Development of Discriminant Analysis and Majority-Voting Based Credit Risk Assessment Classifier

Paulius Danenas[1], **Gintautas Garsva**[2], **and Rimvydas Simutis**[3]
[1,2,3] Department of Informatics, Kaunas Faculty of Humanities, Vilnius University, Kaunas, Lithuania

Abstract - *This article presents a research on a method for credit risk evaluation combining expert majority-based ensemble voting scheme together with discriminant analysis as basis for expert formation and popular machine learning techniques for classification, such as decision trees, rule-based inducers and neural networks. Both single expert and multiple expert evaluations were applied as basis for forming output classes dynamically. Feature selection was applied using correlation-based feature subset evaluator with tabu search. The experiment results form a basis for further research of similar method.*

Keywords: machine learning, credit risk, bankruptcy, majority voting, discriminant analysis, decision trees

1 Introduction

Credit risk evaluation is a widely researched problem which is very important to banks, credit unions and other financial institutions which have to evaluate the possibility of default and to decide whether to satisfy credit request or to reject it. It is especially important in periods of financial crisis as many companies are going bankrupt and this task becomes even more complicated. There are many factors to be evaluated, including current macroeconomic situation, financial ratios that represent the customer's current situation and his financial history. A lot of statistical, econometric, mathematical methods are applied and used in this field, as well as methods based on expert knowledge, artificial intelligence, machine learning. The most widely used techniques are based on scoring and evaluation of probability default. However, many researchers currently focus on intelligent techniques to solve classification and forecasting tasks related to this filed, such as bankruptcy prediction, client evaluation, risk assessment. These methods include neural networks, rule-based classifiers, decision trees, state-of-the-art techniques such as Support Vector Machines (SVM), also modern heuristic techniques such as evolutionary computing or swarm intelligence. These classifiers are applied as standalone and in ensemble metaclassifiers which often obtain better predictive performance and produce better results. Majority voting can also be used while trying to find an optimal solution if more than one expert gives his evaluation on particular problem; it takes the decision my the biggest number of votes. This research proposes a model for credit risk evaluation which combines machine learning classification procedures and majority-voting based expert evaluation principles. It can be used in cases when there are several predictions for each instance.

2 Related work

Many of the earliest researches were based on discriminant analysis. The most widely known and used was developed in 1968 by Altman et al. [1]. Altman obtained 96% and 79% accuracy by using two different samples, however, it is reliable in its predictive ability only in two years, after that the results fall down significantly. Zmijewski [2] examined two estimation biases for financial distress models on non-random samples by using probit (simple probit and bivariate) and maximum likelihood principles. His data set consisted of estimation sample of 40 bankrupt and 800 non-bankrupt companies and a prediction sample of 41 bankrupt and 800 non-bankrupt companies collected from American and New York Stock Exchanges. Springate [3] developed his model using step-wise multiple discriminate analysis to select 4 ratios which best describe a failing company. It obtained an accuracy rate of 92.5% using the 40 companies tested by Springate; later 83.3% and 88% accuracy rates were reported after testing it with other samples [4]. Shumway [5] forecasted bankruptcies with market-driven variables exclusively and by combining market-driven variables with two accounting ratios from Zmijewski's model with data (each firm-year) from 1962 to 1992 (33,621 firm-years and 291 bankruptcies in the first case, 28,664 firm-years and 239 bankruptcies in second case). In the first case 69 percent of bankrupt firms were in the highest probability decile and 95 percent of bankrupt firms above the probability median.

Neural networks have been used for research in credit evaluation field since they were applied as computational technique. Such researches include learning vector quantization (LVQ) network [6], fuzzy neural networks with particle swarm optimization for parameter selection [7], evolutionary neural networks [8] and many other. Self-organizing maps, often referred as Kohonen maps, were also successfully applied [9][10]. Support Vector Machines (SVM) has been extensively researched recently in this field and has been proved to be very efficient obtaining results that can be compared to Neural Networks. Danenas et. al applied LIBLINEAR and SMO algorithms [11], combined with discriminant analysis for evaluation, achieving results similar to Vapnik's SVM classifier results. SVM method has been combined with almost all popular natural computing techniques while applying it in credit risk assessment and bankruptcy prediction; many of these investigations related to credit risk evaluation using SVM-based methods are discussed in [12]

Decision tree is one of the oldest and most widely applied machine learning techniques. One can find numerous applications in various fields, including finance. It comprises a large family of algorithms which were developed on its basis - Classification and Regression Trees (CART), Chi-squared Automatic Interaction Detection (hence CHAID), C4.5 by Quinlan [13]. Modern techniques include functional trees (FT) with logistic regression functions at the inner nodes [14] and logistic model trees (LMT) [15], combination of DT and Naïve Bayes with NB classifiers at leaves [16] as well as forests of random trees [17]. These algorithms tend to show promising results thus they will be used in our experiment as classification methods.

3 Research method

3.1 Binary majority voting evaluation

The "expert" majority evaluation algorithm is based on majority voting principles used in ensemble classifiers, although it has some major changes. This algorithm can be expressed as follows in Figure 1.

Input: *m* – number of "experts" (uncorrelated evaluators), *C* – set representing possible class values ($C \in N$ and $C = N_0 \backslash N_c$, as we analyse only binary classification here, $C = \{0,1\}$), *M* - predictions of experts with values from set *C*, M_j - prediction of *j*-th "expert" such that $Mj \in C$, *j= 1..m*.

if (m = 1)
    $y = M_1$ *(we have single output, nothing to be done)*
else-if (m = 2)
    if ($M_1 \neq M_2$)
        $y = \arg\max_{c \in M} \sum_{i:M_i = c} 1$ *(simple majority selection)*
    else
        $y = rand(\sum_{i:M_i = c} 1)$ *(select value by random)*
else-if (*m = 2n-1* and $n \geq 2$)
    $y = \arg\max_{c \in M} \sum_{i:M_i = c} 1$ *(simple majority selection)*
else-if (*m = 2n* and $n \geq 2$) {
    k_0= size({i:M_i=c_0})
    k_1= size({i:M_i=c_1})
    if ($k_0 \neq k_1$)
        $y = \arg\max_{c \in M} \sum_{i:M_i = c} 1$ *(simple majority selection)*
    else {
        Θ = {} *(init an empty set of "expert" groups)*
        For k=1 to m do {
            M′ = rem(M, k) *(remove k-th element from M)*
            Θ = add(Θ, M′) *(add formed group to set of experts)*
        }
        (remove one ensemble from set by random)
        Θ=remove(Θ, rand(1, m))
        $y = \arg\max_{c \in \Theta} \sum_{i:\Theta_i = c,\, e \in \Theta} \arg\max_{c' \in e} \sum_{j:e_j = c'} 1$
    }
}

Output: *y* - output value for instance D_i of dataset *D*.

Figure 1. Pseudocode for binary majority voting evaluation algorithm

A more detailed explanation of algorithm for case when $m \in \{2n; \forall n \in N; \forall n \geq 2\}$ is as follows: if simple majority evaluation is not possible, we create *m* ensembles (groups of "experts") with *m-1 = 2n-1* members (such that we can apply simple majority voting principle) and randomly select *m-1 = 2n-1* evaluations from here such that expert would participate in this evaluation at least *m-1* times. Thus group majority voting evaluation is decomposed into a set of decisions by subgroups and the evaluation is obtained voting these decisions.

However, it becomes a difficult task to decide which evaluation should be selected if m = 2 and $M_1 = M_2$ as we have two different evaluations and no voting can be applied. Random selection was chosen to in this experiment to solve this problem; however, other options might be application of weights for each of "experts". If evaluators are other classifiers, it might be appropriate to select their accuracy or other evaluation metrics.

3.2 Proposed method

This section describes a method based on genetic search, machine learning technique for classification and discriminant analysis. The main steps are as follows:

1. Evaluate every instance by using *k* evaluators. In this experiment these "experts" are based on discriminant analysis models which values are converted to bankruptcy classes. If there are instances with empty outputs (records, which couldn't be evaluated in Step 1 because of lack of data or division by zero), the evaluation is marked as N/A and is excluded from instance evaluation. If all k evaluations are marked as "N/A", the instance is eliminated.
2. Data preprocessing:

a. Data imputation to eliminate empty values; here missing values are replaced with company's average of the attribute with missing data (for dataset *D* with attributes *X* and its subset D_C as financial records (instances) related to company C, if $D_{Cij} = \{\}$ then $D_{Cij} = average(X_i)$, *i=1..,m, j=1,..,n;* here *m* is the number of attributes, *n* – length of D_C);

b. Create training and testing data by splitting data of selected companies in the sector by particular percentage for hold-out training. These sets are disjoint (for dataset $D = D\_train \cup D\_test$, and $|D\_train| > |D\_test|$);

3. Apply feature selection to select the most relevant ratios;

4. Train classifier using one of machine learning classification algorithms;

5. The created model is tested using testing (holdout) data and results are evaluated.

3.3 Methods for evaluation of instances and results

Discriminant analysis was selected as basis for expert evaluation as widely applied and cited method.

Table I. Discriminant models used in evaluation

	Altman (original)	Springate	Zmijewski	Shumway
w_0	-	-	-4.336	-7.811
w_1	1.2	1.03	-4.513	-6.307
x_1	Working capital/Total assets	Working capital/Total assets	Net Income/Total assets	Net Income/Total assets
w_2	1.4	3.07	5.679	4.068
x_2	Retained earnings/Total assets	Net Profit before Interest and Taxes/Total Assets	Total liabilities/Total assets	Total liabilities/Total assets
w_3	3.3	0.66	0.004	-0.158
x_3	Earnings before interest and taxes/ Total assets	Net Profit before Taxes/Current Liabilities	Current assets / current liabilities	Current assets / current liabilities
w_4	0.6	0.4	-	-
x_4	Book value of Equity/ Book value of total liabilities	Sales/Total Assets	-	-
w_5	0.999	-	-	-
x_5	Net sales/Total assets	-	-	-
Eval	Z>3 – healthy; 2.7<Z<2.99 – non-bankrupt; Z <1.79 - bankrupt	Z>0.862 – healthy, Z<0.862 - bankrupt	Z>0– healthy, Z<0 - bankrupt	Z>0– healthy, Z<0 - bankrupt

Four models used in USA and Canada (as data used in experiment is from EDGAR database), particularly Altman Z-Score, Springate, Zmijewski and Shumway (essentially hazard modification of Zmijewski model) were applied; they are listed in Table I.

Algorithms described in section 3.1 were used in this experiment to train models. The test results are evaluated by using accuracy together with TP (True Positive) and F-Measure rates. As most of the experiment is concluded for two class only (except for single “expert” based on Altman) we give their definitions in terms of binary classification. Accuracy is defined as a proportion of correct predictions to total predictions as

$$acc = \frac{TP + TN}{TP + FP + FN + TN} \quad (1)$$

True Positive rate (also referred as Recall rate) is evaluated as a ratio of true predictions and number of total positive instances:

$$TPR = \frac{TP}{TP + FN} \quad (2)$$

Precision is a rate of predicted positive cases and total number of positive predictions (true positives and false positives):

$$prec = \frac{TP}{TP + FP} \quad (3)$$

F-Measure can be defined as a better option for evaluation of classifier trained with unbalanced data than accuracy and is defined as harmonic mean of precision and recall:

$$F_1 = \frac{2 * prec * recall}{prec + recall} \quad (4)$$

4 The experiment

4.1 Research data

The algorithm described in Section 3.2 was applied on a dataset consisting of entries from 1354 USA service companies with their 2005-2007 yearly financial records (balance and income statement) from financial EDGAR database. Each instance has 62 financial attributes (indices used in financial analysis).

Two types of datasets were formed: three consisting of single “expert” evaluations, and three consisting of majority-based evaluation described in section 3.2. To level the number of classes formed by discriminant analysis based “experts” it was reduced to the smallest number of classes provided by an expert; thus Altman-based “expert” formed only classes “bankrupt” and “good” instead of three own classes by its original evaluation by combining “average” and “healthy” classes. The main statistics of these datasets is given in Table II.

Table 2. Dataset statistics

Dataset	Instances	No of ratios	No classes
Altman	3321	13	3
Springate	3245	12	2
Zmijewski	3245	12	2
Springate-Zmijewski-Shumway	3245	9	2
Altman-Springate-Zmijewski-Shumway	3369	10	2
Altman-Springate-Zmijewski	3372	18	2

Figure 1 gives an overview of evaluation statistics (distribution of classes that is produced by each expert) for the case of all four „experts“. This distribution is clearly unbalanced; this is the case of application in real world when a small number of instances labelled as „bankrupt“ might be sampled together with much larger number of instances labelled as „average“ or „healthy“.

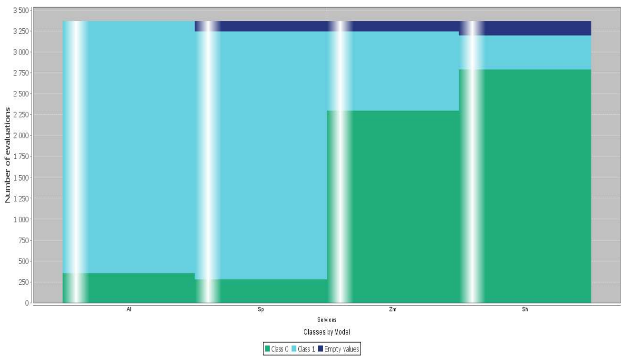

Figure 2. Expert evaluation statistics

There are three classes: Class 0 (mapped to „bankrupt“ class), Class 1 (mapped to „healthy“ class) and „Empty values“ (or unevaluated, marked as “N/A“); the latter one presents the classes that could not be evaluated (e.g., because of lack of particular data or division by zero). This distribution gives just an overview of evaluation; as it represents the distribution after the instances that could not be evaluated (i.e., all evaluations were marked as „N/A“) were removed, it might slightly differ for each case. Main properties of the datasets, together with number of removed (marked as „N/A“) instances, are presented in Table II. It also shows a big difference between each “expert” class distribution, i.e., it can be viewed as two groups each consisting of two experts. This contradiction results in a smaller number of total “agreements” in cases when one particular class totally dominates, and makes it more sensible to explore this principle. However, the case of one class domination might mean that there is an agreement, and thus there is a higher chance that group decision is correct.

4.2 Experiment configuration

The experiment was executed using implementations of neural networks (RBF Network, Multilayer Perceptron), rule based classifiers JRip and FURIA (Fuzzy Unordered Rule Induction Algorithm) and decision tree based algorithms best-first decision tree (BFTree), FT, C4.5, LADTree, LMT, NBTree, RandomForest, REPTree (Weka’s implementation of fast decision tree learner) and SimpleCart (minimal cost-complexity pruning DT) as classifiers. Parameters suggested by default were selected for each of these classifiers. To improve results, Bagging procedure was applied to each of them. All these classifiers are implemented in Weka machine learning framework.

A 7:3 split (70% percent of instances were selected for training) was used in the experiment. Feature selection was applied for each dataset using correlation-based feature subset evaluator with Tabu search. This resulted in reduced number of dimensions and less complexity. Three groups of experts were formed for evaluation: Altman-Springate-Zmijewski-Shumway (further referred as Al-Sp-Zm-Sh), Altman-Springate-Zmijewski (further - Al-Sp-Zm) and Springate-Zmijewski-Shumway (further - Sp-Zm-Sh). Single expert evaluations (Altman, Springate, Zmijewski) were also applied for model training. However, two experts’ evaluation based models were not chosen for evaluation as they might result in too many random evaluations and the model would be inconsistent; thus this selection is left only for evaluation in case of missing values.

4.3 Experiment results

The results of these experiments are presented in Table 3 and Table 4, which include TP rates together with F-Measure ratios. F-Measure is a good option for evaluation as it offers a trade-off between precision and recall (or TP rate). Table 3 presents the results of single “expert” based classifiers. Rule based classifiers and tree-based classifiers outperformed neural network based classifiers.

Result analysis shows that NN-based classifiers performed poorly while identifying particular classes (such as “bankrupt” and “average” in case of Altman-based evaluation) thus they cannot deal with dataset imbalance and special methods for dealing with imbalanced data, such as sampling-based (undersampling, oversampling) or cost-sensitive learning, should be applied together with these methods. The results obtained by decision tree and rule-based classifiers were better and they also identified bankrupt classes more precisely.

Table 4 presents results of majority-based evaluation of multiple expert classifier results. The results show that rule based and tree-based classifiers outperformed neural networks. Both these types of classifiers are better for unbalanced type of data; the results in Table 4 prove this, as true predictions ratio of both “bankrupt” and “healthy” companies” is higher than neural-network based classifiers. “Bad” companies were identified almost perfectly by Sp-Zm-

Sh model; however, ensemble of all four experts performed poorly while identifying bankrupt companies. This might be a consequence of the fact that original Altman model consists of three classes and the combination of the last two as "healthy" class resulted in very high imbalance.

Table 3. Single expert based classifier result

			RBF Network	Multilayer NN	FURIA	JRip	BFTree	FT	C4.5	LAD Tree	LMT	NBTree	Random Forest	REP Tree	Simple Cart
Altman	Accuracy		83,94	82,63	87,75	86,35	85,54	86,75	86,15	84,64	86,15	86,55	87,45	86,75	85,24
	TP	B	0,43	0,32	0,68	0,59	0,58	0,67	0,58	0,56	0,63	0,59	0,64	0,62	0,55
		A	0,00	0,00	0,24	0,22	0,22	0,25	0,25	0,20	0,25	0,25	0,22	0,25	0,20
		G	0,98	0,98	0,97	0,97	0,96	0,96	0,96	0,95	0,95	0,96	0,97	0,96	0,96
	F-Measure	B	0,53	0,42	0,69	0,62	0,61	0,67	0,63	0,56	0,65	0,63	0,67	0,65	0,58
		A	0,00	0,00	0,33	0,30	0,26	0,34	0,31	0,25	0,30	0,35	0,30	0,33	0,25
		G	0,91	0,90	0,94	0,93	0,94	0,93	0,93	0,93	0,94	0,93	0,94	0,94	0,93
Springate	Accuracy		94,35	93,94	97,95	97,64	97,74	95,99	97,84	97,74	97,43	97,43	97,53	97,33	97,95
	TP	B	0,56	0,61	0,85	0,80	0,85	0,78	0,84	0,83	0,78	0,79	0,80	0,81	0,87
		G	0,98	0,97	0,99	0,99	0,99	0,98	0,99	0,99	0,99	0,99	0,99	0,99	0,99
	F-Measure	B	0,64	0,64	0,88	0,86	0,87	0,78	0,87	0,87	0,84	0,85	0,85	0,84	0,88
		G	0,97	0,97	0,99	0,99	0,99	0,98	0,99	0,99	0,99	0,99	0,99	0,99	0,99
Zmijewski	Accuracy		93,83	83,04	97,95	98,15	96,51	99,38	96,51	96,81	99,49	97,23	98,36	97,43	96,40
	TP	B	0,99	0,99	0,99	0,99	0,98	1,00	0,98	0,98	1,00	0,98	0,99	0,98	0,99
		G	0,84	0,49	0,97	0,96	0,93	0,99	0,94	0,94	0,99	0,96	0,98	0,96	0,92
	F-Measure	B	0,96	0,89	0,99	0,99	0,97	1,00	0,97	0,98	1,00	0,98	0,99	0,98	0,97
		G	0,90	0,65	0,97	0,97	0,95	0,99	0,95	0,95	0,99	0,96	0,98	0,96	0,94

However, inclusion of Altman-based expert also did not result in high value of TP for "bankrupt" class in Al-Sp-Zm model which proves that Altman based "expert" was the one that imbalanced the results.

Table 4 shows that best results were obtained by Sp-Zm-Sh and Al-Sp-Zm evaluation based classifiers. It is not surprising as these models used simple majority voting (i.e., there was mostly majority-based consensus, as Figure 1 shows), without the need to form additional inner ensembles. However, the results show that Sp-Zm-Sh based classifier obtained TP for instances labeled as "bankrupt" values close to 1, to compare with values ranging from 0.78-0.85 in case of Springate model applied alone for evaluation. This increase is even higher in case of neural network based classifiers (from 0.56 and 0.61 to 0.98 and 0.96 respectively). However, TP rate values for instances labeled as "good" became lower, especially in case of NN based classifiers. Yet this value did not decrease much in cases of rule-based and tree-based classifiers where it still remained above 0.9. This concludes that multiple "expert" evaluation not only increased reliability of single "expert" based models, but it also sort-of helped to overcome imbalance barrier which NN classifiers seem to have.

Table 4. Single expert based classifier result

	Springate-Zmijewski-Shumway					Altman-Springate-Zmijewski-Shumway					Altman-Springate-Zmijewski				
	Acc	TP		F-Measure		Acc	TP		F-Measure		Acc.	TP		F-Measure	
		B	G	B	G		B	G	B	G		B	G	B	G
RBF Network	88,59	0,98	0,68	0,92	0,79	78,46	0,07	0,99	0,13	0,88	93,38	0,39	0,99	0,51	0,96
Multilayer NN	86,33	0,96	0,66	0,91	0,75	77,67	0,05	0,99	0,10	0,87	91,90	0,41	0,97	0,47	0,96
FURIA	96,40	0,97	0,94	0,97	0,94	78,85	0,18	0,97	0,28	0,88	94,37	0,56	0,98	0,64	0,97
JRip	96,81	0,97	0,96	0,98	0,95	78,56	0,15	0,97	0,23	0,88	94,27	0,54	0,98	0,63	0,97
BFTree	94,76	0,97	0,90	0,96	0,92	78,56	0,16	0,97	0,25	0,88	94,47	0,57	0,98	0,65	0,97
FT	97,02	0,98	0,95	0,98	0,95	79,55	0,22	0,96	0,32	0,88	93,97	0,56	0,98	0,62	0,97
C4.5	95,48	0,98	0,91	0,97	0,93	78,46	0,20	0,96	0,29	0,87	94,37	0,57	0,98	0,64	0,97
LADTree	94,55	0,96	0,91	0,96	0,91	78,76	0,19	0,96	0,29	0,88	94,66	0,57	0,98	0,65	0,97
LMT	96,71	0,97	0,96	0,98	0,95	78,26	0,20	0,95	0,30	0,87	94,07	0,59	0,98	0,64	0,97
NBTree	95,79	0,97	0,93	0,97	0,93	79,55	0,20	0,97	0,30	0,88	94,07	0,48	0,99	0,59	0,97
RandomForest	97,33	0,98	0,97	0,98	0,96	78,85	0,20	0,96	0,30	0,88	95,16	0,59	0,99	0,68	0,97
REPTree	96,10	0,97	0,95	0,97	0,94	78,56	0,22	0,95	0,32	0,87	93,48	0,50	0,98	0,58	0,97
SimpleCart	95,38	0,97	0,92	0,97	0,93	77,37	0,19	0,94	0,28	0,87	94,76	0,58	0,98	0,66	0,97

However, inclusion of Altman evaluation based results resulted in significantly different results. Both evaluation models that were tested (Altman-Springate-Zmijewski-Shumway and Altman-Springate-Zmijewski) obtained results which are worse in terms of TP values that corresponding single "expert" based models. Analysis of TP results for instances labeled as "bankrupt" shows that the most significant influence was the imbalance of data (instances both labeled as "average" and "good" were combined as "good"). Although the Al-Sp-Zm evaluation based classifier results did not suffer much, Al-Sp-Zm-Sh performed poorly in terms of TP rate for "bankrupt" classes. A more concise Altman-based model (i.e., having more classes, which might help to make a better balancing while "splitting" into binary categories) or proper dataset balance might help to overcome this problem.

5 Conclusions and further research

This article presents a research on credit risk which combines machine learning classifiers and multiple majority-voting based "expert" evaluation by discriminant analysis. This method can be complemented by feature selection or extraction procedure; this research used correlation-based feature selection. However, this method currently is only suitable for binary classification; it might be extended for multiclass evaluation in the future. Another important issue is learning from unbalanced data; if modern techniques such as neural networks or SVM are selected as classifiers, techniques to overcome this barrier should be applied, such as internally implemented class-weighting, cost-sensitive learning and evaluation, internal classifier enhancements or sampling techniques. The experiment provided promising results, especially when identifying "bankrupt" companies. Further research might be targeted at multiclass extension, integration of other soft computing techniques such as fuzzy logic or rough sets, as well as optimization of various classifiers by best parameter selection.

6 References

[1] Edward Altman. "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy"; The Journal of Finance, Vol. 23, pp. 589–609, 1968.

[2] Mark Zmijewski. "Methodological Issues Related to the Estimation of Financial Distress Prediction Models"; Journal of Accounting Research 22 (Supplement), pp. 59-86, 1984.

[3] Gordon L. V. Springate. "Predicting the Possibility of Failure in a Canadian Firm"; Unpublished M.B.A. Research Project, Simon Fraser University, 1978.

[4] Earl G. Sands, Gordon L.V. Springate, Turgut Var. "Predicting Business Failures"; CGA Magazine, pp. 24-27, 1983.

[5] Tyler Shumway. "Forecasting Bankruptcy More Accurately: A Simple Hazard Model"; Journal of Business, Vol. 74(1), pp.101-125, 2001.

[6] Yunna Wu, Zhaomin Si. "Application of RBF Neural Network Based on Ant Colony Algorithm in Credit Risk Evaluation of Construction Enterprises"; Proceedings of the 2008 International Conference on Risk Management \& Engineering Management (ICRMEM '08). IEEE Computer Society, Washington, DC, USA, pp.653-658, 2008.

[7] Fu-yuan Huang. "A Particle Swarm Optimized Fuzzy Neural Network for Credit Risk Evaluation"; WGEC '08 Proceedings of the 2008 Second International Conference on Genetic and Evolutionary Computing, pp. 153-157, 2008.

[8] Estefane Lacerda, André C. P. L. F. Carvalho, Antônio Pádua Braga Teresa Bernarda Ludermir, "Evolutionary Radial Basis Functions for Credit Assessment"; Applied Intelligence, Vol. 22, Number 3, pp. 167-181, 2005.

[9] Egidijus Merkevicius, Gintautas Garsva, Rimvydas Simutis. "Neuro-discriminate Model for the Forecasting of Changes of Companies Financial Standings on the Basis of Self-organizing Maps"; Lecture Notes In Computer Science, Vol. 4488, Berlin, Heidelberg: Springer-Verlag, 2007, pp. 439-446.

[10] Egidijus Merkevicius, Gintautas Garsva, Stasys Girdzijauskas. "A hybrid SOM-Altman model for bankruptcy prediction"; Lecture Notes in Computer Science, Vol. 3994, 2006, pp. 364-371.

[11] Paulius Danenas, Gintautas Garsva. "Credit risk evaluation using SVM-based classifier"; Business Information Systems Workshops: Lecture Notes in Business Information Processing, Vol. 57, Part 1, 2010, pp. 7-12.

[12] Paulius Danenas, Gintautas Garsva. "Support Vector Machines and their Application in Credit Risk Evaluation"; Transformations in Business & Economics, Vol. 8, No. 3 (18), 2009, pp. 46-58

[13] Ross Quinlan. C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[14] João Gama. "Functional trees"; Machine Learning, Vol. 55, No. 3, pp. 219-250, 2004.

[15] Niels Landwehr, Mark Hall, Eibe Frank. "Logistic Model Trees"; Machine Learning, Vol. 59, Issue 1-2, pp. 161-205, May 2005.

[16] R. Kohavi. "Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid"; Second International Conference on Knowledge Discovery and Data Mining, pp. 202-207, 1996.

[17] Leo Breiman. "Random Forests"; Machine Learning, Vol. 45(1), pp. 5-32, 2001.