

Predictive Modeling in the Insurance Industry Using SAS Software

Terry J. Woodfield, Ph.D., SAS Institute Inc., Irvine, CA

ABSTRACT

Insurance companies, third party insurance administrators, state insurance funds, state regulatory agencies, insurance industry consultants, and application service providers (ASP) use SAS software in a variety of predictive modeling situations. SAS Enterprise Miner provides tools for modeling continuous responses, such as monetary losses and time off work, and discrete responses, such as fraud, third party subrogation, and litigation. Enterprise Miner includes a rich set of menu driven modeling solutions as well as an underlying array of SAS procedures for modeling insurance data. This paper will address two problems, predicting losses and detecting fraud. Two possible solutions will be presented for each problem, and additional solutions will be suggested. The paper is intended for actuaries, business analysts, and insurance managers who are interested in solving predictive modeling problems in the insurance industry.

INTRODUCTION

Insurance companies, third party insurance administrators, state insurance funds, state regulatory agencies, insurance industry consultants, and application service providers (ASP) use SAS software in a variety of predictive modeling situations. For property and casualty insurance, setting premiums requires an assessment of risk that leads to estimates of loss for a book of business. To improve case management and reduce bulk reserves, insurers have turned to predictive models for predicting individual claim losses. For workers' compensation insurance, loss prediction includes predicting time off of work and total medical expenses. This paper will address the prediction of medical expenses at the claim level. The problem of predicting losses is exacerbated by the increase in fraud by claimants and by health care providers. The problem of claimant fraud detection will also be addressed in this paper. Speights, *et al*, (1999b) address the problem of profiling health care providers, which may indirectly contribute to solving the provider fraud problem.

Statisticians translate business problems into a statistical or mathematical formulation, and then they apply appropriate statistical modeling techniques to help solve the original business problem. For the problem of predicting insurance losses, the equivalent statistical problem is predicting continuous responses (targets, outcomes) given a set of predictor variables (inputs, independent variables). Such problems can be solved using linear or nonlinear regression models, or computer intensive models like neural networks or regression trees. When censoring occurs, survival models are appropriate, and many conventional modeling approaches accommodate censoring, including neural network modeling (DeLaurentis and Ravdin 1994, Faraggi, *et al*, 1995, Biganzoli, *et al*, 1998, Speights, *et al*, 1999a) and decision tree modeling (Ciampi, Negassa, and Lou 1995). Fraud prediction is classified as predicting binary responses, which may be accomplished using logistic regression, decision trees, or neural networks. SAS/STAT software supports a variety of nonlinear regression models for analyses of censored data. One of the challenges of predictive modeling in insurance is obtaining data that can be used to build a predictive model. Woodfield (1994) addresses some of the pitfalls that occur when obtaining and preparing data for modeling. Pyle (1999) gives an excellent overview of data preparation for the general data mining problem. For this paper, hypothetical simulated data will be used to illustrate the techniques employed.

CLAIMANT FRAUD

When a worker is injured while performing the duties of his or her job, the nature of the workers' compensation insurance system assumes liability of the employer while also placing limits of liability depending on the circumstances. The injured party whose accident is covered by an insurance policy is called the claimant. A claimant can commit fraud in a number of ways. The claimant can fake an accident. The claimant can exaggerate the severity of the injury. The claimant can misrepresent the injury to presume coverage, such as the worker who breaks a leg skiing on Sunday, then reports to work on Monday and within a few minutes claims to have fallen and broken the leg on the job. The claimant can conspire with a health care provider to misrepresent when health care was received, the magnitude of the injury, and the nature of the health care received.

Property and Casualty insurance covers a wide spectrum of accidents, including coverage for auto liability bodily injury and workers' compensation medical expenses. When building models to detect fraud, the specific coverage is important, and it is rarely possible to mix data from different insurance lines. Auto liability bodily injury usually involves a one-time payment of a settlement amount that is negotiated based on health care expenses incurred to-date and predicted future expenses. Workers' compensation coverage usually provides for settlement of medical expenses with 45 days of when they are billed, and may include a one-time settlement that is negotiated based on anticipated future expenses. An auto liability settlement is usually a cash payment, whereas a workers' compensation settlement may include an annuity. Auto liability policies usually stipulate maximum coverage, for example, \$1 million per accident, \$100,000 per person. Workers' compensation usually has no limit on medical expenses, although limits are always placed on wage replacement with respect to weekly compensation. These facts will be especially relevant when you consider loss reserving models later.

Statistical models for fraud address two situations. In the first, little or no data is available to target specific claims as having been fraudulent. This becomes an outlier detection problem. In the second situation, data is available in which each claim has one of three flags: (0) definitely no fraud, (1) definitely fraudulent, and (-1) fraud status is unknown. For the second situation, combining the (0) and (-1) flags is often performed in one of two ways: simply treating unknown as no fraud, or treating unknown as the first situation, identifying outliers, classifying some outliers as fraud, and proceeding with only targets (0) and (1) in the data. The problem of unknowns can also be addressed as a *reject inference* problem, with the usual strategy being to randomly select claims and investigate all of them, trying to predict when an unknown claim would have been determined to be fraudulent. In this paper, fraud detection will be addressed using a binary response model, but you should be aware that there are often more complex modeling issues that must be addressed.

USING SAS ENTERPRISE MINER FOR FRAUD DETECTION

To illustrate fraud detection using Enterprise Miner, a simulated fraud data set is employed. As with any data mining problem, at least 50% of the project will be devoted to data collection, data processing, and data repair. Some of the data repair tasks include imputing missing or invalid observations. The initial

modeling data set supplied to Enterprise Miner satisfies the following conditions.

1. Each record (row, observation) represents a unique claimant.
2. All invalid fields (variables, columns) have been corrected or coded as missing.
3. All derived variables, such as transaction roll-ups, statutory limits, and special interactions, have been calculated and placed in the data as unique fields.
4. Variables that cannot be used for modeling have been clearly identified or removed.

The simulated data contains 13 fields typically found in a workers' compensation modeling data set. There are almost always more than 13 fields in practice. The fields are:

1. ClaimantID: The unique ID of the claimant.
2. Fraud: Target, 0 for no fraud, 1 for fraud.
3. AccidentDate: Date that accident occurred.
4. AccidentTime: 24 hour clock time of accident.
5. AccidentNature: Nature of accident, a special industry two byte code.
6. InjuryNature: Nature of Injury, a special industry two byte code.
7. BodyPart: Special code for injured body part.
8. Derived1: Derived variable based on rolling up transactions or applying industry knowledge to different fields. Examples include, number of physician payments in the last 60 days, number of unique providers paid, number of providers more than 20 miles from primary residence.
9. Derived2: Another derived variable.
10. TextDNA1: Numeric derived variable using text mining methods applied to written description of accident.
11. TextDNA2: Another text mining variable.
12. Hour: Derived from AccidentTime, 0 through 23 hour of day that accident occurs.
13. Day: Day of week derived from AccidentDate, 1 through 7.

SAS datetime functions simplify the derivation of variables Hour and Day:

```
Day=weekday(AccidentDate);
Hour=hour(AccidentTime);
```

Derivation of model inputs using the SAS data step simplifies coding and makes it easier for the predictive model to try complex derivations and transactional roll-ups.

When initial models have small lift, you should investigate ways to turn available information, such as free format text fields, into model inputs. This can be as simple creating dummy variables that identify key words that intuitively imply fraud, or as complex as converting text fields into a set of numeric fields. (See Lagus, *et al*, 1999, for a text mining method applied to large document retrieval, or visit the KDD website for a text mining workshop at <http://www.cs.cmu.edu/~dunja/WshKDD2000.html>).

A typical fraud rate in property and casualty insurance is under 1%. Some practitioners recommend oversampling, although there is some empirical evidence that oversampling has little value in improving model accuracy (Jim Georges, personal communication). Rather than oversample the modeling data, you can also identify non-fraud cases that may not be of interest even if they were fraudulent, such as low cost or severity cases, and delete these from the data, thereby increasing the fraud rate. As with any manual data cleaning, you must be careful not to introduce biases into the model. The fraud rate for the simulated data is 2.1%.

Enterprise Miner was used to model the simulated fraud data. A thorough exploratory analysis may reveal interesting relationships between variables, but one of the appealing things about neural

network models is that they accommodate complex interactions without having to identify or specify them in advance. Decision trees are also relatively easy to implement, and they have the added benefit of built-in missing value handling. Logistic regression models often outperform neural networks and decision trees, but they may require more effort with respect to deriving and selecting variables for analysis.

Figure 1 shows an Enterprise Miner flow diagram that fits a logistic regression model, a neural network model, and a decision tree model to the fraud data.

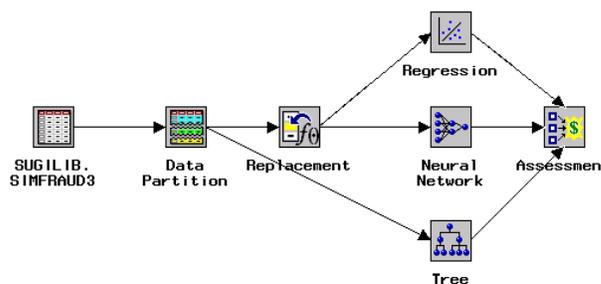


Figure 1

A lift chart compares the three models in Figure 2.

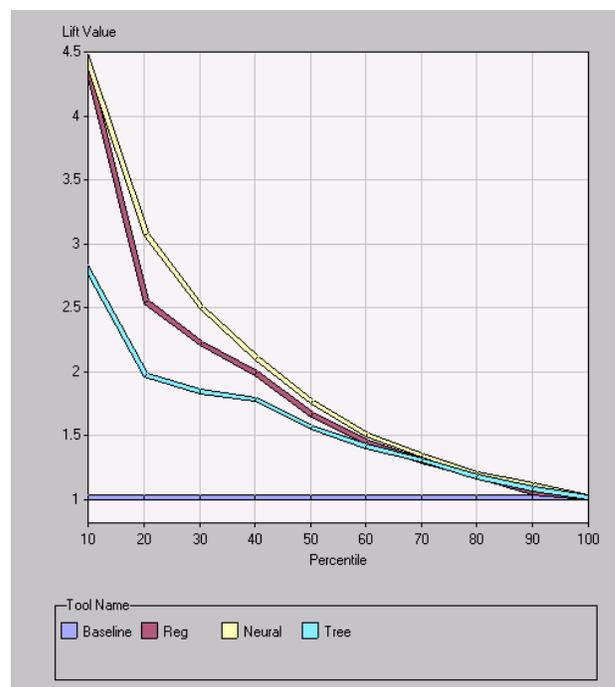


Figure 2

The neural network model and the logistic regression model appear to outperform the decision tree. Analysis of the percent captured response permits the user to make decisions about cutoffs for implementation. Figure 3 presents a percent captured response plot. You can see that by using the scores from the neural network model, a cutoff value at the 30th percentile will capture approximately 75% of the fraud cases. The same percentile cutoff for the decision tree would capture only about 55% of frauds.

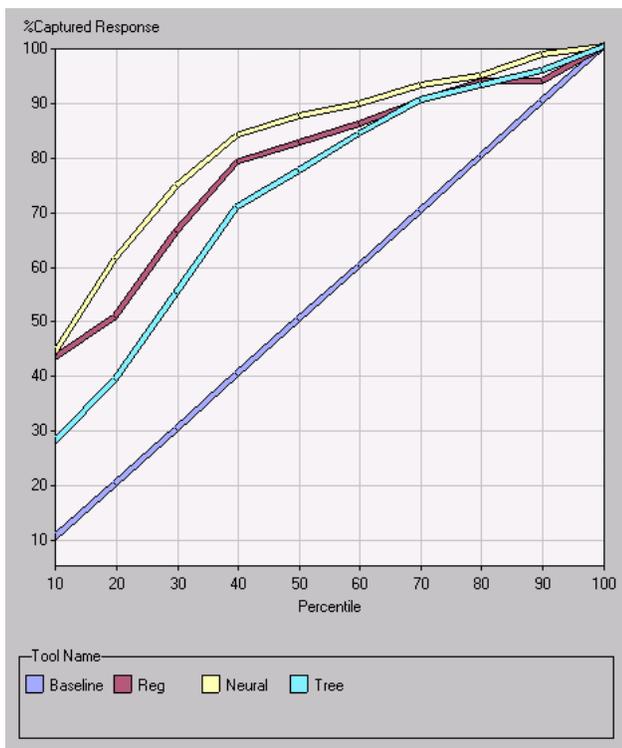


Figure 3

MODEL ASSESSMENT

The lift chart and other graphical tools permit you to assess models and compare different models. You should also scrutinize the goodness-of-fit diagnostics that are provided by Enterprise Miner.

Each model node has a results option. Table 1 shows the results for the neural network model.

| | Fit Statistic | Training | Validation | Test |
|----|--------------------------------|--------------|--------------|--------------|
| 1 | [TARGET=FRAUD] | . | . | . |
| 2 | Average Profit | 0.0211835137 | 0.0217958557 | 0.0201105312 |
| 3 | Misclassification Rate | 0.0210683859 | 0.0217958557 | 0.0201105312 |
| 4 | Average Error | 0.08529009 | 0.090332845 | 0.0924313596 |
| 5 | Average Squared Error | 0.0190019381 | 0.0198203264 | 0.0189396603 |
| 6 | Sum of Squared Errors | 330.10166839 | 258.25885251 | 246.74589434 |
| 7 | Root Average Squared Error | 0.1378475175 | 0.1407846809 | 0.1376214384 |
| 8 | Root Final Prediction Error | 0.1413842924 | . | . |
| 9 | Root Mean Squared Error | 0.1396271038 | 0.1407846809 | 0.1376214384 |
| 10 | Error Function | 1481.6594433 | 1177.0369707 | 1204.1957529 |
| 11 | Mean Squared Error | 0.0194957281 | 0.0198203264 | 0.0189396603 |
| 12 | Maximum Absolute Error | 0.9995808064 | 0.9996639293 | 0.9996297008 |
| 13 | Final Prediction Error | 0.0199895181 | . | . |
| 14 | Divisor for ASE | 17372 | 13030 | 13028 |
| 15 | Model Degrees of Freedom | 220 | . | . |
| 16 | Degrees of Freedom for Error | 8466 | . | . |
| 17 | Total Degrees of Freedom | 8686 | . | . |
| 18 | Frequency of Classified Cases | 8686 | 6515 | 6514 |
| 19 | Sum of Frequencies | 8686 | 6515 | 6514 |
| 20 | Sum Case Weights * Frequencies | 17372 | 13030 | 13028 |
| 21 | Akaike's Information Criterion | 1921.6594433 | . | . |
| 22 | Schwarz's Bayesian Criterion | 3476.9423622 | . | . |

Table 1

The average squared error is similar across training, validation, and test data. For this example, the training/validation/test percentages are 40/30/30. Using the diagnostics from the test data provides an honest assessment of the model's performance. If you examined all three models, you would find that the misclassification rate is at least slightly smaller for the neural network model.

SCORING

The SCORE node produces SAS code to implement the scoring model, including formulas for all derived variables created by

Enterprise Miner. The code for imputing missing values is also included.

Enterprise Miner also supports the C*Score node, which creates C language code to implement the fraud scoring model. Unfortunately, some SAS data step functions may not be available as C code, but in general, the ANSI C code that is produced is easy to implement using most modern C or C++ compilers.

FRAUD SUMMARY

The analysis and modeling of fraud data is not trivial. In most cases, you would not be content to fit three models using all variables and default settings. Rather, an extensive analysis might suggest new variables to consider, such as more complex rollups of transactions. The analysis might also suggest interaction terms to include in the logistic regression model. You may also discover that different options in tree building might improve the tree model. The decision tree node also supports interactive tree building. The neural network node supports a variety of connectionist models as well as an extensive set of options for fitting the models, including traditional backpropagation. Results like those achieved for the hypothetical data are probably rare for a first pass at modeling the data. Good results often require extensive analysis and remodeling, but Enterprise Miner provides the tools necessary for a comprehensive fraud modeling project.

MEDICAL LOSS RESERVING

Property and Casualty insurance often provides coverage for medical expenses that result from an accident. The fraud section discussed how medical payments for auto liability bodily injury and workers' compensation differ substantially, with auto liability often involving a single lump sum payment for all medical expenses, and workers' compensation requiring many medical payments for each medical bill. The nature of how claims are processed and payments are made makes it more difficult to predict auto liability bodily injury, primarily because the model has no medical payment transactions to learn from. For this section, hypothetical data from workers' compensation medical expenses will be modeled.

When modeling medical payments, you should be aware of two inflationary pressures that affect medical expenses. The first inflationary effect is the tendency for a given medical product, such as bandages, prescription drugs, and devices (e.g., crutches), to increase in cost over time. The other inflationary effect reflects a tendency for greater utilization of medical resources, primarily as a psycho-social trend for claimants who are accustomed to receiving good medical benefits from their employer at very low cost. Claimants who historically were less prone to seek medical intervention for relatively mild conditions, now seek intervention because of the perception of low cost. These two inflationary effects are called price inflation and utilization inflation respectively. A third inflationary effect is technology inflation, that is, increasingly sophisticated (and expensive) techniques are being invented to treat injuries. Technology inflation is combined with risk adverse behavior which causes a physician to prescribe a diagnostic test or treatment for fear that failure to do so could result in a lawsuit. For modeling, technology inflation is usually treated as confounded with price and utilization inflation.

The predictive modeler can deal with inflation in several ways. A set of price and utilization inflation indices can be derived, and the dollar amounts can be inflation adjusted before modeling, and then predictions can be inflation adjusted to reflect the time period of the injury.

This paper will ignore inflation by focusing on the duration of medical payments in workers' compensation claims. The actual loss reserve will require implementation of a model to predict average medical payments over the duration predicted, applying appropriate inflation factors.

USING SAS/STAT TO MODEL MEDICAL LOSS DURATION

SAS/STAT software provides a number of procedures for predicting continuous responses. With independent observations, the GLM, MIXED, and REG procedures may be appropriate. When observations are dependent, the MIXED procedure may be used. When response values are censored or truncated, the LIFEREG and PHREG procedures are appropriate. For the medical loss problem, the LIFEREG procedure will be employed. The introduction lists references for neural network and decision tree models for censored data. These have not yet been implemented in Enterprise Miner.

The modeling data set satisfies the same four conditions listed for the fraud data, essentially ensuring that it is ready to be modeled without any additional data pre-processing.

The simulated data contains 17 fields typical of a workers' compensation modeling data set:

1. ClaimantID: The unique ID of the claimant.
2. MedDuration: Duration of medical intervention measured as the number of days from accident date to date of last medical bill.
3. Censor: 1 (censored) if date of last medical bill is within 60 days (an arbitrary cutoff) of data extract, 0 otherwise.
4. AccidentDate: Date that accident occurred.
5. AccidentTime: 24 hour clock time of accident.
6. AccidentNature: Nature of accident, a special industry two byte code.
7. InjuryNature: Nature of Injury, a special industry two byte code.
8. BodyPart: Special code for injured body part.
9. Derived1: Derived variable based on rolling up transactions or applying industry knowledge to different fields. Examples include, number of physician payments in the last 60 days, number of unique providers paid.
10. Derived2: Another derived variable.
11. MedGroup: Ordinal variable that groups treatment zipcode area based on industry medical inflation indices.
12. CPTGroup: Nominal variable that groups CPT medical treatment codes.
13. Age: Claimant age at the time of the accident.
14. Diabetes: 1 if claimant is diabetic (except mild diet controlled diabetes), 0 otherwise.
15. PriorHistory: An ordinal variable that ranks claimants based on prior medical history, with inputs like those often found on a hospital admittance form (anemia, prior hospitalizations, asthma, heart conditions, etc.)
16. Hour: Derived from AccidentTime, 0 through 23 hour of day that accident occurs.
17. Day: Day of week derived from AccidentDate, 1 through 7.

The variable MedGroup is a proxy for inflation, but for convenience, inflation adjustment will not be considered for this problem.

Since Enterprise Miner is licensed with SAS/STAT as a component, you may wish to invoke PROC LIFEREG through an Enterprise Miner SAS code node. Table 2 lists the macro variables that are created and maintained by Enterprise Miner that are relevant for this problem.

| Macro Variable Name | Assigned Value |
|---------------------|---------------------------------|
| __CLASS | Class variables |
| __INTRVL | Continuous (interval) variables |
| __TARGETS | Target variable(s) |
| __INPUTS | All input variables |
| __PREDICT | Predicted variable |
| __TRAIN | Training data set name |
| __VALID | Validation data set name |
| __TEST | Test Data Set Name |
| __PRE | Exported prediction data set |

Table 2

Listing 1 contains the SAS code for performing censored regression using the default Weibull (type 1 extreme value) distribution.

```

/* Define output prediction variable */
%let __PREDICT=P_MedDur;
/* Use censored regression model */
proc lifereg data=__TRAIN;
  class __CLASS;
  model __TARGETS * CENSOR(1)=
          __INTRVL __CLASS;
  output out=__PRE predicted=__PREDICT
         std=STD_Pred;
run;
/
data __PRE;
  set __PRE;
  /* No prediction over 8 years allowed */
  if (__PREDICT>3000) then __PREDICT=.;
  R_MedDur=MedDuration-__PREDICT;
  /* Should convert median to */
  /* conditional mean */
  /* Example for lognormal: */
  /*
NewPred=exp(log(OldPred)+0.5*STD_Pred**2);
  */
run;
/* Add code to score holdout sample */

```

Listing 1

The Enterprise Miner flow diagram is given in Figure 4.

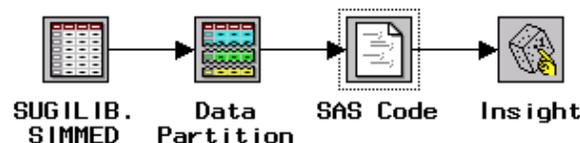


Figure 4

An Insight node has been added to evaluate the results. You still must create code to score the holdout sample so an honest assessment of the model can be performed.

In addition to SAS/INSIGHT, Enterprise Miner supports exploratory graphics. For example, Figure 5 shows a bar chart of accidents by day of the week. Note that weekend accidents are less frequent because for the policies written, the employers tend to have less worker activity on weekends. Also notice that for this data, Monday accidents are the most frequent. This is realistic for a number of manufacturing industries. Of course, the data is hypothetical based on experience working with insurance data.

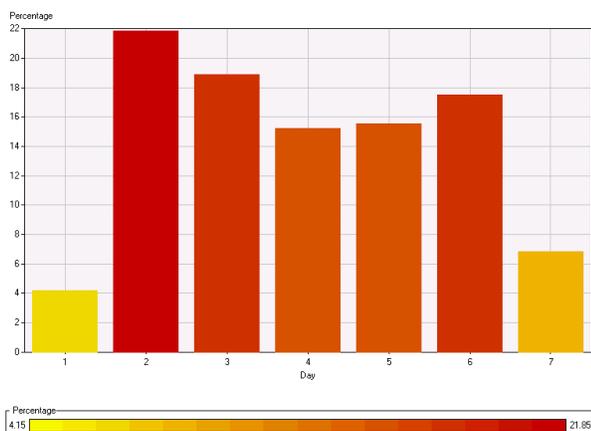


Figure 5

When the SAS code node has completed execution, you can select results and open the output tab to see the printed output produced by the LIFEREG procedure. Output 1 displays a subset of the output information.

Analysis of Parameter Estimates

| Variable | DF | Estimate | Standard Error | Chi-Square | Pr > ChiSq | Label |
|----------------|----|-----------|----------------|------------|------------|-----------|
| Intercept | 1 | 34.81476 | 2574.5 | 0.0002 | 0.9892 | Intercept |
| Age | 1 | 0.01081 | 0.0004729 | 522.4563 | <.0001 | |
| Derived1 | 1 | 0.0053201 | 0.00005124 | 10778.2795 | <.0001 | |
| Derived2 | 1 | 0.0038646 | 0.00005838 | 4382.5827 | <.0001 | |
| AccidentNature | 11 | | | 451.2865 | <.0001 | |
| | 1 | -0.02959 | 0.02114 | 1.9592 | 0.1616 | 10 |
| | 1 | -0.05622 | 0.02107 | 7.1182 | 0.0076 | 20 |
| | 1 | -0.11565 | 0.02586 | 19.9974 | <.0001 | 25 |
| | 1 | -0.15398 | 0.02565 | 36.0252 | <.0001 | 30 |
| | 1 | -0.15007 | 0.02174 | 47.6324 | <.0001 | 39 |

Output 1

Results include a chi-square test of the significance of model coefficients. Since LIFEREG fits nonlinear regression models, the usual problems may arise with the algorithm not converging to a set of maximum likelihood estimates. Creating a general purpose SAS code node to handle such contingencies presents a rather massive programming chore.

MEDICAL RESERVING SUMMARY

As with fraud modeling, predicting loss reserves is a challenging task requiring the imputation of missing values, variable selection, data exploration to determine interactions and candidate derived variables, and model assessment. The code that is provided for the hypothetical insurance data would not be adequate for production work. Fortunately, the Output Delivery System (ODS) permits you to extract parameter estimates so that you can create scoring code to evaluate the holdout sample. Careful attention must be paid to censored values so that they are not erroneously included in accuracy statistics.

Implementation of censored neural network algorithms would be useful to help take advantage of the nonparametric features of neural networks to speed up the modeling process. Even if neural network models cannot be deployed because of regulatory constraints, the models would serve as useful benchmarks to evaluate the traditional parametric survival models that are used.

CONCLUSION

Enterprise Miner serves as a useful environment for modeling insurance data. Two examples illustrate how to construct predictive models using SAS software. Fraud models are supported by a comprehensive set of modeling tools in Enterprise Miner, including logistic regression models, neural network models, and decision tree models. Unfortunately, Enterprise Miner does not currently support analysis of censored data directly, even though algorithms exist for applying neural networks and decision trees to censored data. Instead, you must

create a custom SAS code node to perform the necessary calculations using the underlying SAS/STAT software that is part of the Enterprise Miner product.

REFERENCES

- Biganzoli, E., Boracchi, P., Mariani, L., and Marubini, E. (1998). "Feed forward neural networks for the analysis of censored survival data: A partial logistic regression approach." *Statistics in Medicine*, 17:1169—1186.
- Ciampi, A., Negassa, A. & Lou, Z. Y. (1995). "Tree-structured prediction for censored survival data and the Cox model." *Journal of Clinical Epidemiology*, 48, 675-689.
- DeLaurentis, M., and Ravdin, P.M. (1994). "A technique for using neural network analysis to perform survival analysis of censored data." *Cancer Letters*, Vol. 77, No. 2-3, pp. 127-138 (15 March).
- Faraggi, D., and Simon, R. (1995). "A Neural Network Model for Survival Data". *Statistics in Medicine*. 14, 73-82.
- Faraggi, D., Simon, R., Yaskil, E., and Kramar, A. (1998). "Bayesian Neural Network Models for Censored Data". *Biometrical Journal*, 39, 519-532.
- Lagus, K., Honkela, T., Kaski, S., and Kohonen, T. (1999). "WEBSOM for textual data mining". *Artificial Intelligence Review*, volume 13, issue 5/6, pages 345-364, December 1999, Kluwer Academic Publishers.
- Pyle, Dorian (1999). **Data Preparation for Data Mining**. San Francisco: Morgan Kaufmann Publishers, Inc.
- Speights D., Brodsky J., Chudova D. (1999a) "Using Neural Networks to Predict Duration in the presence of Right Censoring and Covariates". *Casualty Actuarial Society Forum*, Winter 1998-1999.
- Speights, D., Woodfield, T., and Becker, D. (1999b). "Using Resampling Techniques to Measure the Effectiveness of Providers in Workers Compensation". *Casualty Actuarial Society Forum*, Fall 1999.
- Woodfield, Terry (1994). "Data Conversion Pitfalls", 1994 Proceedings of Computer Science and Statistics: 26th Symposium on the Interface, pp. 362-371.

ACKNOWLEDGMENTS

The author would like to thank Jim Georges of SAS for his helpful suggestions.

CONTACT INFORMATION

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

Terry J. Woodfield
 SAS Institute Inc.
 5 Park Plaza, Suite 900
 Irvine, CA 92614
 Work Phone: 949.852.8550x321
 Fax: 949.852.5277
 Email: terry.woodfield@sas.com
 Web: www.sas.com