ELSEVIER

Expert Systems with Applications 29 (2005) 41–47

Expert Systems with Applications

www.elsevier.com/locate/eswa

Building credit scoring models using genetic programming

Chorng-Shyong Ong[a], Jih-Jeng Huang[a], Gwo-Hshiung Tzeng[b,c,*]

[a]*Department of Information Management, National Taiwan University, Taipei, Taiwan*
[b]*Institute of Management of Technology, National Chiao Tung University, Ta-Hsuch Rd, Hsunchu 300, Hsinchu 1001, Taiwan*
[c]*College of Management, Kainan University, Taoyuan, Taiwan*

Abstract

Credit scoring models have been widely studied in the areas of statistics, machine learning, and artificial intelligence (AI). Many novel approaches such as artificial neural networks (ANNs), rough sets, or decision trees have been proposed to increase the accuracy of credit scoring models. Since an improvement in accuracy of a fraction of a percent might translate into significant savings, a more sophisticated model should be proposed to significantly improving the accuracy of the credit scoring mode. In this paper, genetic programming (GP) is used to build credit scoring models. Two numerical examples will be employed here to compare the error rate to other credit scoring models including the ANN, decision trees, rough sets, and logistic regression. On the basis of the results, we can conclude that GP can provide better performance than other models.

Keywords: Credit scoring; Artificial neural network (ANN); Decision trees; Genetic programming (GP); Rough sets

1. Introduction

Credit scoring models have been widely used by financial institutions to determine if loan customers belong to either a good applicant group or a bad applicant group. The advantages of using credit scoring models can be described as the benefit from reducing the cost of credit analysis, enabling faster credit decision, insuring credit collections, and diminishing possible risk (Lee, Chiu, Lu, & Chen, 2002; West, 2000). Since an improvement in accuracy of a fraction of a percent might translate into significant savings (West, 2000), a more sophisticated model should be proposed to significantly improve the accuracy of the credit scoring model in this paper.

In order to obtain a satisfied credit scoring model, numerous methods have been proposed. Roughly, these methods can be classified to parametric statistical methods (e.g. discriminant analysis and logistic regression), non-parametric statistical methods (e.g. k nearest neighbor and decision trees), and soft-computing approaches (e.g. artificial neural network (ANN) and rough sets). Recently, ANNs are the most popular tool used for credit scoring and has been reported that its accuracy is superior to that of traditional statistical methods in dealing with credit scoring problems, especially in regards to non-linear patterns (Desai, Crook, & Overstreet, 1996, 1997; Mahlhotra & Malhotra, 2003; Jensen, 1992; Piramuthu, 1999). However, on the other hand, ANN has been criticized for its poor performance when incorporating irrelevant attributes or small data sets (Castillo, Marshall, Green, & Kordon, 2003; Feraud & Cleror, 2002; Nath, Rajagopalan, & Ryker, 1997).

In order to build an effective discriminant function, two issues should be considered. First, the relationships among attributes and classes may be linear or non-linear. Second, the irrelevant attributes should be removed in order to increase the accuracy of the classification model. In this paper, GP is employed to automatically and heuristically determine the adequate discriminant functions and the valid attributes simultaneously. In addition, unlike ANNs which are only suited for large data sets, GP can perform well even in small data sets (Nath et al., 1997).

In order to efficiently obtain the discriminant function, the data set is preprocessed by discretization. Two real-world

* Corresponding author. Address: Institute of Management of Technology, National Chiao Tung University, Ta-Hsuch Rd, Hsunchu 300, Hsinchu 1001, Taiwan. Tel.: +886 3571212157505; fax: 886 35753926.
E-mail address: ghtzeng@cc.nctu.edu.tw (G.-H. Tzeng).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2005.01.003

cases will be used below to compare the accuracy rate to other classification models including the logistic regression model, ANN, decision trees and rough sets. On the basis of the results, we can conclude that GP can provide better performance than other models.

The rest of this paper is organized as follows. Section 2 describes the models for credit scoring. Discretization and genetic programming are proposed in Section 3. Two real-world examples are used to demonstrate the proposed method in Section 4. Discussions are presented in Section 5 and conclusions are in Section 6.

2. Credit scoring models

In this section, we describe three popular models used in building credit scoring models. The first model is logistic regression, which is mostly used for classification problems in the area of statistics. The second model is ANN, which is known for its excellent ability of learning non-linear relationships in a system. The third model is rough sets, which is one kind of induction based algorithms, and has been widely used in classification problems since 1990s.

2.1. Logistic regression

Logistic regression model is one of the most popular statistical tools for classification problems. Logistic regression model, unlike other statistical tools (e.g. discriminant analysis or ordinary linear regression), can suit various kinds of distribution functions such as Gamble, Poisson, normal, etc. (Press & Wilson, 1978) and is more suitable for the credit scoring problems. In additional, in order to increase its accuracy and flexibility several methods have been proposed to extend the traditional binary logistic regression model, including multinomial logistic regression model (Agresti, 1990; Aldrich & Nelson, 1984; DeMaris, 1992; Knoke & Burke, 1980; Liao, 1994) and logistic regression model for ordered categories (McCullagh, 1980). Therefore, the generalized logistic regression model is the general form of binary logistic regression model and multinomial logistic regression model.

Let a p-dimensional explanatory variables $\boldsymbol{x}' = (x_1, x_2, \ldots, x_p)$ and Y be the response variable with categories $1,2,\ldots,r$. Then the multinomial logistic regression model be given by the equation

$$\text{logistic}(\pi) = \ln\left[\frac{P(Y=j|\boldsymbol{x})}{P(Y=k|\boldsymbol{x})}\right] = \boldsymbol{x}'\beta_j, \quad 0 \le j \le r, \quad j \ne k \tag{1}$$

where β_j is a $(p+1)$ vector of the regression coefficients for the jth variable.

Let the last response level be the reference level and then the response probabilities $\pi_1, \pi_2, \ldots, \pi_r$ can be calculated by the equations

$$\pi_r \equiv P(Y=r|\boldsymbol{x}) = \frac{e^{\boldsymbol{x}'\beta_r}}{\sum_{l=1}^{r} e^{\boldsymbol{x}'\beta_l}} = \frac{e^{\boldsymbol{x}'\beta_r}}{e^{\boldsymbol{x}'\beta_r} + \sum_{l=1}^{r-1} e^{\boldsymbol{x}'\beta_l}} = \frac{1}{1 + \sum_{l=1}^{r-1} e^{\boldsymbol{x}'\beta}} \tag{2}$$

$$\pi_j \equiv P(Y=j|\boldsymbol{x}) = \pi_r e^{\boldsymbol{x}'\beta_j}, \quad 1 \le j \le r-1 \tag{3}$$

where l is a response level, and

$$l = l(\beta_j, 1 \le j \le r, j \ne k) = \sum_{i=1}^{n} \ln(P(Y=y_i|\boldsymbol{x}_i)), \quad l \in [1,2,\ldots,r] \tag{4}$$

is the ln likelihood for the multinomial logistic regression model and $\{(y_i, \boldsymbol{x_i}), 1 \le i \le n\}$ denotes the sample of n objects. When the category is equal to two, the multinomial logistic regression model reduces to a binary logistic regression model.

Although logistic regression model can perform well in many applications, when the relationships of the system are non-linear, the accuracy of logistic regression decreases and ANN has been proposed to deal with this problem.

2.2. Artificial neural network

Artificial neural networks were developed to mimic the neurophysiology of the human brain to be a type of flexible non-linear regression, discriminant, and clustering models The architecture of ANN can usually be represented as a three-layer system, named input, hidden, and output layers. The input layer first processes the input features to the hidden layer. The hidden layer then calculates the adequate weights by using the transfer function such as hyperbolic tangent, softmax, or logistic function before sending to the output layer.

Combining many computing neurons into a highly interconnected system, we can detect the complex non-linear relationship in the data. The simple three-layer perceptron, which is most used in credit scoring problems, can be depicted as shown in Fig. 1.

Recently, ANN has been widely used in credit scoring problems, and it has been reported that its accuracy is superior to the traditional statistical methods such as discriminant analysis and logistic regression (Desai et al., 1996, 1997; Jensen, 1992; Mahlhotra & Malhotra, 2003; Piramuthu, 1999). However, as mentioned previously, ANN has been criticized for its poor performance when existing irrelevant attributes or small data sets. Although many methods have been proposed to deal with the problem of variable selection (Feraud & Cleror, 2002; Nath et al., 1997), it is time waste and makes the model more complicated. In addition, other scholars are criticized

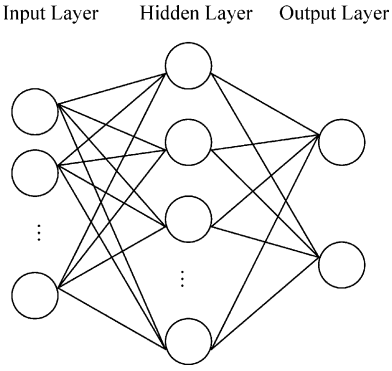

Fig. 1. Three-layer neural network.

the limitations of its long training process in designing the optimal network's topology in credit scoring problems (Chung & Gray, 1999; Craven & Shavlik, 1997).

2.3. Rough sets

Rough sets, originally proposed by Pawlak (1982), is a mathematical tool used to deal with vagueness or uncertainty Compared to fuzzy sets, there are some advantages to rough set theory (Pawlak, Grzymala-Busse, Slowinski, & Ziarko, 1995). One main advantage is that rough sets do not need any pre-assumptions or preliminary information about the data, such as the grade of membership function in fuzzy sets (Grzymala-Busse, 1988). Recently, rough set theory and fuzzy set theory have been used to complement or incorporate (Chakrabarty, Biswas, & Nanda, 2000; Mordeson, 2001; Radzikowska & Kerre, 2002) each other rather than to compete (Dubois & Prade, 1991). More detailed discussion about the process of rough set theory can refer to Walczak and Massart (1999).

The original concept of approximation space in rough sets can be described as follows.

Given an approximation space

$$\text{apr} = (U, A)$$

where U is the universe which is a finite and non-empty set, and A is the set of attributes. Then based on the approximation space, we can define the lower and upper approximations of a set.

Let X be a subset of U and the lower approximation of in A is

$$\underline{\text{apr}}(A) = \{x | x \in U, U/\text{Ind}(A) \subset X\} \tag{5}$$

The upper approximation of X in A is

$$\bar{\text{apr}}(A) = \{x | x \in U, U/\text{Ind}(A) \cap X \neq \phi\} \tag{6}$$

where

$$U/\text{Ind}(A) = \{(x_i, x_j) \in U \cdot U, f(x_i, a) = f(x_j, a) \quad \forall a \in A\} \tag{7}$$

Eq. (5) represents the least composed set in A containing X, called the best upper approximation of X in A, and Eq. (6) represents the greatest composed set in A contained in X, called the best lower approximation.

After constructing upper and lower approximations, the boundary can be represented as

$$BN(A) = \bar{\text{apr}}(A) - \underline{\text{apr}}(A) \tag{8}$$

According to the approximation space, we can calculate reducts and decision rules. Given an information system $\boldsymbol{I} = (U, A)$ then the reduct, RED(B), is a minimal set of attributes $B \subseteq A$ such that $r_B(U) = r_A(U)$ where

$$r_B(U) = \frac{\sum \text{card}(\underline{B}X_i)}{\text{card}(U)} \tag{9}$$

denotes the quality of approximation of U by B.

Once the reducts have been derived, overlaying the reducts on the information system can induce the decision rules. A decision rule can be expressed as $\phi \Rightarrow \theta$, where ϕ denotes the conjunction of elementary conditions, $\Rightarrow$ denotes '*indicates*', and θ denotes the disjunction of elementary decisions.

The advantage of the induction based approaches (e.g. rough sets and decision trees) is that it can provide the intelligible rules for decision-makers (DMs). These intelligible rules can help DMs to realize the contents of data sets. Although these induction methods have been well developed and successfully used in credit scoring problems (Ahn, Cho, & Kim, 2000; Beynon & Peel, 2001; Dimitras, Slowinski, Susmaga, & Zopounidis, 1999), the main problem of induction based methods is the ability of forecasting. It is clear that if a newly entered object does not match any rule, it cannot be determined which class it belongs to. Next, we described the concepts of GP which is used here to build the credit scoring models in Section 3.

3. Genetic programming

Genetic programming was proposed by Koza (1992) to automatically extract intelligible relationships in a system and has been used in many applications such as symbolic regression (Davidson, Savic, & Walters, 2003), and classification (Stefano, Cioppa, & Marcelli, 2002; Zhang & Bhattacharyya, 2004). The representation of GP can be viewed as a tree-based structure composed of the function set and terminal set. The function set is the operators, functions or statements such as arithmetic operators ($\{+, -, \times, |\}$) or conditional statements (If…then…) which are available in the GP. The terminal set contains all inputs, constants and other zero-argument in the GP tree. For example to express $xy + 3/x$, the GP tree can be represented as Fig. 2.

Once we initialize a population of the GP tree, the following procedures are similar to genetic algorithms

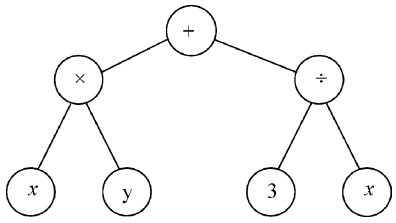

Fig. 2. The representation of a GP tree.

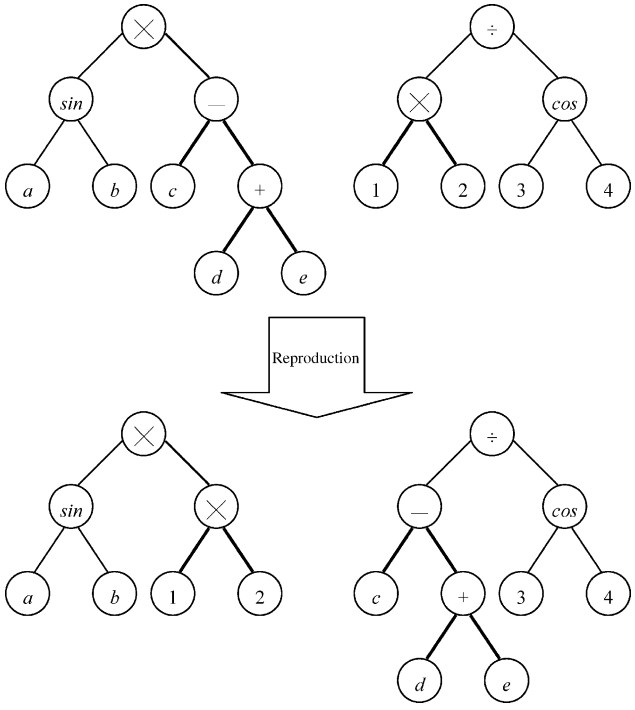

Fig. 3. The crossover operator of GP tree.

(GAs) including defining the fitness function, genetic operators such as crossover, mutation and reproduction, and the termination criterion, etc. Next, we introduce three main operators, crossover, mutation and reproduction, to show the procedures of finding the (approximate) optimal generation.

In GP, the crossover operator is used to swap the subtree from the parents to reproduce the children using mating selection policy rather than exchanging bit strings as in GAs. An example of a crossover in GP is shown in Fig. 3.

Similar to GAs, GP uses the mutation operator in order to avoid falling into the local optimal solution. The mutation operator is used to randomly choose a node in a subtree and replace it with a new created subtree randomly. Finally, a new generation can be reproduced from two parents using the reproduction operator to represent a better solution.

In order to determine the adequate discriminant function, the fitness function of GP can be described as

$$ff_i = \frac{\sum_i^N \mathrm{abs}(o_i - e_i)}{N} \tag{10}$$

where abs(·) denotes the absolute operator, o_i denotes the observed class and e_i denotes the expected class.

It should be highlighted that the function set and the terminal set should be varied enough to represent the relations among independent and response variables. Moreover, in order to satisfy the principle of parsimony, the depth of the GP-tree should also be limited.

In addition, in order to obtain the discriminant function efficiently and effectively, discretization of continuous attributes should be employed before GP. Many discretization algorithms such as Boolean reasoning algorithm, entropy algorithm and naïve algorithm have been proposed to deal with this problem (Shan, Hamilton, Ziarko, & Cercone, 1996; Wu, 1996). In this paper, Boolean reasoning algorithm is employed to determine the adequate discrete values. Next, two empirical cases will be used follow to compare the proposed method and the other models.

4. Empirical analysis

In this section, GP is compared to MLP, classification and regression tree (CART), C4.5, Rough sets, and logistic regression (LR) using two-real world data sets. The first data set includes Australian credit scoring data with 307 examples of credit worthy customers and 383 examples for credit unworthy customers. It contains 14 attributes, where six are continuous attributes and eight are categorical attributes. The second data set, called the German Credit Data Set, was provided by Prof. Hofmann in Hamburg.

Table 1
The discretization of the continuous attributes in Australian data set

Value	1	2	3	4	5	6	7	8	9	10	11
A2	[*, 17.75)	[17.75, 21.21)	[21.21, 22.38)	[22.38, 23.04)	[23.04, 23.34)	[23.34, 24.38)	[24.38, 27.92)	[27.92, 32.38)	[32.38, 37.38)	[37.38, 48.96)	[48.96, *)
A3	[*, 0.480)	[0.480, 1.793)	[1.793, 2.793)	[2.793, 4.020)	[4.020, 6.103)	[6.103, *)					
A7	[*, 0.145)	[0.145, 1.020)	[1.020, 2.145)	[2.145, *)							
A10	[*, 1)	[1, *)									
A13	[*, 23)	[23, 93)	[93, 171)	[171, 262)	[262, *)						
A14	[*, 13)	[13, *)									

Table 2
The parameter settings of GP

Parameter	Value
Population size	40
Fitness function	Eq. (10)
Function set	{+,−,×,sin,cos,≥,=,≤,and,or,not,if}
Terminal set	{Attributes,1,2,3,4,5,6,7,8,9,10,11,12,13,14}
Maximum number of generation	1000
Selection	Lexictour
Crossover rate	0.9
Mutation rate	0.01

It includes customer credit scoring data with 20 features, such as age, gender, marital status, credit history records, job, account, loan purpose, other personal information, etc. There are 700 records judged to be credit worthy and 300 records judged to be credit unworthy. Both data sets are made public from the UCI Repository of Machine Learning Databases, and are mostly used to compare the performance of various classification models.

The first step of the proposed method is to dissect the continuous attributes. For example of Australian data set, the results of the discretization can be shown as in Table 1. The discretization of the continuous attributes in German data set can be described as shown in Appendix A.

Next, we set the GP parameters of Australian data set as shown in Table 2 and the parameters of German data set can also be shown in Appendix B. In order to build the discriminant function as flexible as possible, we incorporate the logic operators into the function set. On the other hand, due to the range of the discretization values is from 1 to 14, we incorporate the constants from 1 to 14 into the terminal set.

Five sub-samples are used to compare the error rate of the credit scoring models. In addition, the holdout method is used for avoiding the problem of overfitting. The error rate of the test sets in both Australian and German data sets can be described as shown in Tables 3 and 4.

On the basis of the results, we can conclude that the proposed method outperforms to other models in our empirical analysis. In addition, ANN and logistic regression also well perform in this study and can be other choices for the credit scoring model. Next, we provide the discussions based on our implementation.

5. Discussions

Due to the huge growth rate of the credit industry, building an effective credit scoring model have been an important task for saving amount cost and efficient decision making. Although many novel approaches have been proposed, more issues should be considered for increasing the accuracy of the credit scoring model.

First, the irrelevant variables will destroy the structure of the data and decreases the accuracy of the discriminant function. Second, the credit scoring model should determine the correct discriminant function (linear or non-linear) automatically. Third, the credit scoring model should be useful in both large and small data sets. For above reasons, GP is used to build the credit scoring models in this paper.

On this basis of the simulated results, we can conclude that GP outperforms than other models. However, ANN and logistic regression can also provide the satisfied solutions and can be other alternatives. The accuracy of the induction based approaches (decision trees and rough set) is inferior in this study. It is clear that the decision rules are derived from

Table 3
The comparison of the credit scoring models in Australian data set

Australian data	Sample 1 (%)	Sample 2 (%)	Sample 3 (%)	Sample 4 (%)	Sample 5 (%)	Overall
GP	0.1111	0.1280	0.1304	0.1207	0.0966	0.1173
MLP	0.1352	0.1256	0.1352	0.1062	0.1014	0.1207
CART	0.1497	0.1256	0.1449	0.1400	0.1497	0.1419
C4.5	0.1594	0.1304	0.1400	0.1014	0.1159	0.1294
Rough sets	0.1382	0.1729	0.1538	0.1718	0.1777	0.1628
LR	0.1497	0.1449	0.1304	0.1304	0.1352	0.1381

Table 4
The comparison of the credit scoring models in German data set

German data	Sample 1 (%)	Sample 2 (%)	Sample 3 (%)	Sample 4 (%)	Sample 5 (%)	Overall
GP	0.2166	0.2266	0.2200	0.2433	0.2266	0.2266
MLP	0.2400	0.2382	0.2500	0.2433	0.2533	0.2449
CART	0.2765	0.2617	0.2435	0.3170	0.3721	0.2941
C4.5	0.2446	0.2500	0.2227	0.2926	0.3318	0.2683
Rough sets	0.2533	0.2649	0.2631	0.2353	0.2551	0.2543
LR	0.2400	0.2421	0.2500	0.2479	0.2500	0.2460

the training set. However, if a newly entered object within the test set does not match any rule, it cannot be determined which class it belongs to.

Compared to other models, we consider that GP is more suitable for the credit scoring problems for the following reasons. Unlike the traditional statistical methods need the assumptions of the data set and the attributes, GP is a non-parametric tool and suitable for any situations and data sets. Compared to ANNs, GP can determine the adequate discriminant function automatically rather than assigned the transfer function by decision-makers. In addition, GP can also select the important variable automatically. Finally, the discriminant function which is derived by GP can provide the better forecasting performance than the induction based algorithms.

6. Conclusions

Building a credit scoring model involves the problems of variable selection and model identification. Although many approaches have been proposed, a flexible and accurate method is limited. In this paper, GP is employed to build the discriminant function for the credit scoring problems. On the basis of the empirical results, we can conclude that GP is more flexible and performs better accuracy in the credit scoring problems significantly.

Appendix A

The discretization of the continuous attributes in Germen data set using Boolean reasoning algorithms can be described as shown in Table A1.

Appendix B

The parameters of German data set can be shown as in Table B1.

Table A1
The discretization of the continuous attributes in German data set

Value	1	2	3	4	5
Checking	[*, 1)	[1, 2)	[2, *)		
Duration	[*, 12)	[12, 23)	[23, 32)	[38, *)	
History	[*, 2)	[2, *)			
Amount	[*, 714)	[714, 1387)	[1387, 2045)	[2045, 3914)	[3914, *)
Saving	[*, 2)	[2, *)			
Employed	[*, 2)	[2, 3)	[3, *)		
Installp	[*, 4)	[4, *)			
Resident	[*, 2)	[2, 4)	[4, *)		
Age	[*, 27)	[27, 33)	[33, *)		
Existcr	[*, 2)	[2, *)			
Job	[*, 1)	[1, *)			

Table B1
The parameter settings of GP

Parameter	Value
Population size	40
Fitness function	Eq. (10)
Function set	{+,−,×,≥,=,≤,and,or,not,if}
Terminal set	{Attributes,1, 2, 3, 4, 5}
Maximum number of generation	40
Selection	Lexictour
Crossover rate	0.9
Mutation rate	0.01

References

Agresti, A. (1990). *Categorical data analysis*. New York: Wiley.

Ahn, B. S., Cho, S. S., & Kim, C. Y. (2000). The integrated methodology of rough set theory and artificial neural network for business failure prediction. *Expert Systems with Applications*, *18*(2), 65–74.

Aldrich, J. H., & Nelson, F. D. (1984). *Linear probability, logit, and probit models*. Beverly Hills, CA: Sage.

Beynon, M. J., & Peel, M. J. (2001). Variable precision rough set theory and data discretisation an application to corporate failure prediction. *OMEGA: the International Journal of Management Science*, *29*(6), 561–576.

Castillo, F., Marshall, K., Green, J., & Kordon, A. (2003). A methodology for combining symbolic regression and design of experiments to improve empirical model building. *Genetic and Evolutionary Computation Conference* , 1975–1985.

Chakrabarty, K., Biswas, R., & Nanda, S. (2000). Fuzziness in rough sets. *Fuzzy Sets and Systems*, *110*(2), 247–251.

Chung, H. M., & Gray, P. (1999). Special section: Data mining. *Journal of Management Information Systems*, *16*(1), 11–16.

Craven, M. W., & Shavlik, J. W. (1997). Using neural networks for data mining. *Future Generation Computer Systems*, *13*(2/3), 221–229.

Davidson, J. W., Savic, D. A., & Walters, G. A. (2003). Symbolic and numerical regression: Experiments and applications. *Information Sciences*, *150*(1/2), 95–117.

DeMaris, A. (1992). *Logit modeling*. Beverly Hills, CA: Sage.

Desai, V., Crook, J., & Overstreet, G. (1996). A comparison of neural networks and linear scoring models in credit union environment. *European Journal of Operations Management*, *95*(1), 24–37.

Desai, V., Crook, J., & Overstreet, G. (1997). Credit scoring models in the credit union environment using neural networks and genetic algorithms. *IMA Journal of Mathematics Applied in Business and Industry*, *8*(4), 324–346.

Dimitras, A. I., Slowinski, R., Susmaga, R., & Zopounidis, C. (1999). Business failure prediction using rough sets. *European Journal of Operational Research*, *144*(2), 263–280.

Dubois, D., & Prade, H. (1991). In Z. Pawlark (Ed.), *Rough sets: Theoretical aspects of reasoning about data*. Dordrecht, The Netherlands: Kluwer.

Feraud, R., & Cleror, F. (2002). A methodology to explain neural network classification. *Neural Network*, *15*(2), 237–246.

Grzymala-Busse, J. W. (1988). Knowledge acquisition under uncertainty—A rough set approach. *Journal of intelligent and Robotic Systems*, *1*(1), 3–16.

Jensen, H. L. (1992). Using neural networks for credit scoring. *Managerial Finance*, *18*(1), 15–26.

Knoke, D., & Burke, P. J. (1980). *Log-linear models*. Beverly Hills, CA: Sage.

Koza, J. (1992). *Genetic programming: On the programming of computers by means of natural selection*. Cambridge, MA: MIT Press.

Lee, T. S., Chiu, C. C., Lu, C. J., & Chen, I. F. (2002). Credit scoring using the hybrid neural discriminant technique. *Expert Systems with Applications*, *23*(3), 245–254.

Liao, T. F. (1994). *Interpreting probability model: Logit, probit, and other generalized linear models*. Beverly Hills, CA: Sage.

Mahlhotra, R., & Malhotra, D. K. (2003). Evaluating consumer loans using neural networks. *OMEGA: The International Journal of Management Science*, *31*(2), 83–96.

McCullagh, P. (1980). Regression model for ordinal data. *Journal of the Royal Statistical Society, Series B*, *42*(2), 109–142.

Mordeson, J. N. (2001). Rough set theory applied to (fuzzy) ideal theory. *Fuzzy Sets and Systems*, *121*(2), 315–324.

Nath, R., Rajagopalan, B., & Ryker, R. (1997). Determining the saliency of input variables in neural network classifiers. *Computers and Operations Researches*, *24*(8), 767–773.

Pawlak, Z. (1982). Rough set. *International Journal of Computer and Information Science*, *11*(5), 341–356.

Pawlak, Z., Grzymala-Busse, J., Slowinski, R., & Ziarko, W. (1995). Rough sets. *Communications of the ACM*, *38*(11), 88–95.

Piramuthu, S. (1999). Financial credit-risk evaluation with neural and neurofuzzy systems. *European Journal of Operational Research*, *112*(2), 310–321.

Press, S. J., & Wilson, S. (1978). Choosing between logistic regression and discriminant analysis. *Journal of the American Statistical Association*, *73*(4), 699–705.

Radzikowska, A. M., & Kerre, E. E. (2002). A comparative study of fuzzy rough sets. *Fuzzy Sets and Systems*, *126*(2), 137–155.

Shan, N., Hamilton, H. J., Ziarko, W., & Cercone, N. (1996). Discretization of continuous valued attributes in attribute-value systems. *Proceeding of the fourth International orkshop on Rough Sets, Fuzzy Sets, and MachineDiscovery, Tokyo, Japan* , 74–81.

Stefano, C. D., Cioppa, A. D., & Marcelli, A. (2002). Character preclassification based on genetic programming. *Pattern Recognition Letters*, *23*(12), 1439–1448.

Walczak, B., & Massart, D. L. (1999). Rough sets theory. *Chemometrics and Intelligent Laboratory Systems*, *47*(1), 1–16.

West, D. (2000). Neural network credit scoring models. *Computers and Operations Research*, *27*(11/12), 1131–1152.

Wu, X. D. (1996). A Bayesian discretizer for real-valued attributes. *The Computer Journal*, *39*(8), 688–691.

Zhang, Y., & Bhattacharyya, S. (2004). Genetic programming in classifying large-scale data: an ensemble method. *Information Science*, *163*(1/3), 85–101.