A Preliminary Investigation Of Decision Tree Models For Classification Accuracy Rates And Extracting Interpretable Rules In The Credit Scoring Task: A Case Of The German Data Set

Jozef Zurada, University of Louisville
Peng C. Lam, Edith Cowan University

ABSTRACT

For many years lenders have been using traditional statistical techniques such as logistic regression and discriminant analysis to more precisely distinguish between creditworthy customers who are granted loans and non-creditworthy customers who are denied loans. More recently new machine learning techniques such as neural networks, decision trees, and support vector machines have been successfully employed to classify loan applicants into those who are likely to pay a loan off or default upon a loan. Accurate classification is beneficial to lenders in terms of increased financial profits or reduced losses and to loan applicants who can avoid overcommitment. This paper examines a historical data set from consumer loans issued by a German bank to individuals whom the bank considered to be qualified customers. The data set consists of the financial attributes of each customer and includes a mixture of loans that the customers paid off or defaulted upon. The paper examines and compares the classification accuracy rates of three decision tree techniques as well as analyzes their ability to generate easy to understand rules.

INTRODUCTION AND PRIOR LITERATURE

Credit scoring process is inherently difficult and unstructured due to the complex nonlinear relationships between independent variables that interact with each other, their combined intricate influence on the dependent variable, and various forms of risks involved. The most harmful risk to the party approving credit is the nonpayment of obligations when they are due. Simultaneously, the payoff associated with a correct credit-risk decision is high. Due to the complexity of credit risk assessment, a bank that provided data for this paper experienced a default rate of thirty percent (30%), even though the bank must have used some credit scoring model to eliminate bad loans. Some of the bad loans could be avoided by using more discriminating credit risk assessment techniques.

In last years the banking industry and lenders have experienced a rapid growth due to large increases in home mortgages, home equity loans, car-loans, and credit card debts to name a few. With this growth, however, there have been significant losses for delinquent and non-performing loans. For example, Manufacturers Hanover's $3.5 billion commercial property portfolio was burdened with $385 million in non-performing loans (Rosenberg and Gleit, 1994). In 1991, $1 billion of Chemical Bank's $6.7 billion in real estate loans were delinquent and the bank held $544 million in foreclosed property. In 1994 in the United Kingdom about 12% of retail expenditure was made using credit cards, amounting to a total of about 36 billion British pounds (Hand and Henley, 1997). This dynamic is also highly relevant to the former East-block Central and East European countries, now members of the European Union. For example, in 2004 alone, Czek and Slovak banks recorded 33.8% and 36.7% increases in their retail loans, respectively (Vojtek and Kocenda, 2006). In response, to support their credit decisions many lenders are developing new credit scoring models based on new techniques such as neural networks, decision trees, support vector machines that complement traditional statistical techniques such as logistic regression and discriminant analysis. The ultimate objective of these new models is to increase accuracy in loan-granting decisions, so that more creditworthy applicants are granted credit, thereby increasing profits, and non-creditworthy applicants are denied credit, thus decreasing losses. Thus, even a slight improvement in making a reliable discrimination, between those who are likely to repay the loan and those who are not, would be highly desired and would lead to significant profits.

The literature on the credit scoring topic is quite abundant and for a more complete literature review, the reader is referred to Zurada (2007). Although we investigated and analyzed a great deal of papers, in this study we only describe several works published recently. Seow and Thomas (2006) contend that credit scoring is employed by financial institutions to minimize the probability of losses (loan defaults) with the ultimate goal of maximizing financial profits that are achieved when customers to whom the loans were extended by lenders actually accept the offers. They develop a model of adaptive dynamic programming to estimate a customer take-up probability distribution and use Bayesian updating learning techniques to ensure loan acceptances which maximize the lender's profit. Yang (2007) uses adaptive kernel learning methods (support vector machines) and kernel attribute selection techniques to enhance the predictive performance of the scoring model and add transparency to the final model, respectively. The author's approach does not require using variable reduction techniques and unlike the traditional logistic regression and discriminant analyses it handles a large number of attributes well, deals efficiently with inherent nonlinearity among variables, and is not extremely sensitive to small sample sizes. Using the 10- and 100-fold cross validation methods, the author tests his/her approach on two real world data sets containing a large number of attributes and samples, and concludes that kernel methods yield the classification results of about 69% and 70% for "good" and "bad" loans, respectively; while the stepwise logistic regression analysis produced the accuracy rates of about 90% and 24% for "good" and "bad" classes, respectively. The 24% classification accuracy rate for "bad" loans is certainly unacceptable. In a series of 18 computer simulation experiments, Huang et al. (2006) compare almost a dozen of various classification techniques including a back-propagation and radial basis function neural network architectures, rule extraction techniques, case-based reasoning methods, Naïve Bayesian classifier, and ensemble classifiers. Using a bank's customers checking account information they obtain the overall classification accuracy rates between 70% and 81% for the dependent variable taking three distinct values: "declined", "risky", and "good". It appears that many other papers we studied concentrate on the overall classification accuracy rates that different tools produce as well as the percentages of "good" and "bad" loans classified correctly. In the available literature, the task of extraction useful rules from the created models as well as important attributes that have the best prediction power seem to be of the secondary importance. Rules could be easily explained to lenders who would need to approve their implementation as well as loan applicants as the reason for denying a loan.

This paper examines and compares the effectiveness of three decision tree models to distinguish between consumers who defaulted upon or paid off a loan. It also attempts to extract useful and easy to understand rules from the created models. Computer simulation has been conducted utilizing a data set provided by a German bank. The paper is organized as follows. The following section discusses the three decision tree techniques used in this study. Next we describe the data set used in this study and present the simulation experiments and results. Finally, the last section concludes the paper and gives some recommendations for future work.

THE DECISION TREE TECHNIQUES

One of a widely used and relatively simple classification method is a decision tree. A tree has three types of nodes: a root node, internal nodes, and terminal (leaf) nodes. In a binary tree, a top node is a root node that has no incoming edges and two outgoing edges (branches). Each internal node has exactly one incoming edge and two outgoing edges. Finally, each leaf or terminal node has exactly one incoming edge and no outgoing edges. Each leaf node is assigned a class label. Edges coming of the root and other internal nodes contain attribute test conditions to separate cases that have different characteristics. Classifying a test record is straightforward once a decision tree has been built. Starting from the root node, one applies the test condition to the case and follows the appropriate branch based on the outcome of the test. This will lead one either to another internal node, for which a new test condition is applied, or to a leaf node. The class label associated with the leaf node is then assigned to the record.

Typical algorithms employ greedy search strategy to induce a reasonably accurate, but suboptimal, decision tree in a small amount of time. They recursively grow a decision tree by making a series of locally optimum decisions about which attribute to use for partitioning the data. The discussion below illustrates the Hunt's algorithm; on which other algorithms such as ID3, C4.5, and CART; are based (Tan *et al*., 2006).

Let D_t be the set of training patterns (examples) that are associated with node t and $y=\{y_1,y_2,\ldots,y_c\}$ be the class labels.

Step 1: If all the patterns in D_t belong to the same class y_t, then t is a leaf node labeled as y_t.

Step 2: If D_t contains patterns that belong to more than one class, an attribute test condition is selected to partition the records into smaller subsets. A child node is created for each outcome of the test condition and the patterns in D_t are distributed to the children based on the outcomes. The algorithm is then recursively applied to each child node.

Algorithms that build a decision tree should address two issues: (1) how should the training patterns be split and (2) how should the splitting procedure stop? Selecting the best split is based on the degree of disorder/impurity of the child nodes. For example, a node which contains only cases of class 1 or class 0 (for binary classification) has the smallest disorder = 0. Similarly, a node that contains an equal number of cases of class 1 and class 0 has the highest disorder = 1. Examples of impurity measures include

$$Entropy(t) = -\sum_{i=0}^{c-1} p(i\,|\,t)\log_2 p(i\,|\,t)$$

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i\,|\,t)]^2,$$

$$Classification\ error\,(t) = 1 - \max_i [p(i\,|\,t)],$$

where c is the number of classes, $0\log_2 0=0$ in entropy calculations, and p(i|t) is the fraction of cases belonging to class i at a given node t. The reference to node t can be omitted and the fraction can be expressed as p_i. In a 2-class problem, the class distribution at any node can be expressed as (p_0,p_1), where

$p_1=1-p_0$.

To find out how good is a test condition, the degree of disorder of the parent node is compared to the degree of disorder in the child nodes. The higher the gain Δ, the better the split.

$$\Delta = I(parent) - \sum_{j=1}^{k} \frac{N(v_j)}{N} I(v_j),$$

where $I(\cdot)$ is the disorder measure of a given node, N is the total number of cases at the parent node, k is the number of attribute values, and $N(v_j)$ is the number of cases associated with the child node, v_j.

Chi-squared splitting criteria measure the reduction in variability of the target distribution in the branch (child) nodes. Specifically, the likelihood ratio Pearson chi-squared test statistic is a measure of association between the categories of the dependent variable and the branch nodes. This test statistic can be used to judge the worth of the split; it measures the difference between the observed cell counts and what would be expected if the branches and target classes were independent. We used a default significance level of 0.20.

For more details on decision trees, refer to (Mitchell, 1997; Han and Kamber, 2001; Giudici, 2003; SAS Enterprise Miner: http://www.sas.com; and Tan et al., 2006).

DATA SET USED IN THE STUDY

To create credit scoring models, lenders have employed data obtained from various loan-granting contexts. Typically, such data sets are unbalanced as the samples representing "good" applicants are overrepresented. In our study we used qualitative variables (Table 1) that are similar to the ones used in several previous studies. For example, we employed a data set provided by an important southern German bank with 1000 loan applicants and 21 variables. It consists of 700 records of customers who paid the loan off and 300 records of customers who defaulted. Fahrmeir and Hamerle (1994) and Giudici (2003) used the same data set in their studies. West (2000) and West et al. (2005) used a very similar, but "enhanced" data set with 24 independent variables and the same number of samples. Desai *et al*.'s (1996) used 3 data sets with 18 independent features and about 900 cases containing data about the customers of three credit unions. The Australian scoring data used by Quinlan (1987) had similar attributes but was more balanced with 307 and 383 observations of each outcome. It is important to keep in mind that the vast majority of data sets used in credit scoring tasks have a bias as they contain only those applicants who actually received a loan. In other words, the models were built on data representing customers that banks deemed to be qualified (creditworthy). There are other individuals who did not obtain a loan and we do not know whether they would have repaid a loan.

Table 1
Characteristics of the German Data Set. (Good_Bad – Dependent Variable, All Others – Independent Variables.)

Variable Name Abbreviation	Variable Full Name
Good_Bad	Credit Rating Status
Checking	Checking Account Balance
Duration	Length of Loan
History	Credit History
Purpose	Reason for Loan Request
Amount	Credit Amount
Savings	Savings Account Balance
Employed	Time at Present Employment
InstallP	Debt as Percent of Disposable Income
Marital	Marital Status and Gender
CoApp	Co-applicant or Guarantor?
Resident	Years at Current Address
Property	Collateral Property for Loan
Age	Age of Applicant [in years]
Other	Other Installment Loans
Housing	Rent/Own
ExistCr	Number of Accounts at this Bank
Job	Employment Status
Depends	Humber of Dependents
Telephon	Has a telephone?
Foreign	Foreign Worker?

COMPUTER EXPERIMENTS AND RESULTS

As mentioned the original data set used in this study contains 21 attributes and 1000 loan applicants divided into 700 “good” loans and 300 of “bad” loans. From this data set, we randomly generated 10 training sets and 10 test sets. We allocated 67% and 33% of samples to each training set and test set, respectively. Each of these data sets had approximately the same proportion of “good” to “bad” loans. We used three decision tree models, i.e., chi-square, entropy reduction, and Gini reduction, to build the trees. We recorded the results for 2 most common probability cutoffs: .3 (30%) and .5 (50%). We averaged the classification accuracy rates across 10 test data sets and computed standard deviations to test the stability/dispersion of the rates for the three methods. The Gini reduction technique outperformed the two other techniques in the overall classification accuracy rates, and the chi-square method was the best at identifying the relative importance of the tree attributes as well as generating simple to interpret rules.

Table 2 presents the classification accuracy rates for each of the 10 random configurations of the test sets for three types of the decision trees for the 0.3 and .5 probability thresholds. The table also presents the average percentage classification accuracy rates measured over 10 runs and the standard deviation of the rates for the three models. For the .3 cut-off level the error in granting a loan to “bad” customer costs 3.3 times more than denying a loan to “good” customer. For the .5 threshold the Type I and II errors have the same cost. For the 0.3 cut-off the Gini reduction method outperforms the other two methods producing the average overall classification accuracy of 72.5%. It also does so for the “good” loans (77.4%). However, this method is slightly worse than the chi-square method in detecting “bad” loans (61% versus 63.6%). By looking at the standard deviations of the rates measured over 10 runs it should be noted that the Gini reduction generates the least dispersion for the overall, “good”, and “bad” classification rates (4.3%, 10%, and 9.4%). For the .5 cut-off level the Gini reduction method seems to be the best than two other methods in the overall classification accuracy (75.1%) and in detecting “bad” loans (43.6%).

Table 2
Percentage Correct Classification Accuracy for 10 Random Test Sets for Three Decision Tree Models at Two Probability Cut-off Levels. (O – Overall, G – Good Loans, B – Bad Loans)

Cutoff	Run #	DT-ChiSq			DT-EntRed			DT-GiniRed		
30%		**O**	**G**	**B**	**O**	**G**	**B**	**O**	**G**	**B**
	1	74.7%	89.7%	40.0%	75.9%	88.4%	47.0%	77.1%	90.9%	45.0%
	2	75.8%	87.0%	49.5%	65.5%	59.7%	78.8%	70.9%	71.6%	68.7%
	3	67.5%	67.2%	68.0%	75.0%	83.2%	56.0%	75.3%	83.6%	56.0%
	4	72.8%	80.1%	56.0%	73.7%	76.6%	67.0%	77.9%	89.2%	52.0%
	5	67.1%	64.1%	74.0%	67.1%	64.1%	74.0%	69.8%	71.4%	66.0%
	6	60.5%	53.0%	78.0%	59.0%	48.3%	84.0%	70.2%	74.6%	60.0%
	7	70.2%	74.6%	60.0%	68.1%	69.4%	65.0%	69.0%	70.3%	66.0%
	8	63.7%	57.3%	78.8%	64.4%	58.2%	78.8%	64.4%	58.2%	78.8%
	9	63.1%	56.5%	78.8%	58.0%	46.6%	84.8%	75.8%	82.8%	59.6%
	10	73.3%	82.3%	52.5%	76.1%	85.3%	54.5%	74.5%	81.8%	57.6%
	Average	**68.9%**	**71.2%**	**63.6%**	**68.3%**	**68.0%**	**69.0%**	**72.5%**	**77.4%**	**61.0%**
	StDev	**5.3%**	**13.4%**	**13.9%**	**6.7%**	**15.1%**	**13.2%**	**4.3%**	**10.0%**	**9.4%**
50%										
	1	74.7%	89.7%	40.0%	75.9%	88.4%	47.0%	77.1%	90.9%	45.0%
	2	75.8%	87.0%	49.5%	75.8%	87.9%	47.5%	77.0%	86.2%	54.5%
	3	72.6%	99.1%	11.0%	75.9%	86.2%	52.0%	75.3%	83.6%	56.0%
	4	74.3%	87.4%	44.0%	74.3%	94.8%	27.0%	78.2%	90.0%	51.0%
	5	72.5%	87.9%	37.0%	72.5%	87.9%	37.0%	72.8%	87.4%	39.0%
	6	75.6%	93.1%	35.0%	59.0%	48.3%	84.0%	70.2%	74.6%	36.0%
	7	70.2%	74.6%	60.0%	74.1%	95.7%	24.0%	73.8%	95.7%	23.0%
	8	72.8%	90.5%	31.3%	74.9%	94.8%	28.3%	73.4%	91.4%	31.3%
	9	72.2%	96.6%	15.2%	71.9%	89.7%	30.3%	77.3%	86.2%	56.6%
	10	75.5%	92.2%	36.4%	77.6%	92.6%	42.4%	75.8%	89.6%	43.4%
	Average	**73.6%**	**89.8%**	**35.9%**	**73.2%**	**86.6%**	**41.9%**	**75.1%**	**87.6%**	**43.6%**
	StDev	**1.8%**	**6.7%**	**14.6%**	**5.3%**	**13.9%**	**17.7%**	**2.5%**	**5.7%**	**11.3%**

As far as the predictive power of variables, all the three methods consistently identify that Checking Account Balance, Length of Loan, Savings Account Balance, and Reason for Loan Request are the four variables that have the highest relative importance (Tables 3-5). Over 10 test sets the chi-square method used only 9 attributes in building the trees, whereas the other two methods employed 11 and 14 attributes. As a result, the trees built using the chi-square method generate simpler and fewer rules than the trees built by other two methods. An example of such a tree generated for run 10 is presented in Figure 1 and the examples of rules that could be read out from the tree are in Table 6. One could see that these rules are easy to interpret and explain. This situation presents a tradeoff between the method that produces the best classification accuracy rates (Gini reduction) and the method that produces easier to understand rules (chi-square). The preliminary results indicate that if simplicity and understandability of the generated rules is of paramount importance, lenders would be willing to use the trees generated by the chi-square method. However, if the overall classification accuracy is more important, financial institutions would be using the trees built by the Gini reduction method.

Table 3
The relative importance of the attributes generated by the Chi – Square Method for 10 random test sets.

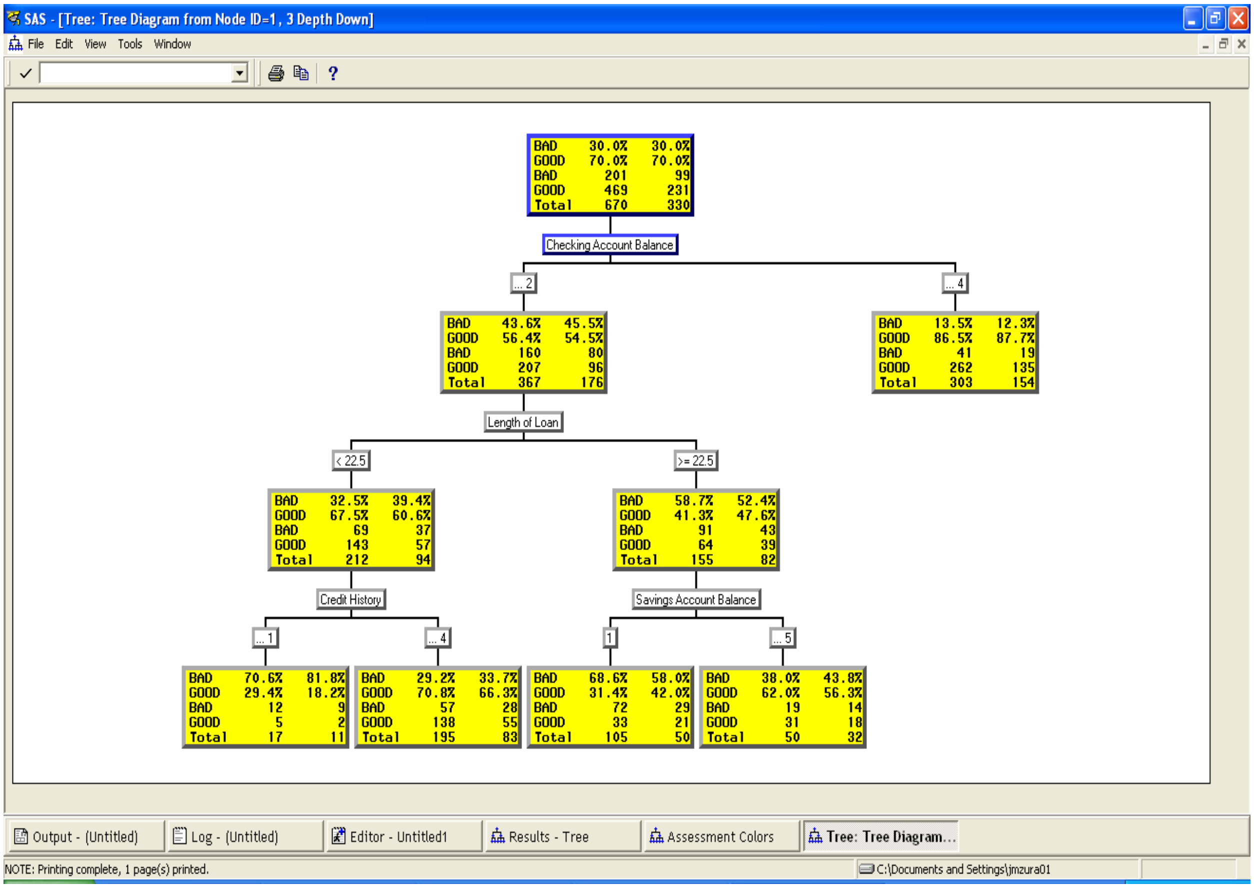
Run #/ Variable Name	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Checking	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Duration	0.704	0.549	0.443	---	0.66	0.486	0.504	0.603	0.416	0.639
Savings	0.562	0.435	0.425	0.445	0.454	0.313	0.488	0.522	0.352	0.459
History	0.555	0.35	---	0.439	---	0.349	---	0.537	---	0.422
Employed	---	0.333	---	---	---	0.365	---	---	---	---
Amount	---	0.283	---	---	---	---	---	---	---	---
Property	---	---	---	0.516	---	---	---	---	0.465	---
Telephon	---	---	---	---	---	---	---	---	---	0.292
Purpose	---	---	---	---	---	---	---	---	---	0.468

Table 4
The relative importance of the attributes generated by the Entropy Reduction Method for 10 random test sets.

Run #/ Variable Name	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Checking	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Duration	0.607	0.59	0.512	0.438	0.66	0.516	0.643	0.698	0.568	0.856
Savings	---	0.445	0.418	0.5	0.454	0.288	0.547	0.522	0.425	0.467
History	0.645	0.363	0.31	0.374	---	---	0.422	0.537	---	0.505
Purpose	---	0.327	0.455	---	---	---	0.27	0.311	---	0.793
Amount	---	---	---	0.373	---	0.397	---	0.294	---	0.689
Employed	---	---	---	---	---	0.359	0.295	---	---	0.294
Other	---	---	0.289	---	---	---	0.327	---	---	0.428
Telephon		0.151	---	---	---	---	---	0.16	---	---
Property	---	---	---	0.484	---	---	---	---	---	---
Age	---	---	---	---	---	---	---	0.248	---	---

Table 5
The relative importance of the attributes generated by the Gini Reduction Method for 10 random test sets.

Run #/ Variable Name	1	2	3	4	5	6	7	8	9	10
Checking	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Duration	0.607	0.559	0.512	0.489	0.66	0.623	0.636	0.602	0.512	0.798
Savings	0.634	0.445	0.51	0.424	0.454	0.313	0.501	0.522	0.4	0.46
History	0.645	0.35	0.31	0.472	0.465	---	0.357	0.537	0.377	0.422
Purpose	0.54	0.483	0.457	---	0.2	0.417	0.231	0.311	0.338	0.643
Amount	0.649	0.285	---	---	---	0.32	---	0.184	0.445	0.378
Telephon	---	0.151	---	0.16	---	---	---	0.16	---	---
Property	---	---	---	---	---	0.255	0.388	---	0.293	---
Age	---	---	0.221	---	---	0.387	---	---	---	---
ExistCr	---	---	---	---	---	---	0.249	---	0.344	---
Other	---	0.285	---	---	---	---	---	---	0.319	---
CoApp	---	---	---	---	0.243	---	---	---	---	---
Employed	---	---	---	---	---	0.365	---	---	---	---
Housing	---	---	---	---	---	---	---	---	---	0.283



NOTE: Printing complete, 1 page(s) printed.

Figure 1
The Tree Generated by the Chi-squared Method for Run 10. (The 2nd and 3rd Number Columns in a Rectangle Represent the Training Set and Test Set, Respectively. Depth of the Tree =3)

Table 6
The Examples of Four Rules Generated by the Tree Shown in Figure 1.

Rule #	Rule

1	IF Checking Account Balance is 3 or 4
	THEN "Bad": 13.5%, "Good": 86.5%

2	IF Credit History is 0 or 1 AND Length of Loan < 22.5 AND Checking Account Balance is 1 or 2
	THEN "Bad": 70.6%, "Good": 29.4%

3	IF Credit History is 2, 3 or 4 AND Length of Loan < 22.5 AND Checking Account Balance is 1 or 2
	THEN "Bad": 29.2%, "Good": 70.8%

4	IF Savings Account Balance is 2, 3, 4 or 5 AND 22.5 <= Length of Loan
	AND Checking Account Balance is 1 or 2
	THEN "Bad": 38.0%, "Good": 62.0%

Notes to Table 6 – The values that the variables take:
(a) Checking Account Balance
1 = less than 0 DM
2 = more than 0 but less than 200 DM
3 = at least 200 DM, and
4 = no checking account
(b) Credit History
0 = no loans taken / all loans paid back in full and on time
1 = all loans at this bank paid back in full and on time
2 = all loans paid back on time until now
3 = late payments on previous loans
4 = critical account / loans in arrears at other banks
(c) Length of Loan [in months]
(d) Savings Account Balance
1 = less than 100 DM
2 = at least 100, but less than 500 DM
3 = at least 500, but less than 1000 DM
4 = at least 1000 DM
5 = unknown / no savings account

SUGGESTIONS FOR FUTURE RESEARCH

In further research we propose to run computer simulation for a larger number of randomly generated training and test sets (about 30 each). We would build about 30 models on each of the training sets and test the classification accuracy rates as well as consistency in identifying important variables on the same number of test sets. We believe that the average classification accuracy rates measured over the number of runs would stabilize after about 30 runs. In addition to the already widely used neural network models, we propose to implement models based on support vector machines and neuron-fuzzy systems. The latter can generate fuzzy rules that one could interpret. We also propose to run computer simulation for several data sets drawn from various credit scoring contexts to make the obtained results more reliable and generalizable.

REFERENCES

1. Desai, V.S., Crook, J.N., and Overstreet, G.A.Jr, A Comparison of Neural Networks and Linear Scoring Models in the Credit Union Environment, *European Journal of Operation Research*, Vol. 95, pp. 24-37, 1996.
2. Fahrmeir, L., and Hamerle, A., *Multivariate Statistical Modeling Based on Generalized Linear Programs*, Springer-Verlag, Berlin, 1994.
3. Giudici, P., *Applied Data Mining: Statistical Methods for Business and Industry*, John Wiley & Sons Ltd., Chichester, West Sussex, England, 2003.
4. Han, J., and Kamber, M., *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, San Francisco, CA, 2001.

5. Hand, D., and Henley, W., Statistical Classification Method in Consumer Scoring: A Review, *Journal of the Royal Statistical Society, Series A (Statistics in Society)*, Vol. 160, 523-541, 1997.
6. Hsieh, N-C, An Integrated Data Mining and Behavioral Scoring Model for Analyzing Bank Customers, *Expert Systems with Applications*, 27, 623-633, 2004.
7. Huang, Y-H., Hung, C-M, Jiau, H.C., Evaluation of Neural Networks and Data mining Methods on a Credit Assessment Task for Class Imbalance Problem, *Nonlinear Analysis: Real Worlds Applications*, 7, 720-747, 2006.
8. Mitchell, T.M., *Machine Learning*, WCB/McGraw-Hill, Boston, Massachusetts, 1997.
9. Quinlan, J.R., Simplifying Decision Trees, *International Journal of Man-Machine Studies*, Vol. 27, pp. 221-234, 1987.
10. Rosenberg, E., and Gleidt, A., Quantitative Methods in Credit Management: A Survey, *Operations Research*, Vol. 42, 589-613, 1994.
11. Seow, H-V., and Thomas, L.C., Using Adaptive Learning in Credit Scoring to Estimate Take-up Probability Distribution, *European Journal of Operational Research*, 173, 880-892, 2006.
12. Tan, P-N., Steinbach, M., and Kumar, V., *Introduction to Data Mining*, Addison-Wesley, 2006.
13. West, D., Neural Network Credit Scoring Models, *Computers & Operations Research*, Vol. 27, pp. 1131-1152, 2000.
14. Vojtek, M, and Kocenda, E, Credit Scoring Methods, *Czech Journal of Economics and Finance*, Vol. 56, Issue 3-4, 152-167, 2006.
15. West, D., Dellana, S., and Qian, J., Neural Network Ensemble Strategies for Financial Decision Applications, *Computers & Operations Research*, Vol. 32, 2543-2559, 2005.
16. Yang, Y., Adaptive Credit Scoring with Kernel Learning Methods, *European Journal of Operational Research*, Vol. 183, Issue 3, 1521-1536, 2007.
17. Zurada, J., Rule Induction Methods for Credit Scoring, *Review of Business Information Systems*, Vol. 11, 2, 11-22, 2007.

NOTES