Paper 153-2011

Ratemaking Using SAS® Enterprise Miner™: An Application Study

Billie Anderson, SAS Institute Inc., Cary, NC

ABSTRACT

Insurance companies gain competitive advantage by offering better rates and services to attract and retain the best customers. Generalized linear models (GLMs) have become popular and proven techniques for ratemaking and actuarial work over the past decade. Claim frequency is typically modeled using a Poisson distribution; severity is modeled using a gamma distribution; and pure premium is modeled using a Tweedie distribution. Insurance providers use these models to accurately estimate losses and set the most competitive rates accordingly. This paper ties the theory of ratemaking using GLMs to case studies that use real insurance data and shows you how to use SAS® Enterprise Miner™ to model claim frequency, severity, and pure premiums.

INTRODUCTION

Ratemaking is the determination of what rates (premiums) to charge for insurance. Traditional ratemaking methods are not statistically sophisticated. Many lines of business are analyzed using one-way analysis. A one-way analysis summarizes insurance statistics such as a loss ratio for each predictor variable without taking into account the effect of the other variables. One-way analyses also do not take into account any interactions that might exist among the predictor variables. For example, premiums can differ between men and women by levels of age (Anderson, et al. 2007).

Over the last decade, more sophisticated methods such as generalized linear models (GLMs) have become more popular as a ratemaking methodology among actuaries. The rating case studies in this paper illustrate how you can use data mining technology and the new Ratemaking node in SAS Enterprise Miner to build predictive ratemaking models. Nodes in SAS Enterprise Miner are visual building blocks for creating a data mining analysis. For example, there are nodes that split the training data set into training, validation, and test data sets; nodes that sample the data; and nodes that perform modeling operations such as regression, decision trees, and neural networks. Each node has a corresponding properties panel that enables you to review or change default settings.

RATEMAKING NODE IN SAS ENTERPRISE MINER 7.1

Figure 1 shows the Ratemaking node located on the **Applications** tab and its properties panel. The Ratemaking node and the **Applications** tab are new in SAS Enterprise Miner 7.1.

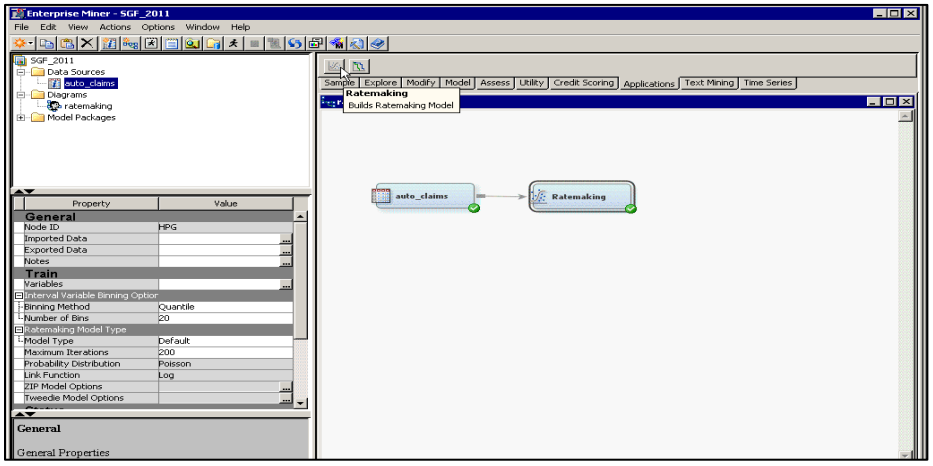


Figure 1: Ratemaking Node

The Ratemaking node uses a fast, highly scalable procedure that builds GLMs. This procedure requires that all rating variables be binned. This requirement is one of the core ingredients of ratemaking.

BINNING METHODOLOGY IN THE RATEMAKING NODE

Identifying the number of bins of a rating variable implies the number of prices in the rating structure and also the granularity of risk segmentation. Setting the numbers and members of each level or bin is critical. Sometimes you might want to bin the variables on an arbitrary or judgmental basis. Alternatively, you can use SAS Enterprise Miner to optimally bin the variables in either of the following ways:

- Use the Decision Tree modeling node.
- Use the **Optimal Binning** option in the Interval Inputs property in the Transform Variables node.

As shown in Figure 2, the Ratemaking node provides two options for creating the bins for the rating variables: quantile and bucket. Quantile binning generates bins using ranked quantiles. Bucket binning generates groups by dividing the data into evenly spaced intervals based on the maximum and minimum values. The number of bins can range from 5 to 50; the default number of bins is 20.

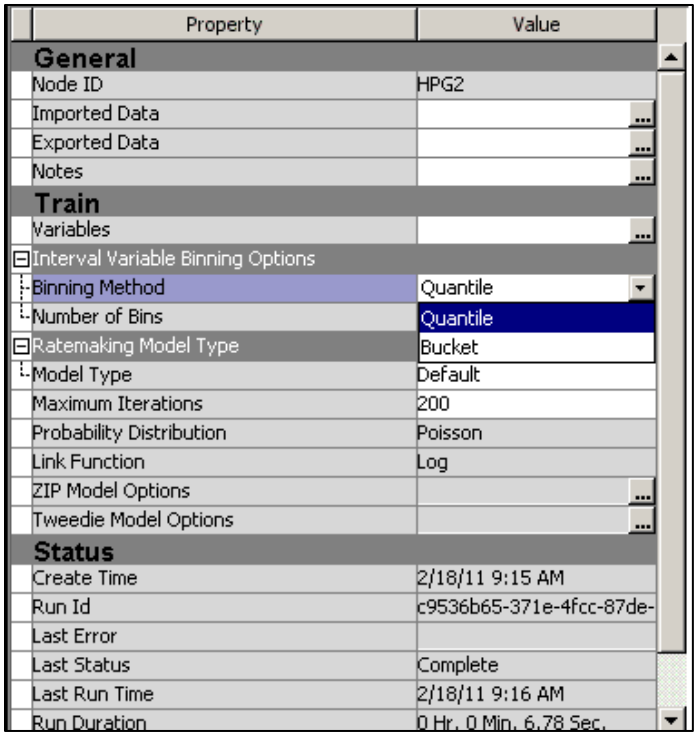


Figure 2: Binning Options in the Ratemaking Node Properties Panel

MODEL TYPES SUPPORTED BY THE RATEMAKING NODE

The most common types of ratemaking models in the insurance industry are frequency, severity, and pure premium models. Frequency models predict how often claims are made, and severity models predict claim amounts. The term pure premium is unique to insurance; it is the portion of the company's expected cost that is "purely" attributed to loss (Werner and Modlin 2009). Pure premium does not include the general expense of doing business, such as overhead and commissions. The Tweedie distribution is used to model pure premium.

The next section shows you the steps for creating a ratemaking project for a case study that uses the Tweedie distribution and homeowner's insurance data to model pure premium. The Ratemaking node and the results produced by it are described in detail. Subsequent sections use real-world auto claims data to describe how the Ratemaking node implements frequency and severity models.

USING SAS ENTERPRISE MINER TO DEVELOP A RATEMAKING PROJECT

This ratemaking case study illustrates how you can use data mining technology to estimate pure premium.

The data set consists of homeowner's insurance policy data with 60 rating variables. Figure 3 shows the distribution of the target variable, which is pure premium. This distribution is typical of pure premium: there is a large spike at 0, and then a wide range of claim amounts. In this case study the Ratemaking node uses a Tweedie distribution to model pure premium for this homeowner's insurance data.

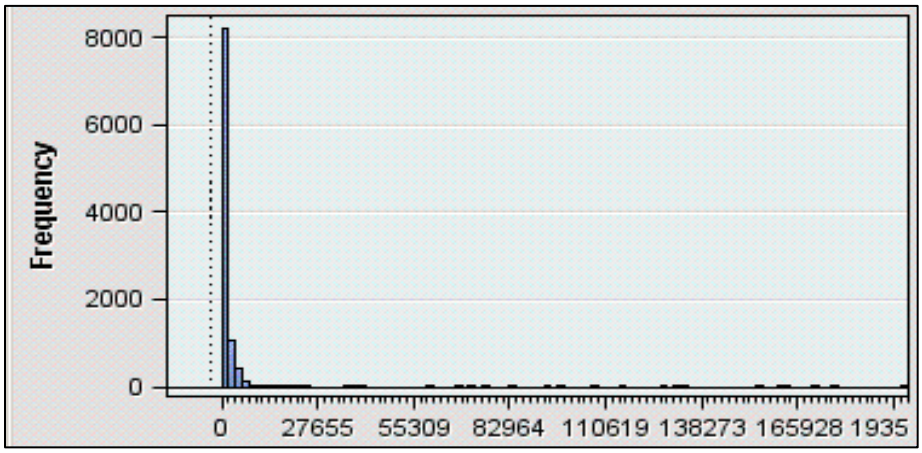
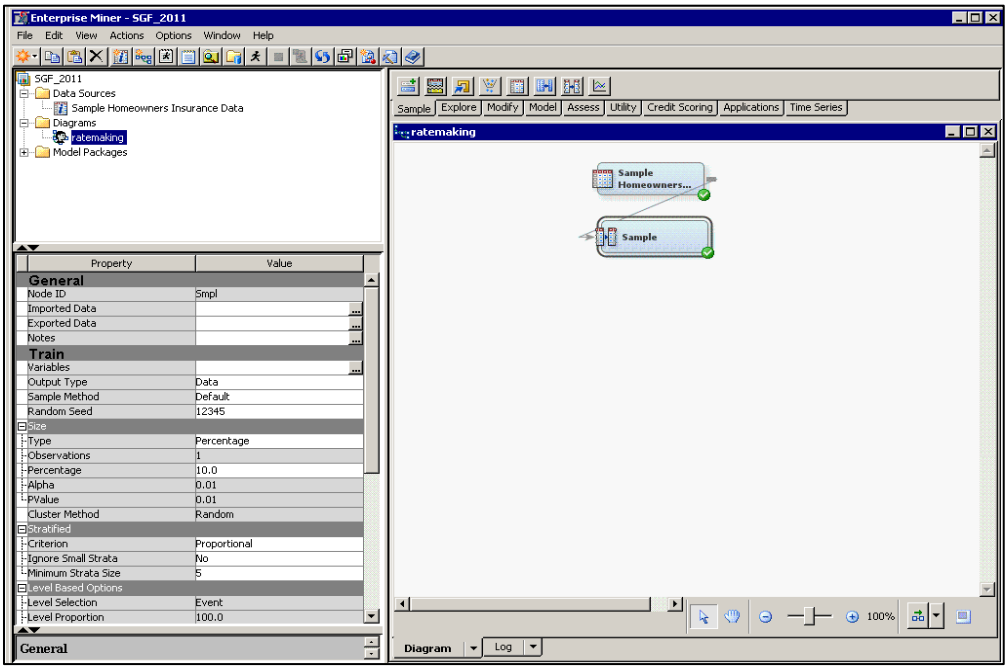


Figure 3: Distribution of Pure Premium

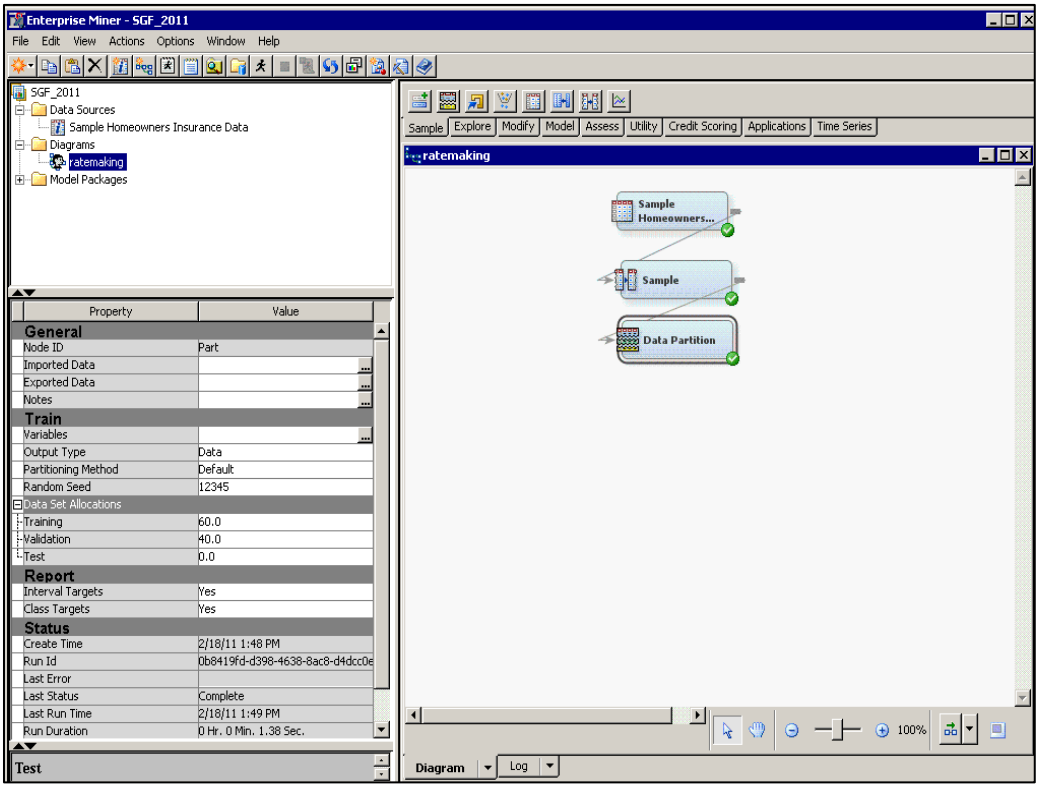
BUILDING THE RATEMAKING MODEL IN SAS ENTERPRISE MINER

To build a pure premium model:

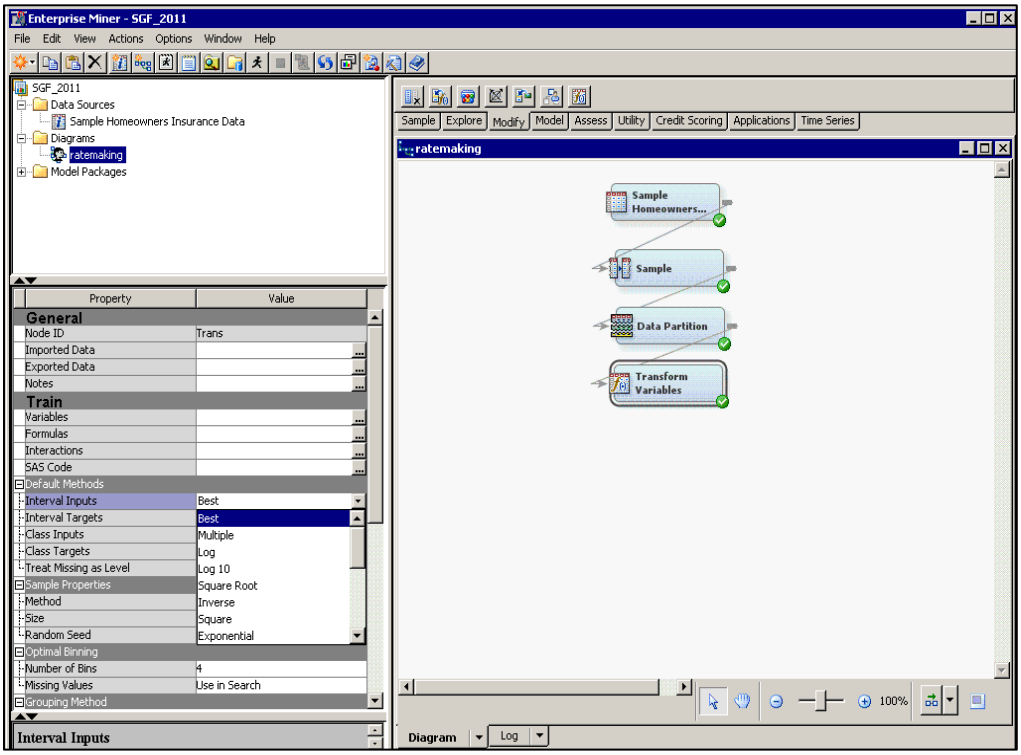
1. The first step of the data mining process is to sample the data. To speed things up, you can use the Sample node to develop your model by using a sample rather than the entire claims table. After the model is developed, you can use the entire claims table to score in order to come up with the predicted pure premium for each policyholder. Available sampling methodologies include random, simple, systematic, cluster, and stratified sampling. For the purposes of this ratemaking model, a random sample is sufficient.



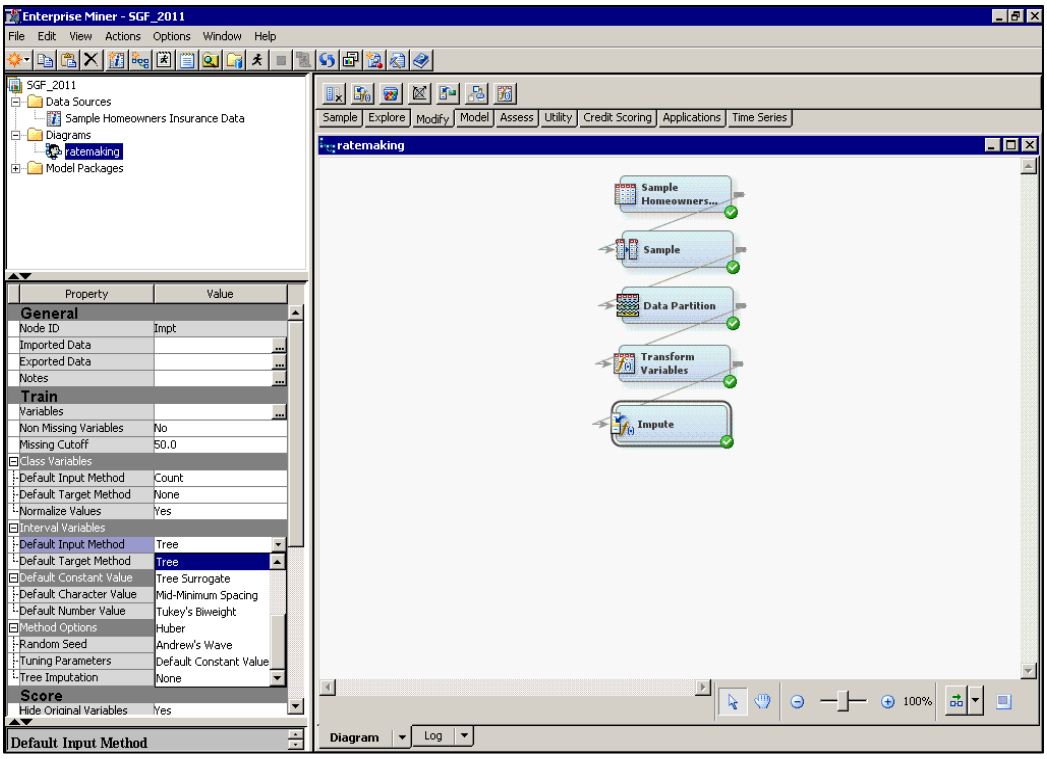
2. Add a Data Partition node to create a training partition and a validation (“hold-out”) partition from the sample. The Ratemaking node uses the training partition to develop the model by estimating the parameter estimates, and uses the validation portion of the data to determine the predictive accuracy of the Tweedie model. For this study, 60% of the data is used for training and 40% is used for validation.



3. Add a Transform Variables node to the analysis. Every data mining project faces the challenge of distributional problems such as skewness, kurtosis, and heteroscedasticity. Manning and Mullahy (2001) showed through simulation that these types of distributional problems can greatly affect the predicted probabilities of GLMs. You can use the Transform Variables node to transform any rating variables that suffer from any of these distributional problems. In the properties panel, select **Best** from the **Interval Inputs** list. This type of transformation examines several commonly used transformations (log, square root, inverse, and so on) and selects the "best" transformation based on an R-square statistic.

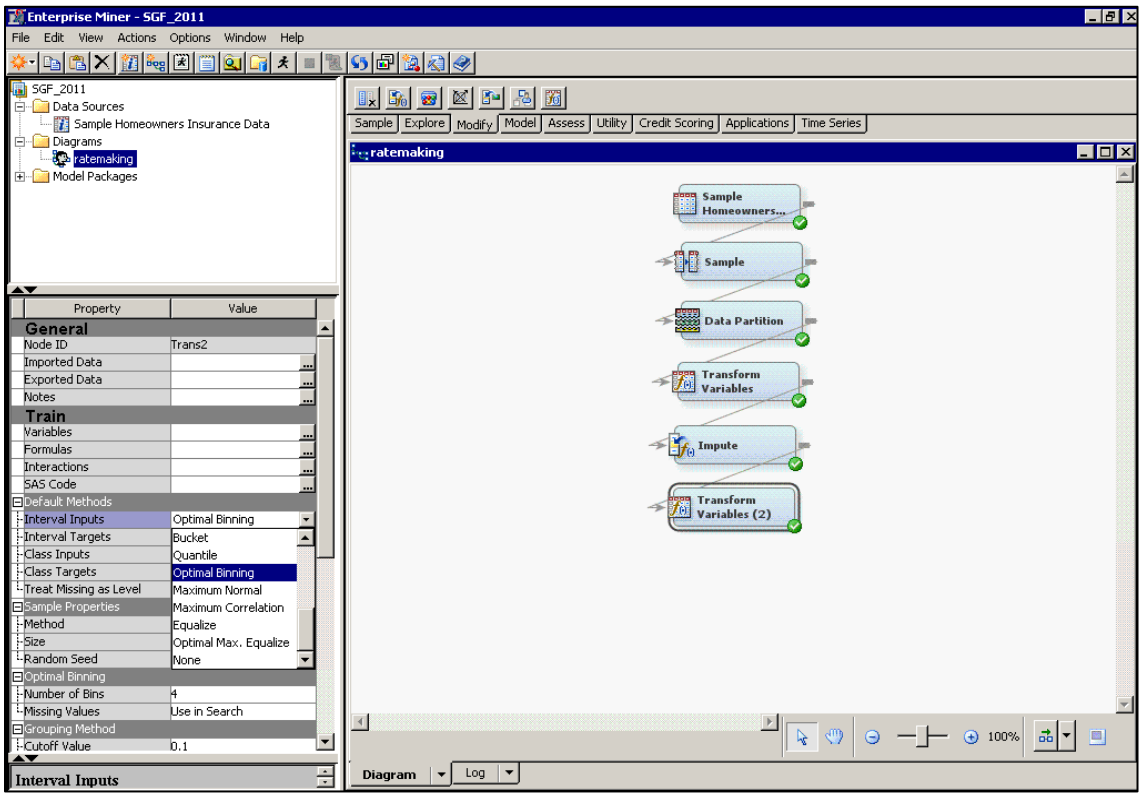


4. Add an Impute node to handle any missing values in the data set. Huge amounts of missing data are common in any data mining project. Select **Tree** from the **Default Input Method** list in the properties panel to use a decision tree to estimate missing values. Although the decision tree imputation method is the most resource-intensive, it represents the most sophisticated imputation method.

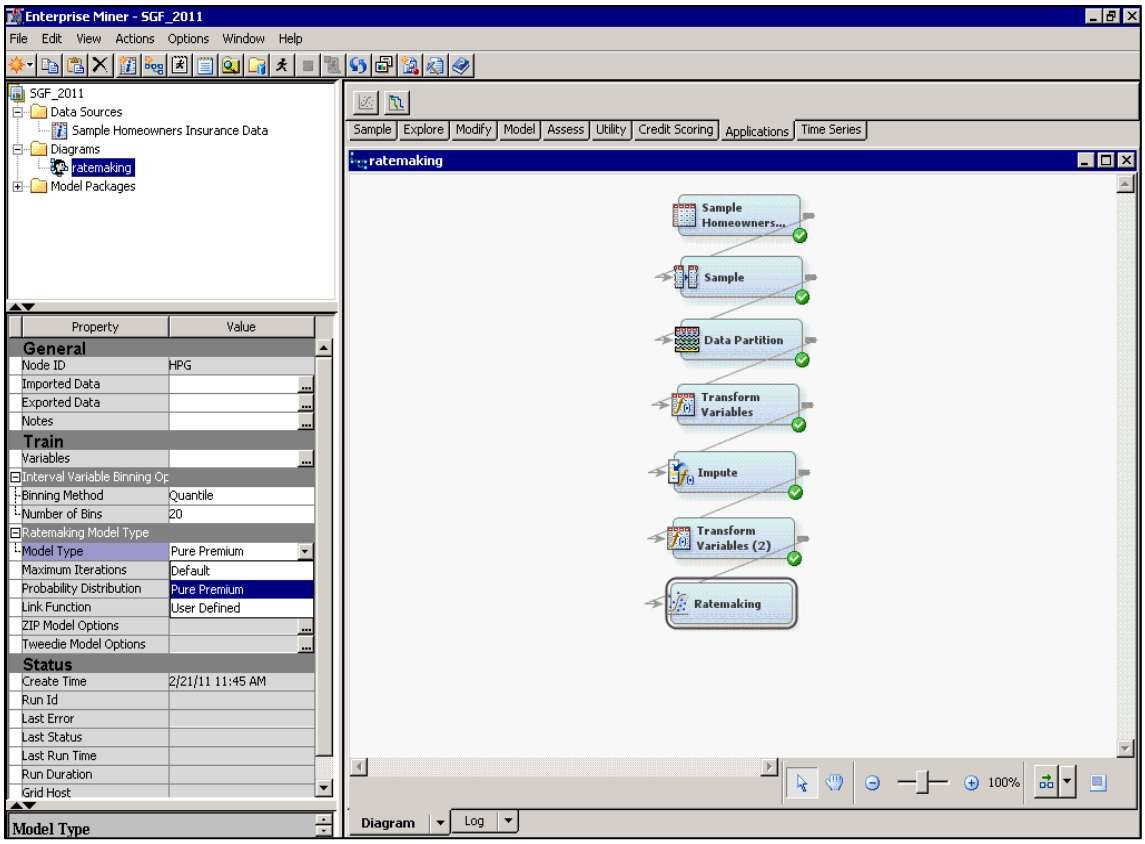


5. Add a second Transform Variables node to the analysis, and select **Optimal Binning** from the **Interval Inputs** list on the properties panel. This option uses a decision tree algorithm to bin the interval variables.

   A common question is how to bin an interval variable such as age. This is particularly important in the insurance field because each bin of the interval variable represents a part of the rating structure. Typically the interval variable is binned into a finite number of categories, policyholders are fixed into the appropriate age category, and their rate is based on that category.



6. Add the Ratemaking node and select **Pure Premium** for the **Model Type** in the properties panel. This setting uses the Tweedie distribution with the automatic optimization method that is described in the section “Pure Premium Model” on page 14.



7. Run the Ratemaking node by right-clicking on the node and selecting **Run**. When the run has completed, click **Results**. The Results window appears (see Figure 4).

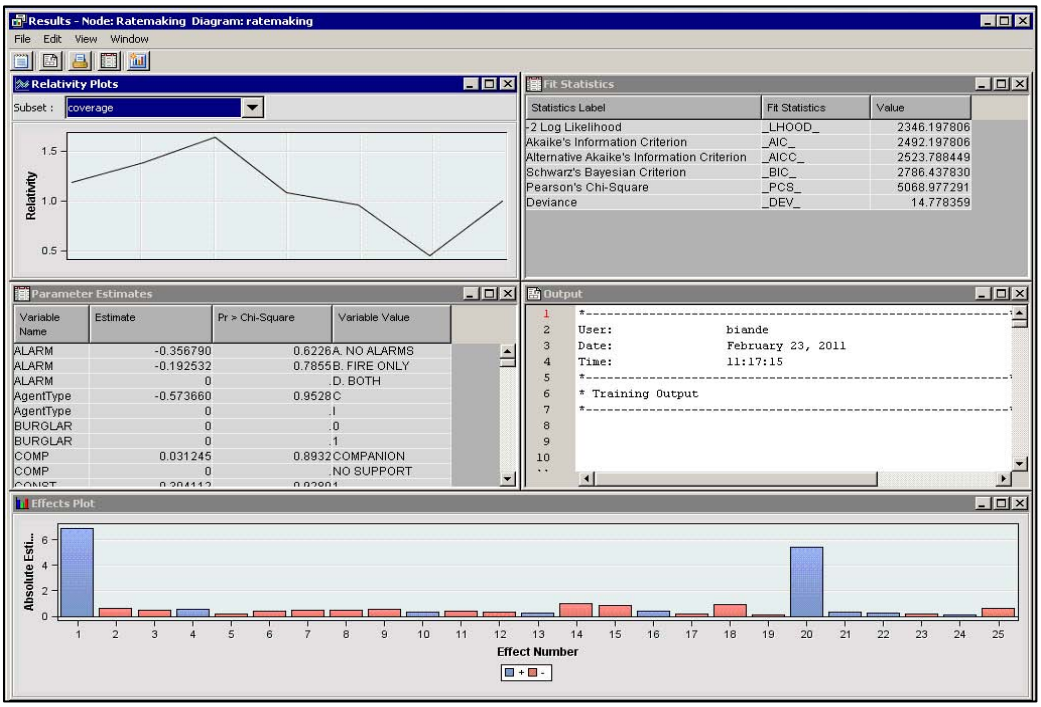


Figure 4: Results from the Ratemaking Node

The results such as the Parameter Estimates, Output, and Effects Plot panes are similar to the results produced by a traditional modeling node in SAS Enterprise Miner. In addition, the Ratemaking node provides the Relativity Plots pane (upper left). The relativities are the exponential of the model coefficients; they are a key factor for setting rates for many insurance lines of business.

For every log link model that is built in the Ratemaking node, the node produces a relativity plot for each variable. Figure 5 shows the relativity plot for the **coverage** rating variable. If you click a bin of the variable, you see the variable value along with the associated relativity.

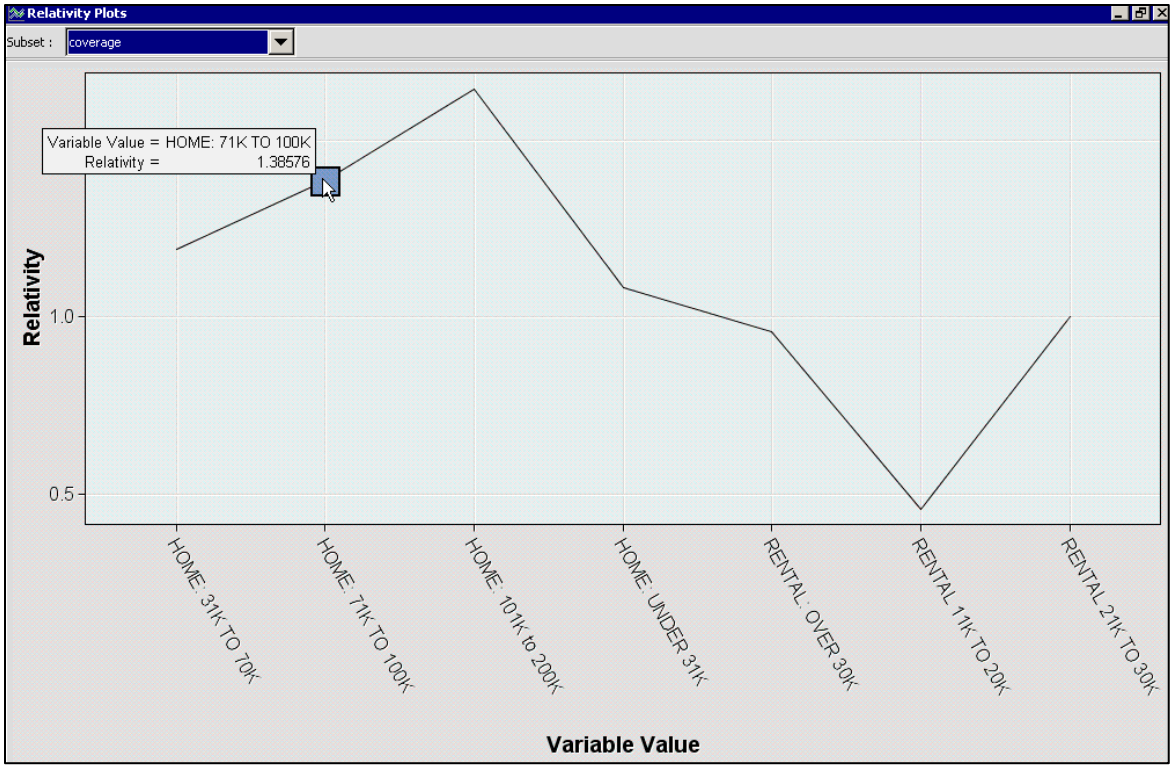


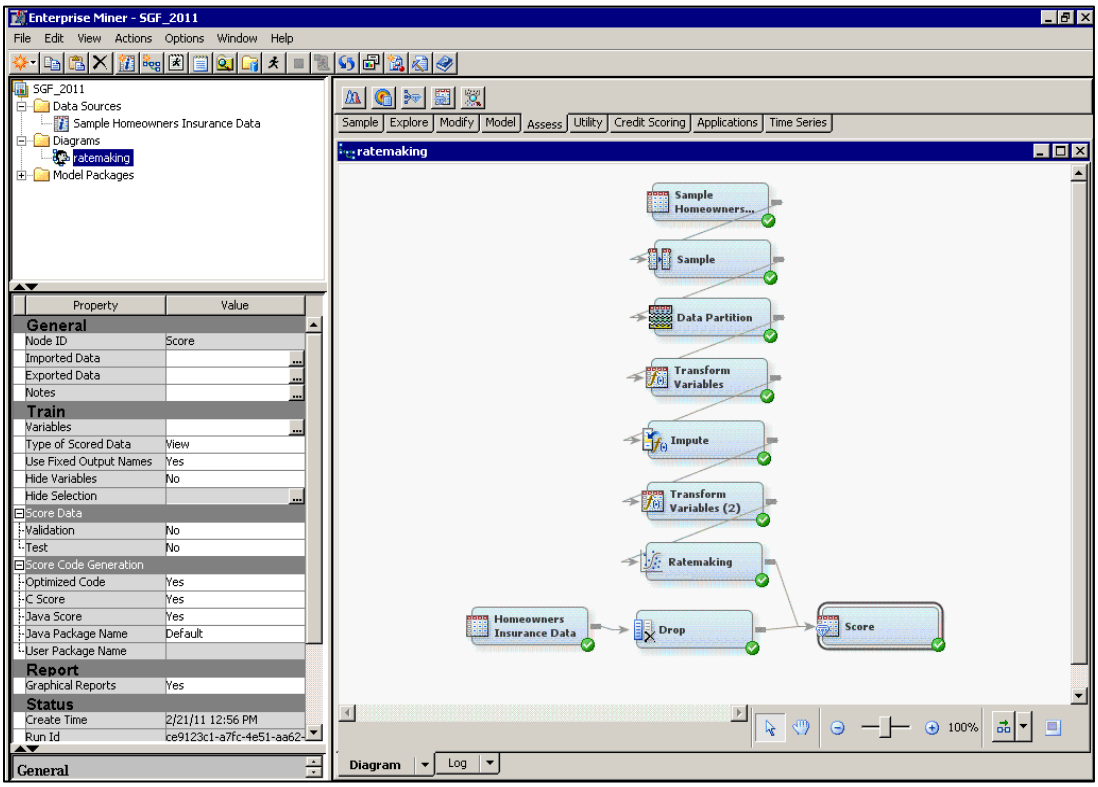
Figure 5: Relativity Plot for the Coverage Rating Variable

The rating variable in Figure 5 is the amount of coverage for the homeowners; it has seven discrete levels. The relativity is plotted on the Y axis, and the levels of the rating variable are plotted on the X axis. The base level, to which all other parameter estimates are expressed relative, is rental property with values in the range $20,000–$30,000. The GLM model gives the statistical effect of the coverage amount on pure premium, all other rating variables being considered. For example, the GLM indicates that a house with a value in the range $71,000–$100,000 has a 39% less indicated pure premium than rental property with a value in the range $21,000–$30,000.

The Fit Statistics table (upper right in Figure 4) is specific to a GLM and is discussed in more detail in the section “Frequency Model” on page 15.

8. Score your model by adding Drop and Score nodes to the analysis.

   The entire homeowner's claims data set is used to obtain a predicted pure premium amount for every policyholder in the data set. The Drop node drops the target variable so that the Score node can apply the score code to the entire claims data set.



The Ratemaking node produces scoring code that computes the linear predictor values needed for the model predictions. The inverse function that produces the model predictions is given at the end of the scoring code. Since this example uses the Tweedie distribution, the link function is log and the inverse function is the exponential. Following is a snapshot of the scoring code that the Ratemaking node produces:

```
/*** Compute Linear Predictor***/
_xbeta=0;
/***Effect:Intercept***/
_xbeta=_xbeta+-8.704146002;
/****Effect:PolicyAge***/
if PolicyAge < 1.79 then do;
_xbeta=_xbeta+(0);
end;
if 1.79 <= PolicyAge AND PolicyAge < 1.95 then do;
_xbeta=_xbeta+(-6.191690779);
end;
……..
/***Scoring Formula***/
_P=exp(_xbeta);
```

DRAWING STATISTICAL INFERENCES FOR THE RATING LEVELS

In a good ratemaking model, each of the rating levels corresponds to a unique distribution and each unique distribution can be fitted with a specific theoretical probability distribution. There are many reasons for using the theoretical distribution rather than the empirical distribution. For example, a theoretical distribution enables you to calculate the probability that losses for a given portfolio will exceed a given amount, enabling you to decide whether certain policyholders should be re-insured.

As a simple example, suppose one of the rating levels includes a policy age of between 12.5 and 13 years for a home that was built after 2007 and has both a fire and burglary alarm. You can subset these observations out of the exported scored data set and use the UNIVARIATE procedure to fit the empirical distribution with a probability that includes some of the most likely candidate distributions, such as exponential, lognormal, and Weibull. Figure 6 shows the empirical distribution for the hypothetical rating level with the exponential, lognormal, and Weibull distributions estimated and superimposed on the empirical distribution.

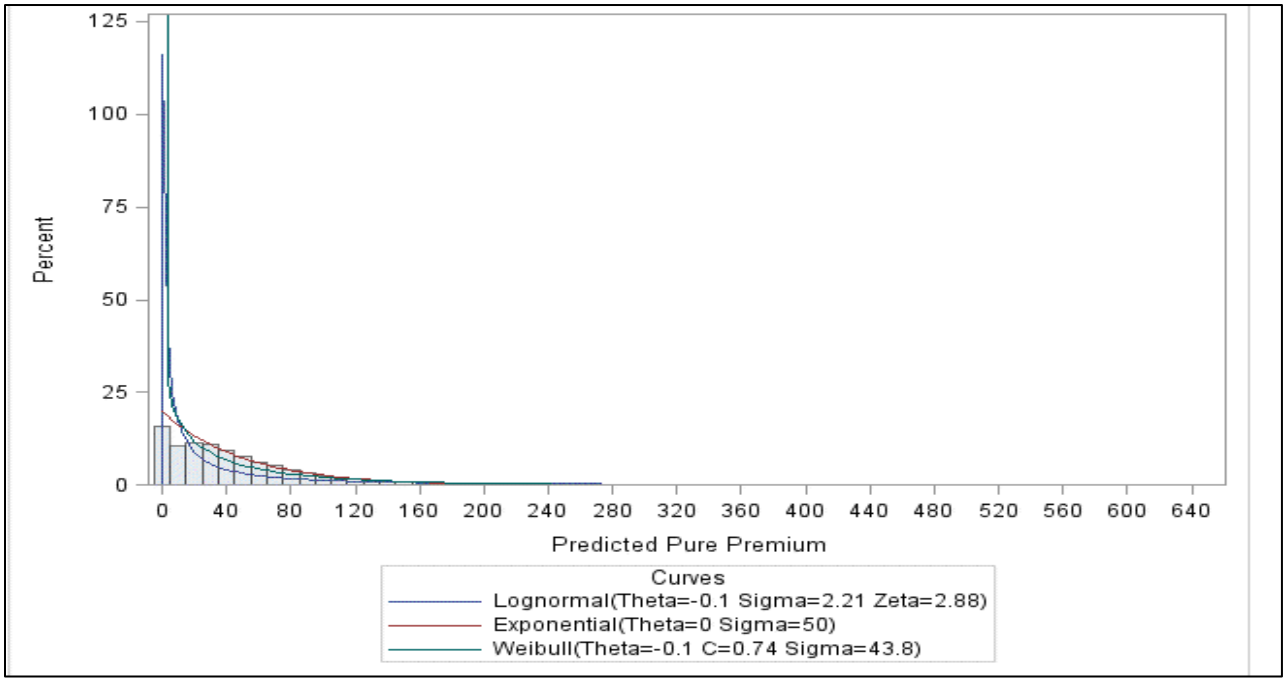


Figure 6: Exponential, Lognormal, and Weibull Distributions Superimposed on the Empirical Rating Level

Figure 7 shows the goodness-of-fit statistics for these distributions. Since the *p*-values for all of the statistics are less than 0.05, you can conclude that all three of these distributions are suitable for this hypothetical rating level. You can now use any of these distributions to draw inferences.

Goodness-of-Fit Tests for Exponential Distribution				
Test	Statistic		p Value	
Kolmogorov-Smirnov	D	0.10791	Pr > D	<0.001
Cramer-von Mises	W-Sq	459.52143	Pr > W-Sq	<0.001
Anderson-Darling	A-Sq	2500.44246	Pr > A-Sq	<0.001

Goodness-of-Fit Tests for Lognormal Distribution				
Test	Statistic		p Value	
Kolmogorov-Smirnov	D	0.20421	Pr > D	<0.010
Cramer-von Mises	W-Sq	1474.82116	Pr > W-Sq	<0.005
Anderson-Darling	A-Sq	9211.38755	Pr > A-Sq	<0.005

Goodness-of-Fit Tests for Exponential Distribution				
Test	Statistic		p Value	
Kolmogorov-Smirnov	D	0.10791	Pr > D	<0.001
Cramer-von Mises	W-Sq	459.52143	Pr > W-Sq	<0.001
Anderson-Darling	A-Sq	2500.44246	Pr > A-Sq	<0.001

Figure 7: Goodness-of-Fit Statistics for the Exponential, Lognormal, and Weibull Distributions

RATEMAKING MODELS

This section shows specific features of the Ratemaking node for the various models it supports.

PURE PREMIUM MODEL

The Ratemaking node has a Tweedie distribution available for modeling pure premium.

Following Kaas (2005), the Tweedie probability density function is of the form

$f_x(x;\theta,\phi) = \exp(\frac{x\theta - b(\theta)}{\phi} + c(x;\phi))$, and the variance is of the form μ^p.

Consider the densities in the preceding probability density function for independent observations $x_1, x_2, \ldots x_n$.

Assume that there are parameters $\beta_1, \beta_2, \ldots$ that lead to means μ_i and associated θ_i through the relations

$\mu = g^{-1}(\mathbf{X}\beta)$ and $\mu(\theta) = b'(\theta)$.

The Ratemaking node offers four choices for the Tweedie optimization process as shown in Figure 8.

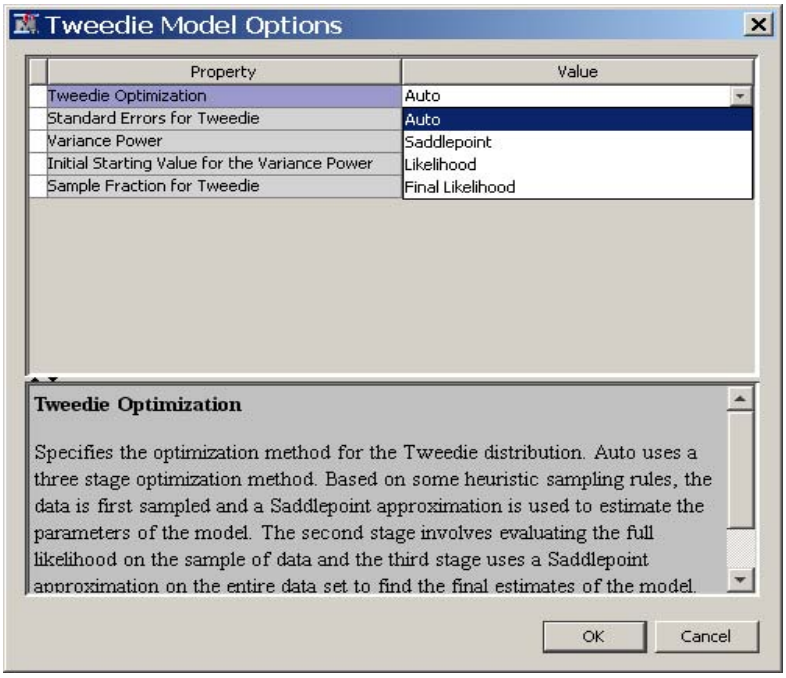


Figure 8: Tweedie Optimization Methods

The Auto, Saddlepoint, and Final Likelihood methods use an extended quasi-likelihood function (Nelder and Pregibon 1987), which enables you to specify only the mean and variance. Using an extended quasi-likelihood function to estimate the parameters has been shown to be approximately equal to using the full likelihood to estimate the parameters (Kaas 2005). The Likelihood method uses the full likelihood function to estimate the parameters of the model.

The following lists the steps for each of the methods:

- **Auto**

  1. Based on some heuristic sampling rules, the data are first sampled and an extended quasi-likelihood is used to estimate β, ϕ, and p.
  2. Using the estimated β, ϕ, and p from the first step as starting parameter values, the full likelihood is

evaluated on the sample of data. Once ϕ and p are estimated at this stage, they are fixed.

3. ϕ and p are fixed from step 2, and the extended quasi-likelihood is used on the entire data set to estimate β.

- **Saddlepoint**

1. The data are sampled, and the extended quasi-likelihood is used to estimate β, ϕ, and p (these parameter values are used as starting values for step 2).
2. The extended quasi-likelihood is used on the entire data set to estimate ϕ and p.

- **Likelihood**

1. The data are sampled, and the full likelihood is used to estimate β, ϕ, and p (these parameter values are used as starting values for step 2).
2. The full likelihood is used on the entire data set to estimate β, ϕ, and p.

- **Final Likelihood**

1. The data are first sampled, and the extended quasi-likelihood is used to estimate β, ϕ, and p.
2. Using the estimated β, ϕ, and p from step 1 as starting parameter values, the full likelihood is evaluated on the sample of data.
3. Using the estimates of β, ϕ, and p from step 2 as the starting parameters, the extended quasi-likelihood is used on the entire data set to estimate β.
4. Using starting values of β, ϕ, and p from step 3, the full likelihood is evaluated on the entire data set to estimate β, ϕ, and p.

FREQUENCY MODEL

This section shows you how to use the Ratemaking node to model automobile insurance claims data. Figure 9 shows the distribution of the number of claims.

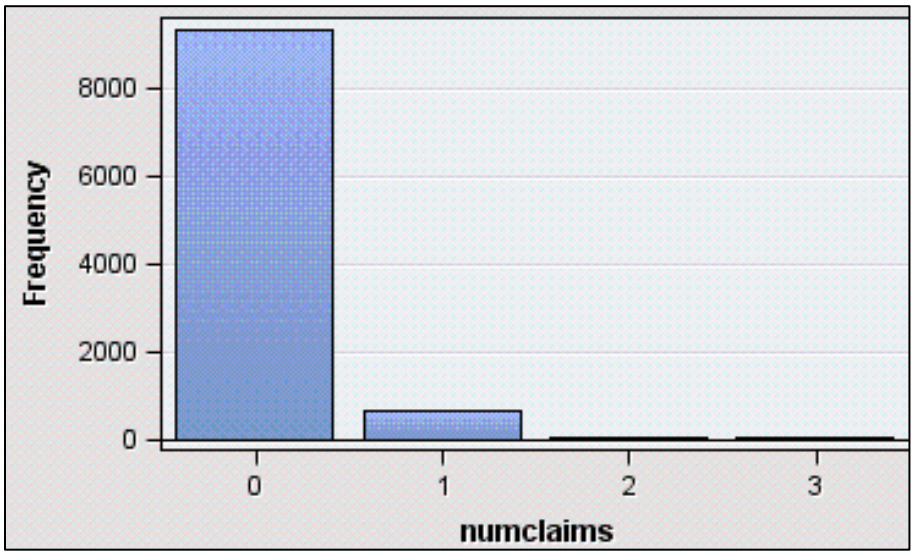


Figure 9: Claim Count for an Auto Insurance Data Set

When you add a Ratemaking node for this type of data, the node automatically builds a frequency model by default. Figure 10 highlights the properties panel of the Ratemaking node and shows the default settings.

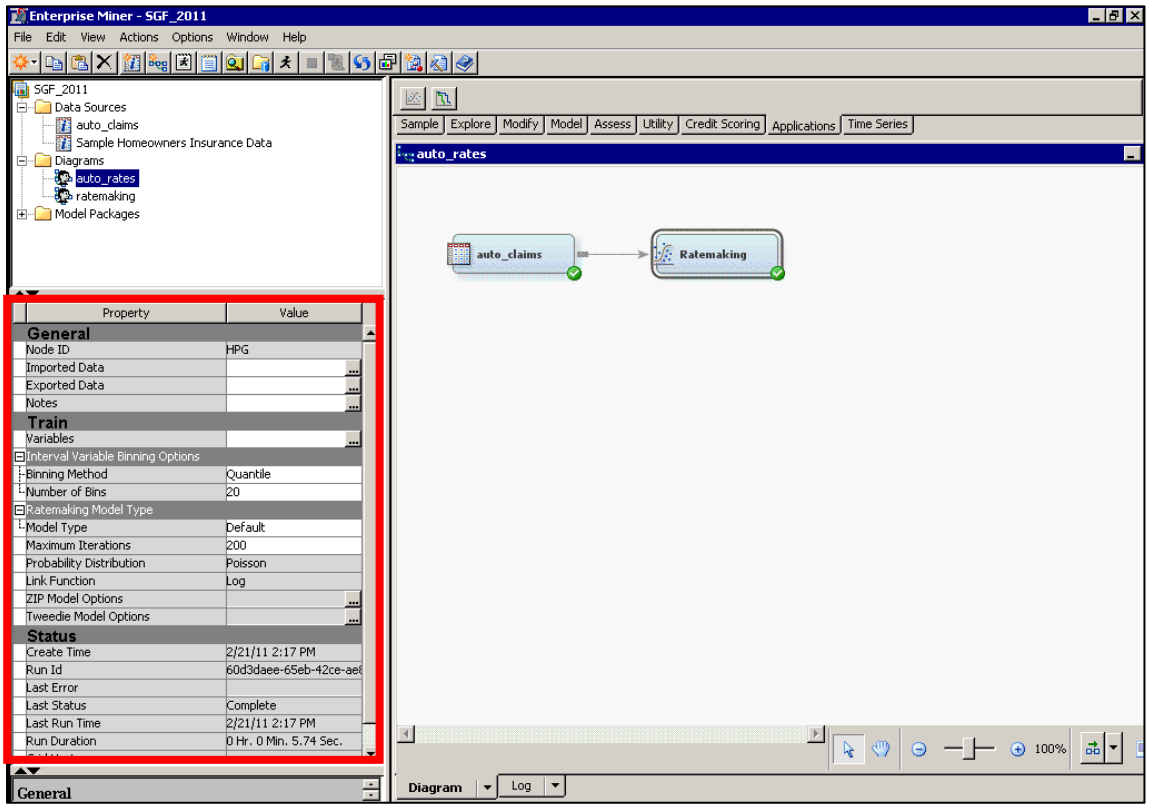


Figure 10: Properties Panel of the Ratemaking Node

When the Model Type property is set to **Default**, the Ratemaking node examines the level of the target variable and automatically sets the distribution and link function. For example, if the level of the target is nominal, a Poisson distribution with a log link function is used; if the level of the target is interval, a gamma distribution with a log link function is used; if the level of the target is binary, a binary distribution with a logit link function is used.

Figure 11 shows the results from the Ratemaking node after the frequency model has been developed.

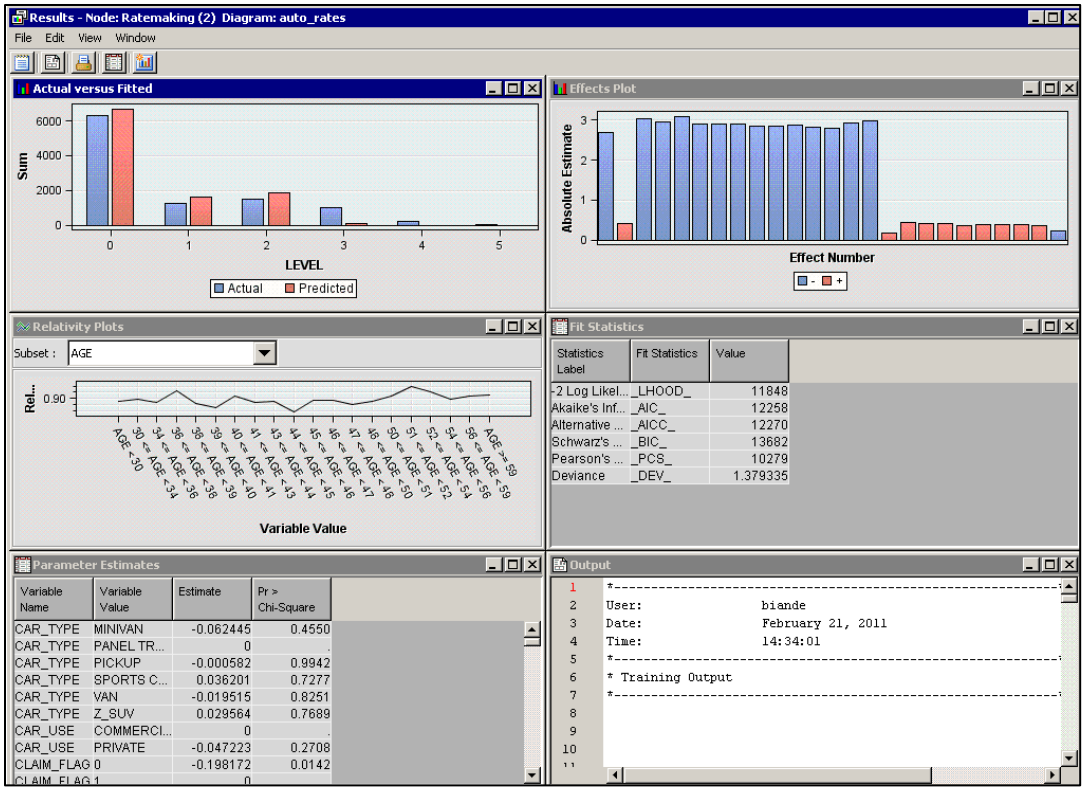


Figure 11: Results for a Frequency Model

For frequency models, the Ratemaking node exports the predicted frequency claim count. The predicted frequency claim count is a rational number, and the raw frequency claim count is an integer. So, the integer function is used to round the predicted frequency claim count, and the actual versus predicted frequency claim counts are displayed in the Actual versus Fitted plot (upper left Figure 11) to see how well the model is predicting the claim counts. This plot indicates how well the frequency model is performing. This plot is displayed every time the target variable is nominal or ordinal and numeric.

The fit statistics that are specific to a GLM are shown in the Fit Statistics table (center right in Figure 12). The value of the **Deviance** statistic for this frequency model indicates that there might be an overdispersion problem. Deviance is the Pearson chi-square statistic divided by the degrees of freedom from the model. If the value of the **Deviance** statistic is near 1, then the distribution might not be overdispersed and Poisson might be appropriate; if it is greater than 1, then the distribution is overdispersed. In this example the value of the **Deviance** statistic is 1.38, indicating there is a problem with overdispersion.

Overdispersion occurs frequently with Poisson count models. In a Poisson distribution, the count mean is equal to the count variance. Real count data are often more spread out; that is, the variance count is larger than the mean count, which means that the data are overdispersed. A possible explanation of the overdispersion might be an important but missing important rating variable. The negative binomial distribution, based on the Poisson distribution, relaxes the assumption that the mean and variance are equal; therefore, it can deal with overdispersed data. In some count modeling situations, the negative binomial distribution might be better than the Poisson distribution.

Overdispersion might not be the only problem with claim count data. Often, count data contain ""certain zeros"; such count data can be hard to fit with Poisson or negative binomial distributions. One approach is to categorize the data into "certain zeros" and all other observations. A "certain zero" can occur for a variety of reasons. For example, there might be elderly policyholders in the insurance book of business who still own their cars but never drive. These policyholders are "certain" to never file an auto accident claim. Mixture models that can handle "certain zeros" might work better than Poisson or negative binomial models. One such mixture model that can handle "certain zeros" is the zero-inflated Poisson distribution.

A zero-inflated Poisson model fits two separate models and then combines them. First, a logit model is fit to the "certain zero" observations, predicting whether the observation is a zero or not. Then, a Poisson claim count model is fit to predict those observations that are not "certain zeros." Finally, the two models are combined.

To specify a zero-inflated Poisson model in the Ratemaking node, select **User Defined** as the **Model Type** and select **Zero-Inflated Poisson** as the **Probability Distribution** as shown in Figure 12.

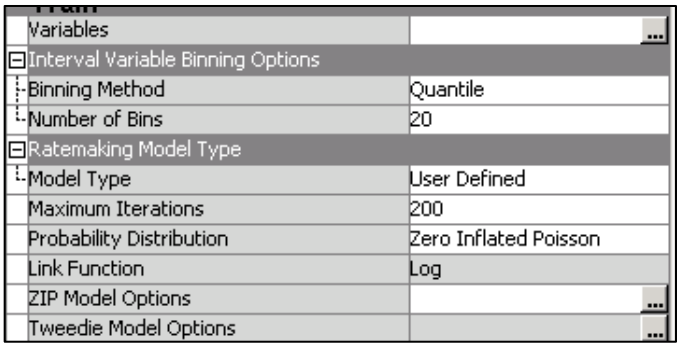


Figure 12: Specifying a Zero-Inflated Poisson Model in the Ratemaking Node

After you have specified a zero-inflated Poisson model, click the ellipsis (...) next to **ZIP Model Options** in the properties panel to review or edit additional zero-inflated Poisson model options. The **ZIP Model Options** dialog box appears, as shown in Figure 13.

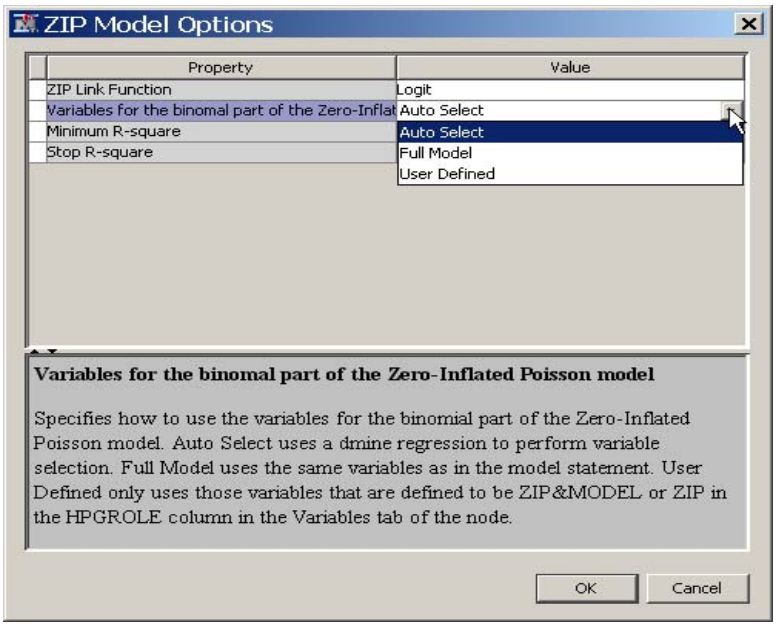


Figure 13: Additional Zero-Inflated Poisson Model Options

The first zero-inflated Poisson model option enables you to specify the link function for the logit part of the model. The second option enables you to specify how you want to deal with the rating variables for the logit part of the model. If the variables are handled using the **Auto Select** method, a stepwise regression is run using the DMINE procedure, and only the rating variables chosen by the stepwise procedure are used in the logit part of the model. The last two options, **Minimum R-square** and **Stop R-square**, correspond to the entry and exit significance levels for the rating variables.

Instead of choosing **Auto Select** for handling the rating variables for the logit part of the model, you can specify **Full Model** or **User Defined**. Both of these options require you to specify values for high-performance generalized linear modeling in a column named HPGRole. To access the HPGRole column, click the ellipsis next to **Variables** in the properties panel. The Variables-HPG dialog box appears, as shown in Figure 15. All possible values for the HPGRole column are shown in Figure 14.

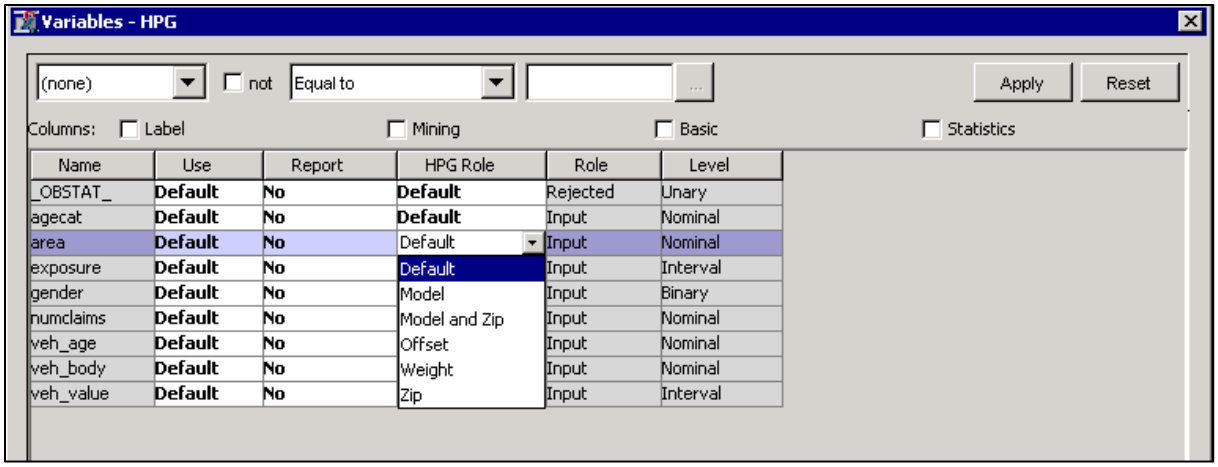


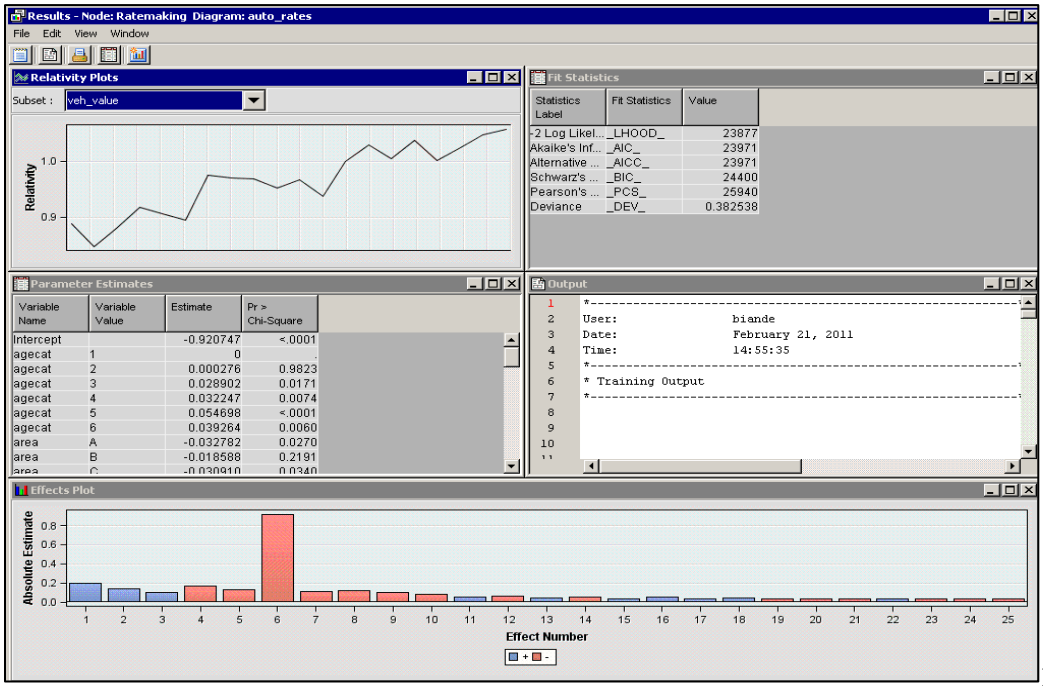
Figure 14: Variables-HPG Dialog Box

If you specify **Full Model** for handling the rating variables of the logit part of the model, then any rating variable that has a HPGRole value of **Default**, **Model**, or **Model and Zip** is used in the logit part of the model. That is, **Full Model** means that all variables in the logit part of the model are the same as in the Poisson part of the model. Selecting **User Defined** specifies certain rating variables only in the logit part of the model and not in the Poisson part of the model. For example, if you specify **User Defined** and then use the HPGRole column to set a rating variable to Zip, then that variable is used only in the logit part of the model and not in the Poisson part of the model.

The HPGRole column is also where you specify which offset and weight variables to use. The offset and weight variables must be numeric and are not used as rating variables in the modeling process. If a weight variable has a negative value, those observations are not processed.

SEVERITY MODEL

A typical way to model severity (claim amount) is by using a gamma distribution with a log link function. The Ratemaking node automatically uses the combination of gamma distribution and log link function if the node is run using the default settings. The same results are produced as are shown for the frequency model except for the Actual versus Predicted plot. Figure 15 shows the results for a severity model.



F

Figure 15: Results for a Severity Model

CONCLUSION

This paper shows how you can use the new Ratemaking node in SAS Enterprise Miner 7.1 to build standard insurance ratemaking models. Examples include pure premium model for homeowner's insurance and frequency and severity models for automobile insurance claims data.

A future release of SAS Enterprise Miner will provide much more functionality related to the relativities. For example, there are plans to add smoothing techniques for the relativity plots and add more functionality for the reference levels of the rating variables. Also, the Ratemaking node will support a stepwise procedure for the GLMs.

ACKNOWLEDGMENTS

The author thanks David Duling, Susan Haller, Dom Latour, and Taiyeong Lee for their guidance and assistance in code reviews to make the code more efficient, Jagruti Kanjia and Nilesh Jakhotiya for their valuable testing and validation input for the Ratemaking node, and Wayne Thompson for providing requirements for the Ratemaking node.

REFERENCES

Anderson, D., Feldblum, S., Modlin, C., Schirmacher, D., Schirmacher, E., and Thandi, N., 2007. A Practitioner's Guide to Generalized Linear Models, http://www.casact.org/pubs/dpp/dpp04/04dpp1.pdf.

Kaas R., 2005. Compound Poisson Distributions and GLM's-Tweedie's Distribution. Lecture, Royal Flemish Academy of Belgium for Science and the Arts, http://www.kuleuven.be/ucs/seminars_events/other/files/3afmd/Kaas.PDF.

Manning, W.G. and Mullahy, J., 2001. Estimating Log Models: To Transform or To Not Transform? Journal of Health Economics 20, 461–494.

Nelder, J.A. and Pregibon, D., 1987. An Extended Quasi-Likelihood Function. Biometrika 74, 221–232.

Werner, G. and Modlin, C., 2009. Basic Ratemaking, http://www.casact.org/pubs/Werner_Modlin_Ratemaking.pdf.

CONTACT INFORMATION

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

Billie Anderson, PhD.
SAS
SAS Campus Drive
Cary, NC 27513
(919)531-3687
Billie.Anderson@sas.com