To Neural or Not to Neural? – This is the question in Cross Selling.

Leon L. Fedenczuk, Gambit Consulting Ltd.

ABSTRACT

Adding new lines of business and adding new services for existing customers is particularly important in deregulated telecommunication industry. Attracting new services to existing customers often translates into happier customers, increased retention of profitable customers, and competitive advantage. Identification of the most profitable customers (who are the most probable buyers of additional or new services) calls for computer based systems that are based on data mining solutions. This paper compares solutions from neural networks to solutions from logistic regression, discriminant analysis, and decision tree analysis for cross selling predictions. This comparison deals with the statistical accuracy of predictions, ability for business interpretation, and the practical implementation of computer systems that can perform outlined tasks. Included in the paper are practical tips on how to use base SAS, SAS/STAT, and SAS Enterprise Miner for model selection and the decision support systems (DSS) implementation.

Keywords: Cross Selling, Modeling, Neural Networks, Logistic Regression, Decision Trees, Discriminant Analysis, Enterprise Miner, DSS systems.

INTRODUCTION

Discriminant analysis and logistic regression are traditionally applied to predict binary response of data. A binary target variable is characterized by two events. They can be of numerical nature *0* and *1* where zero represents non-event and one represents an event. Alternatively, a character string with two outcomes (e.g. No and Yes) is often applied.

This paper presents a history of a model development for predicting cross sales. At first, logistic regression and discriminant function models were developed. Next, a series of models were developed with the SAS Enterprise Miner. These later models included logistic regression, neural net, and decision tree models. The response value *y1=0/1* corresponded to a non-event (no cross sale) and event (cross sale) respectively. Equivalent non-numerical description of (*Y/N*) was applied in some instances. The binary responses were predicted based on eleven independent variables, which were named as *x1-x11* to preserve their confidentiality. These variables represented customer internal characteristics, demographic variables derived based on the customer's postal code, and credit scoring. The last two variable sets were obtained from the external data sources.

The data, models, and results presented in this paper represented a small subset of a larger project. The subset was based on a market segment, which had good success rate in the cross sale program. However, an appetite for better results (return) and the process optimization drove the predictive modeling. The segment's input data set contained *1263* observations that were split into a learning set with *861* observations and a testing set with *402* observations. The project's main goal was to predict a cross sale binary indicator variable (*y1=0/1*) based on the variables *x1-x11*.

This study identified large performance differences between prediction powers of models developed with different modeling methods. Specifically, logistic regression and neural network models outperformed models based on decision tree and discriminant function. In addition, a customized logistic model, which was based on a much smaller subset of variables, outperformed the neural model in the best decile. However, it is not expected that similar projects will support this conclusion. On the contrary, author had experiences with similar data sets where decision tree models outperformed by far the neural and logistic models.

Business requirements and especially interpretability favor logistic or decision tree models because of their interpretation power. At the same time, neural network models should be based on a pre-selected subset of variables that do have predictive powers. Such subsets are traditionally derived with stepwise selection regression, backward elimination regression, or decision tree procedures in the initial stages of data mining projects.

Thus, the neural network models should be implemented in cases when they clearly outperform other types of models and there are no regulatory or interpretation requirements.

TRADITIONAL MODELING

This section presents a modeling portion of a data mining project. It is assumed that reader is familiar with overall techniques required to load and process data before implementing statistical modeling and diagnostics.

Initially, stepwise discriminant analysis and stepwise logistic regression were used to find the smallest subset of the most significant variables that supported classification into two cross sale levels. This development was based on the learning set, which was identified by a categorical variable. Two stepwise analyses (discriminant and logistic) identified the same subset of variables (*x1, x3, x6, x7, x8, x9, and x10*) that significantly contributed to the developed models.

Next, the selected subset of variables was used in building the final discriminant function and logistic regression models from the learning set. Diagnostics for the *x7* variable, in the logistic regression model, indicated that this variable was non-significant. In addition, it was the least important variable based on the stepwise discriminant analysis. Therefore, this variable was excluded from both models.

Finally, the models were applied to predict the cross sale event (*1 or Y*) for all observations in the learning and testing sets. At the end, the estimated probability and generated classifications were compared with the known outcomes of the cross sale program (see Table1 through Table 4).

Table 1. Logistic Regression; Classification Frequency for Learning Set.

Cross sale From/Into	Class Into N	Class Into Y	Total
Class From N	337 39.14%	119 13.82%	456 52.96%
Class From Y	132 15.33%	273 31.71%	405 47.04%
Total	469 54.47%	392 45.53%	861 100.00%

Table 2. Logistic Regression; Classification Frequency for Testing Set.

Cross sale From/Into	Class Into N	Class Into Y	Total
Class From N	132 32.84%	69 17.16%	201 50.0%
Class From Y	57 14.18%	144 35.82%	201 47.04%
Total	189 47.01%	213 52.99%	402 100.00%

Table 3. Discriminant Analysis; Classification Frequency for Learning Set.

Cross sale From/Into	Class Into N	Class Into Y	Total
Class From N	323 37.51%	133 15.45%	456 52.96%
Class From Y	155 18.00%	250 29.04%	405 47.04%
Total	478 55.52%	383 44.48%	861 100.00%

Table 4. Discriminant Analysis; Classification Frequency for Testing Set.

Cross sale From/Into	Class Into N	Class Into Y	Total
Class From N	131 32.59%	70 17.41%	201 50.00%
Class From Y	55 13.68%	146 36.32%	201 50.00%
Total	186 47.27%	216 53.73%	402 100.00%

Overall, the logistic regression model performed better than the discriminant model. This was observed for both the learning and testing sets. However, the differences in the correct classification or misclassification were relatively small (0.5-3%). At this stage no interaction terms were introduced in the logistic model in order to have a fair comparison of these two modeling techniques.

DATA MINING WITH ENTERPRISE MINER

Analysis described in the previous section was characteristic of the tools and methodologies in the last twenty years. Procedures from the SAS/BASE and the data step based programs were used to load, format, summarize, and transform data from the external data sources. The final data set(s) formed a project database. Flat files, Excel tables, and SAS ACCESS to Oracle provided the required interface to corporate data sources. Next, the SAS programs were developed, which called variety of the SAS/STAT procedures[1]. These programs were modified and different options were enabled, as they were required. The log and output printouts were created and some of them were saved as files for future reference. Since, most of the projects involved numerous iterations, the maintenance of programs, outputs, and options represented a serious challenge, even for the best-organized practitioners.

Developing a good model is no longer an isolated one time project. Usually it is only the first step in the development of a decision support system. Such systems usually are vehicles of innovation, cost reduction, and improved decision support. Thus, the implemented model should be best of the best and additional model types (neural networks and decision trees) should be considered.

Unfortunately, the SAS/STAT module does not contain procedures that represent the latest technological advancements. Specifically, neural network and decision tree based modeling are not available outside of the Enterprise Miner. This is inconsistent with a long-standing tradition of adding new developments into the appropriate SAS module.

In the second part of this project, the Enterprise Miner was used to build three different models and to compare their performance. A new logistic model was built again along with a neural model and a decision tree model.

The initial data loads, data cleaning, the data summarization, and segmenting were not repeated. They involved a compression of data into a modeling set. These steps were based on an in-depth analysis of business processes and data flows. Data compression often represents the most important part of any data-mining project.

Descriptive statistics played significant role in summarization processes and generation of categorical variables. These categorical variables had triple purpose. First, they represented initial customer segmentation based on 'known' business knowledge and observed distributions. Secondly, some of these categorical variables were generated for the model performance testing, verification of market segments based on residuals, and verification of untested business hypothesis. Thirdly, some of them represented a hierarchy of business dimensions (geography, product, and time) and were designed to support multidimensional reporting of historical data and model predictions.

PROJECT DIAGRAM FLOW

A simplified project flow diagram for the cross sale customer ranking is presented in Figure 1. At one point this diagram contained two to three neural network models and the same number of decision tree models. They were used to compare between a series of one-type models. In particular, they allowed for the neural network model performance comparison. These models had different number of hidden layers and some had direct links between the input and output layers. Similarly, the tree depth, the business interpretation, and the miss-classification rates were compared before the best tree model was selected.

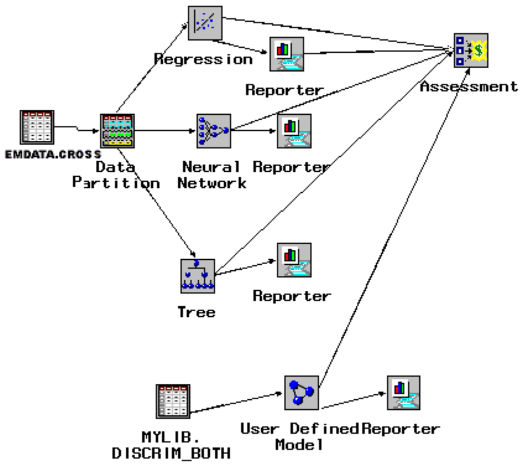


Figure 1. Project Flow Diagram.

The input data source node (EMDATA.CROSS) mapped data from a set with the pre-summarized data and assigned additional variable attributes that were required by modeling nodes (Regression, Neural Network, and Tree). These attributes included the model role for each variable (target, input, rejected) and they were changed from their default assignments when required. Tables of statistics and histograms could be displayed for interval and class variables. Detailed information on the Input Data Source and other nodes can be found in manuals for the Enterprise Miner[2,3] .

The second node, the Sampling node, performed sampling into learning, validation, and testing sets. These data sets resulted from a combination of the user-defined sampling and random sampling (Train, Validation, and Test in the Enterprise Miner). The first two sets were selected from the original learning set of *861* observations. Stratified sampling and user-defined sampling are available in this node.

The next vertical layer of nodes consisted of the Regression (logistic) node, Tree node, the Neural Network node, and the User Defined Model. The first three nodes performed all the steps required to find the most optimal model of the requested model type. The last one applied imported results of the discriminant model from the earlier described analysis with the PROC DISCRIM in the SAS/STAT. This node generated assessment statistics from predicted values in the imported data set (MYLIB.DISCRIM_BOTH).

Each of the modeling nodes in Figure 1 is connected to its own Reporter node. They generated HTML reports that supported structured reporting of each modeling approach. These reports contained the process flow diagram, header information, settings, and results.

The last node in the diagram, the Assessment node compared models and prediction diagnostics for all of four modeling nodes. This comparison was facilitated with a set of charts for lift, profit, return on investment (ROI), receiver operating curves (ROC), and response threshold charts.

Finally, the Score node was used to generate predictions from a trained model and a new input data set (see Mylib.For_Scoring in Figure 2).

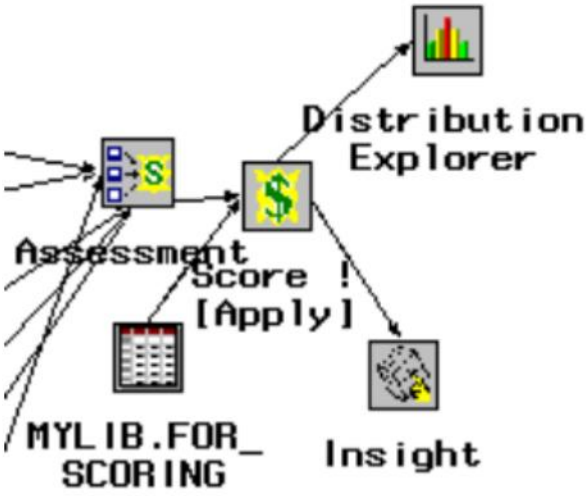


Figure 2. The Assessment and Scoring nodes.

By default, the link between the Assessment and the Score node selected the first model (logistic regression) from the list of models, unless a manual selection was made. Alternatively, a direct link between a specific modeling node can reassure the user's model selection. A potentially stronger solution can be obtained with the Ensemble node that averages the posterior probabilities from multiple models.

TRADITIONAL MODEL ASSESSMENT

In this section two previous models (logistic and discriminant) are compared with the neural and decision tree models. The comparison is based on classification rates in the testing set. The classification rates are shown in Figure 5 for the neural network model and in Table 6 for the decision tree model. The earlier models are shown in Table 2 and Table 4, which present rates for the logistic model and the discriminant model respectively. Overall differences in rates for different models are not large (8 counts for Class = N and 18 counts for Class = Y). The largest differences were attributed to the decision tree model, which turned out to be disappointing.

The best performance was observed for the neural network model, which was closely followed by the logistic and discriminant models. The neural network model accounted correctly for 67% of the known non-events ('N') and for 74% of the known events ('Y'). The corresponding success rates for the decision tree model were 62% and 65% respectively. Counts for the 'Y' group for the best three models differed only by 3 counts and corresponding counts for the 'N' group differed by 5 units. Thus, the neural network model was the initial winner.

Table 5. Neural Network; Classification Frequency for Testing Set.

Cross sale From/Into	Class Into N	Class Into Y	Total
Class From N	134 33.33%	67 16.67%	201 50.0%
Class From Y	52 17.16%	149 37.81%	201 50.0%
Total	186 46.27%	216 53.73%	402 100.00%

Table 6. Decision Tree; Classification Frequency for Testing Set.

Cross sale From/Into	Class Into N	Class into Y	Total
Class From N	126 31.34%	75 18.66%	201 50.0%
Class From Y	70 14.41%	131 32.59%	201 50.0%
Total	196 48.76%	206 51.24%	402 100.00%

ADVANCED ASSESSMENT OF MODELS

Classification tables for learning and testing sets gave approval to the neural network model. However, business decisions require more than just two rates of the correct and erroneous classifications. In a non-discriminatory customer targeting a single criterion or multiple criteria are used to identify the cross sale list. However, the program cost can be lowered substantially if we identify a much smaller portion of customers who are the most likely responders. Identification of a percentage cut-off can be facilitated with lift curves. Reviewing and comparing of the response and lift curves represented more detailed analysis.

Figure 3 shows the percentage of cross sale rates in each decile. A baseline in this figure shows an average cross sale rate in the original sample. Thus, it is a reference for any model, which was developed during this study. The presented curves show the non-cumulative response rate for the sorted deciles from 10 to 100. The first decile (10) shows the cross sale rate (response) for the top 10 percent of the model scores (the most likely cross sales). The second decile shows the responses for the second best 10 percent of the model scores, and so on. These curves allow a user to compare model quality (response rates) in deciles (decreasing quality bins) for different models. In particular, it shows that that the logistic regression model predicts correctly more than 90 percent of the most likely cross sale customers (90% of responders in the best 10%).

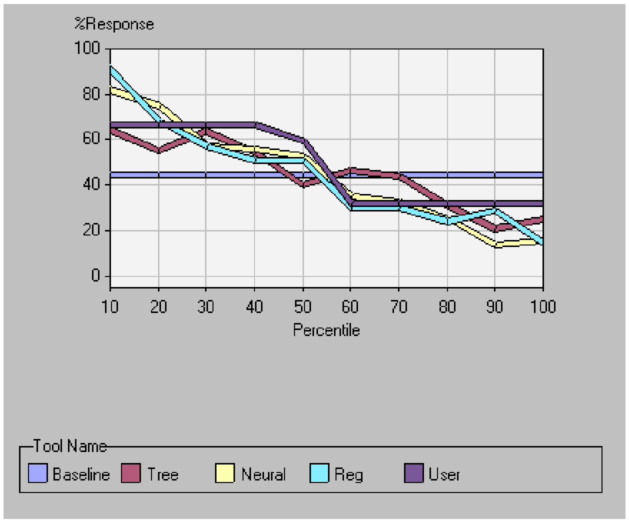


Figure 3. Non-cumulative response curves.

Cumulative lift curves, which correspond to the four models, are showed in Figure 4. A lift curve shows model's effectiveness relative to a baseline, which shows an overall (average) historical cross sale rate (horizontal line). The non-cumulative lift curves are shown in Figure 5 and they enhance visual comparison of model's performance in each decile. Figure 4 and Figure 5 show the lift curves in a relative scale where the baseline corresponds to one.

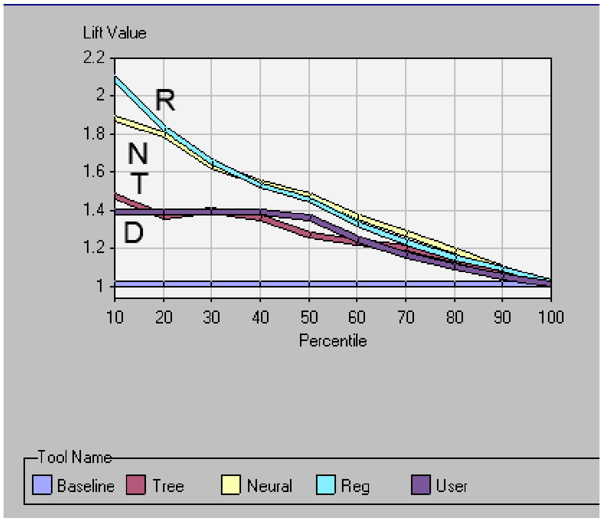


Figure 4. Cumulative lift curves. R = Logistic regression; N = Neural network; T = Decision Tree; D = Discriminant function.

Finally beyond the fifth decile, all models performed below the overall average, which corresponded to a random choice for a potential cross sale prospect (see Figure 5). A more sophisticated predictive system could apply a combination of the two or three best models. In the first decile the logistic regression model would produce the best results, while the neural model would produce better predictions in the second decile. Finally, the discriminant model would produce the best results in the third and fourth deciles. At this point we can imagine faces of developers and the model's maintenance team.

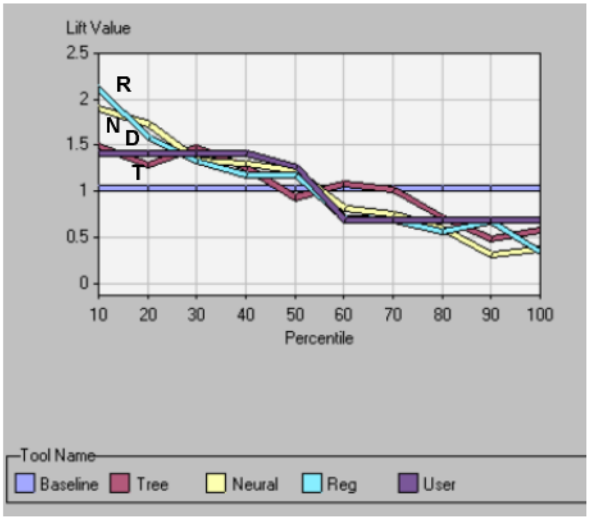


Figure 5. Non-cumulative lift curves. R = Logistic regression; N = Neural network; T = Decision Tree; D = Discriminant function.

Both sets of lift curves showed that the logistic and neural models significantly outperformed the other two models (discriminant and tree). The non-cumulative lift curves in Figure 5 showed that the best two models performance dropped fast from around 2 to 1.5 between the second and the third best decile. In the next two deciles the discriminant model performed better.

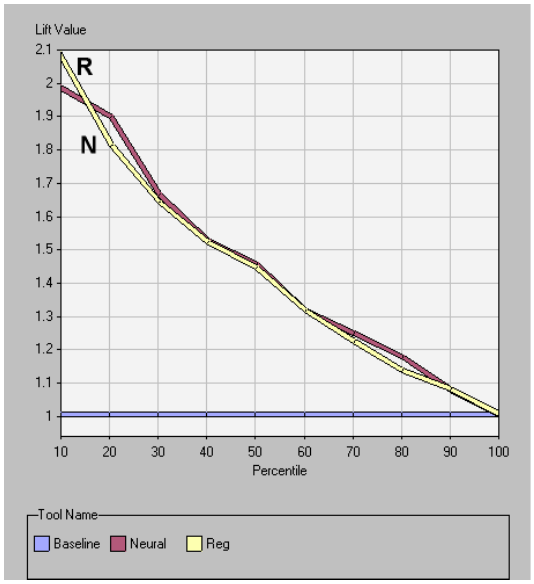


Figure 6. Cumulative lift curves. R = Logistic regression; N = Neural network with additional variables and direct links.

As with any type of modeling adding more variables should increase the neural network performance. The same effect can be achieved by adding more layers or adding direct links. Figure 6 shows a comparison of lift curves between the logistic regression model and the neural network model with the extra variables and direct links. The logistic model was based on the selected subset of the most important variables and it was still better in the first decile than the more complicated neural network model. These findings enforce a general rule that simpler is better.

Many modeling decisions and the model selection depended on the misclassification rates. Figure 7 shows a threshold-based chart and agreements between the actual and predicted cross sale counts for the neural model at the 50% threshold value. This diagram helps with verifying the agreement between the actual and predicted classes at different threshold levels. The threshold level is the cutoff, which is applied in classifying observations based on the evaluated posterior probabilities. If a predicted score was below the threshold value, then the predicted sale class was assigned to zero (no cross sale), otherwise the class was assigned to one (cross sale).

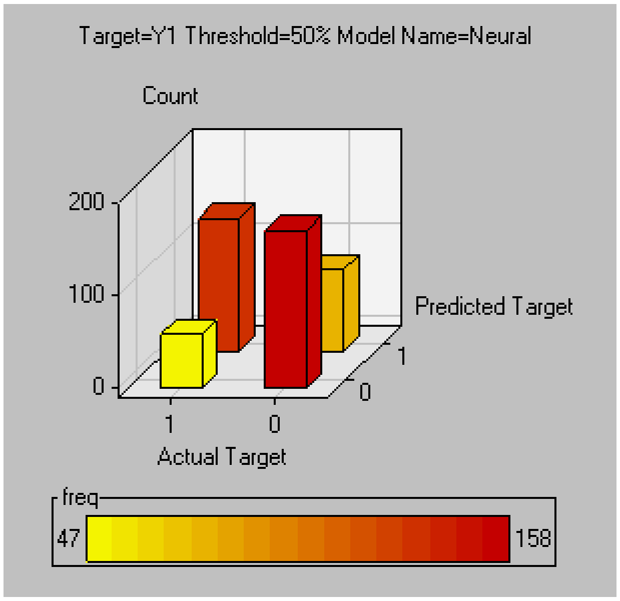
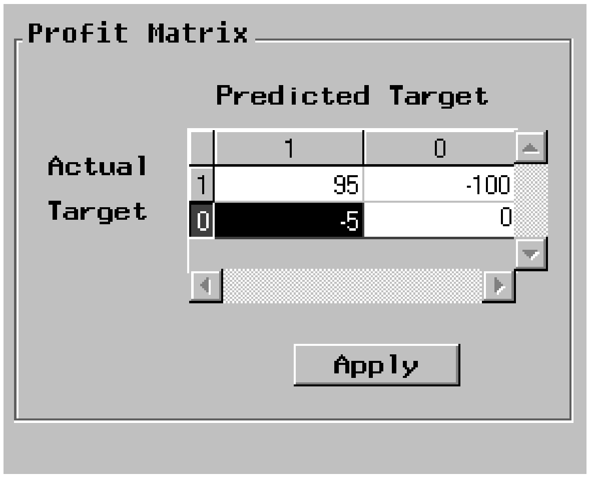


Figure 7. Threshold chart for the neural network model.

From the threshold-based chart in the Enterprise Miner a user can request an interactive profit chart. This chart enables observing the relationship between the return and the threshold value for a specified profit matrix. Cross sale efforts and marketing programs had associated costs and returns on investment for every case of four outcomes between the predicted and the actual classes. Figure 8 presents a profit matrix for these four outcomes between the actual and the predicted classes.

A simple (*0/1* or *N/Y*) decision schema had two cases of misclassification and two cases of correct classification. The assigned fix profit was based on a simple principle that a successful cross sale, which was identified as a prospect, would generate 100 units less 5 units of the fixed costs (see 1/1 cell with return=95). A non-successful cross sale, which was classified as a prospect, had a negative return related to the fixed cost (-5). The predicted non-events were classified in a similar way, where -100 was assigned for the 1/0 case (missed revenue), and 0 for the 0/0 case (the correct prediction of the non-event). This was one of many scenarios that were tested with the model. The presented values have been changed from the original values to preserve the confidentiality of information. The corresponding profit (return) chart in Figure 9 shows a relationship of the estimated return versus the classification threshold value (if posterior probability >= threshold, then class=1).

This diagram proved that the cross sale program should target all of the customers. In other words, the zero threshold should generate the highest average return. This example shows how business costs associated with different model's decisions seemed to negate the modeling efforts. However, these conclusions would be impossible without the model and its estimates. A chart in Figure 9 shows that the average return would slowly drop for the threshold between 0 and 45 percent. Beyond a flex-point at 45% the return would drop fast and a break-even point would be reached at 60 percent.



Profit Matrix

Predicted Target

Actual Target

	1	0
1	95	-100
0	-5	0

Figure 8. Profit matrix.

The cross sale results presented here characterized a very successful market segment, where the additional service was a natural fit. In general, much lower initial cross sale success rates and different shapes characterized profit charts for other customer segments.

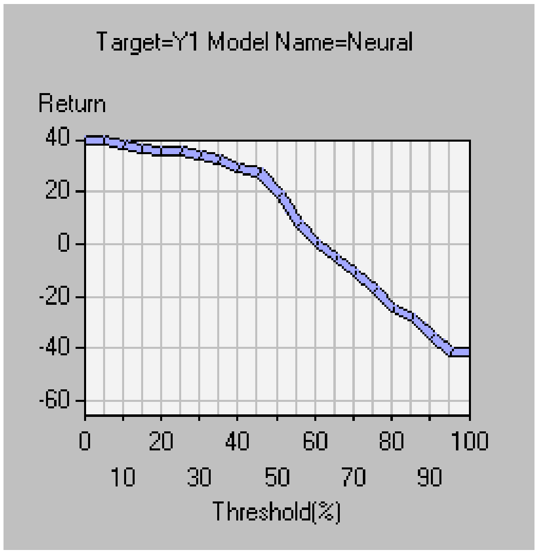


Figure 9. Return (av. profit); Neural model; Profit matrix from Figure 8.

A fine-tuning of a threshold value, which is used in a final model, is specifically important in projects with nonrandom samples (e.g. rare case sampling). During such studies a user can observe relationships between the predicted and actual target values as a function of the threshold values. Figure 10 presents a different behavior of the average return, which was based on a different profit matrix. In this example, the expected return per customer would reach a maximum at a threshold level of 25%.

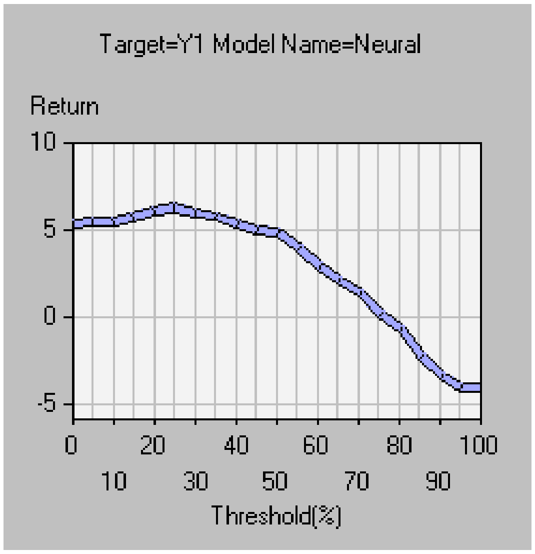


Figure 10. Return (av. profit); Neural model; Profit matrix (1/1=25; 1/0=-10; 0/1=-10; 0/0=0)

TO NEURAL OR NOT TO NEURAL?

Adding extra variables in the neural network model (Figure 6) or adding more nodes or hidden layers did not produce much better models nor produced desired improvements. Thus, the final model and the production system utilized a formula that was based on the logistic regression.

A neural network development requires a significant statistical analysis in order to understand the data and process flows. Most practitioners apply the stepwise, the backward regression, and decision trees before the neural network modeling. Furthermore, the neural network models cannot be directly applied in business interpretation processes, which can eliminate the neural model from the consideration[4]. Therefore, only significantly better prediction rates can justify a neural network model.

TOOLS AND MODELING DECISIONS

The case described in this paper is not complete. Hopefully it provides enough arguments for scientific based modeling and methodology selection, which should be driven by results and their diagnostics.

Process modeling and data mining are not for amateurs. The use and abuse of the advanced mining tools, including the Enterprise Miner, can lead to disastrous results. Even, a complete rookie can create a model and generate predictions using the SAS Enterprise Miner. This advanced tool relies heavily on defaults and their combinations along the process flow that can make a big difference in the final outcome. Furthermore, the Enterprise Miner diagnostics is not as visible as the traditional display of diagnostics in the SAS/STAT procedures. User has to look for diagnostics and in some cases it is not as extensive or buried in reports. The statistical diagnostics has to compete for user's attention with new more business oriented diagnostics in a graphical format. Finally, erroneous data manipulation and transformation can render the modeling part useless.

COMPUTER DECISION SYSTEMS

Production implementation of decision support systems introduces an extra layer of complexity. The system requirements in cases when humans are replaced by intelligent systems expand to lesser-known regions of the systems development. User interfaces, data gathering and simple reports are not enough. Typically, when models are developed the implementation follows without delays. However, after an initial period, the system sponsors, business analysts, and support people start to realize that model implementation was just a good start along a bumpy road.

In a manual process, a series of rules, user's perceptions, and unwritten rules are applied to each transaction. Mistakes are made, though their impact is minimized because in manual processes these errors are inconsistent. For example, different users apply different variations of rules and their business knowledge changes in time. Thus, even a weak set of rules does not have to generate wide spread problems. Contrary, computer based models make decisions for many transactions in a consistent way. However, such systems require their models to be replaced or modified when the maintenance is scheduled. What happens if there are drastic changes in the environment and the maintenance is not applied? The answer is simple. The non-maintained system happily generates more or less useless predictions, which are applied as designed. Furthermore, even the best models make mistakes, and an array of non-believers and the 'old guard' team will find plenty of examples 'proving' that the system is useless.

Thus, decision systems should be equipped with a reporting and visualization infrastructure based on business dimensions, multidimensional reporting, graphics, statistical diagnostics, and the system performance diagnostics. The business dimensions (product, customer/geography, management level, time, and performance diagnostics) should drive interactive reporting from a multidimensional database at a user defined intersection of these dimensions (e.g. EIS based reporting).

CONCLUSIONS

Model based computer decision systems require appropriate model selection, model diagnostics, model maintenance schedule, and information visualization. The last one includes visualization in form of tables and graphs at intersection of requested business dimension and time. Diagnostics of model performance at different levels of management must be carefully designed and implemented.

Neural network models should be considered along with other model types (decision trees, regression models, etc.). The final model should be selected based on the performance and whether the model type is appropriate for the considered problem. In addition, separate models should be considered for market segments that are characterized by different factors. Furthermore, simplicity, interpretation ability, maintenance requirements, and stability of models should influence both the modeling and the development approach.

REFERENCES

1. SAS/STAT User's Guide, Version 6, Fourth Edition, ISBN 1-55544-376-1.
2. Enterprise Miner Software, Online Tutorial (SAS V8).
3. Data Mining Using Enterprise Miner Software: A Case Study Approach, First Edition, ISBN-58025-641-4, February 2000.
4. Michael J.A. Berry and Gordon Linoff, Data Mining Techniques, Data Marketing, Sales, and Customer Support, John Wiley & Sons, Inc., ISBN 0-471-17980-9, 1997.

CONTACT INFORMATION

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

Leon L. Fedenczuk
Gambit Consulting Ltd.
268 Silvercrest Dr. N.W.
Calgary, Alberta, T3B 3A4 Canada
Tel: (403) 288-6764
Email: leon@gambitconsulting.com