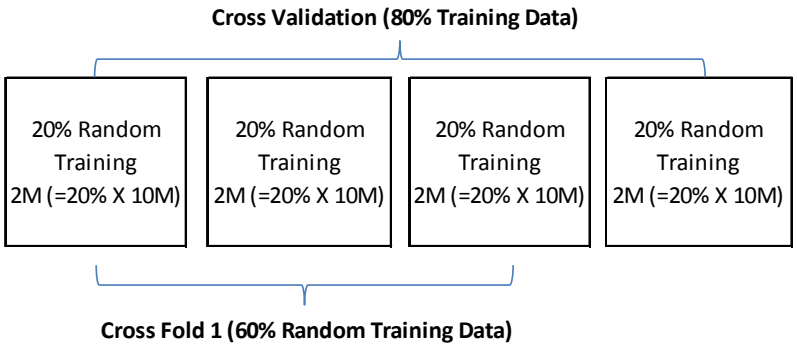


Figure 15. A Diagram of Creating Cross Four Fold Subset Data Sets for Cross Validation

In SAS Enterprise Miner, run the model fitting (Logistics Regression or Decision Tree, etc.) on each of the cross fold subset data sets to get parameter estimates. Then examine the four sets of parameter estimates side by side to see if they are stable. A macro could be created and utilized to run the same model fitting process on four cross fold subset data sets.

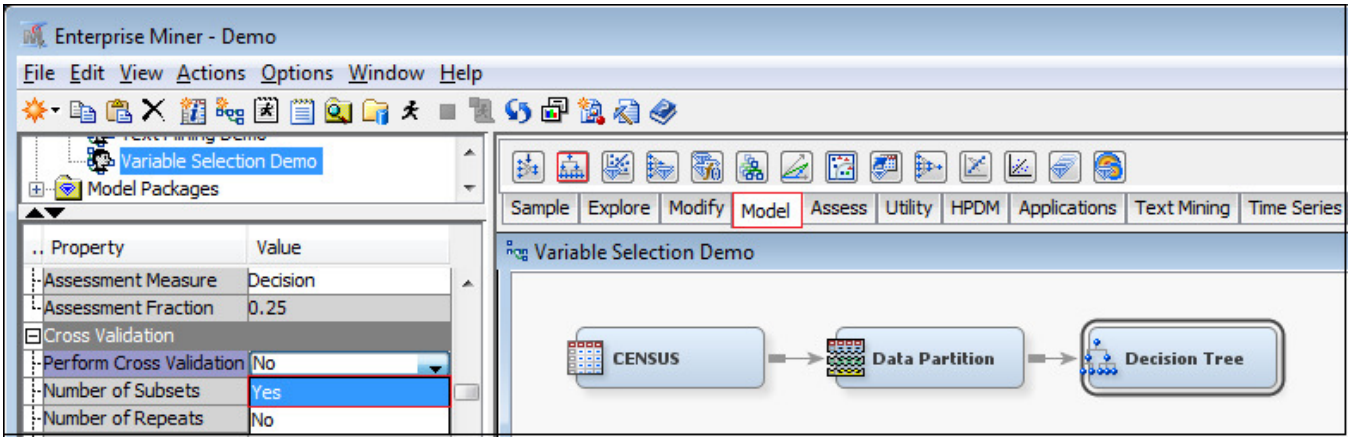


Figure 16. A Cross Validation Example Using ***Decision Tree*** node in SAS Enterprise Miner

The following common fit statistics are reviewed for model validation:

- Akaike's Information Criterion
- Average Squared Error
- Average Error Function
- Misclassification Rate
- Mean Square

3.8 MODEL TESTING

The overall model performance on validation data could be overstated because the validation data has been used to select the best model. Therefore, model testing is necessarily performed to further evaluate the model performance and provide a final unbiased assessment of model performance. Model testing methods are similar as model validation but using holdout testing data and/or new data (See Figure 6).

The following diagram (Figure 17) shows a predictive modeling process with major stages starting from prepared data to data partition, variable selection, building logistic regression model, and testing (scoring) new data in SAS Enterprise Miner.

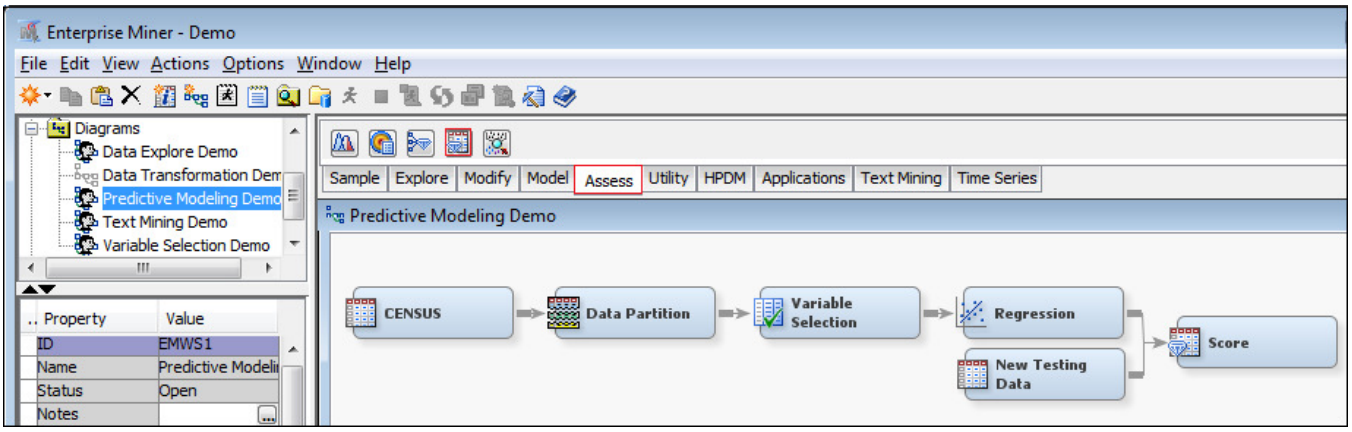


Figure 17. A Simplified Predictive Modeling Flow Chart in SAS Enterprise Miner

3.9 MODEL IMPLEMENTATION

The last but very important stage of any predictive modeling project is the model implementation which is the stage to turn all the modeling work into action to achieve the business goal and/or solve the business problem. Usually before model implementation, a model pilot would be helpful to get some sense of the model performance and prepare for the model implementation appropriately. If there is an existing model, we would also like to conduct a model champion challenge to understand the benefit of implementing the new model over the old one. There are numeric ways to implement the models which largely depends on what type of models and the collaboration with IT support at the organization.

4.0 SOME SUCCESSFUL MODELS

Pricing Model is to predict how much premium should be charged for each individual policy. Retention Model is to predict whether the policy will stay with the insurance company or not. Fraud Model is to predict whether the policy will be involved in fraud or not. Large Loss Model is to predict if the loss will be exceeding a certain threshold or not.

5.0 CONCLUSION

While we are aware that there are always multiple ways to build models to accomplish the same business goals, the primary goal of this paper is to introduce a common P&C insurance predictive modeling process in SAS Enterprise Guide and Enterprise Miner. P&C insurance predictive analytics, just like any other industry, has been further developing and evolving along with new technology, new data sources, and new business problems. There will be more advanced predictive analytics to solve more business problems using SAS products in the future.

6.0 TRADEMARK CITATIONS

7.0 REFERENCES:

Census Data Source (http://www.census.gov/)

SAS Product Documentation (http://support.sas.com/documentation/)

8.0 CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Mei Najim
Company: Sedgwick Claim Management Services
E-mail: mei.najim@sedgwick.com/yumei100@gmail.com