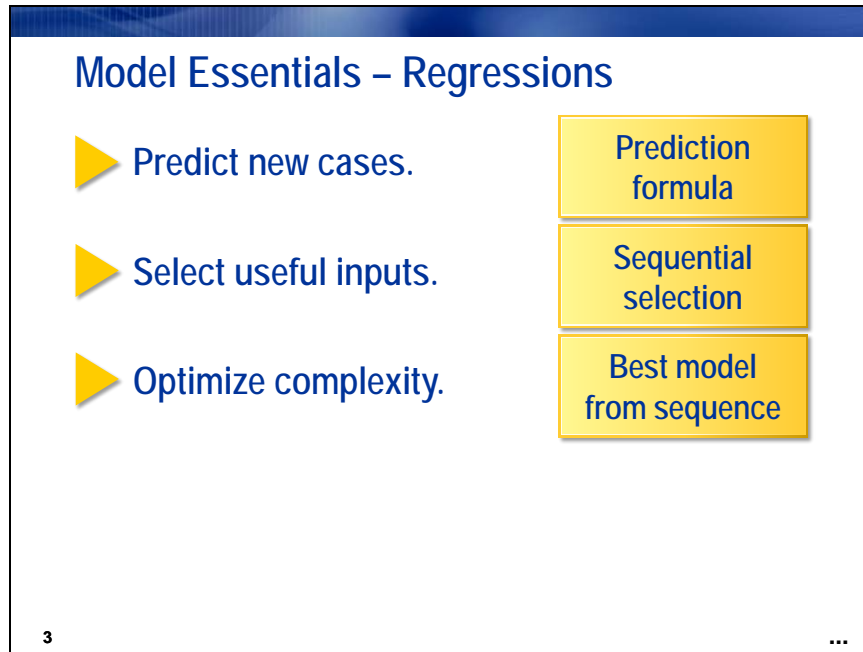


Chapter 4 Introduction to Predictive Modeling: Regressions

0.1	Introduction.....	Error! Bookmark not defined.
0.2	A Section Title.....	Error! Bookmark not defined.
	Demonstration: <Type title of demo here.>.....	Error! Bookmark not defined.
	Exercises	Error! Bookmark not defined.
0.3	Chapter Summary.....	Error! Bookmark not defined.
0.4	Solutions	Error! Bookmark not defined.
	Solutions to Exercises	Error! Bookmark not defined.
	Solutions to Student Activities (Polls/Quizzes)	Error! Bookmark not defined.

4.1 Introduction



Regressions offer a different approach to prediction compared to decision trees. Regressions, as parametric models, assume a specific association structure between inputs and target. By contrast, trees, as predictive algorithms, do not assume any association structure; they simply seek to isolate concentrations of cases with like-valued target measurements.

The regression approach to the model essentials in SAS Enterprise Miner is outlined over the following pages. Cases are scored using a simple mathematical *prediction formula*. One of several heuristic *sequential selection* techniques is used to pick from a collection of possible inputs and creates a series of models with increasing complexity. Fit statistics calculated from validation data select the *best model from the sequence*.

Model Essentials – Regressions

▶ Predict new cases.

Prediction
formula

▶ Select useful inputs.

Sequential
selection

▶ Optimize complexity.

Best model
from sequence

5

Regressions predict cases using a mathematical equation involving values of the input variables.

Linear Regression Prediction Formula

$$\hat{y} = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2$$

prediction estimate
input measurement
intercept estimate *parameter estimate* *parameter estimate*

Choose intercept and parameter estimates to *minimize*:

$$\sum_{\text{training data}} (y_i - \hat{y}_i)^2$$

squared error function

6

...

In standard linear regression, a prediction estimate for the target variable is formed from a simple linear combination of the inputs. The intercept centers the range of predictions, and the remaining parameter estimates determine the trend strength (or slope) between each input and the target. The simple structure of the model forces changes in predicted values to occur in only a single direction (a vector in the space of inputs with elements equal to the parameter estimates).

Intercept and *parameter estimates* are chosen to minimize the squared error between the predicted and observed target values (least squares estimation). The prediction estimates can be viewed as a linear approximation to the expected (average) value of a target conditioned on observed input values.

Linear regressions are usually deployed for targets with an interval measurement scale.

Logistic Regression Prediction Formula

$$\log \left(\frac{\hat{p}}{1 - \hat{p}} \right) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2 \quad \text{logit scores}$$

8

...

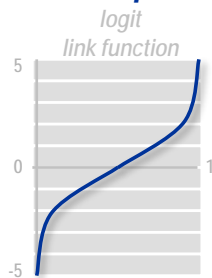
Logistic regressions are closely related to linear regressions. In logistic regression, the expected value of the target is transformed by a link function to restrict its value to the unit interval. In this way, model predictions can be viewed as primary outcome probabilities. A linear combination of the inputs generates a *logit score*, the log of the odds of primary outcome, in contrast to the linear regression's direct prediction of the target.



If your interest is ranking predictions, linear and logistic regressions yield virtually identical results.

Logit Link Function

$$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2 \quad \text{logit scores}$$



The logit link function transforms probabilities (between 0 and 1) to logit scores (between $-\infty$ and $+\infty$).

9

...

For binary prediction, any monotonic function that maps the unit interval to the real number line can be considered as a link. The logit link function is one of the most common. Its popularity is due, in part, to the interpretability of the model.

Logit Link Function

$$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2 = \text{logit}(\hat{p})$$

$$\hat{p} = \frac{1}{1 + e^{-\text{logit}(\hat{p})}}$$

To obtain prediction estimates, the logit equation is solved for \hat{p} .

11

...

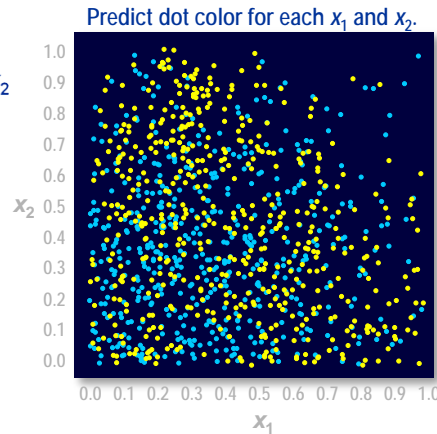
The predictions can be decisions, rankings, or estimates. The logit equation produces a ranking or logit score. To get a decision, you need a threshold. The easiest way to get a meaningful threshold is to convert the prediction ranking to a prediction estimate. You can obtain a prediction estimate using a straightforward transformation of the logit score, the logistic function. The *logistic function* is simply the inverse of the logit function. You can obtain the logistic function by solving the logit equation for p .

Simple Prediction Illustration – Regressions

$$\text{logit}(\hat{p}) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2$$

$$\hat{p} = \frac{1}{1 + e^{-\text{logit}(\hat{p})}}$$

You need intercept and parameter estimates.



14

...

To demonstrate the properties of a logistic regression model, consider the two-color prediction problem introduced in Chapter 3. As before, the goal is to predict the target color, based on the location in the unit square. To make use of the prediction formulation, you need estimates of the intercept and other model parameters.

Simple Prediction Illustration – Regressions

$$\text{logit}(\hat{p}) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2$$

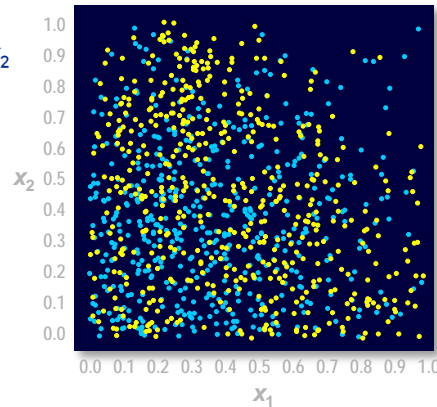
$$\hat{p} = \frac{1}{1 + e^{-\text{logit}(\hat{p})}}$$

Find parameter estimates by maximizing

$$\sum \log(\hat{p}_i) + \sum \log(1 - \hat{p}_i)$$

primary outcome training cases secondary outcome training cases

log-likelihood function



16

...

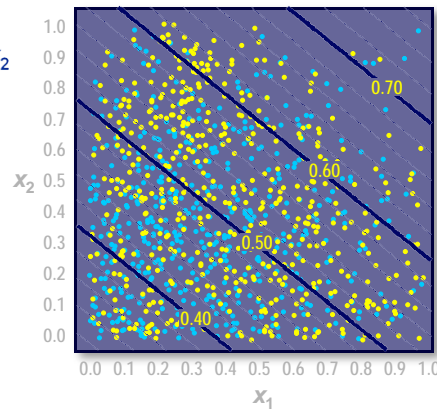
The presence of the logit link function complicates parameter estimation. Least squares estimation is abandoned in favor of maximum likelihood estimation. The likelihood function is the joint probability density of the data treated as a function of the parameters. The maximum likelihood estimates are the values of the parameters that maximize the probability of obtaining the training sample.

Simple Prediction Illustration – Regressions

$$\text{logit}(\hat{p}) = -0.81 + 0.92x_1 + 1.11x_2$$

$$\hat{p} = \frac{1}{1 + e^{-\text{logit}(\hat{p})}}$$

Using the maximum likelihood estimates, the prediction formula assigns a logit score to each x_1 and x_2 .



18

...

Parameter estimates are obtained by maximum likelihood estimation. These estimates can be used in the logit and logistic equations to obtain predictions. The plot on the right shows the prediction estimates from the logistic equation. One of the attractions of a standard logistic regression model is the simplicity of its predictions. The contours are simple straight lines. (In higher dimensions, they would be hyperplanes.)

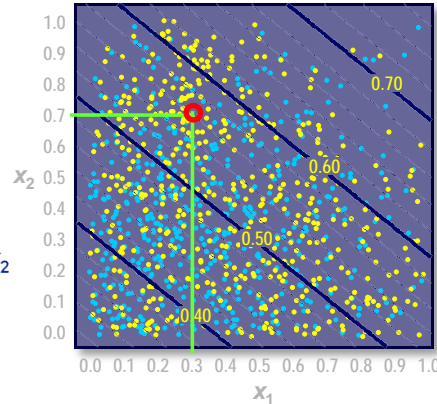
4.01 Multiple Choice Poll

What is the logistic regression prediction for the indicated point?

- a. -0.243
- b. 0.56
- c. yellow
- d. It depends ...

$$\text{logit}(\hat{p}) = -0.81 + 0.92x_1 + 1.11x_2$$

$$\hat{p} = \frac{1}{1 + e^{-\text{logit}(\hat{p})}}$$



20

To score a new case, the values of the inputs are plugged into the logit or logistic equation. This action creates a logit score or prediction estimate. Typically, if the prediction estimate is greater than 0.5 (or equivalently, the logit score is positive), cases are usually classified to the primary outcome. (This assumes an equal misclassification cost.)

The answer to the question posed is, of course, it depends.

- Answer A, the logit score, is reasonable if the goal is ranking.
- Answer B, the prediction estimate from the logistic equation, is appropriate if the goal is estimation.
- Answer C, a classification, is a good choice if the goal is deciding dot color.

Regressions: Beyond the Prediction Formula

- ▶ Manage missing values.
- ▶ Interpret the model.
- ▶ Handle extreme or unusual values.
- ▶ Use nonnumeric inputs.
- ▶ Account for nonlinearities.

22

...

While the prediction formula would seem to be the final word in scoring a new case with a regression model, there are actually several additional issues that must be addressed.

- What should be done when one of the input values used in the prediction formula is missing? You might be tempted to simply treat the missing value as zero and skip the term involving the missing value. While this approach can generate a prediction, this prediction is usually biased beyond reason.
- How do you interpret the logistic regression model? Certain inputs influence the prediction more than others. A means to quantify input importance is needed.
- How do you score cases with unusual values? Regression models make their best predictions for cases near the centers of the input distributions. If an input can have (on rare occasion) extreme or *outlying* values, the regression should respond appropriately.
- What value should be used in the prediction formula when the input is not a number? Categorical inputs are common in predictive modeling. They did not present a problem for the rule-based predictions of decision trees, but regression predictions come from algebraic formulas that require numeric inputs. (You cannot multiply marital status by a number.) A method to include nonnumeric data in regression is needed.
- What happens when the relationship between the inputs and the target (or rather logit of the target) is not a straight line? It is preferable to be able to build regression models in the presence of nonlinear (and even nonadditive) input target associations.

Regressions: Beyond the Prediction Formula

- ▶ Manage missing values.
- ▶ Interpret the model.
- ▶ Handle extreme or unusual values.
- ▶ Use nonnumeric inputs.
- ▶ Account for nonlinearities.

23

...



The above issues affect both model construction and model deployment. The first of these, handling missing values, is dealt with immediately. The remaining issues are addressed, in turn, at the end of this chapter.

Missing Values and Regression Modeling

Training Data

			inputs				target

Problem 1: Training data cases with missing values on inputs used by a regression model are ignored.

24

...

Missing values present two distinct problems. The first relates to model construction. The default method for treating missing values in most regression tools in SAS Enterprise Miner is *complete-case analysis*. In complete-case analysis, only those cases without any missing values are used in the analysis.

Missing Values and Regression Modeling

Training Data

			inputs				target

Consequence: Missing values can significantly reduce your amount of training data for regression modeling!

26

...

Even a smattering of missing values can cause an enormous loss of data in high dimensions. For instance, suppose that each of the k input variables is missing at random with probability α . In this situation, the expected proportion of complete cases is as follows:

$$(1 - \alpha)^k$$

Therefore, a 1% probability of missing ($\alpha=.01$) for 100 inputs leaves only 37% of the data for analysis, 200 leaves 13%, and 400 leaves 2%. If the “missingness” were increased to 5% ($\alpha=.05$), then <1% of the data would be available with 100 inputs.

Missing Values and the Prediction Formula

$$\text{logit}(\hat{p}) = -0.81 + 0.92 \cdot x_1 + 1.11 \cdot x_2$$

Predict: (x1, x2) = (0.3, ?)

Problem 2: Prediction formulas cannot score cases with missing values.

27

...

Missing Values and the Prediction Formula

$$\text{logit}(\hat{p}) = ?$$

Problem 2: Prediction formulas cannot score cases with missing values.

30

...

The second missing value problem relates to model deployment or using the prediction formula. How would a model built on the complete cases score a new case if it had a missing value? To decline to score new incomplete cases would be practical only if there were a very small number of missing values.

Missing Value Issues

Manage missing values.

Problem 1: Training data cases with missing values on inputs used by a regression model are ignored.

Problem 2: Prediction formulas cannot score cases with missing values.

31

...

A remedy is needed for the two problems of missing values. The appropriate remedy depends on the reason for the missing values.

Missing Value Causes

► Manage missing values.

- ☐ Non-applicable measurement
- ☐ No match on merge
- ☐ Non-disclosed measurement

33

...

Missing values arise for a variety of reasons. For example, the time since last donation to a card campaign is meaningless if someone did not donate to a card campaign. In the **PVA97NK** data set, several demographic inputs have missing values in unison. The probable cause was no address match for the donor. Finally, certain information, such as an individual's total wealth, is closely guarded and is often not disclosed.

Missing Value Remedies

► Manage missing values.

- ☐ Non-applicable measurement
- ☐ No match on merge
- ☐ Non-disclosed measurement

Synthetic distribution



Estimation

$$x_i = f(x_1, \dots, x_p)$$

34

...

The primary remedy for missing values in regression models is a missing value replacement strategy. Missing value replacement strategies fall into one of two categories.

***Synthetic
distribution
methods***

use a one-size-fits-all approach to handle missing values. Any case with a missing input measurement has the missing value replaced with a fixed number. The net effect is to modify an input's distribution to include a point mass at the selected fixed number. The location of the point mass in synthetic distribution methods is not arbitrary. Ideally, it should be chosen to have minimal impact on the magnitude of an input's association with the target. With many modeling methods, this can be achieved by locating the point mass at the input's mean value.

***Estimation
methods***

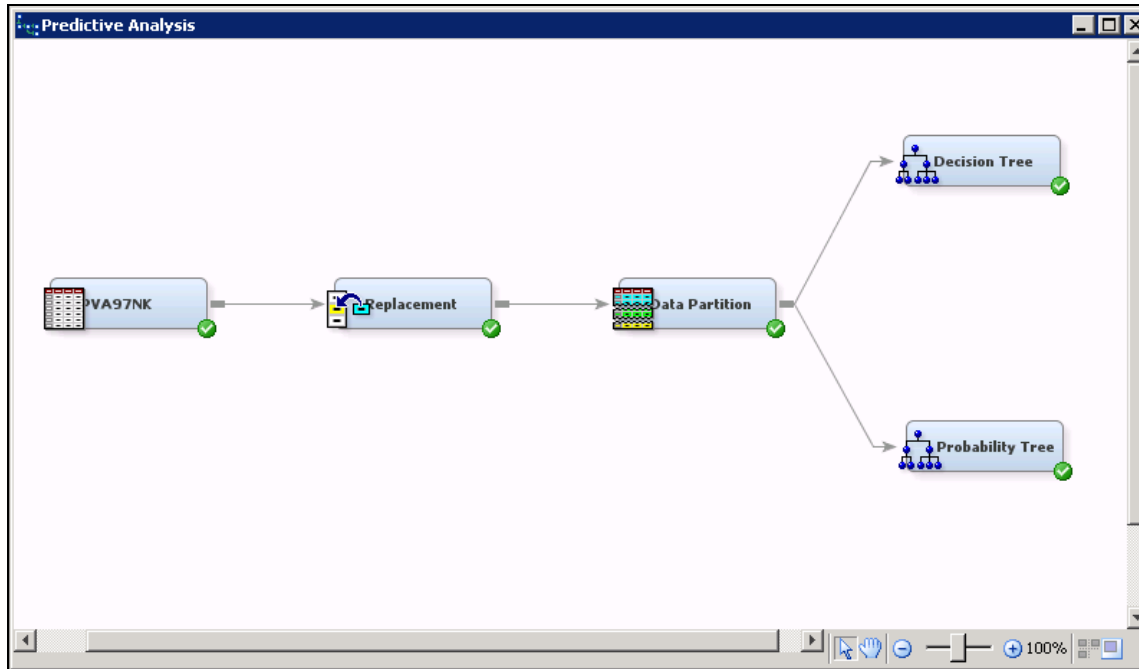
eschew the one-size-fits-all approach and provide tailored imputations for each case with missing values. This is done by viewing the missing value problem as a prediction problem. That is, you can train a model to predict an input's value from other inputs. Then, when an input's value is unknown, you can use this model to predict or estimate the unknown missing value. This approach is best suited for missing values that result from a lack of knowledge, that is, no-match or nondisclosure, but it is not appropriate for not-applicable missing values.

Because predicted response might be different for cases with a missing input value, a binary imputation indicator variable is often added to the training data. Adding this variable enables a model to adjust its predictions in the situation where "missingness" itself is correlated with the target.




Managing Missing Values

The demonstrations in this chapter build on the demonstrations of Chapters 2 and 3. At this point, the process flow diagram has the following structure:



Data Assessment

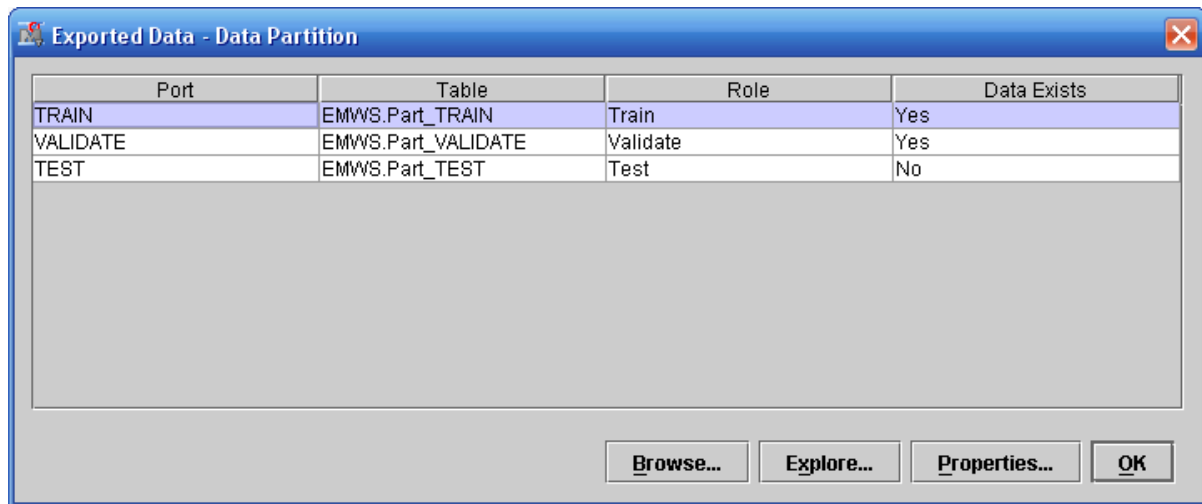
Continue the analysis at the Data Partition node. As discussed above, regression requires that a case have a complete set of input values for both training and scoring. Follow these steps to examine the data status after the partition.

1. Select the **Data Partition** node.
2. Select **Exported Data** ⇨  from the Data Partition node property sheet.

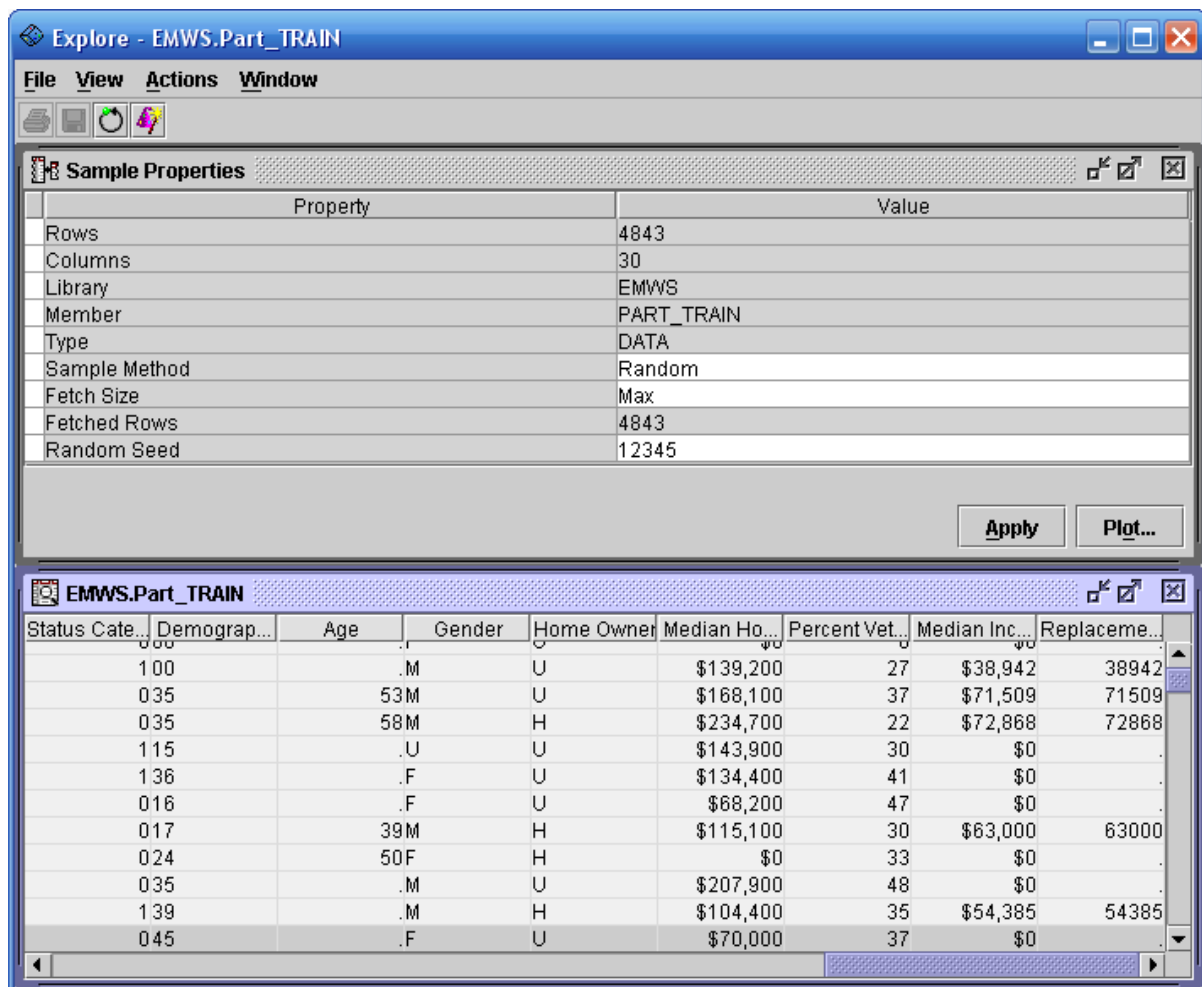
Property	Value
General	
Node ID	Part
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Output Type	Data
Partitioning Method	Default
Random Seed	12345
Data Set Allocations	
Training	50.0
Validation	50.0
Test	0.0

The Exported Data window opens.

3. Select the **TRAIN** data port and select **Explore...**.



There are several inputs with a noticeable frequency of missing values, for example, **Age** and the replaced value of median income.



There are several ways to proceed:

- **Do nothing.** If there are very few cases with missing values, this is a viable option. The difficulty with this approach comes when the model must predict a new case that contains a missing value. Omitting the missing term from the parametric equation usually produces an extremely biased prediction.
- **Impute** a synthetic value for the missing value. For example, if an interval input contains a missing value, replace the missing value with the mean of the nonmissing values for the input. This eliminates the incomplete case problem but modifies the input's distribution. This can bias the model predictions.

Making the missing value imputation process part of the modeling process allays the modified distribution concern. Any modifications made to the training data are also made to the validation data and the remainder of the modeling population. A model trained with the modified training data will not be biased if the same modifications are made to any other data set that the model might encounter (and the data has a similar pattern of missing values).

- Create a **missing indicator** for each input in the data set. Cases often contain missing values for a reason. If the reason for the missing value is in some way related to the target variable, useful predictive information is lost.

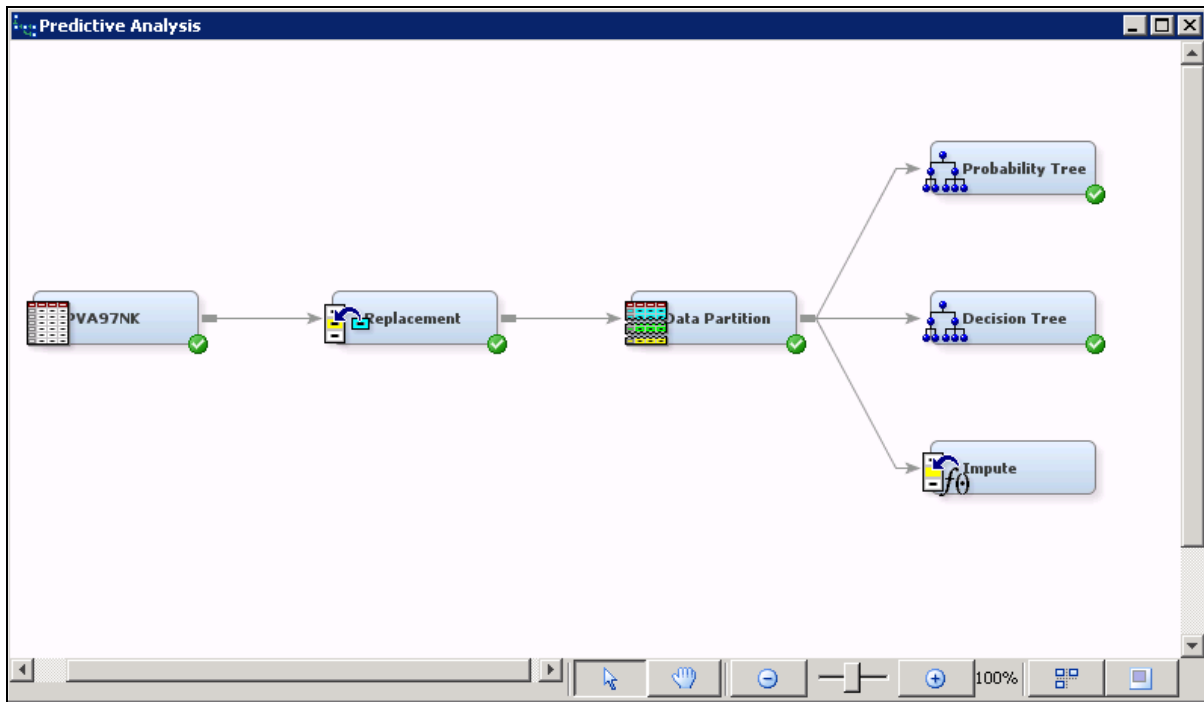
The missing indicator is 1 when the corresponding input is missing and 0 otherwise. Each missing indicator becomes an input to the model. This enables modeling of the association between the target and a missing value on an input.

4. Close the Explore and Exported Data windows.

Imputation

To address missing values in the **PVA97NK** data set, use the following steps to impute synthetic data values and create missing value indicators:

1. Select the **Modify** tab.
2. Drag an **Impute** tool into the diagram workspace.
3. Connect the **Data Partition** node to the **Impute** node. In the display, below, the Decision Tree modeling nodes are repositioned for clarity.



4. Select the **Impute** node and examine the Properties panel.

Property	Value
General	
Node ID	Impt
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Non Missing Variables	No
Missing Cutoff	50.0
Class Variables	
Default Input Method	Count
Default Target Method	None
Normalize Values	Yes
Interval Variables	
Default Input Method	Mean
Default Target Method	None
Default Constant Value	
Default Character Value	
Default Number Value	.
Method Options	
Random Seed	12345
Tuning Parameters	...
Tree Imputation	...
Score	
Hide Original Variables	Yes
Indicator Variables	
Type	Unique
Source	Imputed Variables

The defaults of the Impute node are as follows:

- For interval inputs, replace any missing values with the mean of the nonmissing values.
- For categorical inputs, replace any missing values with the most frequent category.

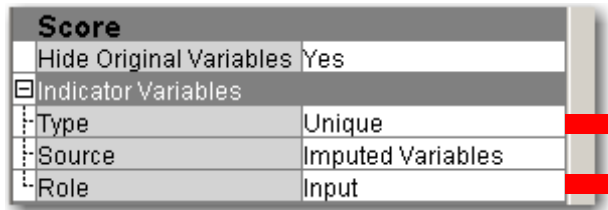


These are acceptable default values and are used throughout the rest of the course.

With these settings, each input with missing values generates a new input. The new input named *IMP_original_input_name* will have missing values replaced by a synthetic value and nonmissing values copied from the original input.

Missing Indicators

Use the following steps to create missing value indicators. The settings for missing value indicators are found in the Score property group.



1. Select **Indicator Variables** ⇒ **Type** ⇒ **Unique**.
2. Select **Indicator Variables** ⇒ **Role** ⇒ **Input**.

With these settings, new inputs named *M_original_input_name* will be added to the training data to indicate the synthetic data values.

Imputation Results

Run the Impute node and review the Results window. Three inputs had missing values.

Imputation Summary

Variable Name	Impute Method	Imputed Variable	Indicator Variable	Impute Value	Role	Measurement Level	Label	Number of Missing for TRAIN
DemAge	MEAN	IMP_DemA...	M_DemAge	59.26291	INPUT	INTERVAL	Age	1203
GiftAvgCard...	MEAN	IMP_GiftAvg...	M_GiftAvgC...	14.2049	INPUT	INTERVAL	Gift Amount...	910
REP_Dem...	MEAN	IMP_REP_...	M_REP_De...	53570.85	INPUT	INTERVAL	Replaceme...	1191

Output

```

*-----*
User:
Date:
Time:
*-----*
* Training Output
*-----*

Variable Summary
  
```

With all of the missing values imputed, the entire training data set is available for building the logistic regression model. In addition, a method is in place for scoring new cases with missing values. (See Chapter 7.)

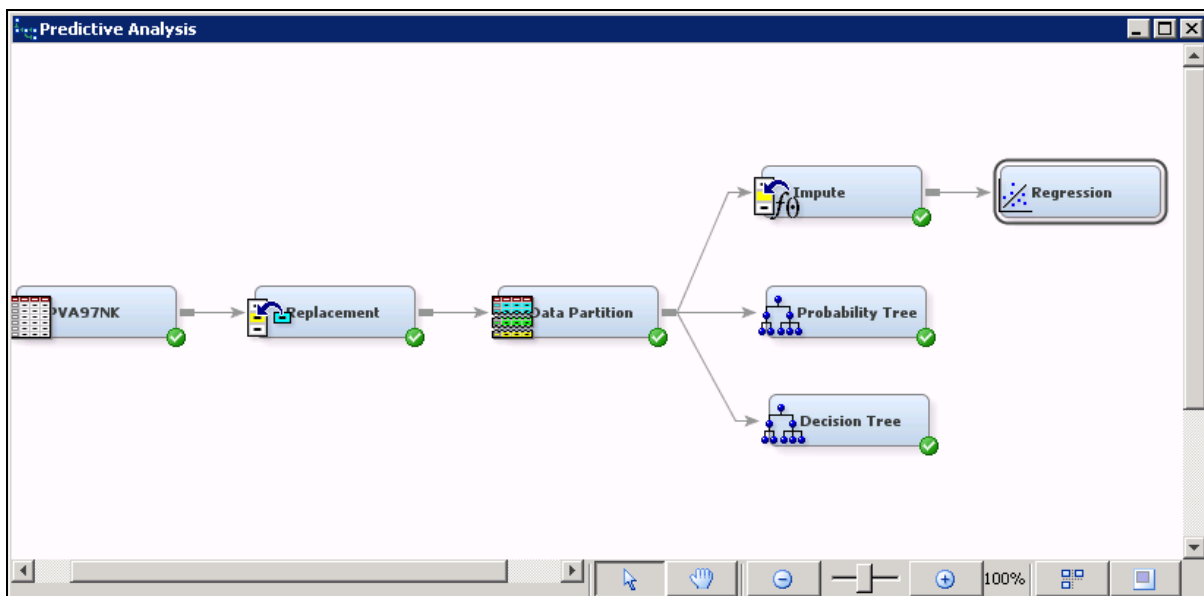


Running the Regression Node

There are several tools in SAS Enterprise Miner to fit regression or regression-like models. By far, the most commonly used (and, arguably, the most useful) is the simply named Regression tool.

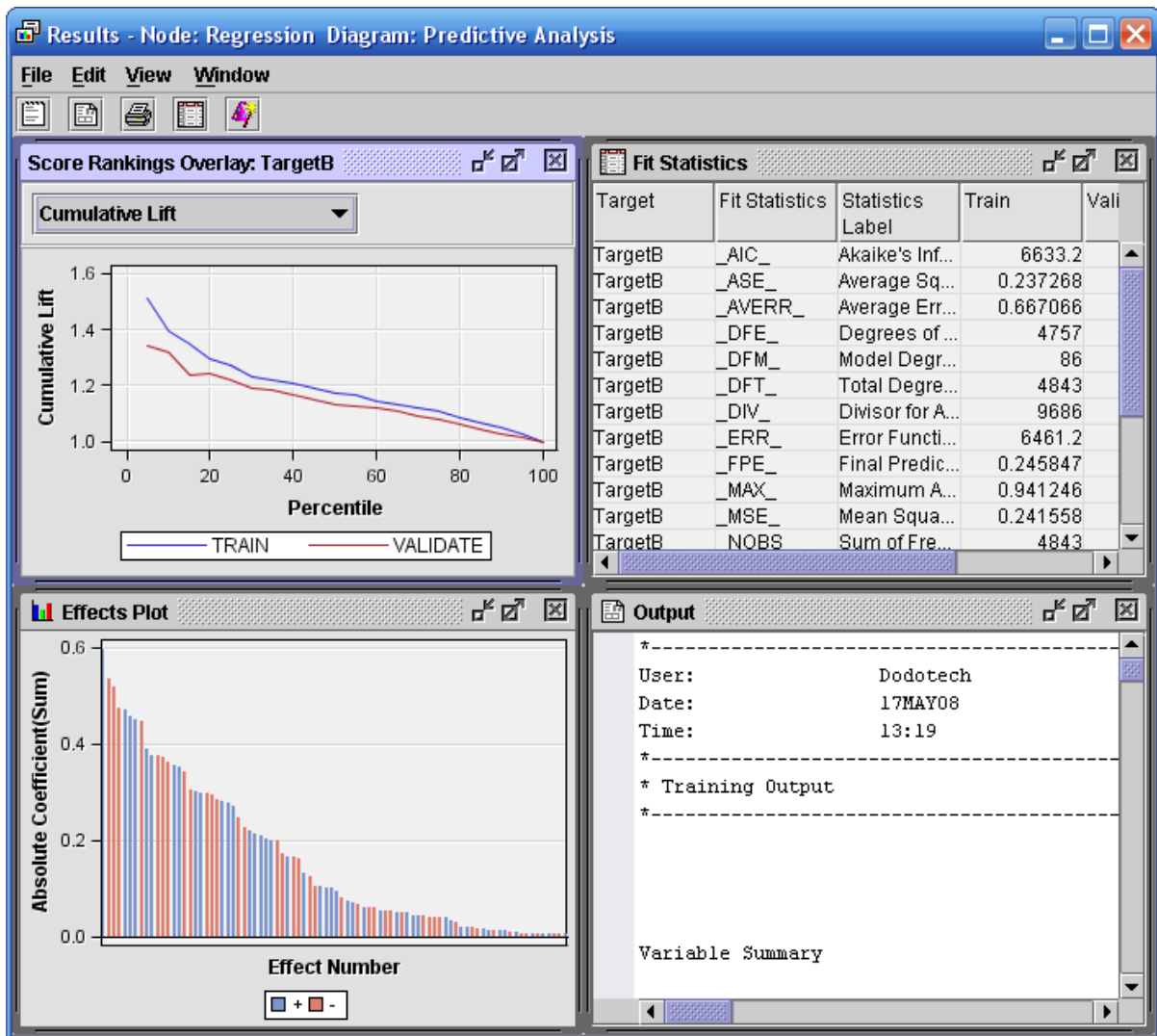
Use the following steps to build a simple regression model.

1. Select the **Model** tab.
2. Drag a **Regression** tool into the diagram workspace.
3. Connect the **Impute** node to the **Regression** node.



The Regression node can create several types of regression models, including linear and logistic. The type of default regression type is determined by the target's measurement level.

- Run the Regression node and view the results. The Results - Node: Regression Diagram window opens.



- Maximize the Output window by double-clicking its title bar.

The initial lines of the Output window summarize the roles of variables used (or not) by the Regression node.

Variable Summary		
ROLE	LEVEL	COUNT
INPUT	BINARY	5
INPUT	INTERVAL	20
INPUT	NOMINAL	3
REJECTED	INTERVAL	2
TARGET	BINARY	1

The fit model has 28 inputs that predict a binary target.

Ignore the output related to model events and predicted and decision variables. The next lines give more information about the model, including the training data set name, target variable name, number of target categories, and most importantly, the number of model parameters.

Model Information	
Training Data Set	EMWS2.IMPT_TRAIN.VIEW
DMDB Catalog	WORK.REG_DMDB
Target Variable	TargetB (Target Gift Flag)
Target Measurement Level	Ordinal
Number of Target Categories	2
Error	MBernoulli
Link Function	Logit
Number of Model Parameters	86
Number of Observations	4843

Based on the introductory material about logistic regression that is presented above, you might expect to have a number of model parameters equal to the number of input variables. This ignores the fact that a single nominal input (for example, **DemCluster**) can generate scores of model parameters.

Next, consider maximum likelihood procedure, overall model fit, and the Type 3 Analysis of Effects.

The Type 3 Analysis tests the statistical significance of adding the indicated input to a model that already contains other listed inputs. A value near 0 in the Pr > ChiSq column approximately indicates a significant input; a value near 1 indicates an extraneous input.

Type 3 Analysis of Effects			
Effect	DF	Wald Chi-Square	Pr > ChiSq
DemCluster	53	58.9098	0.2682
DemGender	2	0.5088	0.7754
DemHomeOwner	1	0.1630	0.6864
DemMedHomeValue	1	2.4464	0.1178
DemPctVeterans	1	5.2502	0.0219
GiftAvg36	1	1.6709	0.1961
GiftAvgAll	1	0.0339	0.8540
GiftAvgLast	1	0.0026	0.9593
GiftCnt36	1	1.2230	0.2688
GiftCntAll	1	0.1308	0.7176
GiftCntCard36	1	1.0244	0.3115
GiftCntCardAll	1	0.0061	0.9380
GiftTimeFirst	1	1.6064	0.2050
GiftTimeLast	1	21.5351	<.0001
IMP_DemAge	1	0.0701	0.7911
IMP_GiftAvgCard36	1	0.0476	0.8273
IMP_REP_DemMedIncome	1	0.1408	0.7074
M_DemAge	1	3.0616	0.0802
M_GiftAvgCard36	1	0.9190	0.3377
M_REP_DemMedIncome	1	0.6228	0.4300
PromCnt12	1	3.2335	0.0721
PromCnt36	1	1.0866	0.2972
PromCntAll	1	1.9715	0.1603
PromCntCard12	1	0.0294	0.8639
PromCntCard36	1	0.0049	0.9441
PromCntCardAll	1	2.9149	0.0878
StatusCat96NK	5	11.3434	0.0450
StatusCatStarAll	1	1.7487	0.1860

The statistical significance measures a range from <0.0001 (highly significant) to 0.9593 (highly dubious). Results such as this suggest that certain inputs can be dropped without affecting the predictive prowess of the model.

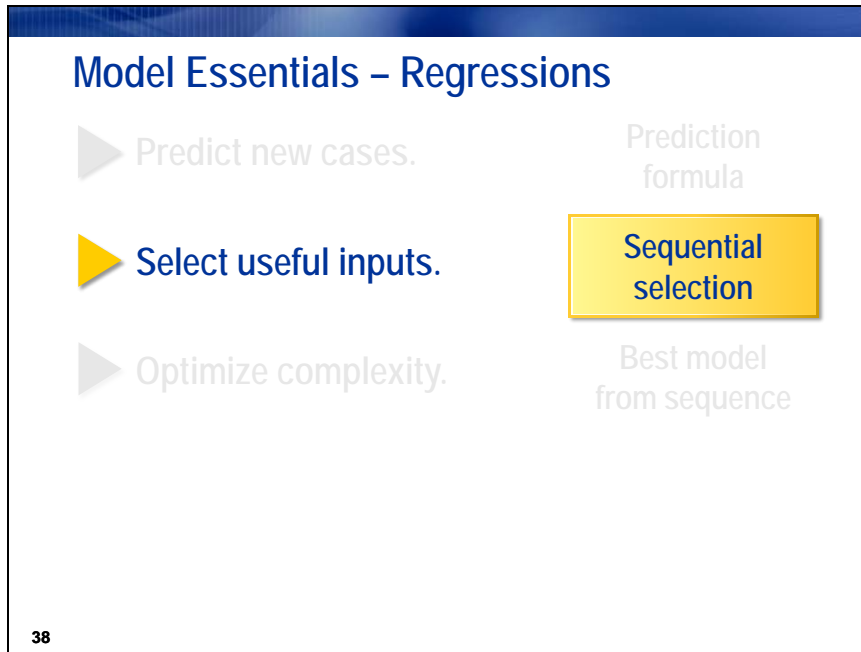
6. Restore the Output window to its original size by double-clicking its title bar. Maximize the Fit Statistics window.

Fit Statistics					
Target	Fit Statistics	Statistics Label	Train	Validation	Test
TargetB	_AIC_	Akaike's Inf...	6633.2	.	.
TargetB	_ASE_	Average Sq...	0.237268	0.24381	.
TargetB	_AVERR_	Average Err...	0.667066	0.680861	.
TargetB	_DFE_	Degrees of ...	4757	.	.
TargetB	_DFM_	Model Degr...	86	.	.
TargetB	_DFT_	Total Degre...	4843	.	.
TargetB	_DIV_	Divisor for A...	9686	9686	.
TargetB	_ERR_	Error Functi...	6461.2	6594.821	.
TargetB	_FPE_	Final Predic...	0.245847	.	.
TargetB	_MAX_	Maximum A...	0.941246	0.841531	.
TargetB	_MSE_	Mean Squa...	0.241558	0.24381	.
TargetB	_NOBS_	Sum of Fre...	4843	4843	.
TargetB	_NW_	Number of ...	86	.	.
TargetB	_RASE_	Root Avera...	0.487102	0.493771	.
TargetB	_RFPE_	Root Final ...	0.49583	.	.
TargetB	_RMSE_	Root Mean ...	0.491485	0.493771	.
TargetB	_SBC_	Schwarz's ...	7190.935	.	.
TargetB	_SSE_	Sum of Squ...	2298.179	2361.545	.
TargetB	_SUMW_	Sum of Cas...	9686	9686	.
TargetB	_MISC_	Misclassific...	0.411522	0.431964	.

If the decision predictions are of interest, model fit can be judged by misclassification. If estimate predictions are the focus, model fit can be assessed by average squared error. There appears to be some discrepancy between the values of these two statistics in the train and validation data. This indicates a possible overfit of the model. It can be mitigated by using an input selection procedure.

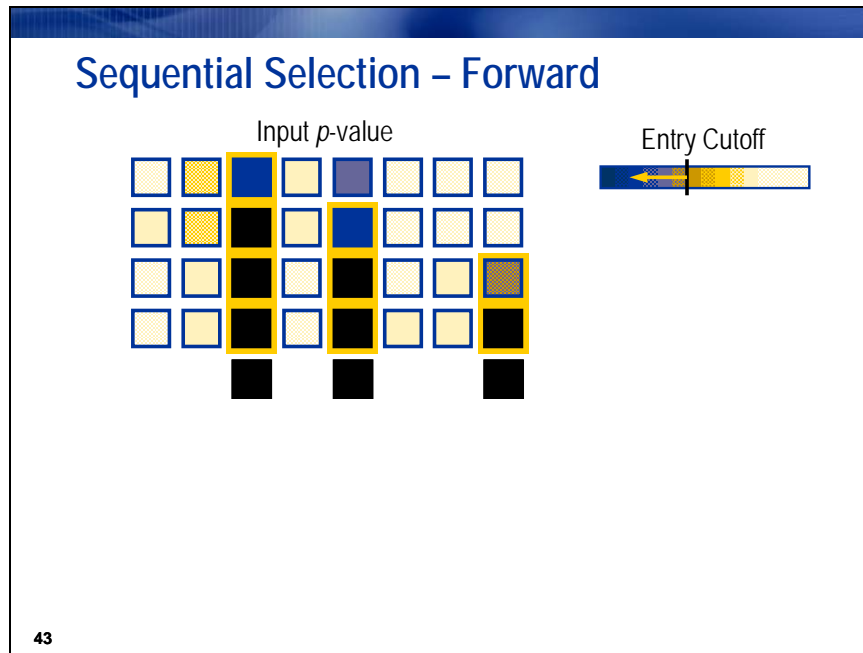
7. Close the Results window.

4.2 Selecting Regression Inputs



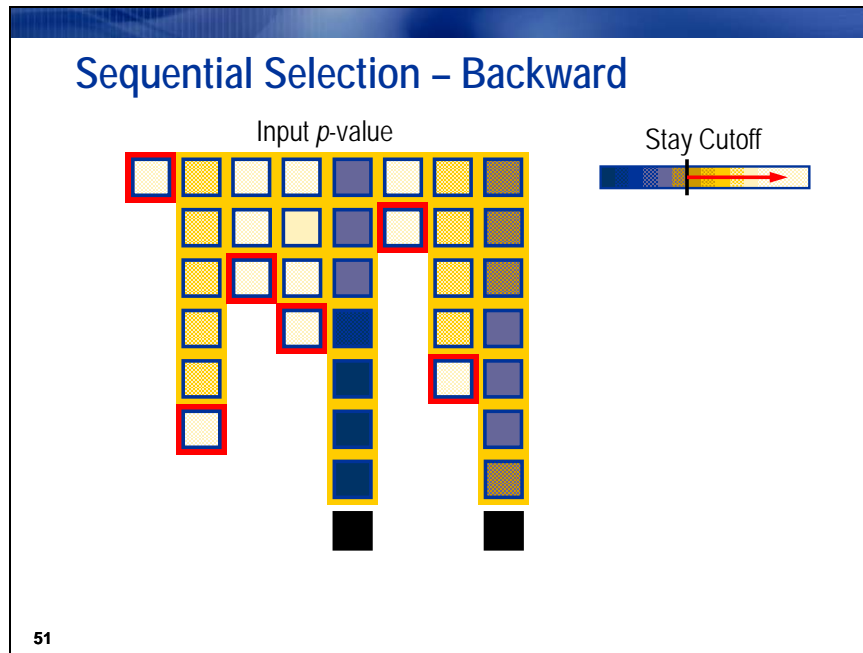
The second task that all predictive models should perform is input selection. One way to find the optimal set of inputs for a regression is simply to try every combination. Unfortunately, the number of models to consider using this approach increases exponentially in the number of available inputs. Such an exhaustive search is impractical for realistic prediction problems.

An alternative to the exhaustive search is to restrict the search to a sequence of improving models. While this might not find the single best model, it is commonly used to find models with good predictive performance. The Regression node in SAS Enterprise Miner provides three sequential selection methods.

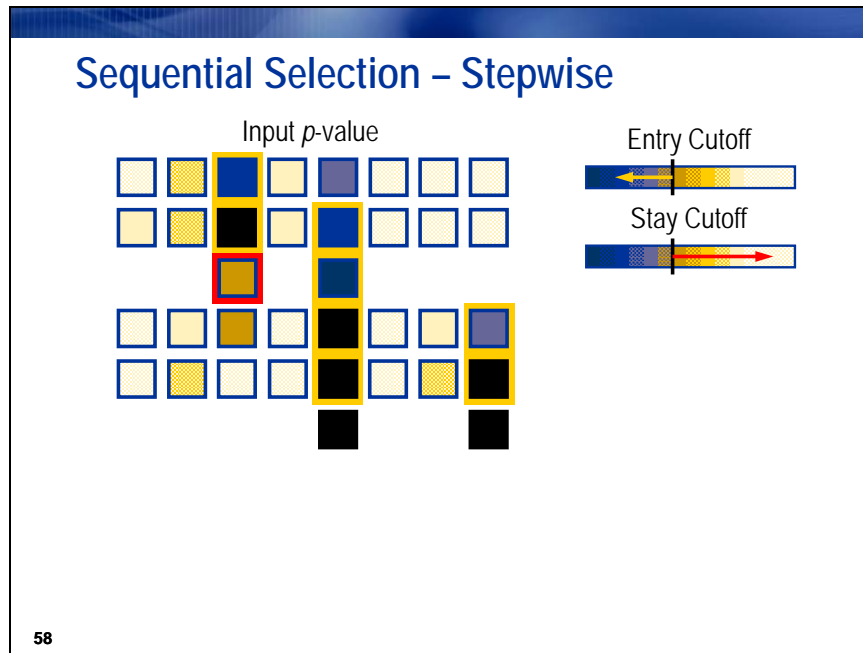


Forward selection creates a sequence of models of increasing complexity. The sequence starts with the baseline model, a model predicting the overall average target value for all cases. The algorithm searches the set of one-input models and selects the model that most improves upon the baseline model. It then searches the set of two-input models that contain the input selected in the previous step and selects the model showing the most significant improvement. By adding a new input to those selected in the previous step, a nested sequence of increasingly complex models is generated. The sequence terminates when no significant improvement can be made.

Improvement is quantified by the usual statistical measure of significance, the p -value. Adding terms in this nested fashion always increases a model's overall fit statistic. By calculating the change in the fit statistic and assuming that the change conforms to a chi-squared distribution, a significance probability, or p -value, can be calculated. A large fit statistic change (corresponding to a large chi-squared value) is unlikely due to chance. Therefore, a small p -value indicates a significant improvement. When no p -value is below a predetermined entry cutoff, the forward selection procedure terminates.



In contrast to forward selection, *backward selection* creates a sequence of models of **decreasing** complexity. The sequence starts with a saturated model, which is a model that contains all available inputs, and therefore, has the highest possible fit statistic. Inputs are sequentially removed from the model. At each step, the input chosen for removal least reduces the overall model fit statistic. This is equivalent to removing the input with the highest p -value. The sequence terminates when all remaining inputs have a p -value that is less than the predetermined stay cutoff.



Stepwise selection combines elements from both the forward and backward selection procedures. The method begins in the same way as the forward procedure, sequentially adding inputs with the smallest p -value below the entry cutoff. After each input is added, however, the algorithm reevaluates the statistical significance of all included inputs. If the p -value of any of the included inputs exceeds the stay cutoff, the input is removed from the model and reentered into the pool of inputs that are available for inclusion in a subsequent step. The process terminates when all inputs available for inclusion in the model have p -values in excess of the entry cutoff and all inputs already included in the model have p -values below the stay cutoff.



Selecting Inputs

Implementing a sequential selection method in the Regression node requires a minor change to the Regression node settings.

1. Select **Selection Model** ⇒ **Stepwise** on the Regression node property sheet.

Train	
Variables	
Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	No
Polynomial Degree	2
User Terms	No
Term Editor	
Class Targets	
Regression Type	Logistic Regression
Link Function	Logit
Model Options	
Suppress Intercept	No
Input Coding	Deviation
Model Selection	
Selection Model	Stepwise
Selection Criterion	Default
Use Selection Defaults	Yes
Selection Options	

The Regression node is now configured to use stepwise selection to choose inputs for the model.

2. Run the Regression node and view the results.
3. Maximize the Output window.
4. Hold down the CTRL key and type **G**. The Go To Line window opens.

Go To Line	
Enter line number:	OK
79	Cancel

5. Type **79** in the Enter line number field and select **OK**.

The stepwise procedure starts with Step 0, an intercept-only regression model. The value of the intercept parameter is chosen so that the model predicts the overall target mean for every case. The parameter estimate and the training data target measurements are combined in an objective function. The objective function is determined by the model form and the error distribution of the target. The value of the objective function for the intercept-only model is compared to the values obtained in subsequent steps for more complex models. A large decrease in the objective function for the more complex model indicates a significantly better model.

Step 0: Intercept entered.							
The DMREG Procedure							
Newton-Raphson Ridge Optimization							
Without Parameter Scaling							
Parameter Estimates							
1							
Optimization Start							
Active Constraints	0	Objective Function	3356.9116922				
Max Abs Gradient Element	5.707879E-12						
Optimization Results							
Iterations	0	Function Calls	3				
Hessian Calls	1	Active Constraints	0				
Objective Function	3356.9116922	Max Abs Gradient Element	5.707879E-12				
Ridge	0	Actual Over Pred Change	0				
Convergence criterion (ABSGCONV=0.00001) satisfied.							
Likelihood Ratio Test for Global Null Hypothesis: BETA=0							
-2 Log Likelihood Likelihood							
Intercept	Intercept &	Ratio					
Only	Covariates	Chi-Square	DF	Pr > ChiSq			
6713.823	6713.823	0.0000	0	.			
Analysis of Maximum Likelihood Estimates							
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept	1	-0.00041	0.0287	0.00	0.9885		1.000

Step 1 adds one input to the intercept-only model. The input and corresponding parameter are chosen to produce the largest decrease in the objective function. To estimate the values of the model parameters, the modeling algorithm makes an initial guess for their values. The initial guess is combined with the training data measurements in the objective function. Based on statistical theory, the objective function is assumed to take its minimum value at the correct estimate for the parameters. The algorithm decides whether changing the values of the initial parameter estimates can decrease the value of the objective function. If so, the parameter estimates are changed to decrease the value of the objective function and the process iterates. The algorithm continues iterating until changes in the parameter estimates fail to substantially decrease the value of the objective function.

Step 1: Effect GiftCnt36 entered.									
The DMREG Procedure									
Newton-Raphson Ridge Optimization									
Without Parameter Scaling									
Parameter Estimates								2	
Optimization Start									
Active Constraints				0	Objective Function		3356.9116922		
Max Abs Gradient Element				89.678463762					
									Ratio
									Between
									Actual
									and
									Predicted
Iter	Restarts	Function Calls	Active Constraints	Objective Function	Objective Function Change	Max Abs Gradient Element	Ridge	Change	
1	0	2	0	3316	41.4036	2.1746	0	1.014	
2	0	3	0	3315	0.0345	0.00690	0	1.002	
3	0	4	0	3315	2.278E-7	4.833E-8	0	1.000	
Optimization Results									
Iterations				3	Function Calls		6		
Hessian Calls				4	Active Constraints		0		
Objective Function				3315.473573	Max Abs Gradient Element		4.833086E-8		
Ridge				0	Actual Over Pred Change		0.999858035		
Convergence criterion (GCONV=1E-6) satisfied.									

The output next compares the model fit in Step 1 with the model fit in Step 0. The objective functions of both models are multiplied by 2 and differenced. The difference is assumed to have a chi-squared distribution with one degree of freedom. The hypothesis that the two models are identical is tested. A large value for the chi-squared statistic makes this hypothesis unlikely.

The hypothesis test is summarized in the next lines.

Likelihood Ratio Test for Global Null Hypothesis: BETA=0					
-2 Log Likelihood Intercept Only	-2 Log Likelihood Intercept & Covariates	Likelihood Ratio Chi-Square	DF	Pr > ChiSq	
6713.823	6630.947	82.8762	1	<.0001	

The output summarizes an analysis of the statistical significance of individual model effects. For the one input model, this is similar to the global significance test above.

Type 3 Analysis of Effects			
Effect	DF	Wald Chi-Square	Pr > ChiSq
GiftCnt36	1	79.4757	<.0001

Finally, an analysis of individual parameter estimates is made. (The standardized estimates and the odds ratios merit special attention and are discussed in the next section of this chapter.)

Analysis of Maximum Likelihood Estimates							
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept	1	-0.3956	0.0526	56.53	<.0001		0.673
GiftCnt36	1	0.1250	0.0140	79.48	<.0001	0.1474	1.133
Odds Ratio Estimates							
Effect	Point Estimate						
GiftCnt36	1.133						

The standardized estimates present the effect of the input on the log-odds of donation. The values are standardized to be independent of the input's unit of measure. This provides a means of ranking the importance of inputs in the model.

The odds ratio estimates indicate by what factor the odds of donation increase for each unit change in the associated input. Combined with knowledge of the range of the input, this provides an excellent way to judge the practical (as opposed to the statistical) importance of an input in the model.

The stepwise selection process continues for eight steps. After the eighth step, neither adding nor removing inputs from the model significantly changes the model fit statistic. At this point the Output window provides a summary of the stepwise procedure.

- Go to line 850 to view the stepwise summary.

The summary shows the step in which each input was added and the statistical significance of each input in the final eight-input model.

Summary of Stepwise Selection						
Step	Effect Entered	DF	Number In	Score Chi-Square	Wald Chi-Square	Pr > ChiSq
1	GiftCnt36	1	1	81.6807		<.0001
2	GiftTimeLast	1	2	23.2884		<.0001
3	DemMedHomeValue	1	3	16.9872		<.0001
4	GiftAvgAll	1	4	14.8514		0.0001
5	StatusCat96NK	5	5	18.2293		0.0027
6	DemPctVeterans	1	6	7.4187		0.0065
7	M_GiftAvgCard36	1	7	7.1729		0.0074
8	M_DemAge	1	8	4.6501		0.0311

NOTE: No (additional) effects met the 0.05 significance level for entry into the model.

The default selection criterion selects the model from Step 8 as the model with optimal complexity. As the next section shows, this might not be the optimal model, based on the fit statistic that is appropriate for your analysis objective.

The selected model, based on the CHOOSE=NONE criterion, is the model trained in Step 8. It consists of the following effects:

Intercept DemMedHomeValue DemPctVeterans GiftAvgAll GiftCnt36 GiftTimeLast M_DemAge M_GiftAvgCard36 StatusCat96NK

For convenience, the output from Step 8 is repeated. An excerpt from the analysis of individual parameter estimates is shown below.

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept	1	0.2727	0.2024	1.82	0.1779		1.314
DemMedHomeValue	1	1.385E-6	3.009E-7	21.18	<.0001	0.0763	1.000
DemPctVeterans	1	0.00658	0.00261	6.38	0.0115	0.0412	1.007
GiftAvgAll	1	-0.0136	0.00444	9.33	0.0023	-0.0608	0.987
GiftCnt36	1	0.0587	0.0187	9.79	0.0018	0.0692	1.060
GiftTimeLast	1	-0.0376	0.00770	23.90	<.0001	-0.0837	0.963
M_DemAge	0 1	0.0741	0.0344	4.65	0.0311		1.077
M_GiftAvgCard36	0 1	0.1112	0.0411	7.30	0.0069		1.118
StatusCat96NK	A 1	-0.0880	0.0927	0.90	0.3423		0.916
StatusCat96NK	E 1	0.4974	0.1818	7.48	0.0062		1.644
StatusCat96NK	F 1	-0.4570	0.1303	12.30	0.0005		0.633
StatusCat96NK	L 1	0.1456	0.3735	0.15	0.6966		1.157
StatusCat96NK	N 1	-0.1206	0.1323	0.83	0.3621		0.886

The parameter with the largest standardized estimate (in absolute value) is **GiftTimeLast**.

7. Restore the Output window and maximize the Fit Statistics window.

Target	Fit Statistics	Statistics Label	Train	Validation	Test
TargetB	_AIC_	Akaike's Inf...	6563.093		
TargetB	_ASE_	Average Sq...	0.240919	0.242336	
TargetB	_AVERR_	Average Err...	0.674901	0.678517	
TargetB	_DFE_	Degrees of ...	4830		
TargetB	_DFM_	Model Degr...	13		
TargetB	_DFT_	Total Degr...	4843		
TargetB	_DIV_	Divisor for A...	9686	9686	
TargetB	_ERR_	Error Functi...	6537.093	6572.12	
TargetB	_FPE_	Final Predic...	0.242216		
TargetB	_MAX_	Maximum A...	0.965413	0.998582	
TargetB	_MSE_	Mean Squa...	0.241567	0.242336	
TargetB	_NOBS_	Sum of Fre...	4843	4843	
TargetB	_NW_	Number of ...	13		
TargetB	_RASE_	Root Avera...	0.490835	0.492277	
TargetB	_RFPE_	Root Final ...	0.492154		
TargetB	_RMSE_	Root Mean ...	0.491495	0.492277	
TargetB	_SBC_	Schwarz's ...	6647.402		
TargetB	_SSE_	Sum of Squ...	2333.54	2347.27	
TargetB	_SUMWV_	Sum of Cas...	9686	9686	
TargetB	_MISC_	Misclassific...	0.421433	0.42453	

The simpler model improves on both the validation misclassification and average squared error measures of model performance.

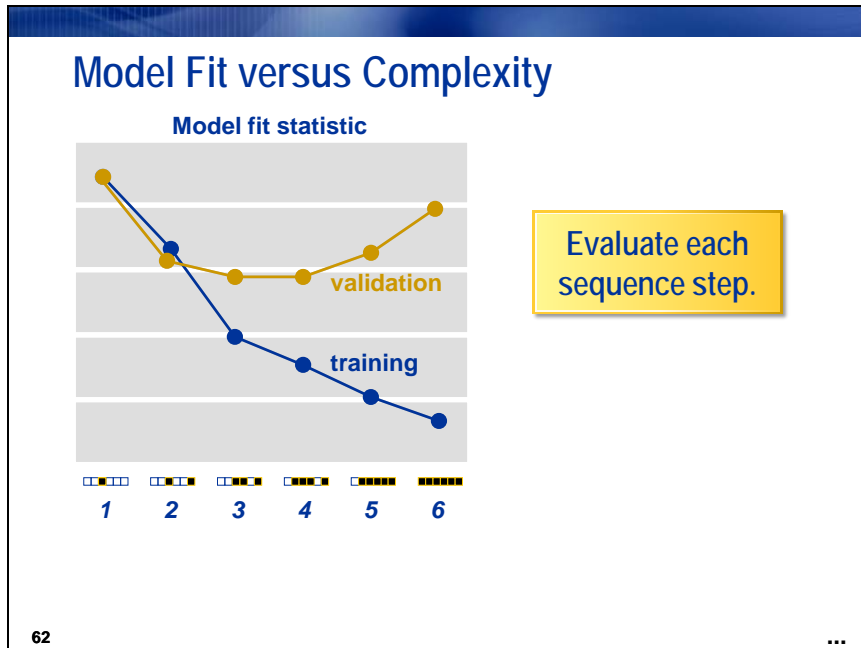
4.3 Optimizing Regression Complexity

Model Essentials – Regressions

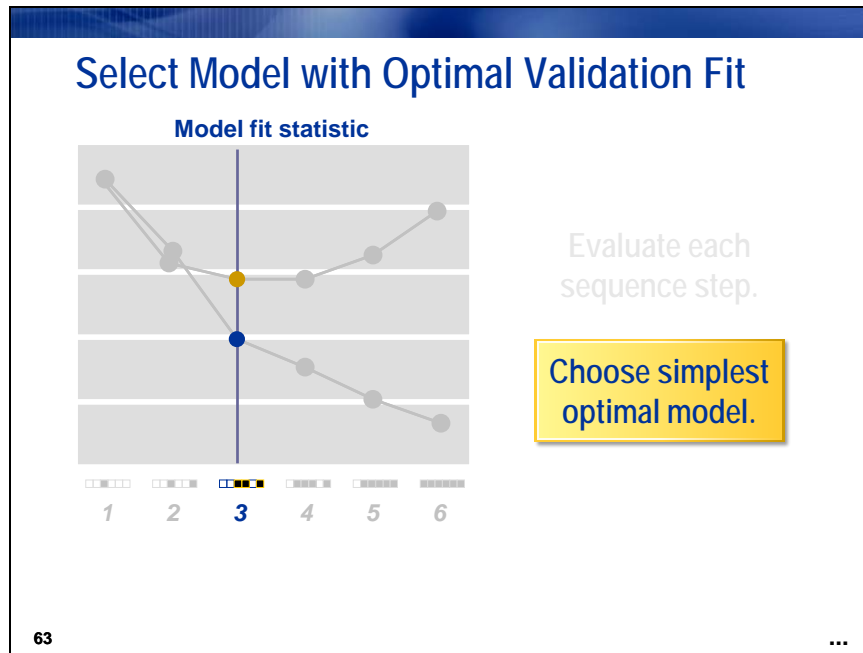
- Predict new cases. Prediction formula
- Select useful inputs. Sequential selection
- Optimize complexity.** **Best model from sequence**

61 ...

Regression complexity is optimized by choosing the optimal model in the sequential selection sequence.



The process involves two steps. First, fit statistics are calculated for the models generated in each step of the selection process. Both the training and validation data sets are used.



Then, as with the decision tree in Chapter 3, the simplest model (that is, the one with the fewest inputs) with the optimal fit statistic is selected.



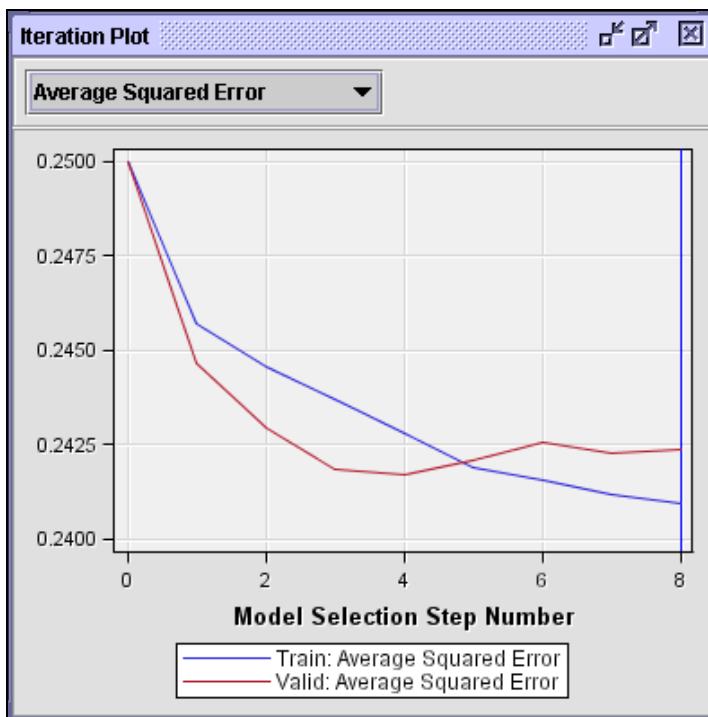
Optimizing Complexity

Iteration Plot

The following steps illustrate the use of the iteration plot in the Regression tool Results window.

In the same manner as the decision tree, you can tune a regression model to give optimal performance on the validation data. The basic idea involves calculating a fit statistic for each step in the input selection procedure and selecting the step (and corresponding model) with the optimal fit statistic value. To avoid bias, of course, the fit statistic should be calculated on the validation data set.

1. Select **View** ⇒ **Model** ⇒ **Iteration Plot**. The Iteration Plot window opens.

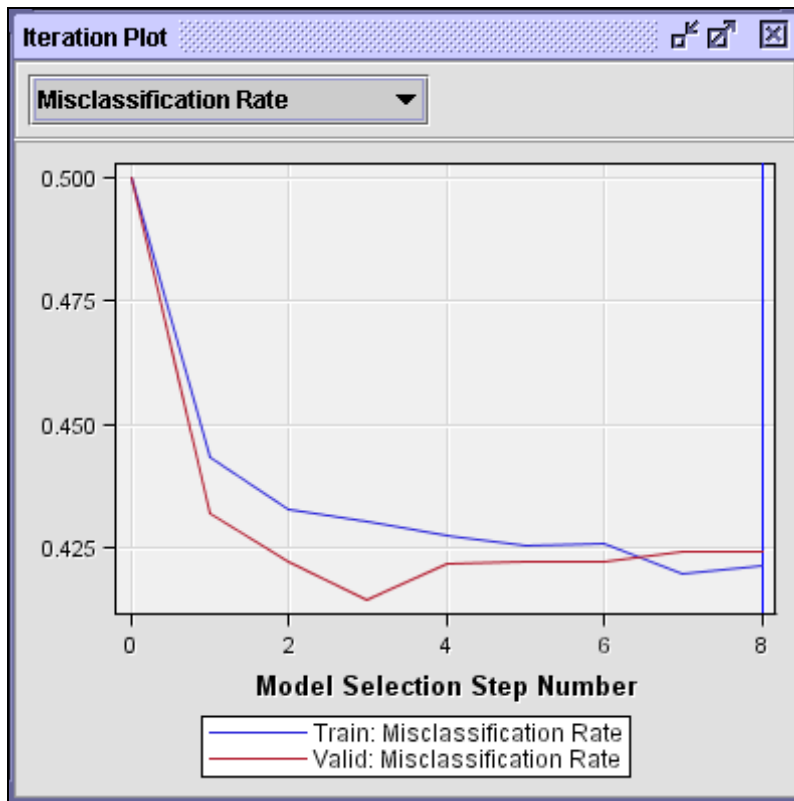


The Iteration Plot window shows (by default) average squared error (training and validation) from the model selected in each step of the stepwise selection process.



Surprisingly, this plot contradicts the naïve assumption that a model fit statistic calculated on training data will **always** be better than the same statistic calculated on validation data. This concept, called the *optimism principle*, is correct only *on the average*, and usually manifests itself only when overly complex (overly flexible) models are considered. It is not uncommon for training and validation fit statistic plots to cross (possibly several times). These crossings illustrate unquantified variability in the fit statistics.

Apparently, the smallest average squared error occurs in Step 4, rather than in the final model, Step 8. If your analysis objective requires estimates as predictions, the model from Step 4 should provide slightly less biased ones.

2. Select **Select Chart** ⇒ **Misclassification Rate**.






The iteration plot shows that the model with the smallest misclassification rate occurs in Step 3. If your analysis objective requires decision predictions, the predictions from the Step 3 model are as accurate as the predictions from the final Step 8 model.


The selection process stopped at Step 8 to limit the amount of time spent running the stepwise selection procedure. In Step 8, no more inputs had a chi-squared p -value below 0.05. The value 0.05 is a somewhat arbitrary holdover from the days of statistical tables. With the validation data available to gauge overfitting, it is possible to eliminate this restriction and obtain a richer pool of models to consider.

Full Model Selection

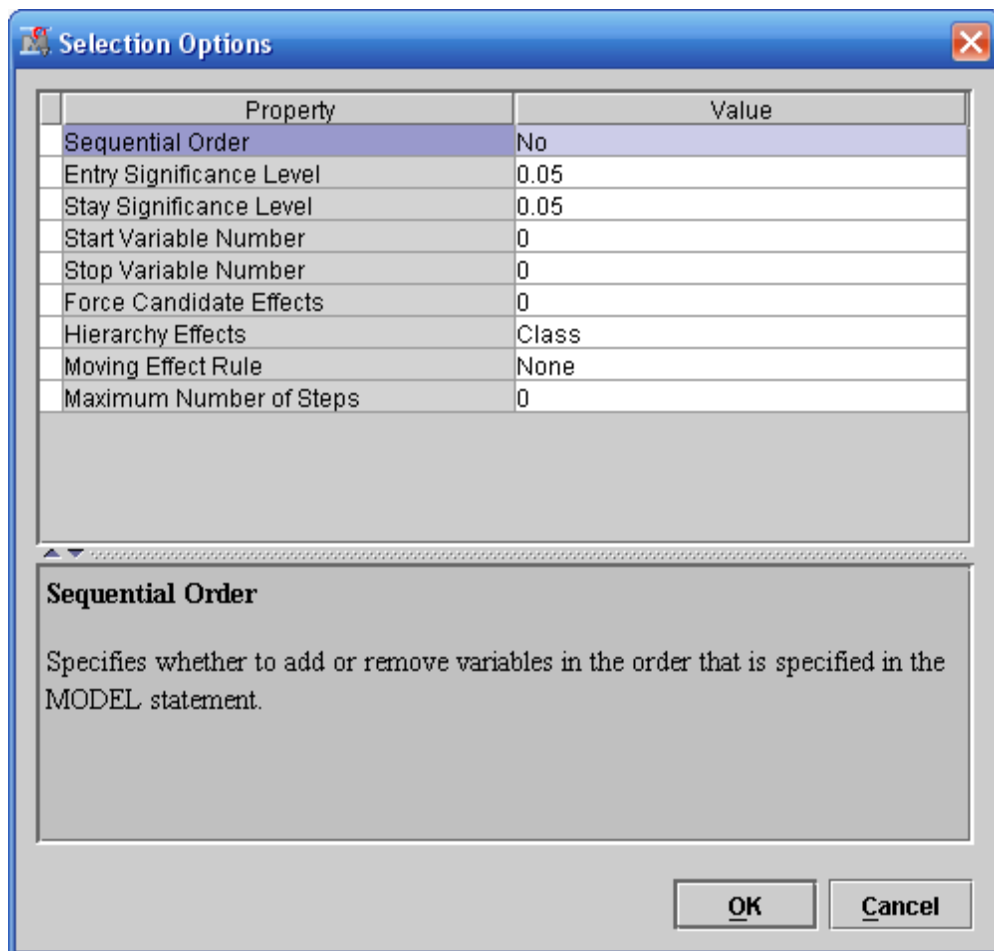
Use the following steps to build and evaluate a larger sequence of regression models:

1. Close the Results - Regression window.
2. Select **Use Selection Default** ⇒ **No** from the Regression node Properties panel.

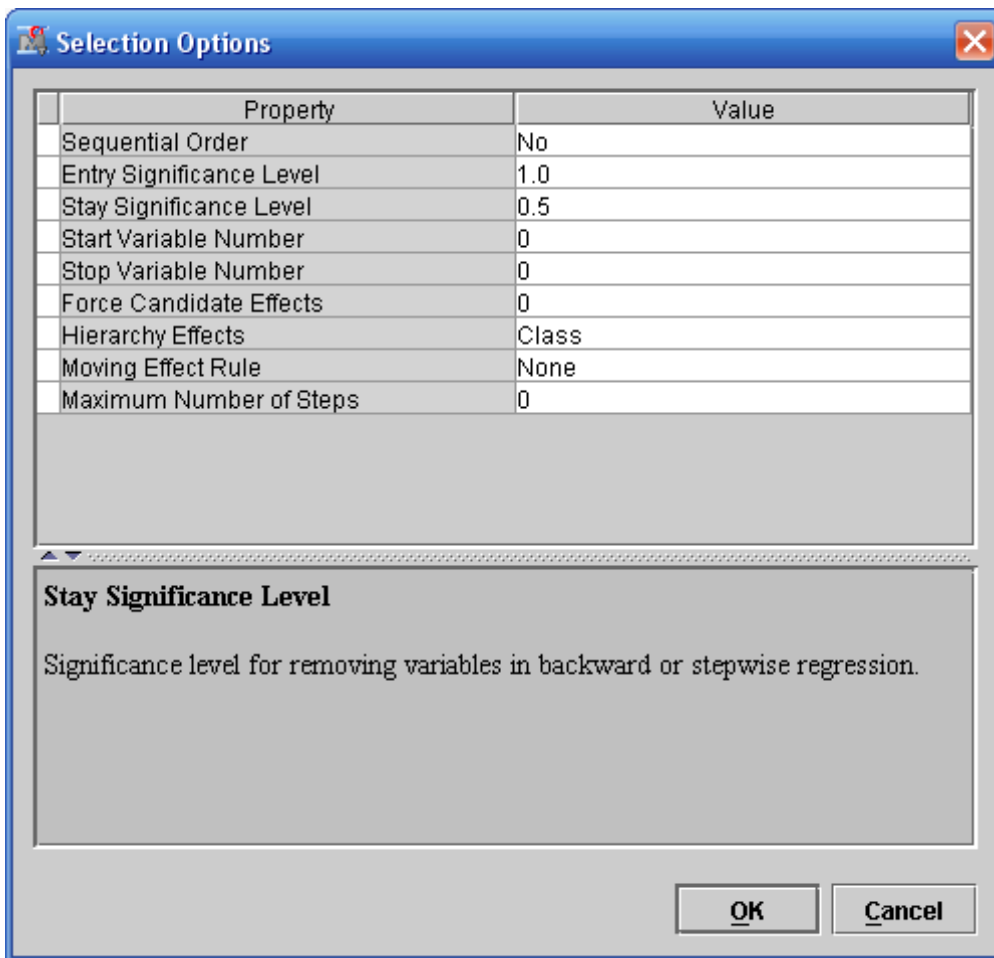
Train	
Variables	
<input checked="" type="checkbox"/> Equation	
Main Effects	Yes
Two-Factor Interaction	No
Polynomial Terms	No
Polynomial Degree	2
User Terms	No
Term Editor	
<input checked="" type="checkbox"/> Class Targets	
Regression Type	Logistic Regression
Link Function	Logit
<input checked="" type="checkbox"/> Model Options	
Suppress Intercept	No
Input Coding	Deviation
<input checked="" type="checkbox"/> Model Selection	
Selection Model	Stepwise
Selection Criterion	Default 
Use Selection Default	No 
Selection Options	

3. Select **Selection Options** ⇒ .

The Selection Options window opens.



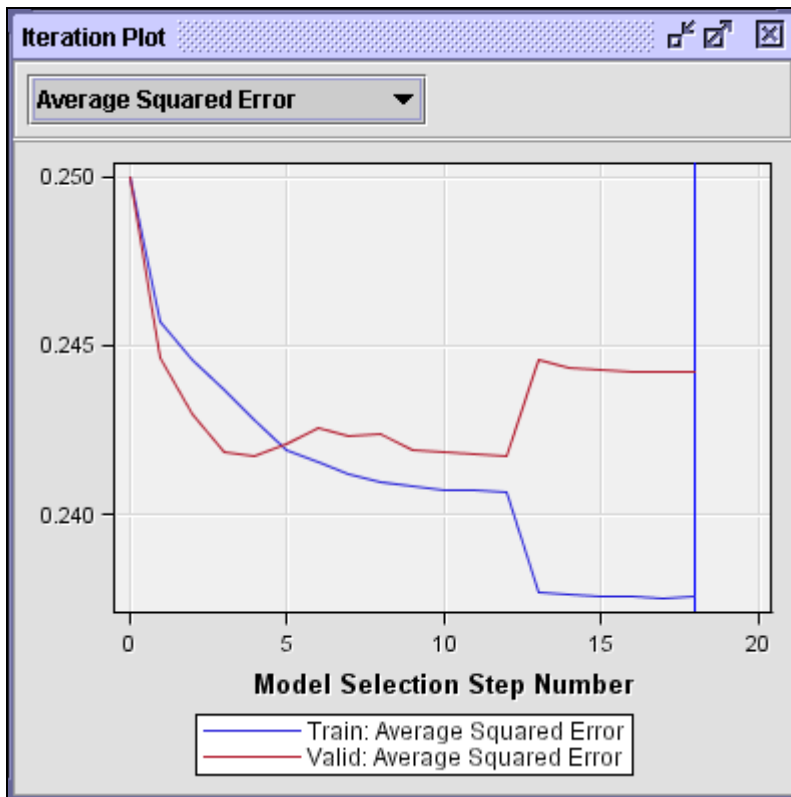
4. Type **1.0** as the Entry Significance Level value.
5. Type **0.5** as the Stay Significance Level value.



The Entry Significance value enables any input in the model. (The one chosen will have the smallest p -value.) The Stay Significance value keeps any input in the model with a p -value less than 0.5. This second choice is somewhat arbitrary. A smaller value can terminate the stepwise selection process earlier, while a larger value can maintain it longer. A Stay Significance of 1.0 forces stepwise to behave in the manner of a forward selection.

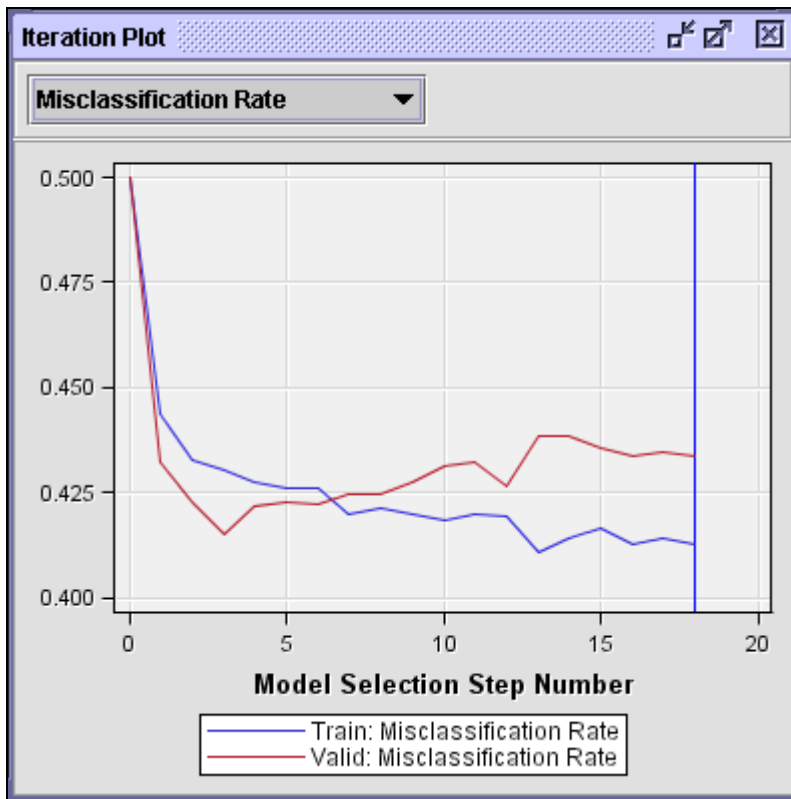
6. Run the Regression node and view the results.

7. Select **View** ⇒ **Model** ⇒ **Iteration Plot**. The Iteration Plot window opens.



The iteration plot shows the smallest average squared errors occurring in Steps 4 or 12. There is a significant change in average squared error in Step 13, when the **DemCluster** input is added. Inclusion of this nonnumeric input improves training performance but hurts validation performance.

8. Select **Select Chart** ⇒ **Misclassification Rate**.



The iteration plot shows that the smallest validation misclassification rates occur at Step 3. Notice that the change in the assessment statistic in Step 13 is much less pronounced.

Best Sequence Model

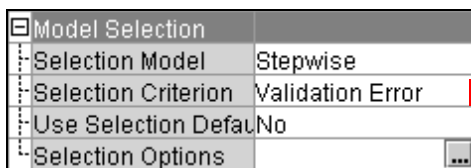
You can configure the Regression node to select the model with the smallest fit statistic (rather than the final stepwise selection iteration). This method is how SAS Enterprise Miner optimizes complexity for regression models.

1. Close the Results - Regression window.
2. If your predictions are decisions, use the following setting:

Select **Selection Criterion** ⇒ **Validation Misclassification**. (Equivalently, you can select **Validation Profit/Loss**. The equivalence is demonstrated in Chapter 6.)

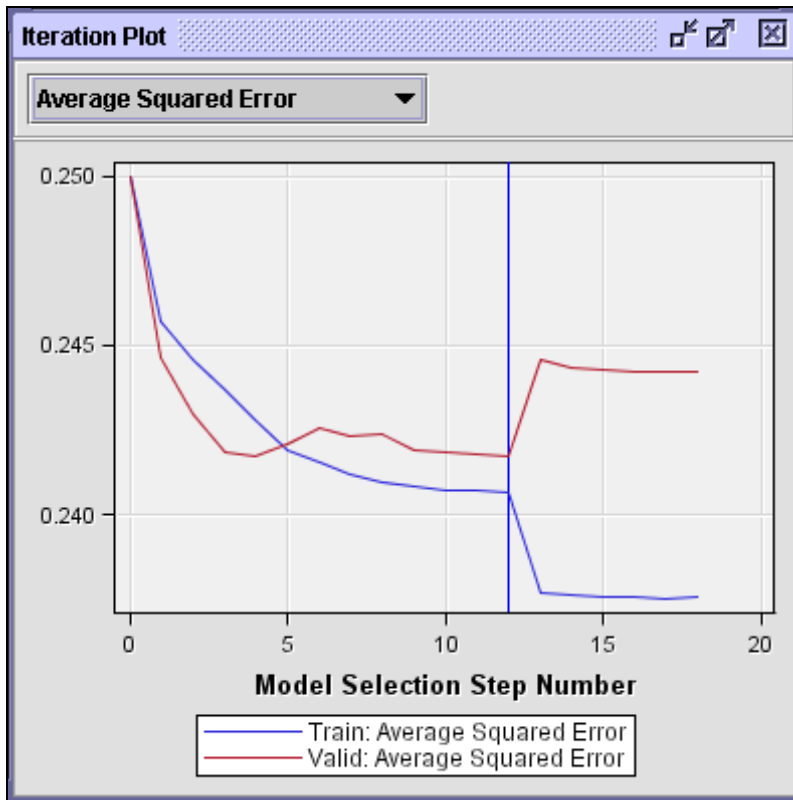
3. If your predictions are estimates (or rankings), use the following setting:

Select **Selection Criterion** ⇒ **Validation Error**.



The continuing demonstration assumes validation error selection criteria.

4. Run the **Regression** node and view the results.
5. Select **View** ⇒ **Model** ⇒ **Iteration Plot**.



The vertical blue line shows the model with the optimal validation error (Step 12).

6. Go to line 2690.

Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept	1	0.4999	0.2575	3.77	0.0522		1.649
DemMedHomeValue	1	1.416E-6	3.011E-7	22.12	<.0001	0.0781	1.000
DemPctVeterans	1	0.00651	0.00261	6.23	0.0126	0.0407	1.007
GiftAvg36	1	-0.0101	0.00355	8.02	0.0046	-0.0561	0.990
GiftCnt36	1	0.0574	0.0197	8.53	0.0035	0.0677	1.059
GiftTimeLast	1	-0.0415	0.00829	25.07	<.0001	-0.0923	0.959
M_DemAge	0	1	0.0720	0.0345	4.36	0.0367	1.075
M_GiftAvgCard36	0	1	0.1126	0.0412	7.46	0.0063	1.119
PromCntCard12	1	-0.0381	0.0281	1.85	0.1740	-0.0282	0.963
StatusCat96NK	A	1	-0.0353	0.0957	0.14	0.7122	0.965
StatusCat96NK	E	1	0.4010	0.1950	4.23	0.0398	1.493
StatusCat96NK	F	1	-0.4485	0.1314	11.66	0.0006	0.639
StatusCat96NK	L	1	0.1733	0.3743	0.21	0.6433	1.189
StatusCat96NK	N	1	-0.0988	0.1353	0.53	0.4649	0.906
StatusCatStarAll	0	1	-0.0701	0.0367	3.64	0.0563	0.932

While not all the p -values are less than 0.05, the model seems to have a better validation average squared error (and misclassification) than the model selected using the default Significance Level settings.

In short, there is nothing sacred about 0.05. It is not unreasonable to override the defaults of the Regression node to enable selection from a richer collection of potential models. On the other hand, most of the reduction in the fit statistics occurs during inclusion of the first three inputs. If you seek a parsimonious model, it is reasonable to use a smaller value for the Stay Significance parameter.

4.4 Interpreting Regression Models

Beyond the Prediction Formula

- ▶ Manage missing values.
- ▶ **Interpret the model.**
- ▶ Handle extreme or unusual values.
- ▶ Use nonnumeric inputs.
- ▶ Account for nonlinearities.

66 ...

After you build a model, you might be asked to interpret the results. Fortunately regression models lend themselves to easy interpretation.

Odds Ratios and Doubling Amounts

$$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2 \quad \text{logit scores}$$

	Δx_i	consequence	
Doubling amount: Input change is required to double odds.	1	$\Rightarrow odds \times \exp(w_i)$	Odds ratio: Amount odds change with unit change in input.
	$\frac{0.69}{w_i}$	$\Rightarrow odds \times 2$	

69

...

There are two equivalent ways to interpret a logistic regression model. Both relate changes in input measurements to changes in odds of primary outcome.

- An *odds ratio* expresses the increase in primary outcome odds associated with a unit change in an input. It is obtained by exponentiation of the parameter estimate of the input of interest.
- A *doubling amount* gives the amount of change required for doubling the primary outcome odds. It is equal to $\log(2) \approx 0.69$ divided by the parameter estimate of the input of interest.



If the predicted logit scores remain in the range -2 to +2, linear and logistic regression models of binary targets are virtually indistinguishable. Balanced stratified sampling (Chapter 6) often ensures this. Thus, the prevalence of balanced sampling in predictive modeling might, in fact, be a vestigial practice from a time when maximum likelihood estimation was computationally extravagant.



Interpreting a Regression Model

The following steps demonstrate how to interpret a model using odds ratios:

1. Go to line 2712 of the regression model output.

Odds Ratio Estimates		
Effect		Point Estimate
DemMedHomeValue		1.000
DemPctVeterans		1.007
GiftAvg36		0.990
GiftCnt36		1.059
GiftTimeLast		0.959
M_DemAge	0 vs 1	1.155
M_GiftAvgCard36	0 vs 1	1.253
PromCntCard12		0.963
StatusCat96NK	A vs S	0.957
StatusCat96NK	E vs S	1.481
StatusCat96NK	F vs S	0.633
StatusCat96NK	L vs S	1.179
StatusCat96NK	N vs S	0.898
StatusCatStarAll	0 vs 1	0.869

This output includes most of situations you will encounter when you build a regression model.

For **GiftAvg36**, the odds ratio estimate equals 0.990. This means that for each additional dollar donated (on average) in the past 36 months, the odds of donation on the 97NK campaign change by a factor of 0.99, a 1% decrease.

For **GiftCnt36**, the odds ratio estimate equals 1.059. This means that for each additional donation in the past 36 months, the odds of donation on the 97NK campaign change by a factor of 1.059, a 5.9% increase.

For **M_DemAge**, the odds ratio (0 versus 1) estimate equals 1.155. This means that cases with a 0 value for **M_DemAge** are 1.155 times more likely to donate than cases with a 1 value for **M_DemAge**.



The unusual value of 1.000 for the **DemMedHomeValue** odds ratio has a simple explanation. Unit (that is, single dollar) changes in home value do not change the odds of response by an amount captured in three significant digits. To obtain a more meaningful value for this input's effect on response odds, you can multiply the parameter estimate by 1000 and exponentiate the result. You then have the change in response odds based on 1000 dollar changes in median home value. Equivalently, you could use the Transform Variables node to replace **DemMedHomeValue** with **DemMedHmVal1000=DemMedHomeValue/1000**, and a unit increase on that new input would represent a \$1000 increase in the **DemMedHomeValue**.

2. Close the Results window.

4.5 Transforming Inputs

Beyond the Prediction Formula

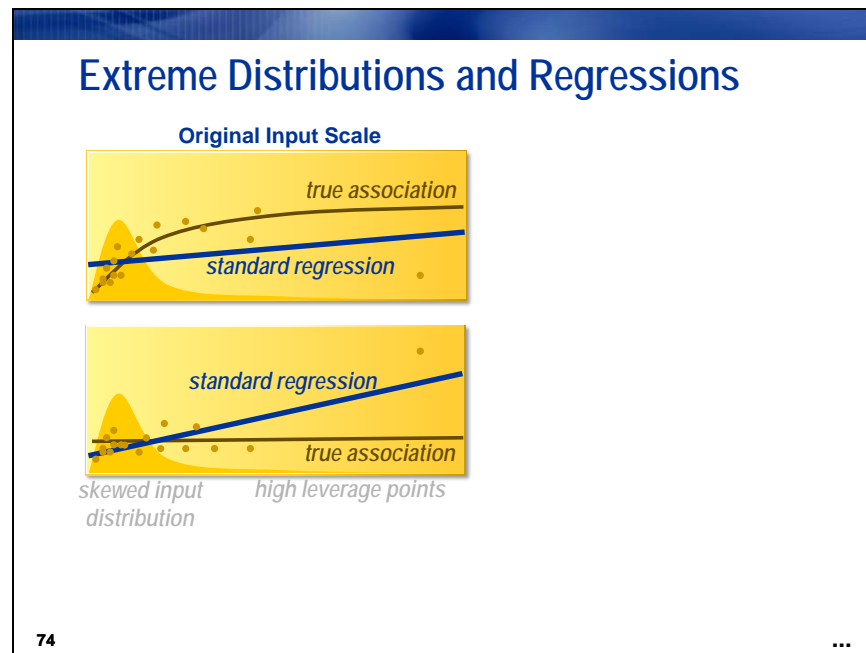
- ▶ Manage missing values.
- ▶ Interpret the model.
- ▶ **Handle extreme or unusual values.**
- ▶ Use nonnumeric inputs.
- ▶ Account for nonlinearities.

72

...

Classical regression analysis makes no assumptions about the distribution of inputs. The only assumption is that the expected value of the target (or some function thereof) is a linear combination of fixed input measurements.

Why should you worry about extreme input distributions?



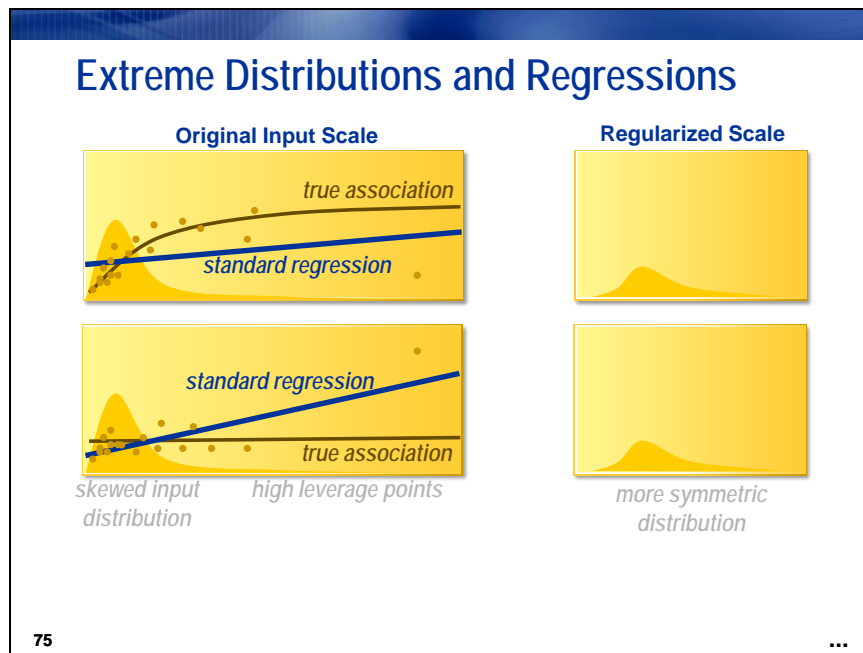
There are at least two compelling reasons.

- First, in most real-world applications, the relationship between expected target value and input value does not increase without bound. Rather, it typically tapers off to some horizontal asymptote. Standard regression models are unable to accommodate such a relationship.
- Second, as a point expands from the overall mean of a distribution, the point has more influence, or *leverage*, on model fit. Models built on inputs with extreme distributions attempt to optimize fit for the most extreme points at the cost of fit for the bulk of the data, usually near the input mean. This can result in an exaggeration or an understating of an input's association with the target.

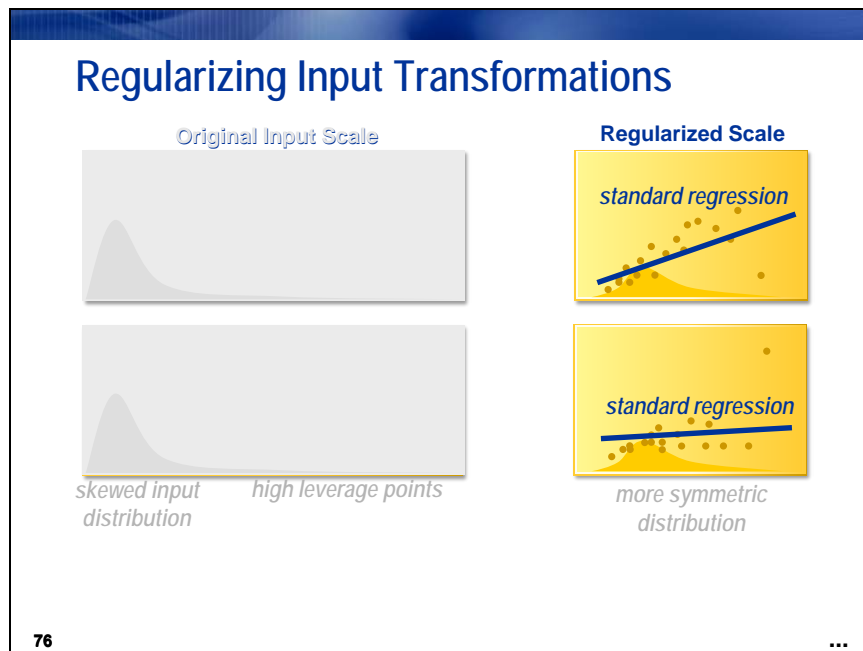
Both of these phenomena are seen in the above slide.



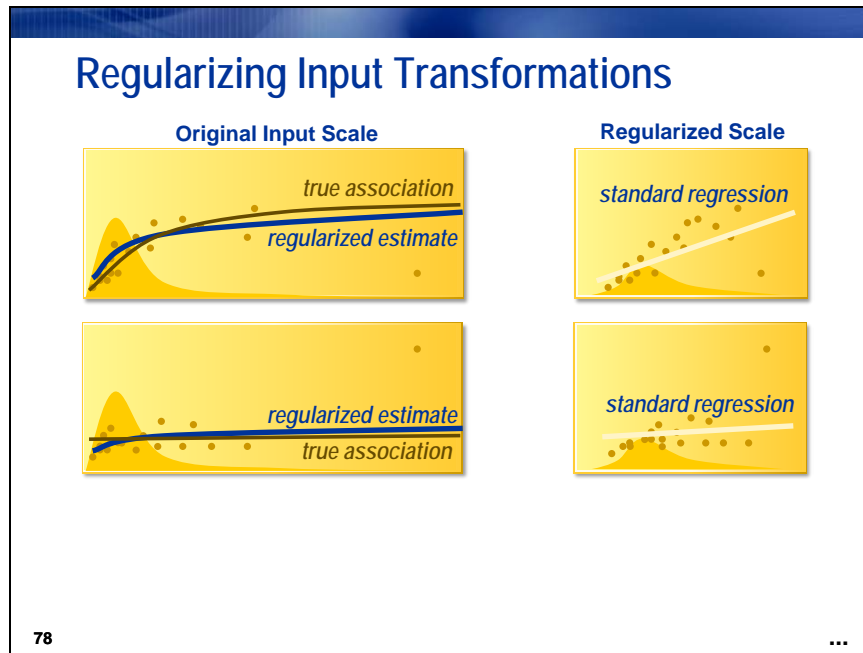
The first concern can be addressed by abandoning standard regression models for more flexible modeling methods. Abandoning standard regression models is often done at the cost of model interpretability and, more importantly, failure to address the second concern of leverage.



A simpler and, arguably, more effective approach transforms or regularizes offending inputs in order to eliminate extreme values.



Then, a standard regression model can be accurately fit using the transformed input in place of the original input.



Often this can solve both problems mentioned above. This not only mitigates the influence of extreme cases, but also creates the desired asymptotic association between input and target on the original input scale.



Transforming Inputs

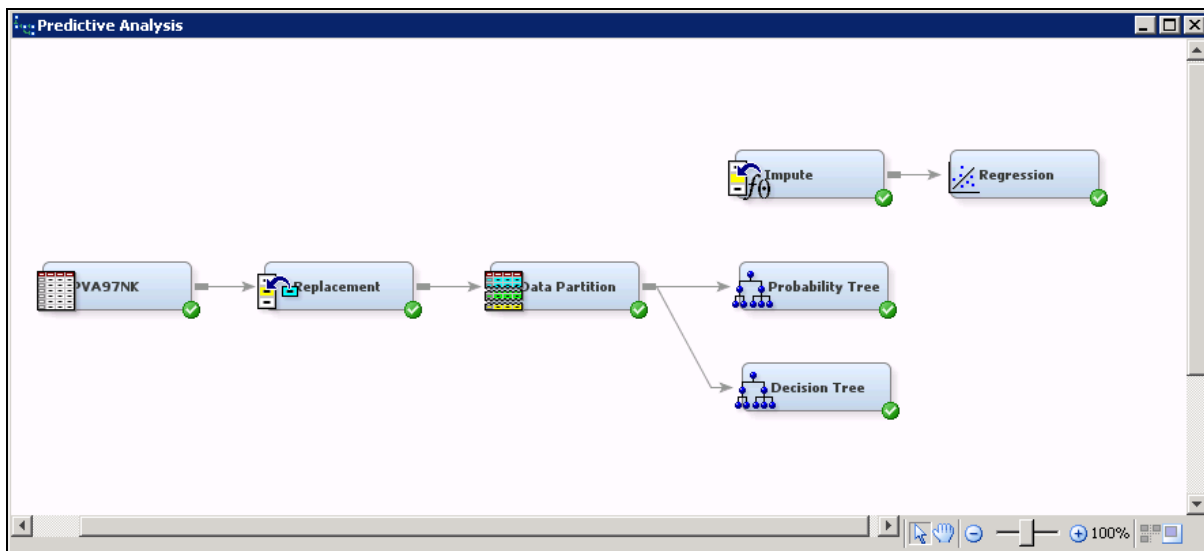
Regression models are sensitive to extreme or outlying values in the input space. Inputs with highly skewed or highly kurtotic distributions can be selected over inputs that yield better overall predictions. To avoid this problem, analysts often regularize the input distributions using a simple transformation. The benefit of this approach is improved model performance. The cost, of course, is increased difficulty in model interpretation.

The Transform Variables tool enables you to easily apply standard transformations (in addition to the specialized ones seen in Chapter 9) to a set of inputs.

The Transform Variables Tool

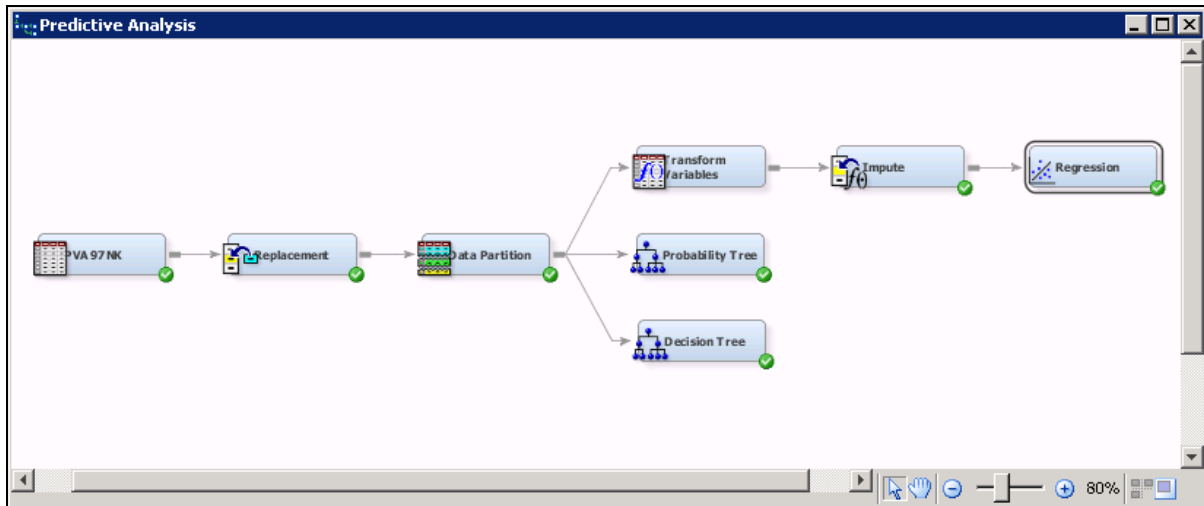
Use the following steps to transform inputs with the Transform Variables tool:

1. Remove the connection between the Data Partition node and the Impute node.











2. Select the **Modify** tab.
3. Drag a **Transform Variables** tool into the diagram workspace.
4. Connect the **Data Partition** node to the **Transform Variables** node.
5. Connect the **Transform Variables** node to the **Impute** node.

6. Adjust the diagram icons for aesthetics. (So that you can see the entire diagram, the zoom level is reduced.)



The Transform Variables node is placed before the Impute node to keep the imputed values at the average (or center of mass) of the model inputs.

7. Select the **Variables** ⇒  property of the Transform Variables node.

Property	Value
General	
Node ID	Trans
Imported Data	
Exported Data	
Notes	
Train	
Variables	 
Formulas	
Interactions	
SAS Code	
Default Methods	
Interval Inputs	None
Interval Targets	None
Class Inputs	None
Class Targets	None
Sample Properties	
Method	First N
Size	Default
Random Seed	12345

The Variables - Trans window opens.

Variables - Trans

(none) ☐ not Equal to ...

Name	Method	Number of Bins	Role	Level	Type	Order	Label
DemAge	Default	4.0	Input	Interval	Numeric		Age
DemCluster	Default	4.0	Input	Nominal	Character		Demographi
DemGender	Default	4.0	Input	Nominal	Character		Gender
DemHomeOw	Default	4.0	Input	Binary	Character		Home Owne
DemMedHom	Default	4.0	Input	Interval	Numeric		Median Hom
DemMedIncon	Default	4.0	Rejected	Interval	Numeric		Median Incon
DemPctVetera	Default	4.0	Input	Interval	Numeric		Percent Vete
GiftAvg36	Default	4.0	Input	Interval	Numeric		Gift Amount,
GiftAvgAll	Default	4.0	Input	Interval	Numeric		Gift Amount,
GiftAvgCard36	Default	4.0	Input	Interval	Numeric		Gift Amount,
GiftAvgLast	Default	4.0	Input	Interval	Numeric		Gift Amount I
GiftCnt36	Default	4.0	Input	Interval	Numeric		Gift Count 36
GiftCntAll	Default	4.0	Input	Interval	Numeric		Gift Count Al
GiftCntCard36	Default	4.0	Input	Interval	Numeric		Gift Count C:
GiftCntCardAll	Default	4.0	Input	Interval	Numeric		Gift Count C:
GiftTimeFirst	Default	4.0	Input	Interval	Numeric		Times Since
GiftTimeLast	Default	4.0	Input	Interval	Numeric		Time Since L
PromCnt12	Default	4.0	Input	Interval	Numeric		Promotion C
PromCnt36	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntAll	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard	Default	4.0	Input	Interval	Numeric		Promotion C
REP_DemMed	Default	4.0	Input	Interval	Numeric		Replaceme
StatusCat96N	Default	4.0	Input	Nominal	Character		Status Categ
StatusCatStar	Default	4.0	Input	Binary	Numeric		Status Categ

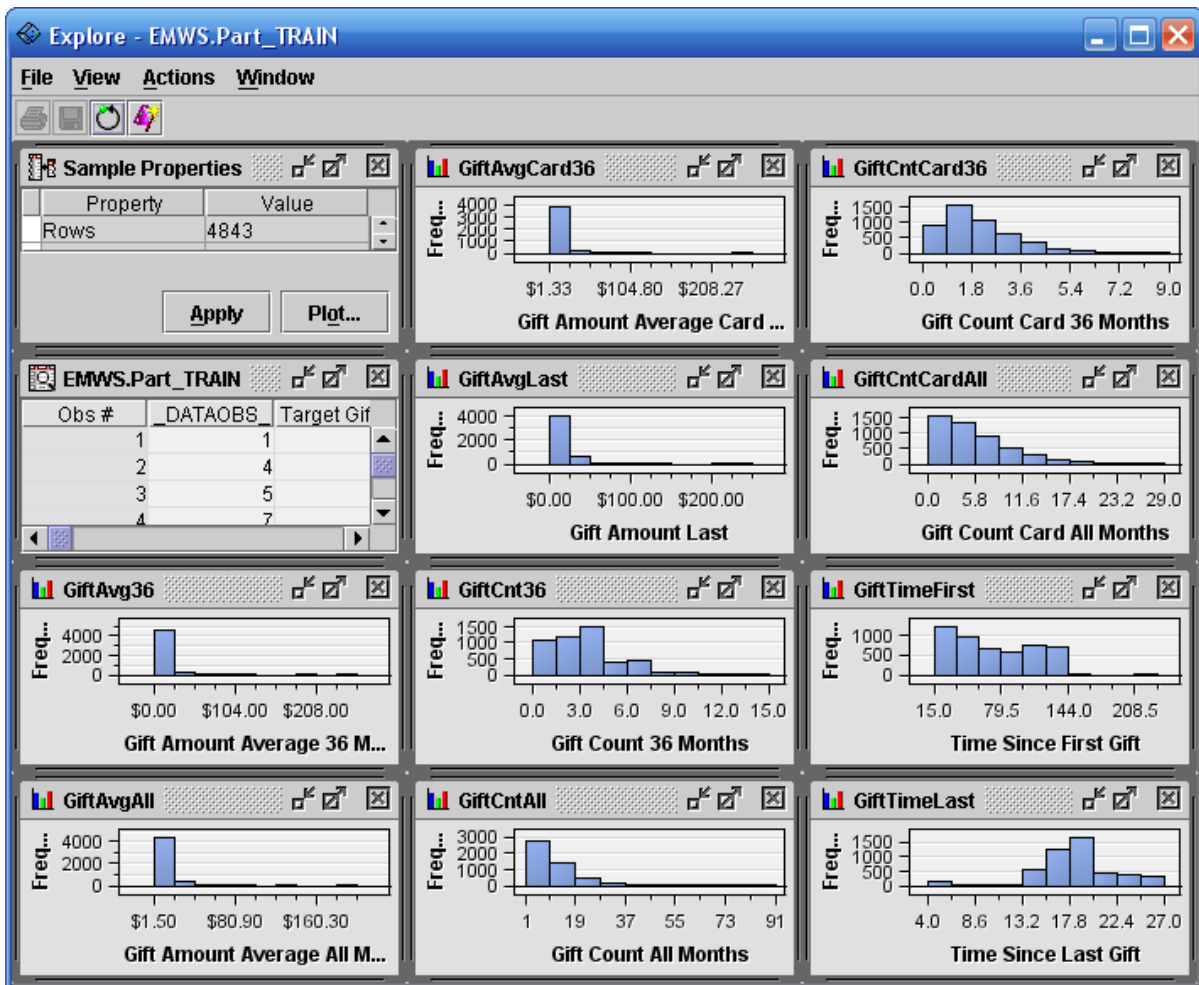
8. Select all inputs with **Gift** in the name.

Variables - Trans

(none) ☐ not Equal to

Name	Method	Number of Bins	Role	Level	Type	Order	Label
DemAge	Default	4.0	Input	Interval	Numeric		Age
DemCluster	Default	4.0	Input	Nominal	Character		Demographi
DemGender	Default	4.0	Input	Nominal	Character		Gender
DemHomeOw	Default	4.0	Input	Binary	Character		Home Owne
DemMedHom	Default	4.0	Input	Interval	Numeric		Median Hom
DemMedInco	Default	4.0	Rejected	Interval	Numeric		Median Inco
DemPctVete	Default	4.0	Input	Interval	Numeric		Percent Vete
GiftAvg36	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftAvgAll	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftAvgCard36	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftAvgLast	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftCnt36	Default	4.0	Input	Interval	Numeric		Gift Count 36
GiftCntAll	Default	4.0	Input	Interval	Numeric		Gift Count All
GiftCntCard36	Default	4.0	Input	Interval	Numeric		Gift Count C
GiftCntCardAll	Default	4.0	Input	Interval	Numeric		Gift Count C
GiftTimeFirst	Default	4.0	Input	Interval	Numeric		Times Since
GiftTimeLast	Default	4.0	Input	Interval	Numeric		Time Since
PromCnt12	Default	4.0	Input	Interval	Numeric		Promotion C
PromCnt36	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntAll	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard36	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCardAll	Default	4.0	Input	Interval	Numeric		Promotion C
REP_DemMed	Default	4.0	Input	Interval	Numeric		Replaceme
StatusCat96N	Default	4.0	Input	Nominal	Character		Status Cate
StatusCatStar	Default	4.0	Input	Binary	Numeric		Status Cate

9. Select **Explore....** The Explore window opens.



The **GiftAvg** and **GiftCnt** inputs show some degree of skew in their distribution. The **GiftTime** inputs do not. To regularize the skewed distributions, use the log transformation. For these inputs, the order of magnitude of the underlying measure predicts the target rather than the measure itself.

10. Close the Explore window.

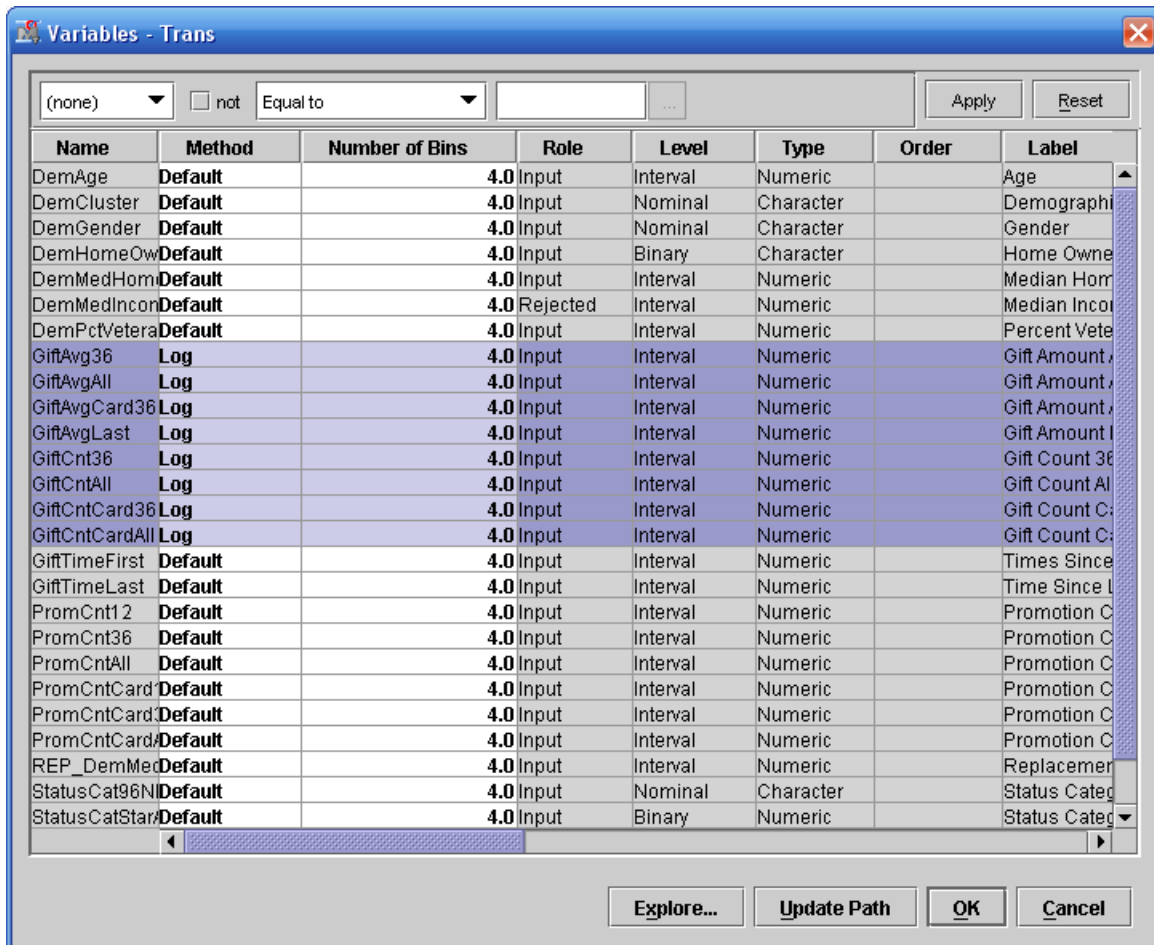
11. Deselect the two inputs with **GiftTime** in their names.

Variables - Trans

(none) ☐ not Equal to ...

Name	Method	Number of Bins	Role	Level	Type	Order	Label
DemAge	Default	4.0	Input	Interval	Numeric		Age
DemCluster	Default	4.0	Input	Nominal	Character		Demographi
DemGender	Default	4.0	Input	Nominal	Character		Gender
DemHomeOw	Default	4.0	Input	Binary	Character		Home Owne
DemMedHom	Default	4.0	Input	Interval	Numeric		Median Hom
DemMedIncon	Default	4.0	Rejected	Interval	Numeric		Median Inco
DemPctVetera	Default	4.0	Input	Interval	Numeric		Percent Vete
GiftAvg36	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftAvgAll	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftAvgCard36	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftAvgLast	Default	4.0	Input	Interval	Numeric		Gift Amount
GiftCnt36	Default	4.0	Input	Interval	Numeric		Gift Count 36
GiftCntAll	Default	4.0	Input	Interval	Numeric		Gift Count Al
GiftCntCard36	Default	4.0	Input	Interval	Numeric		Gift Count C:
GiftCntCardAll	Default	4.0	Input	Interval	Numeric		Gift Count C:
GiftTimeFirst	Default	4.0	Input	Interval	Numeric		Times Since
GiftTimeLast	Default	4.0	Input	Interval	Numeric		Time Since L
PromCnt12	Default	4.0	Input	Interval	Numeric		Promotion C
PromCnt36	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntAll	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCard36	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCardAll	Default	4.0	Input	Interval	Numeric		Promotion C
PromCntCardLast	Default	4.0	Input	Interval	Numeric		Promotion C
REP_DemMed	Default	4.0	Input	Interval	Numeric		Replaceme
StatusCat96N	Default	4.0	Input	Nominal	Character		Status Categ
StatusCatStar	Default	4.0	Input	Binary	Numeric		Status Categ

12. Select **Method** ⇒ **Log** for one of the remaining selected inputs. The selected method changes from Default to Log for the **GiftAvg** and **GiftCnt** inputs.



13. Select **OK** to close the Variables - Trans window.
14. Run the Transform Variables node and view the results.
15. Maximize the Output window and go to line 28.

Input Name	Role	Input Level	Name	Level	Formula
GiftAvg36	INPUT	INTERVAL	LOG_GiftAvg36	INTERVAL	$\log(\text{GiftAvg36} + 1)$
GiftAvgAll	INPUT	INTERVAL	LOG_GiftAvgAll	INTERVAL	$\log(\text{GiftAvgAll} + 1)$
GiftAvgCard36	INPUT	INTERVAL	LOG_GiftAvgCard36	INTERVAL	$\log(\text{GiftAvgCard36} + 1)$
GiftAvgLast	INPUT	INTERVAL	LOG_GiftAvgLast	INTERVAL	$\log(\text{GiftAvgLast} + 1)$
GiftCnt36	INPUT	INTERVAL	LOG_GiftCnt36	INTERVAL	$\log(\text{GiftCnt36} + 1)$
GiftCntAll	INPUT	INTERVAL	LOG_GiftCntAll	INTERVAL	$\log(\text{GiftCntAll} + 1)$
GiftCntCard36	INPUT	INTERVAL	LOG_GiftCntCard36	INTERVAL	$\log(\text{GiftCntCard36} + 1)$
GiftCntCardAll	INPUT	INTERVAL	LOG_GiftCntCardAll	INTERVAL	$\log(\text{GiftCntCardAll} + 1)$

Notice the Formula column. While a log transformation was specified, the actual transformation used was $\log(\text{input} + 1)$. This default action of the Transform Variables tool avoids problems with 0-values of the underlying inputs.

16. Close the Transform Variables - Results window.

Regressions with Transformed Inputs

The following steps revisit regression, and use the transformed inputs:

1. Run the diagram from the Regression node and view the results.
2. Go to line 3754 the Output window.

Summary of Stepwise Selection							
Step	Effect		DF	Number		Score	
	Entered	Removed		In	Chi-Square	Wald Chi-Square	Pr > ChiSq
1	LOG_GiftCnt36		1	1	95.0275		<.0001
2	GiftTimeLast		1	2	21.1330		<.0001
3	DemMedHomeValue		1	3	17.7373		<.0001
4	LOG_GiftAvgAll		1	4	21.7306		<.0001
5	DemPctVeterans		1	5	7.0742		0.0078
6	StatusCat96NK		5	6	13.7906		0.0170
7	LOG_GiftCntCard36		1	7	5.9966		0.0143
8	M_DemAge		1	8	5.0301		0.0249
9	DemCluster		53	9	61.2167		0.2049
10	StatusCatStarAll		1	10	1.2431		0.2649
11	PromCntCard12		1	11	1.4604		0.2269
12	PromCntAll		1	12	1.0022		0.3168
13	LOG_GiftCntAll		1	13	2.2990		0.1295
14	PromCnt12		1	14	0.8158		0.3664
15	PromCntCardAll		1	15	1.8875		0.1695
16		PromCntCard12	1	14		0.0358	0.8500
17	M_REP_DemMedIncome		1	15	0.6075		0.4357
18	LOG_GiftAvg36		1	16	0.4691		0.4934
19	M_LOG_GiftAvgCard36		1	17	0.6226		0.4301
20	GiftTimeFirst		1	18	0.3972		0.5285
21		GiftTimeFirst	1	17		0.3971	0.5286

The selected model, based on the CHOOSE=ERROR criterion, is the model trained in Step 4. It consists of the following effects:

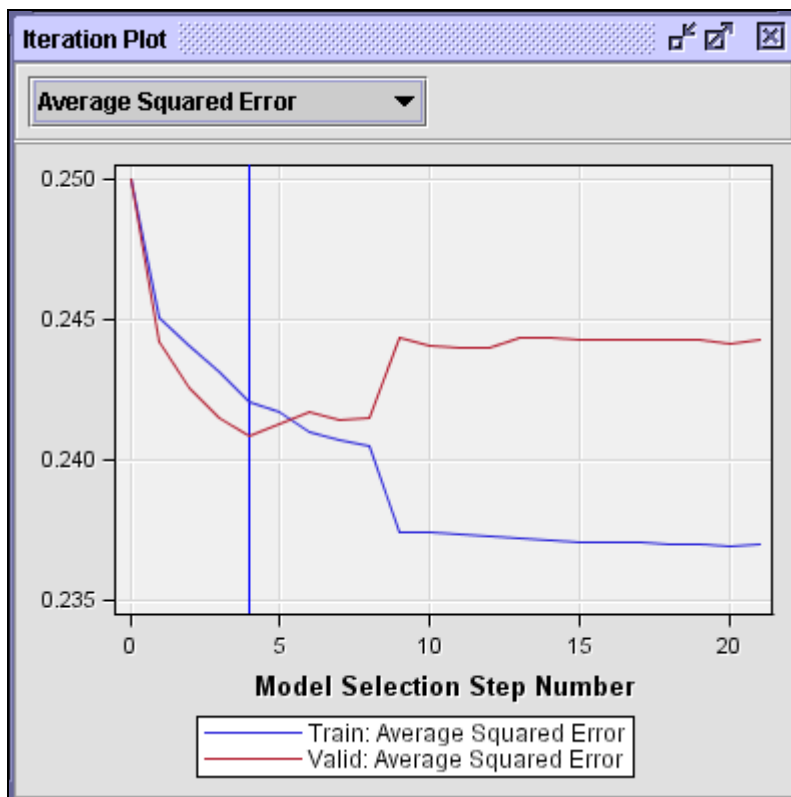
Intercept DemMedHomeValue GiftTimeLast LOG_GiftAvgAll LOG_GiftCnt36

The stepwise selection process took 21 steps, and the selected model came from step 4. Notice that half of the selected inputs are log transformations of the original gift variables.

3. Go to line 3800 to view more statistics from the selected model.

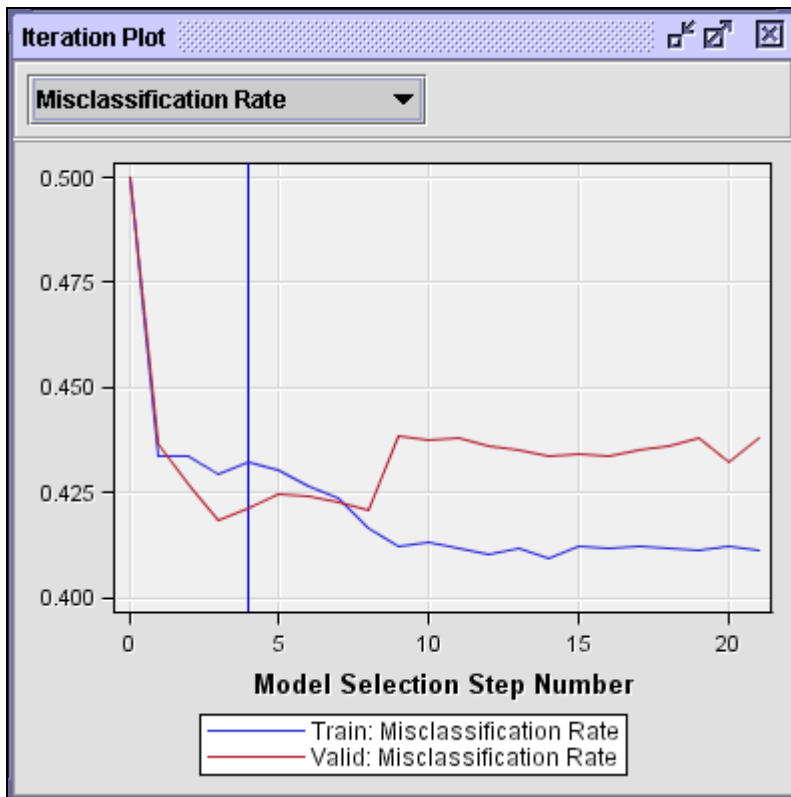
Analysis of Maximum Likelihood Estimates							
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq	Standardized Estimate	Exp(Est)
Intercept	1	0.8251	0.2921	7.98	0.0047		2.282
DemMedHomeValue	1	1.448E-6	3.002E-7	23.26	<.0001	0.0798	1.000
GiftTimeLast	1	-0.0341	0.00756	20.33	<.0001	-0.0758	0.966
LOG_GiftAvgAll	1	-0.3469	0.0747	21.58	<.0001	-0.0895	0.707
LOG_GiftCnt36	1	0.3736	0.0728	26.34	<.0001	0.0998	1.453
Odds Ratio Estimates							
Effect		Point Estimate					
DemMedHomeValue		1.000					
GiftTimeLast		0.966					
LOG_GiftAvgAll		0.707					
LOG_GiftCnt36		1.453					

4. Select **View** ⇒ **Model** ⇒ **Iteration Plot**.



The selected model (based on minimum error) occurs in Step 4. The value of average squared error for this model is slightly lower than that for the model with the untransformed inputs.

5. Select **Select Chart** ⇒ **Misclassification Rate**.



The misclassification rate with the transformed input model is nearly the same as that for the untransformed input model. The model with the lowest misclassification rate comes from Step 3. If you want to optimize on the misclassification rate, you must change this property in the Regression node's property sheet.

6. Close the Results window.

4.6 Categorical Inputs

Beyond the Prediction Formula

- ▶ Manage missing values.
- ▶ Interpret the model.
- ▶ Handle extreme or unusual values.
- ▶ **Use nonnumeric inputs.**
- ▶ Account for nonlinearities.

81

...

Using nonnumeric or categorical inputs presents another problem for regressions. As was seen in the earlier demonstrations, inclusion of a categorical input with excessive levels can lead to overfitting.

Nonnumeric Input Coding

<i>Level</i>	<i>D_A</i>	<i>D_B</i>	<i>D_C</i>	<i>D_D</i>	<i>D_E</i>	<i>D_F</i>	<i>D_G</i>	<i>D_H</i>	<i>D_I</i>
A	1	0	0	0	0	0	0	0	0
B	0	1	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	0	0
D	0	0	0	1	0	0	0	0	0
E	0	0	0	0	1	0	0	0	0
F	0	0	0	0	0	1	0	0	0
G	0	0	0	0	0	0	1	0	0
H	0	0	0	0	0	0	0	1	0
I	0	0	0	0	0	0	0	0	1

83

...

Coding Redundancy

Level	D_A	D_B	D_C	D_D	D_E	D_F	D_G	D_H	D_I
A	1	0	0	0	0	0	0	0	0
B	0	1	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	0	0
D	0	0	0	1	0	0	0	0	0
E	0	0	0	0	1	0	0	0	0
F	0	0	0	0	0	1	0	0	0
G	0	0	0	0	0	0	1	0	0
H	0	0	0	0	0	0	0	1	0
I	0	0	0	0	0	0	0	0	1

84

...

To represent these nonnumeric inputs in a model, you must convert them to some sort of numeric values. This conversion is most commonly done by creating design variables (or *dummy* variables), with each design variable representing approximately one level of the categorical input. (The total number of design variables required is, in fact, one less than the number of inputs.) A single categorical input can vastly increase a model's degrees of freedom, which, in turn, increases the chances of a model overfitting.

Coding Consolidation

Level	D_A	D_B	D_C	D_D	D_E	D_F	D_G	D_H	D_I
A	1	0	0	0	0	0	0	0	0
B	0	1	0	0	0	0	0	0	0
C	0	0	1	0	0	0	0	0	0
D	0	0	0	1	0	0	0	0	0
E	0	0	0	0	1	0	0	0	0
F	0	0	0	0	0	1	0	0	0
G	0	0	0	0	0	0	1	0	0
H	0	0	0	0	0	0	0	1	0
I	0	0	0	0	0	0	0	0	1

85

...

Coding Consolidation

Level	D_{ABCD}	D_B	D_C	D_D	D_{EF}	D_F	D_{GH}	D_H	D_I
A	1	0	0	0	0	0	0	0	0
B	1	1	0	0	0	0	0	0	0
C	1	0	1	0	0	0	0	0	0
D	1	0	0	1	0	0	0	0	0
E	0	0	0	0	1	0	0	0	0
F	0	0	0	0	1	1	0	0	0
G	0	0	0	0	0	0	1	0	0
H	0	0	0	0	0	0	1	1	0
I	0	0	0	0	0	0	0	0	1

86

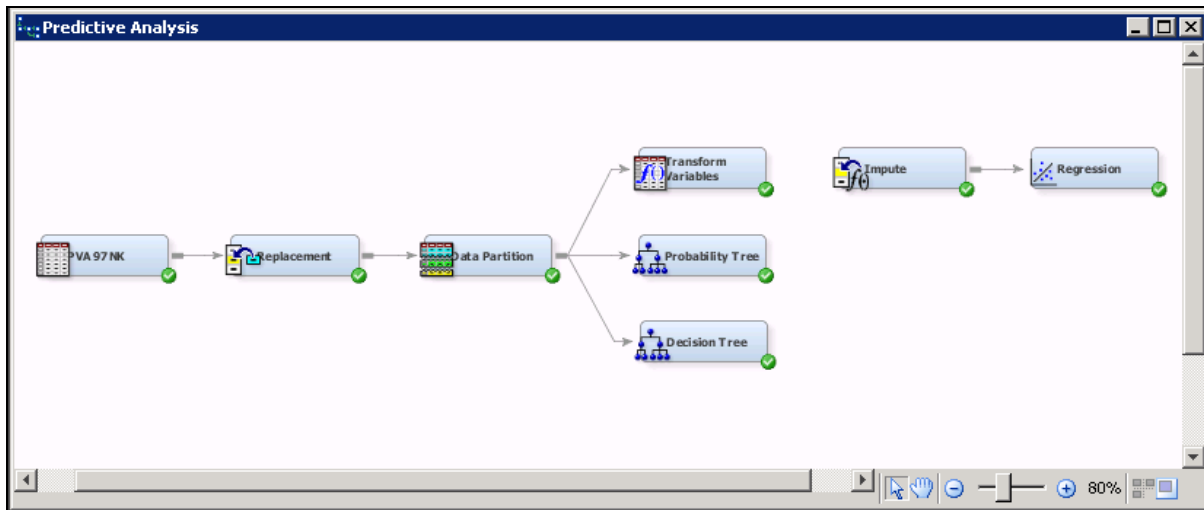
There are many remedies to this problem. One of the simplest remedies is to use domain knowledge to reduce the number of levels of the categorical input. In this way, level-groups are encoded in the model in place of the original levels.



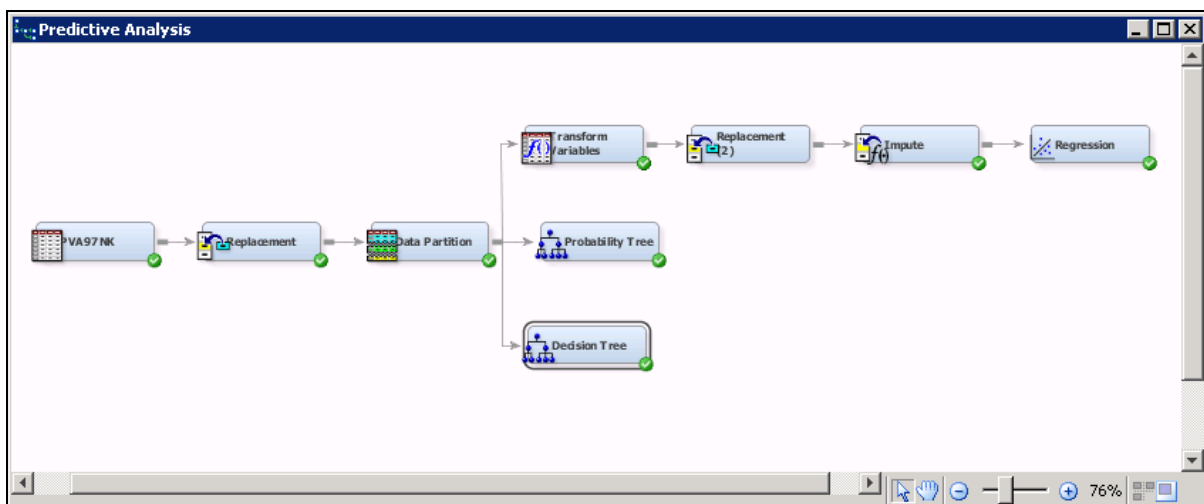
Recoding Categorical Inputs

In Chapter 2, you used the Replacement tool to eliminate an inappropriate value in the median income input. This demonstration shows how to use the Replacement tool to facilitate combining input levels of a categorical input.

1. Remove the connection between the Transform Variables node and the Impute node.



2. Select the **Modify** tab.
3. Drag a **Replacement** tool into the diagram workspace.
4. Connect the **Transform Variables** node to the **Replacement** node.
5. Connect the **Replacement** node to the **Impute** node.



You need to change some of the node's default settings so that the replacements are limited to a single categorical input.

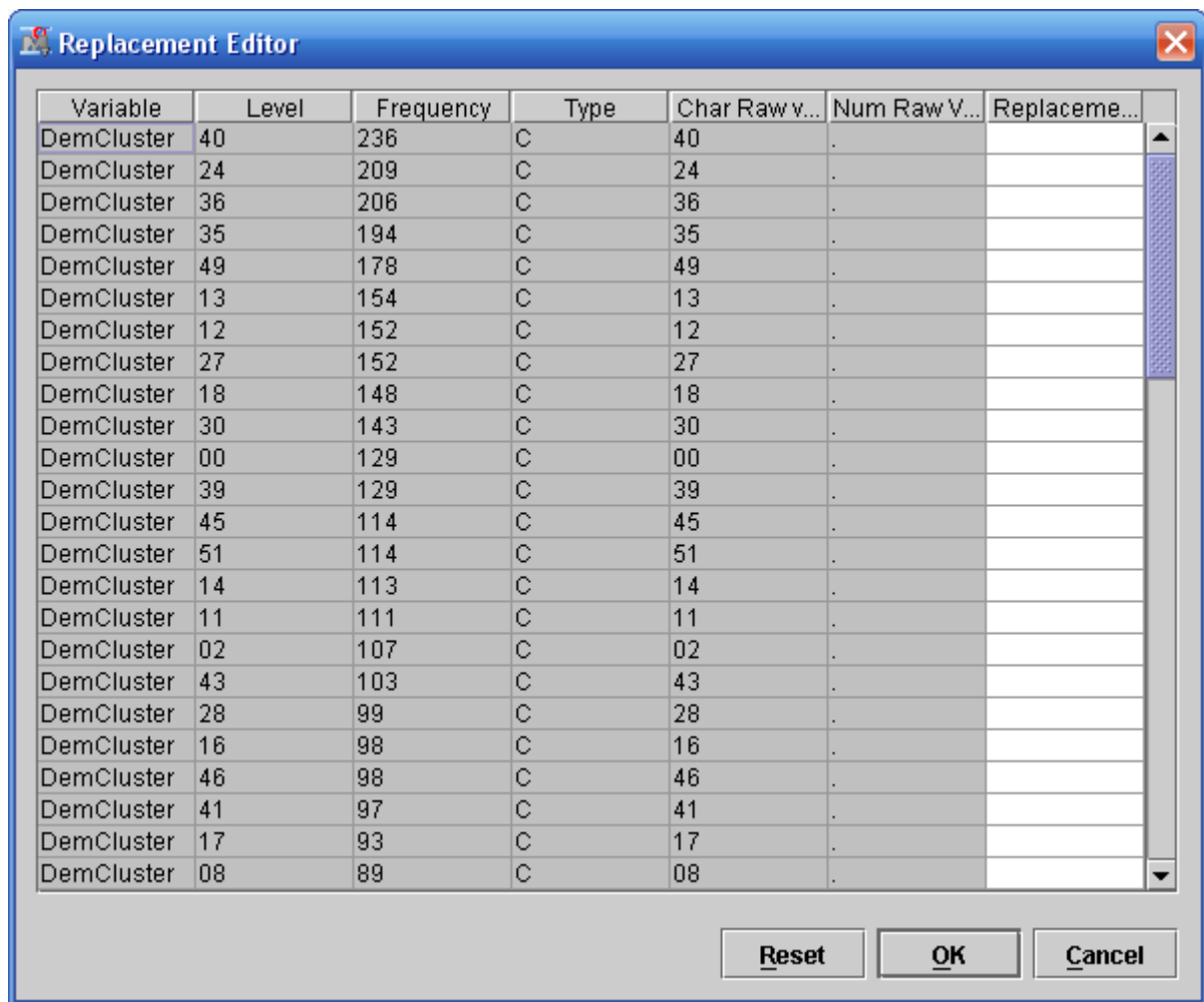
6. In the Interval Variables property group, select **Default Limits Method** ⇒ **None**.

Property	Value
General	
Node ID	Repl2
Imported Data	...
Exported Data	...
Notes	...
Train	
Interval Variables	
Replacement Editor	...
Default Limits Method	None
Cutoff Values	...
Class Variables	
Replacement Editor	...
Unknown Levels	Ignore

7. In the Class Variables property group, select **Replacement Editor** ⇒  from the Replacement node Properties panel.

Property	Value
General	
Node ID	Repl2
Imported Data	...
Exported Data	...
Notes	...
Train	
Interval Variables	
Replacement Editor	...
Default Limits Method	None
Cutoff Values	...
Class Variables	
Replacement Editor	...
Unknown Levels	Ignore

The Replacement Editor opens.

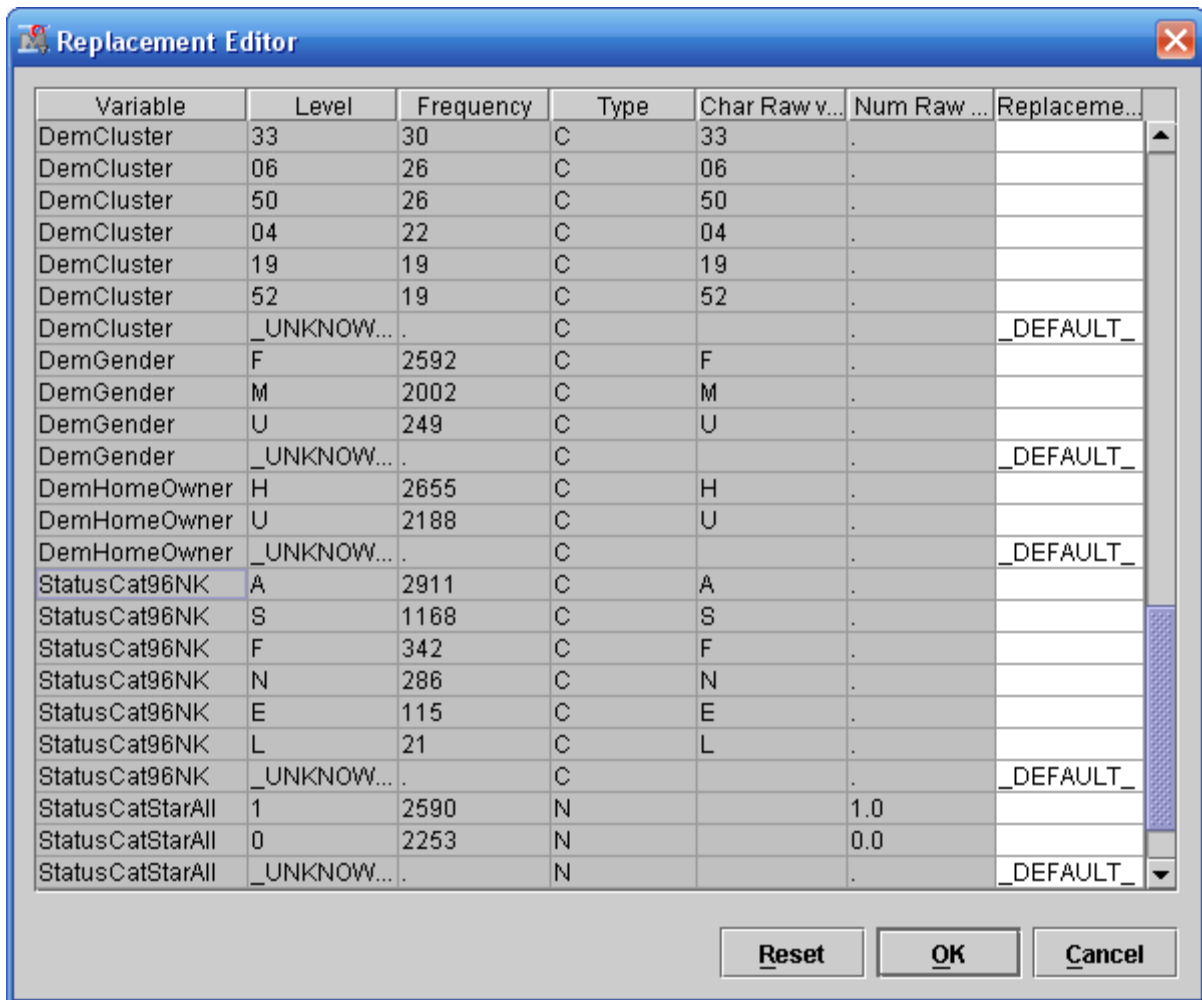


The categorical input Replacement Editor lists all levels of each binary, ordinal, and nominal input. You can use the Replacement column to reassign values to any of the levels.

The input with the largest number of levels is **DemCluster**, which has so many levels that consolidating the levels using the Replacement Editor would be an arduous task. (Another, autonomous method for consolidating the levels of **DemCluster** is presented as a special topic in Chapter 8.)

For this demonstration, you combine the levels of another input, **StatusCat96NK**.

8. Scroll the Replacement Editor to view the levels of **StatusCat96NK**.



Variable	Level	Frequency	Type	Char Raw v...	Num Raw ...	Replaceme...
DemCluster	33	30	C	33	.	
DemCluster	06	26	C	06	.	
DemCluster	50	26	C	50	.	
DemCluster	04	22	C	04	.	
DemCluster	19	19	C	19	.	
DemCluster	52	19	C	52	.	
DemCluster	_UNKNOWN...	.	C		.	_DEFAULT_
DemGender	F	2592	C	F	.	
DemGender	M	2002	C	M	.	
DemGender	U	249	C	U	.	
DemGender	_UNKNOWN...	.	C		.	_DEFAULT_
DemHomeOwner	H	2655	C	H	.	
DemHomeOwner	U	2188	C	U	.	
DemHomeOwner	_UNKNOWN...	.	C		.	_DEFAULT_
StatusCat96NK	A	2911	C	A	.	
StatusCat96NK	S	1168	C	S	.	
StatusCat96NK	F	342	C	F	.	
StatusCat96NK	N	286	C	N	.	
StatusCat96NK	E	115	C	E	.	
StatusCat96NK	L	21	C	L	.	
StatusCat96NK	_UNKNOWN...	.	C		.	_DEFAULT_
StatusCatStarAll	1	2590	N		1.0	
StatusCatStarAll	0	2253	N		0.0	
StatusCatStarAll	_UNKNOWN...	.	N		.	_DEFAULT_

The input has six levels, plus a level to represent unknown values (which do not occur in the training data). The levels of **StatusCat96NK** will be consolidated as follows:

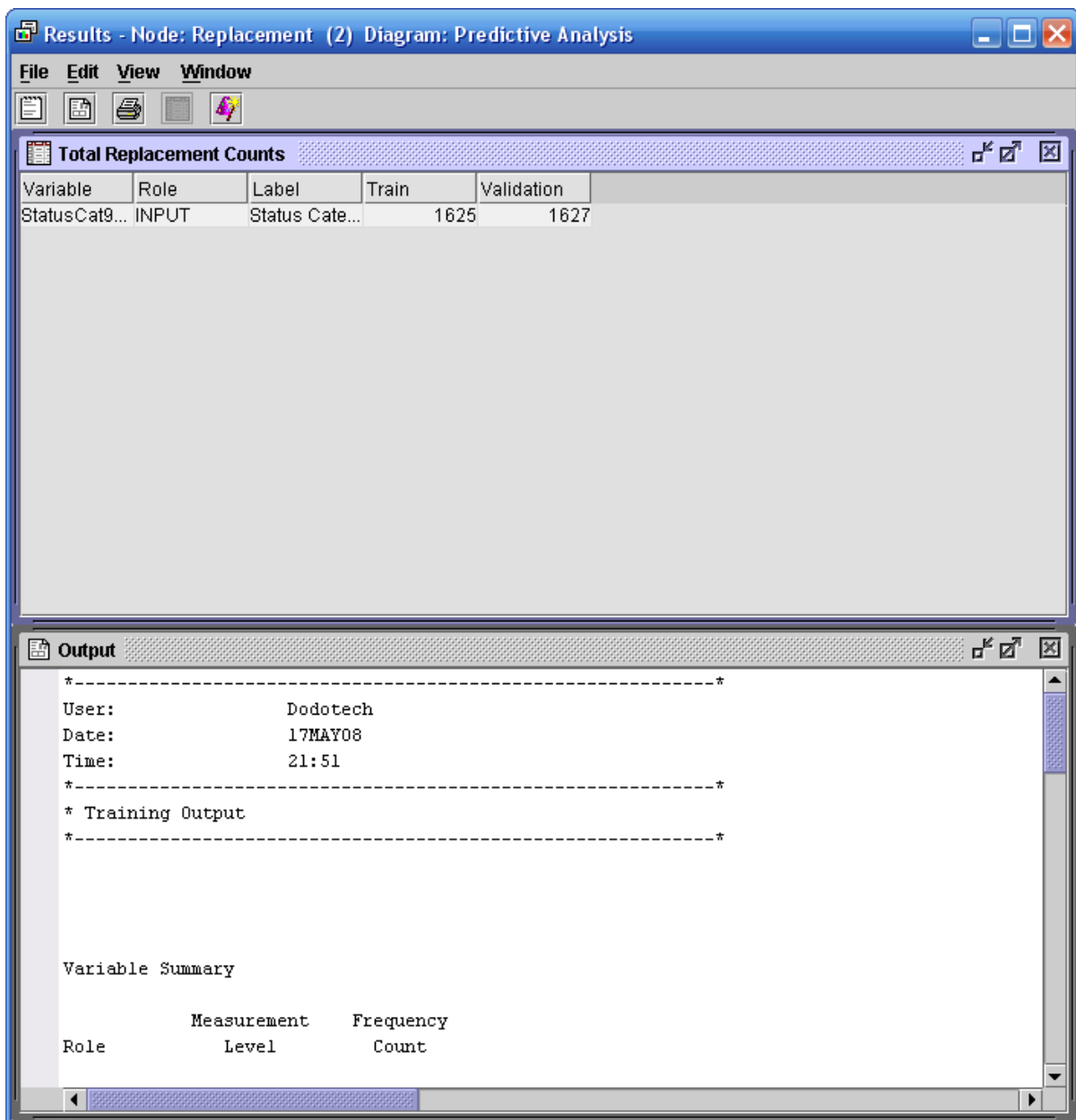
- Levels A and S (active and star donors) indicate consistent donors and are grouped into a single level, A.
- Levels F and N (first-time and new donors) indicate new donors and are grouped into a single level, N.
- Levels E and L (inactive and lapsing donors) indicate lapsing donors and are grouped into a single level L.

9. Type **A** as the Replacement level for **StatusCat96NK** levels A and S.
10. Type **N** as the Replacement level for **StatusCat96NK** levels F and N.
11. Type **L** as the Replacement level for **StatusCat96NK** levels L and E.

StatusCat96NK	A	2911	C	A	.	A
StatusCat96NK	S	1168	C	S	.	A
StatusCat96NK	F	342	C	F	.	N
StatusCat96NK	N	286	C	N	.	N
StatusCat96NK	E	115	C	E	.	L
StatusCat96NK	L	21	C	L	.	L
StatusCat96NK	_UNKNOWN...	.	C	.	.	_DEFAULT_

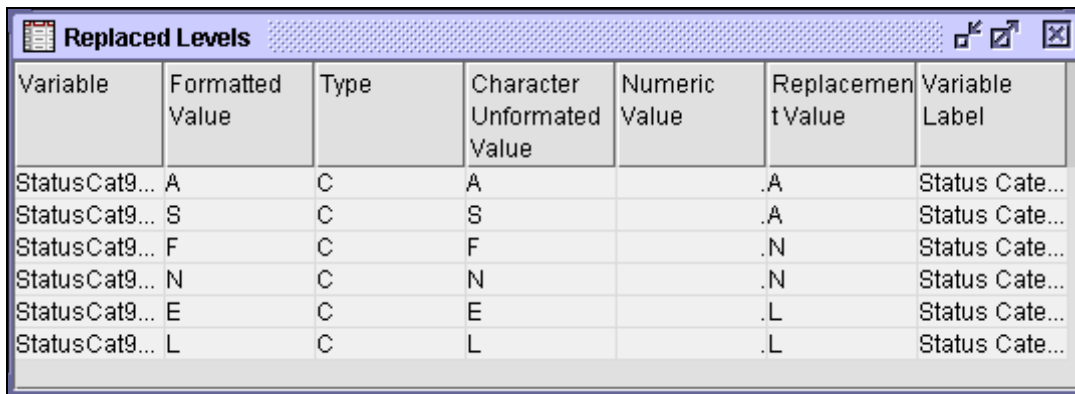
12. Select **OK** to close the Replacement Editor.

13. Run the Replacement node and view the results.



The Total Replacement Counts window shows the number of replacements that occur in the training and validation data.

14. Select **View** ⇒ **Model** ⇒ **Replaced Levels**. The Replaced Levels window opens.



Variable	Formatted Value	Type	Character Unformatted Value	Numeric Value	Replacement Value	Variable Label
StatusCat96NK	A	C	A		.A	Status Cat96NK
StatusCat96NK	S	C	S		.A	Status Cat96NK
StatusCat96NK	F	C	F		.N	Status Cat96NK
StatusCat96NK	N	C	N		.N	Status Cat96NK
StatusCat96NK	E	C	E		.L	Status Cat96NK
StatusCat96NK	L	C	L		.L	Status Cat96NK

The replaced level values are consistent with expectations.

15. Close the Results window.

16. Run the Regression node and view the results.

17. Go to line 3659 of the Output window.

Summary of Stepwise Selection								
Step	Entered	Effect		DF	Number		Score	
		Removed			In		Chi-Square	Wald
							Chi-Square	Pr > ChiSq
1	LOG_GiftCnt36			1	1	95.0275		<.0001
2	GiftTimeLast			1	2	21.1330		<.0001
3	DemMedHomeValue			1	3	17.7373		<.0001
4	LOG_GiftAvgAll			1	4	21.7306		<.0001
5	DemPctVeterans			1	5	7.0742		0.0078
6	REP_StatusCat96NK			2	6	9.7073		0.0078
7	LOG_GiftCntCard36			1	7	6.2112		0.0127
8	M_DemAge			1	8	4.8754		0.0272
9	DemCluster			53	9	61.7834		0.1910
10	StatusCatStarAll			1	10	1.6743		0.1957
11	PromCntCard12			1	11	1.3961		0.2374
12	PromCntAll			1	12	1.1442		0.2848
13	LOG_GiftCntAll			1	13	1.8685		0.1717
14	PromCnt12			1	14	0.6761		0.4109
15	PromCntCardAll			1	15	2.0585		0.1514
16		PromCntCard12		1	14		0.0216	0.8830
17	LOG_GiftAvg36			1	15	0.7608		0.3831
18	M_LOG_GiftAvgCard36			1	16	0.7343		0.3915
19	M_REP_DemMedIncome			1	17	0.5853		0.4443
20	GiftTimeFirst			1	18	0.3821		0.5365
21		GiftTimeFirst		1	17		0.3821	0.5365

The **REP_StatusCat96NK** input (created from the original **StatusCat96NK** input) is included in Step 6 the Stepwise Selection process. The three-level input is represented by two degrees of freedom.

18. Close the Results window.

4.7 Polynomial Regressions (Self-Study)

Beyond the Prediction Formula

- ▶ Manage missing values.
- ▶ Interpret the model.
- ▶ Handle extreme or unusual values.
- ▶ Use nonnumeric inputs.
- ▶ **Account for nonlinearities.**

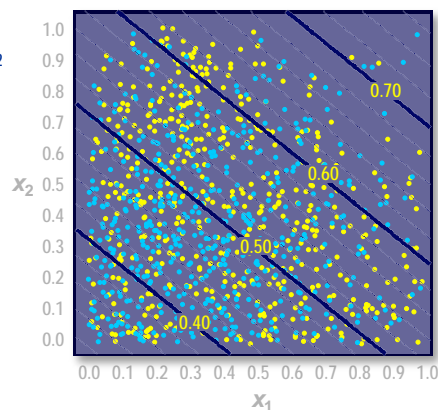
90

...

The Regression tool assumes (by default) a linear and additive association between the inputs and the logit of the target. If the true association is more complicated, such an assumption might result in biased predictions. For decisions and rankings, this bias can (in some cases) be unimportant. For estimates, this bias appears as a higher value for the validation average squared error fit statistic.

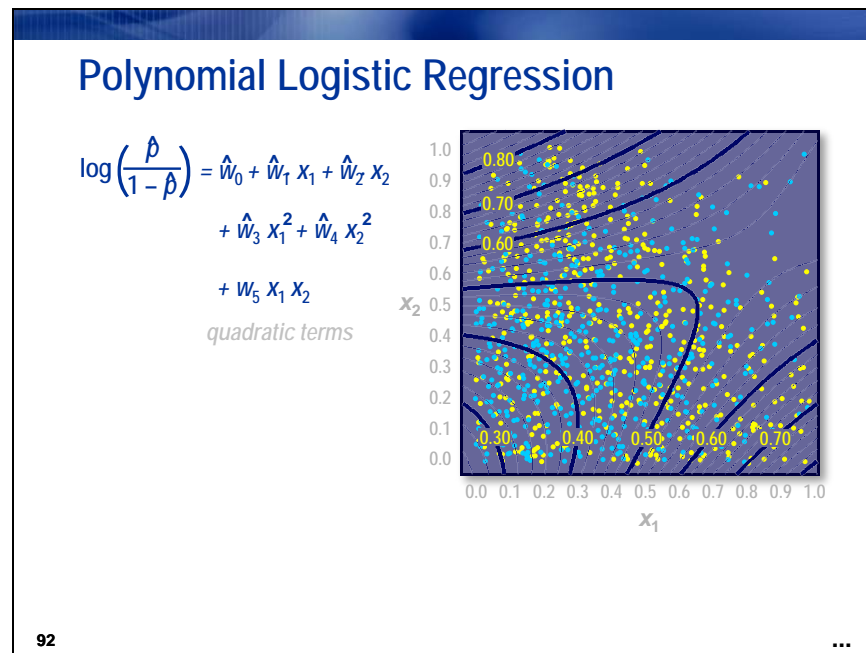
Standard Logistic Regression

$$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = \hat{w}_0 + \hat{w}_1 x_1 + \hat{w}_2 x_2$$



91

In the dot color problem, the (standard logistic regression) assumption that the concentration of yellow dots increases toward the upper right corner of the unit square seems to be suspect.



When minimizing prediction bias is important, you can increase the flexibility of a regression model by adding polynomial combinations of the model inputs. This enables predictions to better match the true input/target association. It also increases the chances of overfitting while simultaneously reducing the interpretability of the predictions. Therefore, polynomial regression must be approached with some care.

In SAS Enterprise Miner, adding polynomial terms can be done selectively or autonomously.

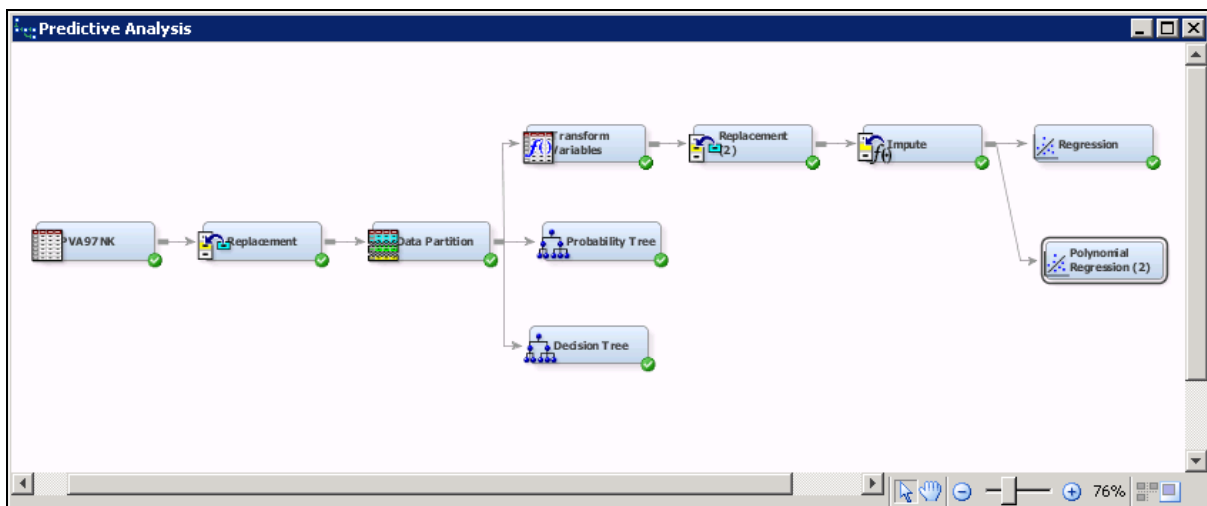


Adding Polynomial Regression Terms Selectively

This demonstration shows how to use the Term Editor window to selectively add polynomial regression terms.

You can modify the existing Regression node or add a new Regression node. If you add a new node, you must configure the Polynomial Regression node to perform the same tasks as the original. An alternative is to make a copy of the existing node.

1. Right-click the **Regression** node and select **Copy** from the menu.
2. Right-click the diagram workspace and select **Paste** from the menu. A new Regression node is added with the label **Regression (2)** to distinguish it from the existing one.
3. Select the **Regression (2)** node. The properties are identical to the existing node.
4. Rename the new regression node **Polynomial Regression (2)**. The (2) is retained to help with model identification in later chapters.
5. Connect the **Polynomial Regression (2)** node to the **Impute** node.



To add polynomial terms to the model, you use the Term Editor. To use the Term Editor, you need to enable User Terms.

6. Select **User Terms** ⇒ **Yes** in the Polynomial Regression (2) property panel.

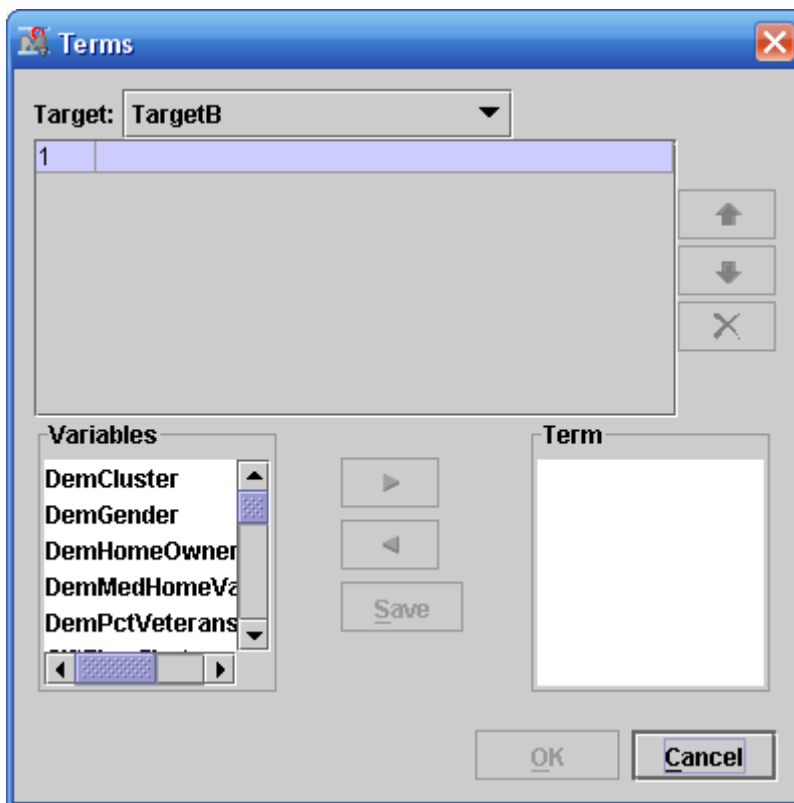
Property	Value
General	
Node ID	Reg2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	No
Polynomial Degree	2
User Terms	Yes
Term Editor	...

The Term Editor is now unlocked and can be used to add specific polynomial terms to the regression model.

3. Select **Term Editor** ⇒ ... from the Polynomial Regression Properties panel.


Property	Value
General	
Node ID	Reg2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Equation	
Main Effects	Yes
Two-Factor Interactions	No
Polynomial Terms	No
Polynomial Degree	2
User Terms	Yes
Term Editor	...

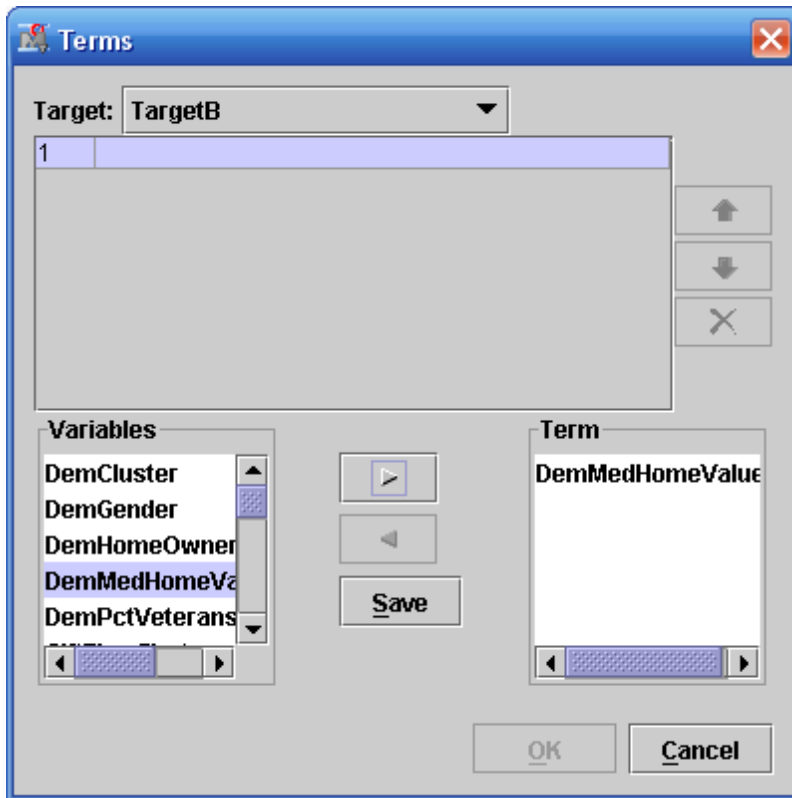
The Terms window opens.



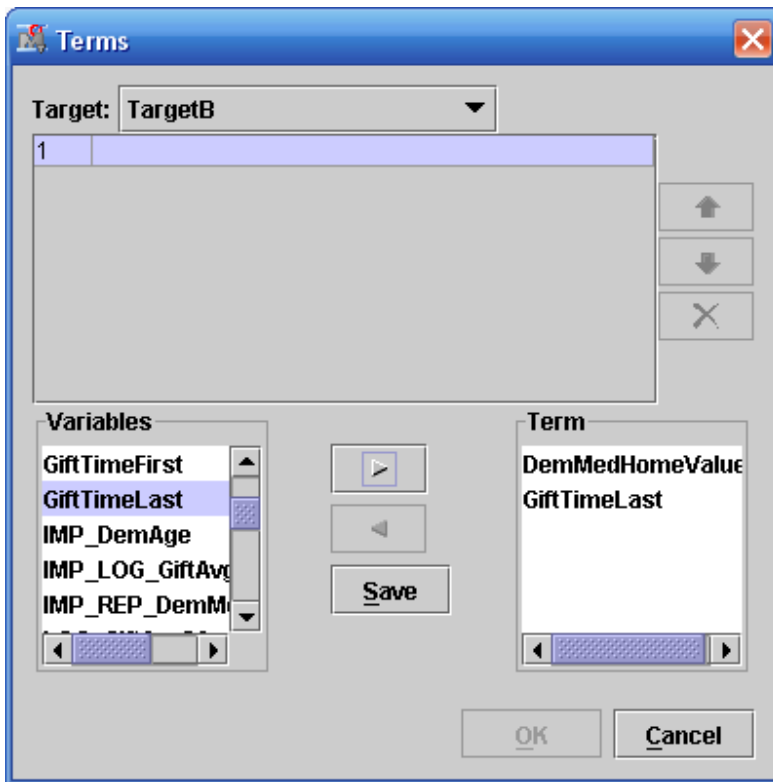
Interaction Terms

Suppose that you suspect an interaction between home value and time since last gift. (Perhaps a recent change in property values affected the donation patterns.)

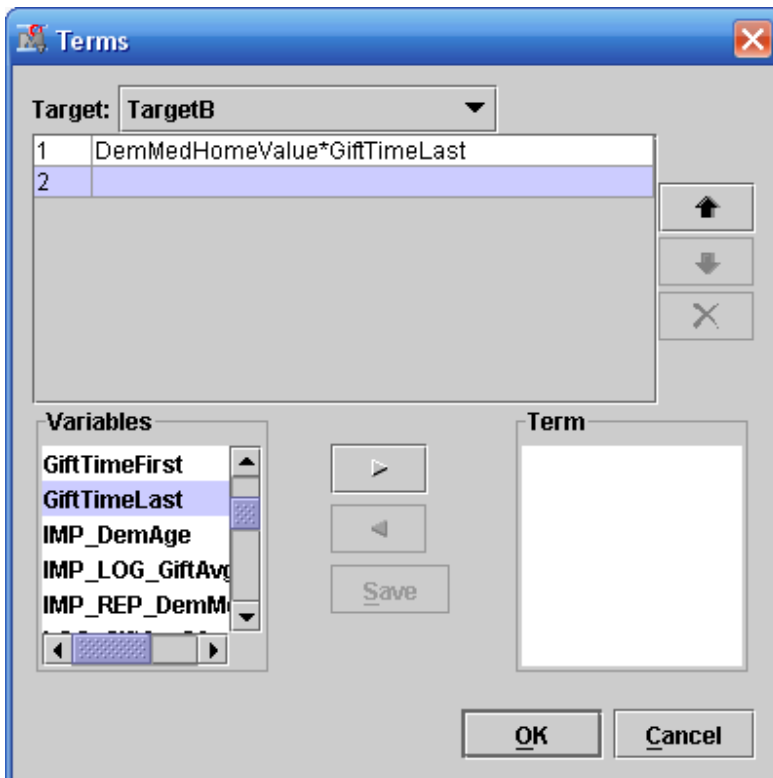
1. Select **DemMedHomeValue** in the Variables panel of the Terms dialog box.
2. Select the Add button, . The **DemMedHomeValue** input is added to the Term panel.



- Repeat the previous step to add **GiftTimeLast**.





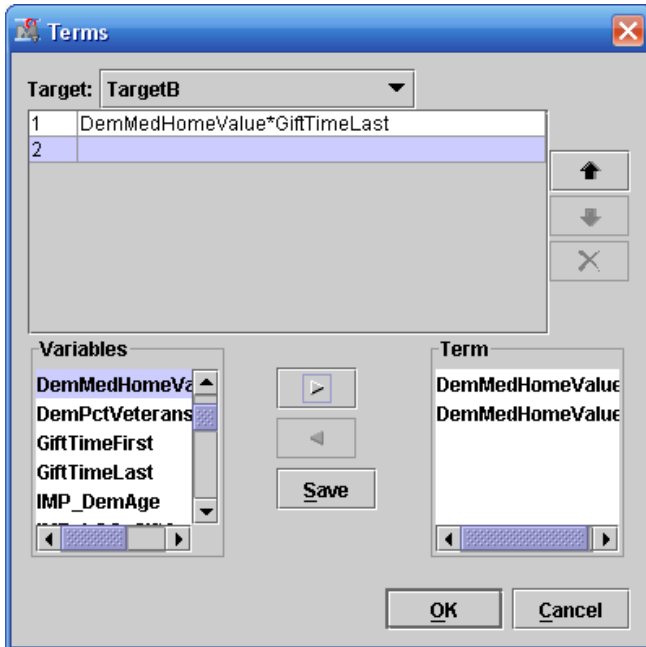
- Select **Save**. An interaction between the selected inputs is now available for consideration by the Regression node.



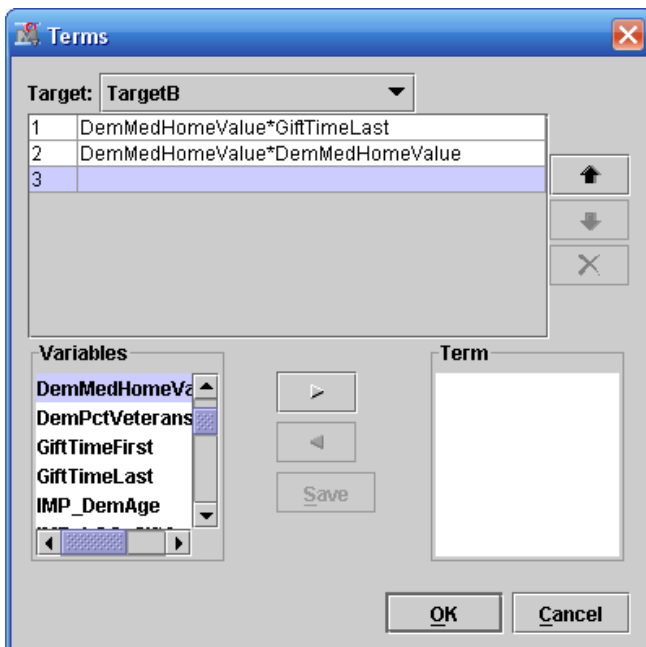
Quadratic Terms

Similarly, suppose that you suspect a parabola-shaped relationship between the logit of donation probability and median home value.

1. Select **DemMedHomeValue**.
2. Select the Add button, . The **DemMedHomeValue** input is added to the Term panel.
3. Select  again. Another **DemMedHomeValue** input is added to the Term panel.



4. Select **Save**. A quadratic median home value term is available for consideration by the model.



5. Select **OK** to close the Terms dialog box.
6. Run the Polynomial Regression node and view the results.
7. Go to line 3752 in the Output window.

Summary of Stepwise Selection								
Step	Entered	Effect	Removed	DF	Number In	Score Chi-Square	Wald Chi-Square	Pr > ChiSq
1	LOG_GiftOnt36			1	1	95.0275		<.0001
2	GiftTimeLast			1	2	21.1330		<.0001
3	DemMedHomeValue*GiftTimeLast			1	3	19.6032		<.0001
4	LOG_GiftAvgAll			1	4	21.8432		<.0001
5	DemPctVeterans			1	5	7.0965		0.0077
6	REP_StatusCat96NK			2	6	9.7708		0.0076
7	LOG_GiftOntCard36			1	7	6.2012		0.0128
8	M_DemAge			1	8	4.9143		0.0266
9	DemMedHomeValue*DemMedHomeValue			1	9	3.6530		0.0560
10	StatusCatStarAll			1	10	1.8153		0.1779
11	PromOntCard12			1	11	1.2570		0.2622
12	PromOntAll			1	12	1.3799		0.2401
13		StatusCatStarAll		1	11		0.4504	0.5021
14	DemCluster			53	12	58.7308		0.2736
15	LOG_GiftOntAll			1	13	1.0539		0.3046
16	StatusCatStarAll			1	14	1.2548		0.2626
17	PromOnt12			1	15	0.6591		0.4169
18	PromOntCardAll			1	16	2.0806		0.1492
19		PromOntCard12		1	15		0.0180	0.8931
20	LOG_GiftAvg36			1	16	0.7426		0.3888
21	M_REP_DemMedIncome			1	17	0.6424		0.4228
22	M_LOG_GiftAvgCard36			1	18	0.6013		0.4381
23	GiftTimeFirst			1	19	0.3946		0.5299
24		GiftTimeFirst		1	18		0.3945	0.5299

The stepwise selection summary shows the interaction term added in Step 3 and the quadratic term in Step 9.

8. Close the Results window.

This raises the obvious question: How do you know which nonlinear terms to include in a model? Unfortunately, there is no simple solution to this question in SAS Enterprise Miner, other than including all polynomial and interaction terms in the selection process.



Adding Polynomial Regression Terms Autonomously (Self-Study)

SAS Enterprise Miner has the ability to add **every** polynomial combination of inputs to a regression model. Obviously, this feature must be used with some care, because the number of polynomial input combinations increases rapidly with input count.

For instance, the **PVA97NK** data set has 20 interval inputs. If you want to consider every quadratic combination of these 20 inputs, your selection procedure must sequentially plod through more than 200 inputs. This is not an overwhelming task for today's fast computers.

Follow these steps to explore a full two-factor stepwise selection process:

1. Select **Two-Factor Interaction** ⇒ **Yes** in the Polynomial Regression property panel.
2. Select **Polynomial Terms** ⇒ **Yes** in the Polynomial Regression Properties panel.

Property	Value
General	
Node ID	Reg2
Imported Data	...
Exported Data	...
Notes	...
Train	
Variables	...
Equation	
Main Effects	Yes
Two-Factor Interaction	Yes
Polynomial Terms	Yes
Polynomial Degree	2
User Terms	Yes
Term Editor	...

3. Run the Polynomial Regression (2) node and view the results. (In general, this might take longer than most activities.)

4. Go to line 1774 of the Output window.

Summary of Stepwise Selection							
Step	Entered	Effect	Removed	Number DF	Score In Chi-Square	Wald Chi-Square	Pr > ChiSq
1		LOG_GiftCnt36*LOG_GiftCntCardAll		1	1	101.0902	<.0001
2		GiftTimeLast*LOG_GiftAvgLast		1	2	33.9163	<.0001
3		DemMedHomeValue*DemPctVeterans		1	3	25.2441	<.0001
4		REP_StatusCat96NK		2	4	10.2804	0.0059
5		DemHomeOwner*M_LOG_GiftAvgCard36		1	5	5.8659	0.0154
6		DemCluster*DemGender		106	6	134.9632	0.0302
7		GiftTimeLast*PromCnt12		1	7	5.6507	0.0174
8		LOG_GiftCntCard36*PromCnt12		1	8	3.7134	0.0540
9		LOG_GiftAvgAll		1	9	5.8292	0.0158
10		DemCluster		50	10	64.6125	0.0801
11		DemCluster	DemCluster	53	9	39.8737	0.9086



Surprisingly, the selection process takes only 11 steps. This is the result of the 106 degree-of-freedom **DemCluster** and **DemGender** interaction in Step 6. As the iteration plot shows below, the model is hopelessly overfit after this step. Inputs with many levels are problematic for predictive models. It is a good practice to reduce the impact of these inputs either by consolidating the levels or by simply excluding them from the analysis.

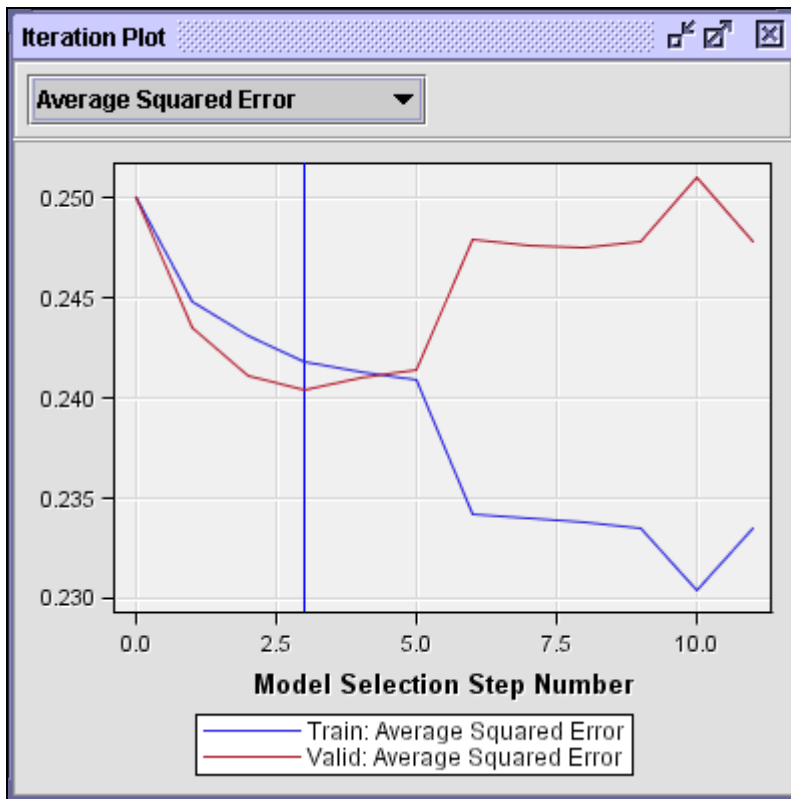
5. Scroll down in the Output Window.

The selected model, based on the CHOOSE=ERROR criterion, is the model trained in Step 3. It consists of the following effects:

Intercept DemMedHomeValue*DemPctVeterans GiftTimeLast*LOG_GiftAvgLast LOG_GiftCnt36*LOG_GiftCntCardAll

The selected model includes only three terms!

6. Select **View** ⇒ **Model** ⇒ **Iteration Plot**.



The validation average squared error of the three-term model is lower than any other model considered to this point.



Exercises

1. Predictive Modeling Using Regression

- a. Return to the Chapter 3 Organics diagram. Attach the StatExplore tool to the **ORGANICS** data source and run it.
- b. In preparation for regression, is any missing values imputation needed? _____
 If yes, should you do this imputation before generating the decision tree models? _____
 Why or why not? _____
- c. Add an **Impute** node to the diagram and connect it to the **Data Partition** node. Set the node to impute **U** for unknown class variable values and the overall mean for unknown interval variable values. Create imputation indicators for all imputed inputs.
- d. Add a **Regression** node to the diagram and connect it to the **Impute** node.
- e. Choose the stepwise selection and validation error as the selection criterion.
- f. Run the Regression node and view the results.
 Which variables are included in the final model? _____
 Which variables are important in this model? _____
 What is the validation ASE? _____
- g. In preparation for regression, are any transformations of the data warranted? _____
 Why or why not? _____
- h. Disconnect the **Impute** node from the **Data Partition** node.
- i. Add a **Transform Variables** node to the diagram and connect it to the **Data Partition** node.
- j. Connect the **Transform Variables** node to the **Impute** node.
- k. Apply a log transformation to the **DemAffl** and **PromTime** inputs.
- l. Run the **Transform Variables** node. Explore the exported training data. Did the transformations result in less skewed distributions? _____
- m. Rerun the **Regression** node.
 Do the selected variables change? _____
 How about the validation ASE? _____
- n. Create a full second-degree polynomial model. How does the validation average squared error for the polynomial model compare to the original model? _____

4.8 Chapter Summary

Regression models are a prolific and useful way to create predictions. New cases are scored using a prediction formula. Inputs are selected via a sequential selection process. Model complexity is controlled by fit statistics calculated on validation data.

To use regression models, there are several issues with which to contend that go beyond the predictive modeling essentials.

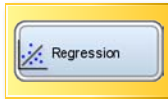
1. A mechanism for handling missing input values must be included in the model development process.
2. A reliable way to interpret the results is needed.
3. Methods for handling extreme or outlying predictions should be included.
4. The level-count of a categorical should be reduced to avoid overfitting.
5. The model complexity might need to be increased beyond what is provided by standard regression methods.

One approach to this is polynomial regression. Polynomial regression models can be fit manually with specific interactions in mind. They can also be fit autonomously by selecting polynomial terms from a list of all polynomial candidates.

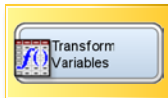
Regression Tools Review



Replace missing values for interval (means) and categorical data (mode). Create a unique replacement indicator.



Create linear and logistic regression models. Select inputs with a sequential selection method and appropriate fit statistic. Interpret models with odds ratios.



Regularize distributions of inputs. Typical transformations control for input skewness via a log transformation.

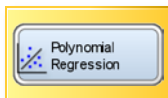
96

continued...

Regression Tools Review



Consolidate levels of a nonnumeric input using the Replacement Editor window.



Add polynomial terms to a regression either by hand or by an autonomous exhaustive search.

97

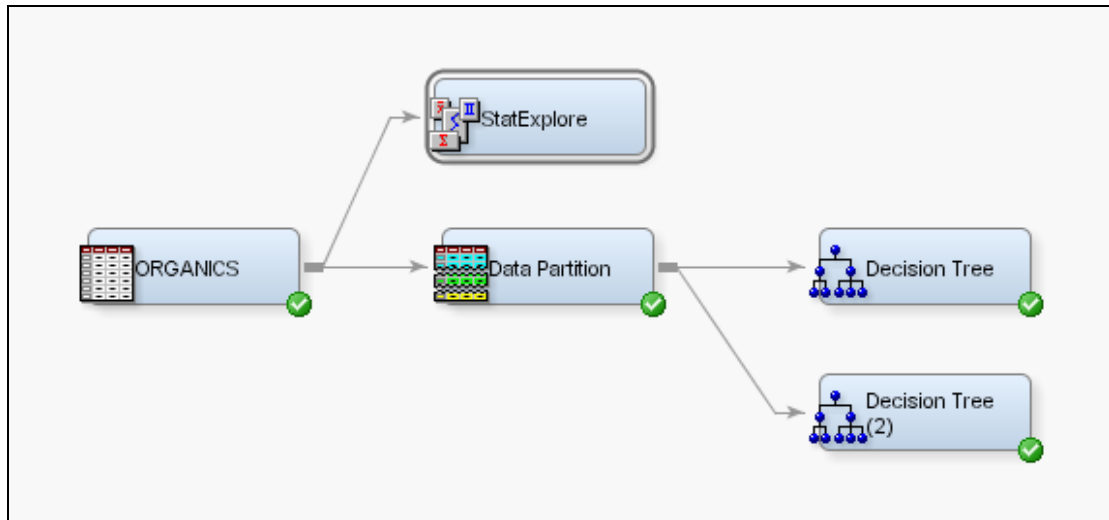
4.9 Solutions

Solutions to Exercises

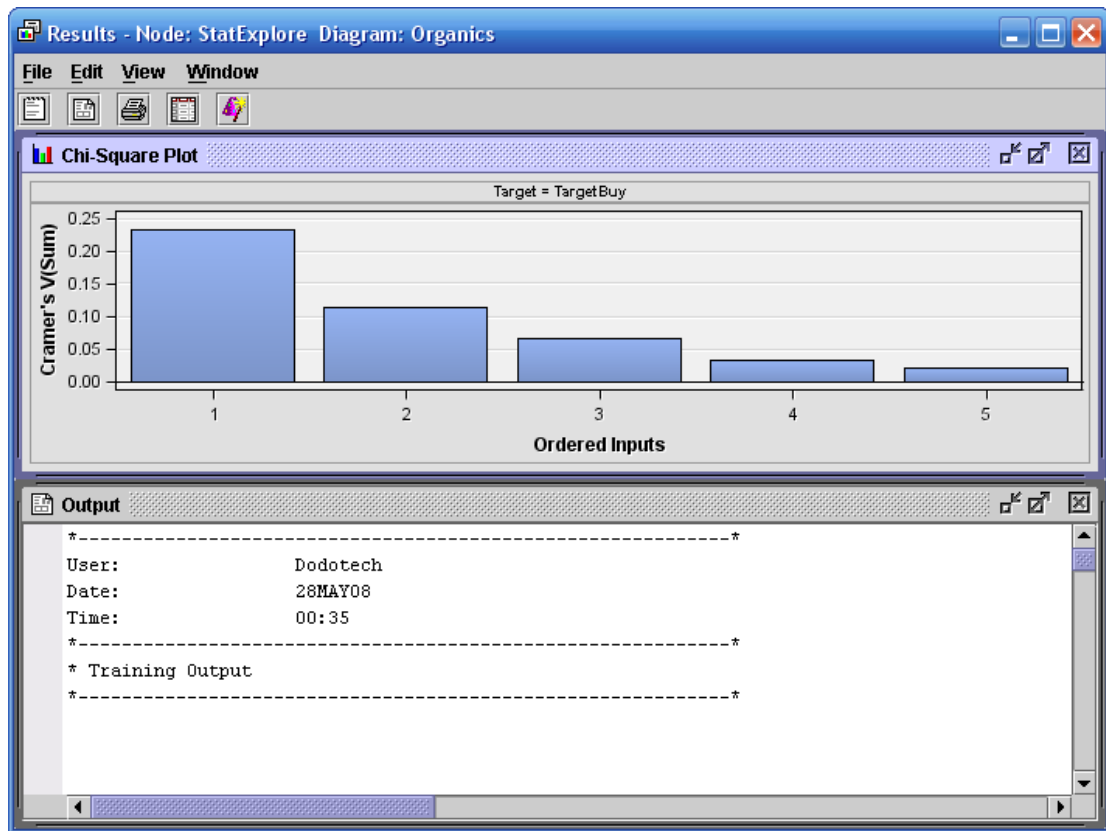
1. Predictive Modeling Using Regression

- a. Return to the Chapter 4 Organics diagram in the **Exercises** project. Use the StatExplore tool on the **ORGANICS** data source.

- 1) Connect the StatExplore node to the ORGANICS node as shown.



- 2) Run the StatExplore node and view the results.



- b. In preparation for regression, is any missing values imputation needed? If yes, should you do this imputation before generating the decision tree models? Why or why not?

Go to line 38 in the Output window. Several of the class inputs have missing values.

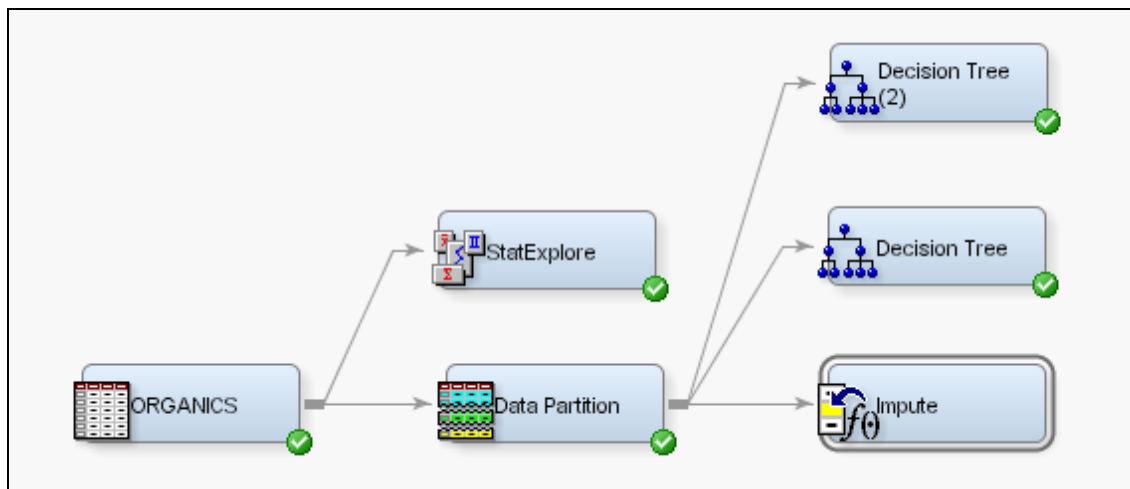
Class Variable Summary Statistics							
Variable	Role	Number of Levels	Missing	Mode	Mode Percentage	Mode2	Mode2 Percentage
DemClusterGroup	INPUT	8	674	C	20.55	D	19.70
DemGender	INPUT	4	2512	F	54.67	M	26.17
DemReg	INPUT	6	465	South East	38.85	Midlands	30.33
DemTVReg	INPUT	14	465	London	27.85	Midlands	14.05
PromClass	INPUT	4	0	Silver	38.57	Tin	29.19
TargetBuy	TARGET	2	0	0	75.23	1	24.77

Go to line 65 of the Output window. Most of the Interval inputs also have missing values.

Interval Variable Summary Statistics								
Variable	ROLE	Mean	Std. Deviation	Non Missing	Missing	Minimum	Median	Maximum
DemAffl	INPUT	8.71	3.42	21138	1085	0.00	8	34.00
DemAge	INPUT	53.80	13.21	20715	1508	18.00	54	79.00
PromSpend	INPUT	4420.59	7559.05	22223	0	0.01	2000	296313.85
PromTime	INPUT	6.56	4.66	21942	281	0.00	5	39.00

You do not need to impute before the Decision Tree node. Decision trees have built-in ways to handle missing values. (See Chapter 3.)

- c. Add an **Impute** node to the diagram and connect it to the **Data Partition** node. Set the node to impute **U** for unknown class variable values and the overall mean for unknown interval variable values. Create imputation indicators for all imputed inputs.



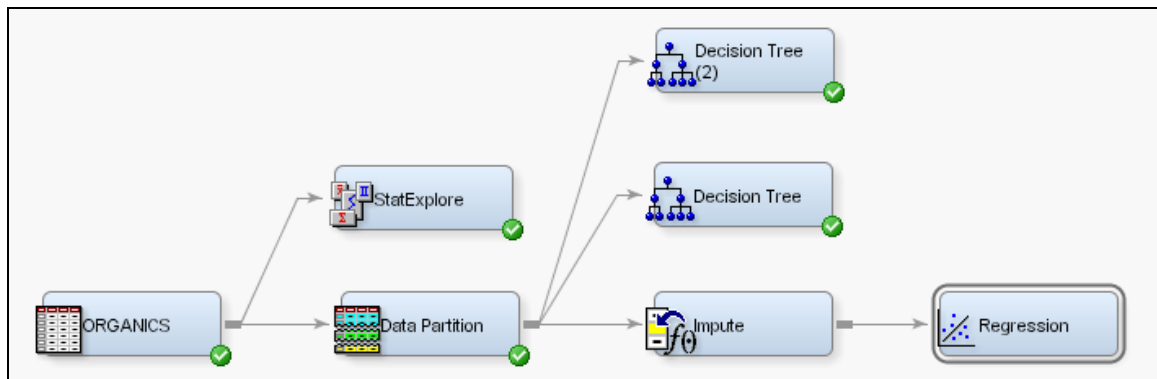
- 1) Select **Default Input Method** ⇒ **Default Constant Value**.
- 2) Type **U** for the Default Character Value.

Class Variables	
Default Input Method	Default Constant Value
Default Target Method	None
Normalize Values	Yes
Interval Variables	
Default Input Method	Mean
Default Target Method	None
Default Constant Value	
Default Character Value	U
Default Number Value	.

- 3) Select **Indicator Variable Type** ⇒ **Unique**.
- 4) Select **Indicator Variable Role** ⇒ **Input**.

Score	
Hide Original Variables	Yes
Indicator Variables	
Type	Unique
Source	Imputed Variables
Role	Input

- d. Add a **Regression** node to the diagram and connect it to the **Impute** node.

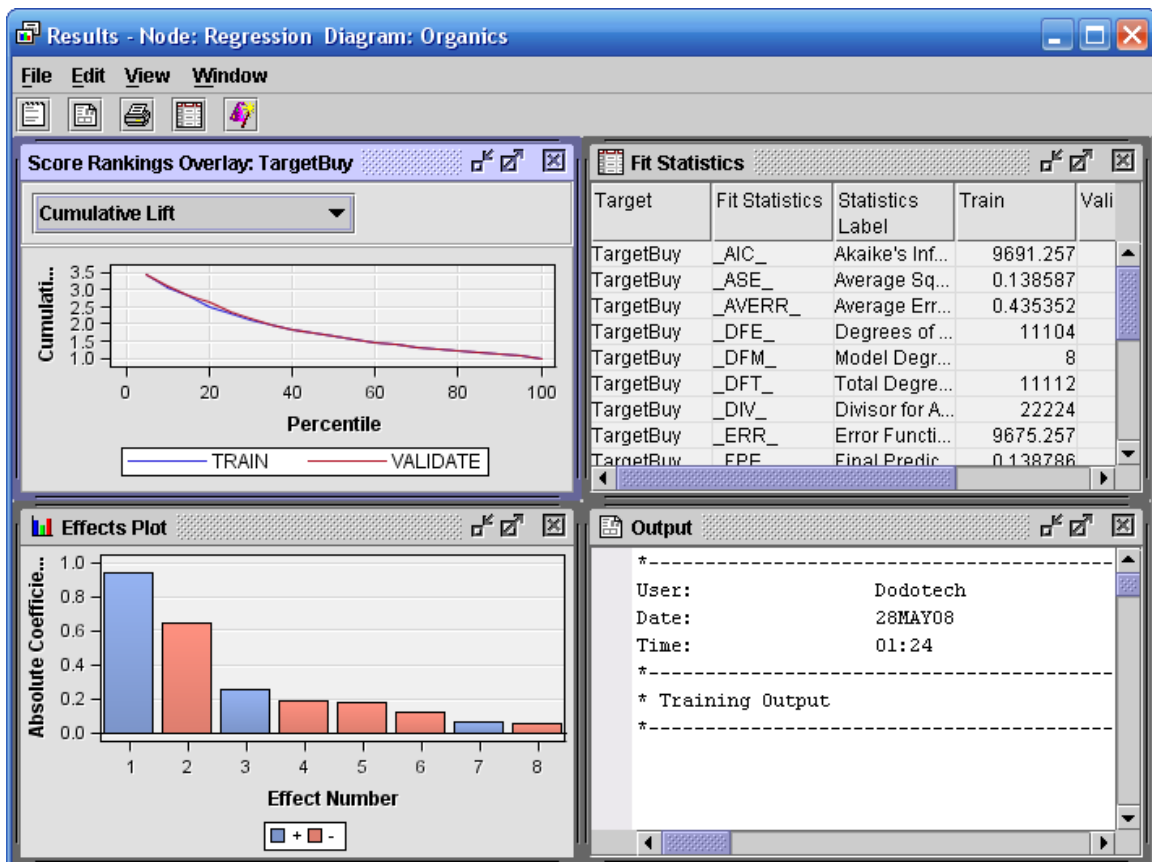


- e. Select the **Stepwise** selection and **Validation Error** as the selection criterion.

Model Selection	
Selection Model	Stepwise
Selection Criterion	Validation Error
Use Selection Defaults	Yes
Selection Options	...

- f. Run the Regression node and view the results. Which variables are included in the final model? Which variables are important in this model? What is the validation ASE?

- 1) The Results window opens.



- 2) Go to line 664 in the Output window.

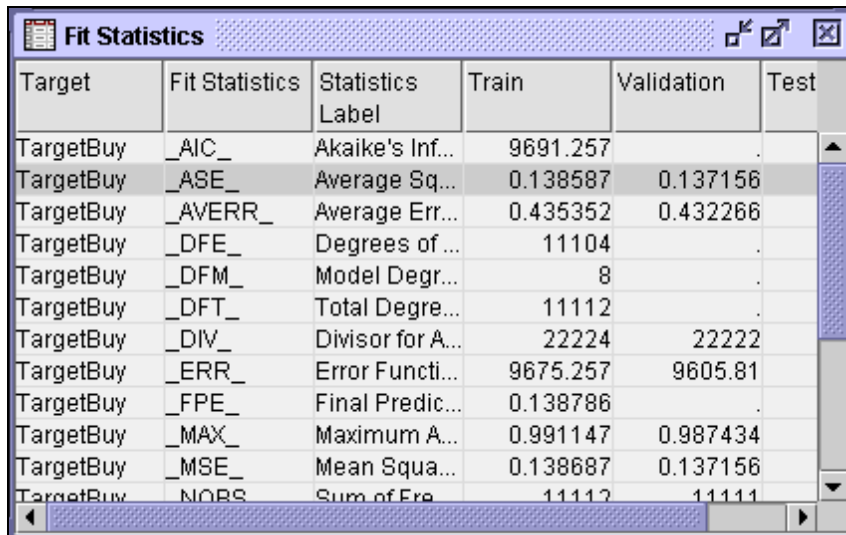
The selected model, based on the CHOOSE=ERROR criterion, is the model trained in Step 6. It consists of the following effects:

Intercept IMP_DemAff1 IMP_DemAge IMP_DemGender M_DemAff1 M_DemAge M_DemGender

- 3) The odds ratios indicate the effect that each input has on the logit score.

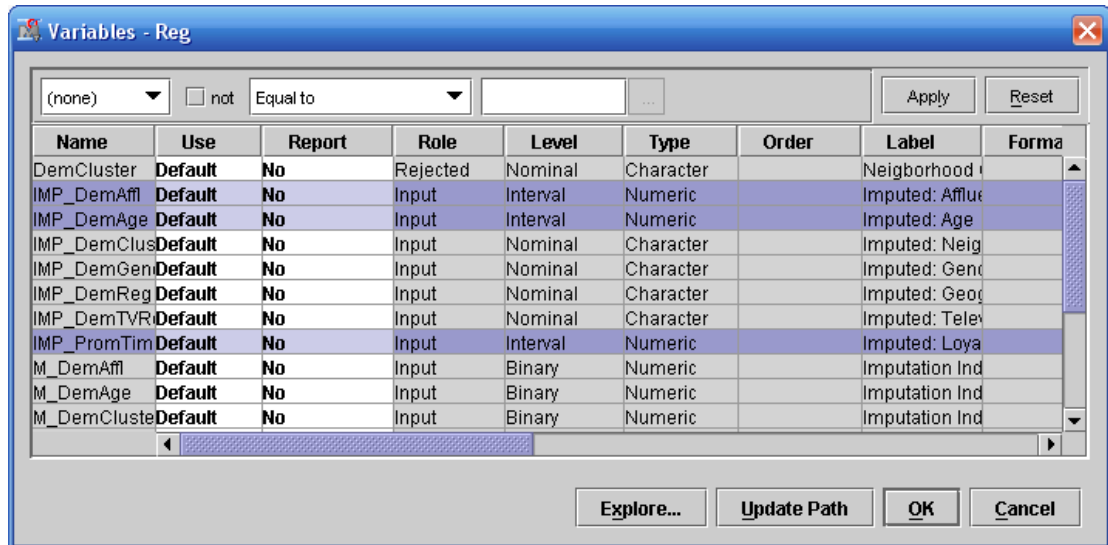
			Point
Effect			Estimate
IMP_DemAff1			1.283
IMP_DemAge			0.947
IMP_DemGender	F vs U		6.967
IMP_DemGender	M vs U		2.899
M_DemAff1	0 vs 1		0.708
M_DemAge	0 vs 1		0.796
M_DemGender	0 vs 1		0.685

- 4) The validation ASE is given in the Fit Statistics window.

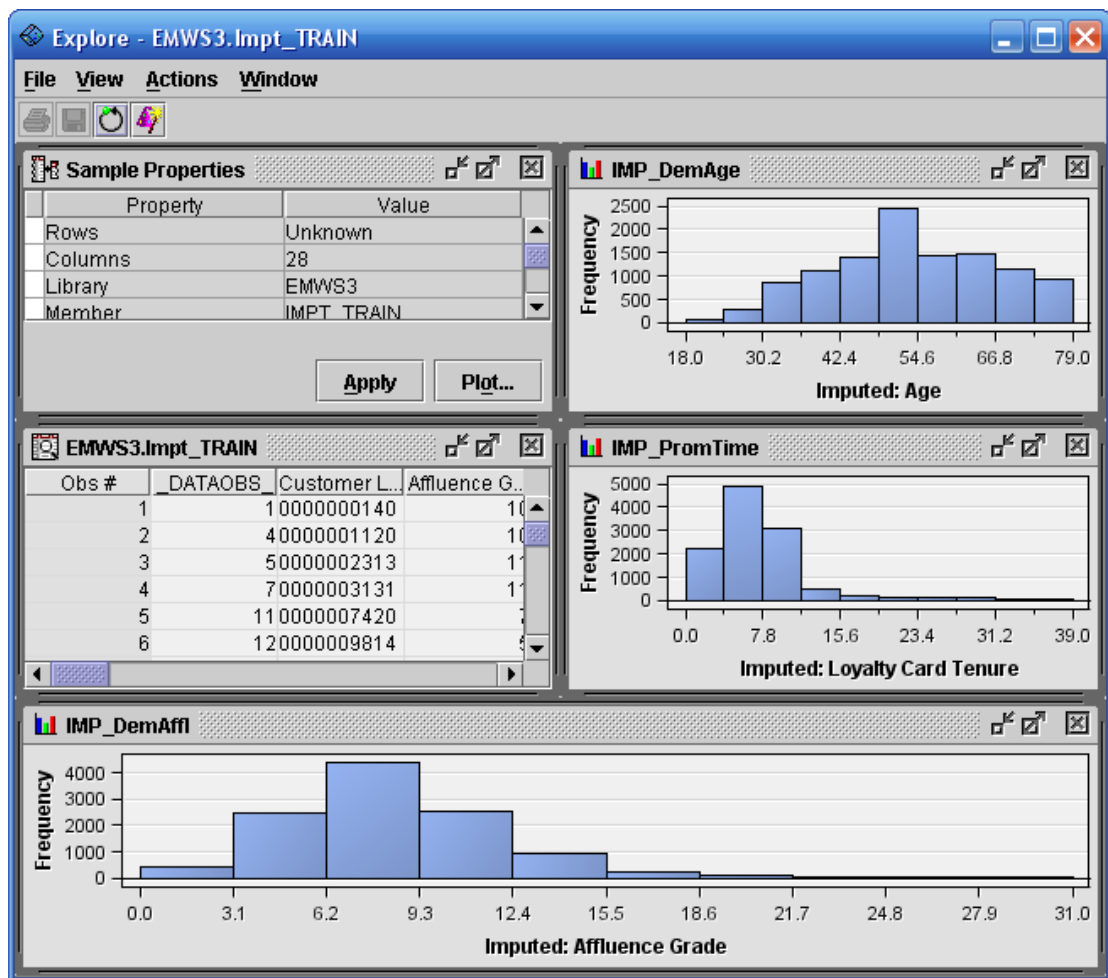


Target	Fit Statistics	Statistics Label	Train	Validation	Test
TargetBuy	_AIC_	Akaike's Inf...	9691.257	.	.
TargetBuy	_ASE_	Average Sq...	0.138587	0.137156	.
TargetBuy	_AVERR_	Average Err...	0.435352	0.432266	.
TargetBuy	_DFE_	Degrees of ...	11104	.	.
TargetBuy	_DFM_	Model Degr...	8	.	.
TargetBuy	_DFT_	Total Degre...	11112	.	.
TargetBuy	_DIV_	Divisor for A...	22224	22222	.
TargetBuy	_ERR_	Error Functi...	9675.257	9605.81	.
TargetBuy	_FPE_	Final Predic...	0.138786	.	.
TargetBuy	_MAX_	Maximum A...	0.991147	0.987434	.
TargetBuy	_MSE_	Mean Squa...	0.138687	0.137156	.
TargetBuy	_NOBS_	Sum of Ere...	11112	11111	.

- 1) Open the Variables window of the Regression node.
- 2) Select all Interval inputs.

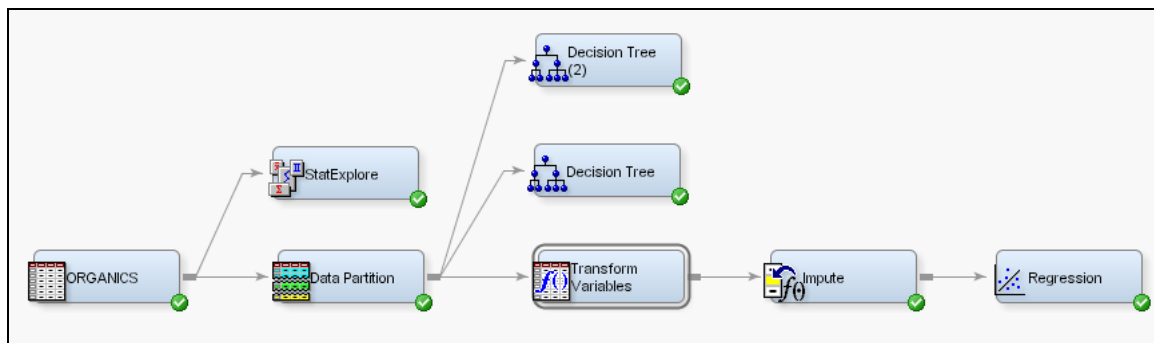


3) Select **Explore...**. The Explore window opens.

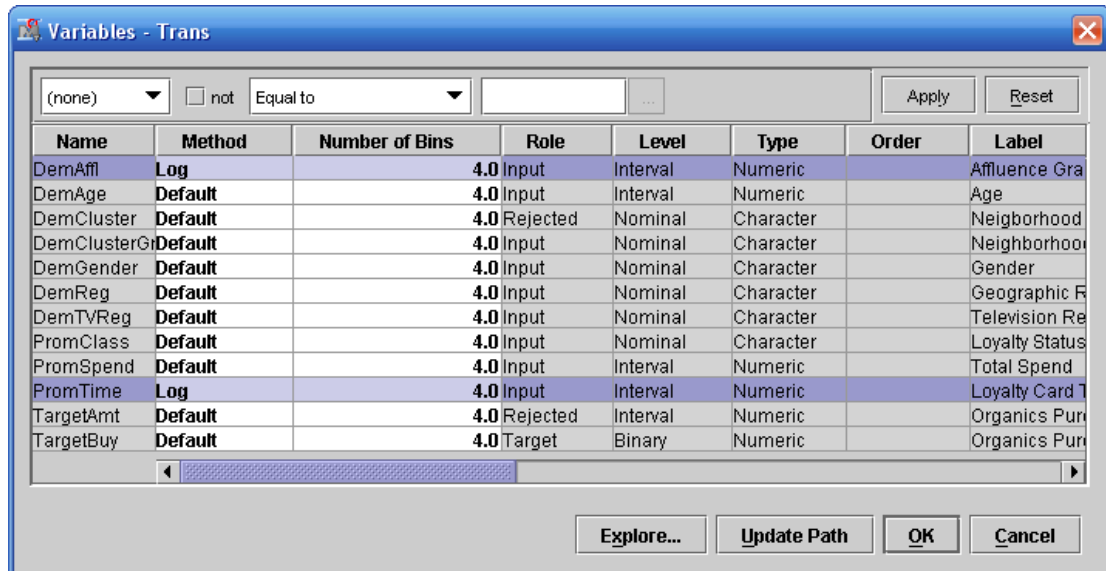


Both Card Tenure and Affluence Grade have moderately skewed distributions. Applying a log transformation to these inputs might improve the model fit.

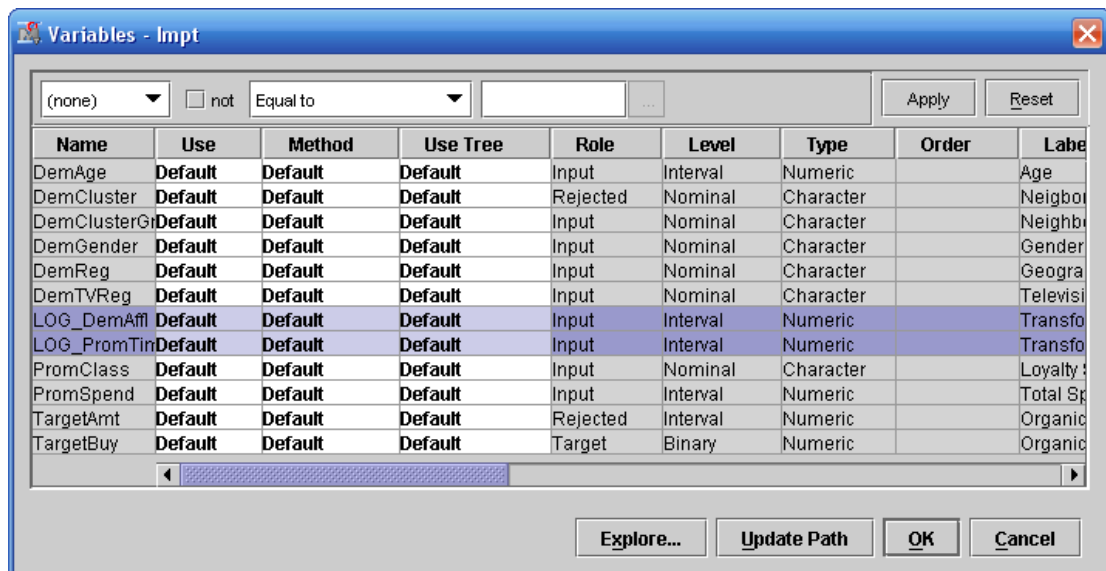
- h. Disconnect the **Impute** node from the **Data Partition** node.
- i. Add a **Transform Variables** node to the diagram and connect it to the **Data Partition** node.
- j. Connect the **Transform Variables** node to the **Impute** node.



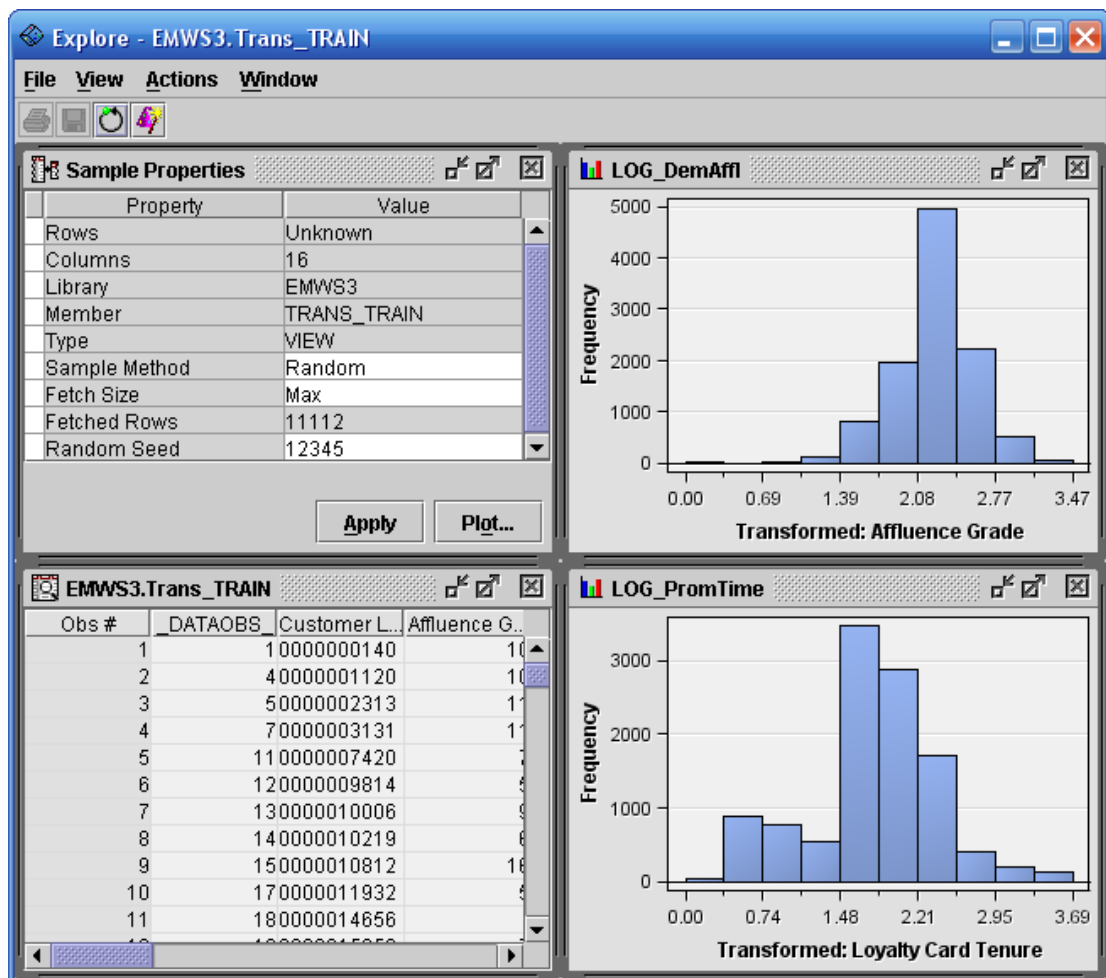
- k. Apply a log transformation to the **DemAffl** and **PromTime** inputs.
- 1) Open the Variables window of the Transform Variables node.
 - 2) Select **Method** ⇒ **Log** for the **DemAffl** and **PromTime** inputs.



- 3) Select **OK** to close the Variables window.
- l. Run the **Transform Variables** node. Explore the exported training data. Did the transformations result in less skewed distributions?
- 1) The easiest way to explore the created inputs is to open the Variables window in the subsequent Impute node. Make sure that you update the Impute node before opening its Variables window.



- 2) With the **LOG_DemAffl** and **LOG_PromTime** inputs selected, select **Explore...**.



The distributions are nicely symmetric.

- m. Rerun the **Regression** node. Do the selected variables change? How about the validation ASE?

- 1) Go to line 664 of the Output window.

The selected model, based on the CHOOSE=VERROR criterion, is the model trained in Step 6. It consists of the following effects:

Intercept IMP_DemAge IMP_DemGender IMP_LOG_DemAffl M_DemAge M_DemGender M_LOG_DemAffl

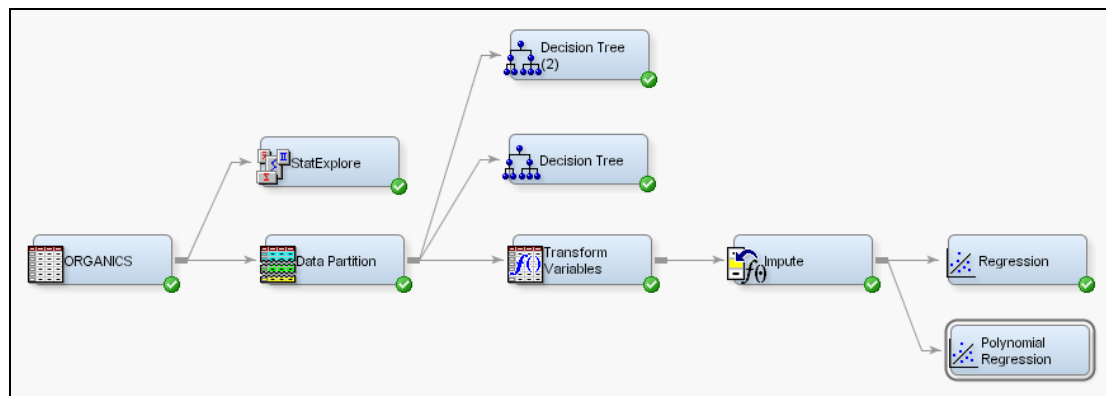
- 2) **IMP_LOG_DemAffl** and **M_LOG_DemAffl** replace **IMP _DemAffl** and **M_ _DemAffl**, respectively.

- 3) Apparently the log transformation actually reduced the validation ASE slightly.

Target	Fit Statistics	Statistics Label	Train	Validation	T
TargetBuy	_AIC_	Akaike's Inf...	9758.609	.	
TargetBuy	_ASE_	Average Sq...	0.139545	0.138204	
TargetBuy	_AVERR_	Average Err...	0.438382	0.43599	
TargetBuy	_DFE_	Degrees of ...	11104	.	
TargetBuy	_DFM_	Model Degr...	8	.	
TargetBuy	_DFT_	Total Degre...	11112	.	
TargetBuy	_DIV_	Divisor for A...	22224	22222	
TargetBuy	_ERR_	Error Functi...	9742.609	9688.581	
TargetBuy	_FPE_	Final Predic...	0.139746	.	
TargetBuy	_MAX_	Maximum A...	0.992317	0.994405	
TargetBuy	_MSE_	Mean Squa...	0.139646	0.138204	
TargetBuy	_NOBS_	Sum of Fre...	11112	11111	

- n. Create a full second-degree polynomial model. How does the validation average squared error for the polynomial model compare to the original model?

- 1) Add another Regression node to the diagram and rename it **Polynomial Regression**.



- 2) Make the indicated changes to the Polynomial Regression Properties panel.

Train	
Variables	...
<input checked="" type="checkbox"/> Equation	
Main Effects	Yes
Two-Factor Interactions	Yes
Polynomial Terms	Yes
Polynomial Degree	2
User Terms	No
Term Editor	...
<input checked="" type="checkbox"/> Class Targets	
Regression Type	Logistic Regression
Link Function	Logit
<input checked="" type="checkbox"/> Model Options	
Suppress Intercept	No
Input Coding	Deviation
<input checked="" type="checkbox"/> Model Selection	
Selection Model	Stepwise
Selection Criterion	Validation Error
Use Selection Defaults	Yes
Selection Options	...

- 3) Go to line 1598

The selected model, based on the CHOOSE=VERROR criterion, is the model trained in Step 7. It consists of the following effects:

Intercept IMP_DemAge IMP_DemGender IMP_LOG_DemAff1 M_DemAge
M_DemGender*M_LOG_DemAff1 IMP_DemAge*IMP_DemAge IMP_LOG_DemAff1*IMP_LOG_DemAff1

- 4) The Polynomial Regression node adds additional interaction terms.
5) Examine the Fit Statistics window.

Fit Statistics					
Target	Fit Statistics	Statistics Label	Train	Validation	T
TargetBuy	_AIC_	Akaike's Inf...	9529.938		
TargetBuy	_ASE_	Average Sq...	0.136407	0.134038	
TargetBuy	_AVERR_	Average Err...	0.428003	0.421824	
TargetBuy	_DFE_	Degrees of ...	11103		
TargetBuy	_DFM_	Model Degr...	9		
TargetBuy	_DFT_	Total Degre...	11112		
TargetBuy	_DIV_	Divisor for A...	22224	22222	
TargetBuy	_ERR_	Error Functi...	9511.938	9373.784	
TargetBuy	_FPE_	Final Predic...	0.136628		
TargetBuy	_MAX_	Maximum A...	0.985718	0.986233	

The additional terms reduce the validation ASE slightly.

Solutions to Student Activities (Polls/Quizzes)

4.01 Multiple Choice Poll – Correct Answer

What is the logistic regression prediction for the indicated point?

- a. -0.243
- b. 0.56
- c. yellow
- d. It depends ...

$$\text{logit}(\hat{p}) = -0.81 + 0.92x_1 + 1.11x_2$$

$$\hat{p} = \frac{1}{1 + e^{-\text{logit}(\hat{p})}}$$

