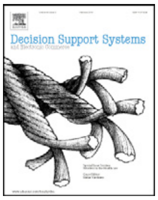
Decision Support Systems 52 (2012) 539–547

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss



Multistage classification by using logistic regression and neural networks for assessment of financial condition of company

Bartosz Swiderski [a], Jarosław Kurek [a], Stanislaw Osowski [b,c,*]

[a] University of Life Sciences, Poland
[b] Warsaw University of Technology, Poland
[c] Military University of Technology, Poland

ARTICLE INFO

Article history:
Received 20 October 2010
Received in revised form 5 October 2011
Accepted 19 October 2011
Available online 26 October 2011

Keywords:
Multinomial ordinary regression
Assessment of financing condition
Neural networks
Support vector machine

ABSTRACT

The paper presents the new approach to the automatic assessment of the financial condition of the company. We develop the computerized classification system applying WOE representation of data, logistic regression and Support Vector Machine (SVM) used as the final classifier. The applied method is a combination of a classical binary scoring approach and Support Vector Machine classification. The application of this method to the assessment of the financial condition of companies, classified into five classes, has shown its superiority with respect to classical approaches.

1. Introduction

The problem of assessment of the financial condition of company is very crucial to avoid customers, who may cause problems with financial liquidity or are the potential bankrupts [1, 2, 19]. To decrease the risk of transaction the assessment of financial condition of the company is necessary. Company credit ratings are very costly to obtain, since they require to invest large amount of time and human resources to perform analysis of the company risk status. Such report is based on various aspects, ranging strategic competitiveness to the operational level details [2, 10].

Although rating agencies emphasize the importance of analysts' assessment in determining credit ratings, there is a parallel way of developing the automatic methods relying on the artificial intelligence approach. Nowadays the most often used methods apply the artificial neural networks, being the universal tools able to do both classification and prediction tasks [6, 11, 13, 17]. The important point in such application is the availability of large amount of historical data of the company, regarding the financial reports and embedded valuable expertise of the agencies in evaluating companies' credit risk levels. The objective of credit rating prediction is to build the mathematical models that can extract knowledge of credit risk evaluation from past observations and to apply it to evaluate credit risk of companies with much broader scope.

However besides the prediction the modeling of the process can deliver another valuable information to the user. These studies can help the user to capture the fundamental characteristics of the most important dependencies between different financial ratings and the company risk status. Such analysis can also simplify the process of the assessment of the risk status of the company, by eliminating some parameters that are loosely associated with the level of the company risk.

The most important point in the credit report is the assessment of the financial condition of company by using many factors (not only financial). Important are also such information as changes of board of directors, status of the company, location and other registry information. Very often this information can be available online in Internet. The other source of fresh information about company is the interview via phone with subject or other companies with which our subject cooperates e.g. suppliers, sister company, parent company, affiliates etc. On the basis of this we can obtain additional information about e.g. payment history. Other sources of data are the agents distributed all over the world. This additional information can be added to the set of attributes, enriching in this way the input information taken into account at taking decision.

On the basis of all gathered information we can create the diagnostic features describing the financial state of the company, and then associate them with one of few classes, representing the level of insolvency risk. The insurance companies apply different number of classes. In this paper we assume 5 classes of insolvency risk [10]:

- excellent (without any risk)
- good

---

* Corresponding author at: Warsaw University of Technology, 00-661 Warsaw, Koszykowa 75, Poland. Tel.: +48 22 234 7235; fax: +48 22 234 5642.
E-mail address: sto@iem.pw.edu.pl (S. Osowski).

0167-9236/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.dss.2011.10.018

- satisfactory
- passable
- poor.

The problem that arises is to provide the unified way of representing the information as an input to the computer system performing the role of automatic extraction of knowledge and undertaking the final decision of assessment of insolvency risk of the company. In most application the numerical data is either represented directly in numerical form or converted to some classes, while the other (non-numerical) data is somehow coded in a binary way. In this paper we apply the unified way of representing data by using the weight of evidence (WOE) concept [17] associated with each feature. The data represented by WOE will be classified by us in two step procedure. In the first step we apply the binary logistic regression associating the data with seven models of 2-class classification. The results in the form of probability of membership to these 7 models are applied as the input attributes for the second stage of classification recognizing the final class (one of 5 already defined). As the classifier we apply the support vector machine (SVM), generally regarded as the most efficient classification tool [6, 15]. The novelty of the paper may be characterized in the following points.

- Development of continuous representation of financial data, very efficient in practical application and leading to the improvement of the quality of the classification system.
- Proposing the 2-step classification of financial data by applying the binary classification systems in both stages. We will show that such solution leads to the significant improvement of the accuracy of classification.

2. Binary logistic regression and WOE approach

In the first step of proposed approach we transform the multiclass task into few simple binary models of classification (similar to decision tree) and then in the second stage apply the additional classifier responsible for undertaking the final decision of the classification. We will show that such dissolution of classification task into two steps is profitable and leads to the increase of the accuracy of an automatic system of assessment of financial condition of company. In this work we substitute the entries of vector **x** representing any real financial data by their numerical codes in the form of weights of evidence (WOE). Weights of evidence is a quantitative method for combining evidence in support of a hypothesis [4, 7].

Suppose we have only two classes labeled by either zero or one. Let us assume that the input features (represented by WOE) defined for the data under classification are organized in the form of vector **x** of the dimension N, where N denotes the number of input features. We assume that the actual target y has the binomial distribution i.e. $y_i \sim \mathrm{B}(n_i, pd_i)$ where n_i denotes the number of trials and pd_i probability of success. This distribution is dependent on the set of input variables (features) x_i. Our goal is to estimate the conditional probability

$$pd(y_i = 1) = E\left(\frac{y_i}{n_i}\middle| x_i\right) \tag{1}$$

The model of logistic regression is now defined in the form [7, 14]

$$\ln\left(\frac{pd(y_i = 1)}{1 - pd(y_i = 1)}\right) = \mathbf{x}_i\boldsymbol{\alpha} \tag{2}$$

where $\mathbf{x}_i$ denotes the vector of input variables. From this model by solving the set of linear equations written for the learning data we can estimate the unknown vector $\boldsymbol{\alpha}$. To the most popular approaches to this estimation belongs the maximum likelihood method [2, 7]. Alternatively, one can express the previous formula in a probability category as

$$pd(\mathrm{y_i}) = pd(\mathrm{y}_i = 1) = \frac{1}{1 + \exp(-\boldsymbol{\alpha}\mathbf{x}_i)} \tag{3}$$

Then the output $pd(y_i)$ can be interpreted as the probability of belonging the input vector $\mathbf{x_i}$ to the class labeled by 1.

In the proposed approach the input vector **x** is represented by weights of evidence. Weights of evidence is a quantitative method for combining evidence in support of a hypothesis [2, 4]. Such representation enables to apply this model to the evidential themes with binary (presence/absence) classes as well as to multi-class maps. Binary evidence is relatively straightforward to interpret and we limit its definition for two classes. Let us assume for example that we deal with the input data called ‘type of consolidation’ of the company, recognizing three types of it: consolidated, non-consolidated and group consolidated. In each group there is some quantity of companies belonging to class 1 and some quantity of data classified as class 0 (binary representation of classes). Let us denote by odds_i for ith attribute of the feature the ratio of the number of all companies belonging to class 0 and to the class 1, that is

$$odds_i = \frac{rate_i(y = 1)}{rate_i(y = 0)} \tag{4}$$

where the $rate_i(y = a)$ is calculated as the number of examples of representatives of class a ($a = 0$ or $a = 1$) related to the total population of the members of this class for all attributes of the particular feature (in the case of type of consolidation it will be three groups counted together: consolidated, non-consolidated and group consolidated). The weights of evidence for this category variable is defined for each ith attribute in the form

$$WOE_i = \ln(odds_i) \tag{5}$$

We illustrate the procedure of calculation of WOE for the input feature ‘type of consolidation’ on the exemplary data presented in Table 1.

After transformation, the input variable ‘type of consolidation’ is represented by its attribute's WOE. For example $\mathrm{WOE}_1 = 1.427$, $\mathrm{WOE}_2 = -0.088$ or $\mathrm{WOE}_3 = -1.073$ will represent the consolidated, non-consolidated or group consolidated data, according to the particular type of the considered company.

We can use this approach to order any type of variables, including the category as well as numerical data. In the case of category data if some input variable has less than 20 categories we treat it as a standard one and code directly each category by WOE. In the case of higher number of categories we may merge the groups characterized by similar values of WOE, keeping the number of categories not higher than 20. In the case when the class contains less than 5% of data we may merge it with the class closest in respect to WOE and then we calculate the new value of WOE for the merged group.

Table 1
The example of determination of the WOE for the feature ‘type of consolidation’.

i	Type of consolidation	Class 0	Class 1	Rate (y = 0)	Rate (y = 1)	Odds	WOE
1	Consolidated	30	5	0.375	0.090	4.166	1.427
2	Non-consolidated	40	30	0.500	0.546	0.915	− 0.088
3	Group consolidated	10	20	0.125	0.364	0.342	− 1.073
Sum		80	55	1	1		

In the case of numerical input variable we divide the numerical data into intervals (bins) each containing c.a. 5% of observations and compute share of ones and zeros in all bins. If some ranges have similar values of WOE we can join these bins together, reducing in this way the quantity of input attributes. Next we compute the respective value of WOE for each bin. This process can be done easily in an automatic way.

In our solution at converting the continuous variable to WOE we have applied special strategy, which in spite of applied discretization, provides as much as possible the continuity of the resulting representation at changing the class. First we transform the continuous value of variable x into quintile representation using kernel density estimation providing the range (0–1) for the new transformed variable x_q. This operation reduces the sensitivity of the results with respect to the extremity of values of feature x, that we deal with. Each particular value x_q corresponding to the continuous attribute is associated with the probability $\mathrm{pd}(y_i=1|\, x_q)$. The theoretical estimation of this probability might be expressed for example by the approximate formula

$$pd\left(y_i = 1\middle|x_q\right) = \frac{\#_{1,x_q}}{\#_{1,x} + \#_{0,x_q}} \tag{6}$$

in which $\#_{i,xq}$ ($i=1$ or 0) means the theoretical cardinality of i-th class in the imaginary set for the quantiled variable x_q. In this way each value of x transformed first to x_q is associated with its probability of belonging to class 1. Then we approximate the value of WOE corresponding to x by using logistic model, in which the continuous value of the variable x can be represented by WOE as follows [2, 14]

$$WOE(x) = \ln\left(\frac{pd(\mathrm{y}_i = 1|x_q)}{1 - pd(y_i = 1|x_q}\right)\frac{0,x}{1,x}) \tag{7}$$

where $\#_{i,x}$ ($i=1$ or 0) is the total cardinality of ith class in the learning set for the variable x. In this way any continuous variable x may be transformed into its WOE representation. On the basis of this representation of the input variables we will classify them into proper class (one of five already defined).

We should note that application of WOE in data representation provides the uniformity of input data irrespective of its character. It allows also to avoid the problem of increasing the dimension of the input vector at growing the number of subclasses used to represent some features.

3. Classification of data

3.1. Binary logistic regression for multiclass problem

Our first step in solving the multiclass classification task is to convert the multiclass problem into binary one. We will do it by applying the properly defined subclasses $\mathrm{M}_1, \mathrm{M}_2, \ldots, \mathrm{M}_7$, each representing the decision of matching the input data to two classes only. It resembles the decision tree. The general scheme of the data preprocessing for the class M_i ($i=1, 2, \ldots, 7$) is presented in Fig. 1.

We recognize 7 models of classes M_i ($i=1, 2,\ldots,7$) for which we perform independently the feature selection procedure. The model M_1 corresponds to the case when the actual data represented by the selected vector **x** belongs to either class 1 or to the class higher than 1. Its output defined by Eq. (3) represents the probability of belonging the data to the class higher than or equal to 1. In similar way model M_2 generates the output signal representing the probability of belonging the input data to classes higher than 2. Model M_3 represents the probability of data belonging to classes higher than 3 and model M_4 corresponds to data belonging to classes higher than 4. Additionally on the next three levels we build the models of exact membership to the particular class (model M_5 —> probability of belonging to class 2, model M_6 —> to class 3 and similarly model M_7 —> to class 4). Each model may be fed by different set of features. Selection of these features is done in an introductory step by applying the procedure of feature validation and selection (independently for each model). We have done it by using well known stepwise method [8, 14, 15], preceding the true logistic regression performed independently for each model.

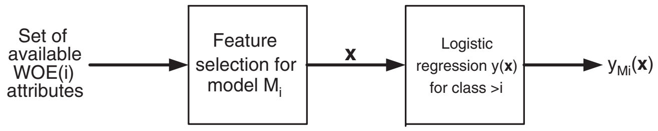

Set of available WOE(i) attributes
Feature selection for model M_i
x
Logistic regression y(**x**) for class >i
$\mathrm{y}_{\mathrm{Mi}}(\mathbf{x})$

Fig. 1. The general scheme of data pre-classification using the binary logistic regression for *i*th model.

Each model is created independently from the other. It means seven repetitions of feature selection procedure as well as seven logistic regressions applied for each model. After performing all these steps of model building with the vectors **x** representing the selected input attributes (expressed in WOE form) we get 7 models associated with the proper values of probability defined by the value $\mathrm{y}_{\mathrm{Mi}}(\mathbf{x})$. In this way the model M_1 will be further represented by the probability value $\mathrm{y}_{\mathrm{M1}}(\mathbf{x})$, M_2 by the value of $\mathrm{y}_{\mathrm{M2}}(\mathbf{x})$, etc. These probability values will form the inputs to the next stage classification of data, which is performed in our solution by the neural SVM classifier. In this way the final classifying network will be fed by 7 input features, all in the range (0, 1).

This two step approach has many advantages.

- The logistic regression defined by Eq. (3), used as the fist step of classification is fed by the input variables coded in WOE, which naturally reflect the information about proportion of the classes and their association with the input attributes.
- In natural way the classification problem has been distributed into few smallest models, each responsible for two classes only. This way of solution is much easier to control.
- For the model on each level we can apply the dedicated feature selection method such as stepwise or forward and backward regression, leading to the simplification of the undertaken decision and at the same time to better understanding the basis relationships between the attributes and the recognized classes.

3.2. Final classification of data

The binary logistic regression discussed in the previous subsection delivers the information of the probability of belonging the input vector **x** (representing the financial data characterizing the company) to predefined cumulative classes, which do not represent our final classification problem. The last step is to assign the particular data to the final class (one of five defined in the introduction). The input data for this step will form seven probabilities denoting the degree of belonging the input data to the models $\mathrm{M}_1, \mathrm{M}_2,\ldots, \mathrm{M}_7$. Denoting them by $\mathrm{y}_{\mathrm{M1}}(\mathbf{x})$, $\mathrm{y}_{\mathrm{M2}}(\mathbf{x}),\ldots,\mathrm{y}_{\mathrm{M7}}(\mathbf{x})$ we form the input vector for the last step classifier as the 7-entries vector $\mathbf{x}_{\mathrm{fin}} = [\mathrm{y}_{\mathrm{M1}}(\mathbf{x}), \mathrm{y}_{\mathrm{M2}}(\mathbf{x}),\ldots,\mathrm{y}_{\mathrm{M7}}(\mathbf{x})]$.

The basic classifier proposed by us to solve this problem is the Support Vector Machine. The SVM is a feedforward network of one hidden layer (the kernel function layer). It is known as an excellent classifier of good generalization ability [9, 18, 20]. The learning problem of SVM is formulated as the task of separating the learning vectors into two classes of the destination values either $\mathrm{d_i}=1$ (one class) or $\mathrm{d_i}=-1$ (the opposite class), with the maximal separation margin. The separation margin, formed in the learning stage according to the assumed value of the regularization constant C, provides some immunity of this classifier to the noise, inevitably contained in the real data under testing. The great advantage of SVM is the unique formulation of learning problem leading to the quadratic programming with linear constraints, which is very easy to solve. The SVM of the Gaussian kernel has been used in our application. The hyperparameters σ of the Gaussian function and the regularization constant C have been adjusted by repeating the learning experiments for the set of their predefined values and choosing the best one at the validation data sets. The optimal values of these parameters were $\sigma=2.5$ and $C=100$. To deal with a problem of many classes we have applied the strategy *one against one* [9]. In this approach the few SVM networks are trained to recognize between all combinations of two classes of data. For M classes we have to train $M(M\text{-}1)/2$ individual SVM networks. In the retrieval mode the input vector belongs to the class of the highest number of winnings in all combinations of classes.

To compare the efficiency of this classifier we have tried also other classifying tools. Among them are the well known decision tree approach [5], fuzzy KNN classifier [12] and ordinal regression [3].

The decision tree is very well known approach to multiclass classification of data. On each level it creates the model that predicts the class name by comparing the values of input attributes with the assumed bias. This is done top-down from a root node and involves partitioning the data into subsets which contain instances that have similar values. Each non-terminal node is split based on the actual values of attributes and each leaf represents a value of the target, given the values of the input variables represented by the path from the root to the leaf. The method is very simple to understand and interpret. It requires little data preparation and is able to handle both numerical and categorical data of even large size in a short time.

The ordinal regression [3, 16] is a special kind of statistical linear technique used by us to solve the final classification problem on the basis of the results of logistic regressions of all binary models from M_1 to M_7 treated as input variables. In this method of data processing we estimate the order of the response category on the basis of the weighted combination of the input variables. The important role fulfill here the regression coefficients α and β used to predict the probability of the outcome (class to which the input data belongs), while the outcomes have the ordinal character. At the presentation of ith input vector $\mathbf{x}_i$ the parameters α and β fulfill the linear equation

$$\ln\left(\frac{\pi_j(\mathbf{x}_i)}{1-\pi_j(\mathbf{x}_i)}\right)=\alpha_j+\left(-\beta_1 y_{M1}(\mathbf{x}_i)-\beta_2 y_{M2}(\mathbf{x}_i)-\ldots-\beta_7 y_{M7}(\mathbf{x}_i)\right) \tag{8}$$

The variable $\pi_j(\mathbf{x}_i)$ is the probability that $pd(y_i\le class_j|\mathbf{x}_i)$ and $j=1, 2, \ldots, 5$ represent classes. This model generates as many outputs as is the number of classes. Each output signal is generated on the basis of the same input signals and the same parameters β_j. Only parameters α_j ($j=1, 2, \ldots, 5$) are different for all 5 classes. The adaptation of parameters α and β is made on the basis of known class membership concerning the learning data, and these parameters are then used in classification mode for the current data under testing [3].

The last method used in our comparison is fuzzy KNN classifier. According to the principle of operation of this classifier we find K nearest neighbors of the actually applied input vector $\mathbf{x}_{fin}$ of unknown class. These neighbors are selected from the vectors of known class destination. Next we determine the value of the Gaussian membership function of the vector $\mathbf{x}_{fin}$ to each of the classes represented by the chosen neighbors. Then we assign the input vector $\mathbf{x}_{fin}$ to the class, which has the highest sum of membership among the vectors identified in the previous step [12, 14].

The best choice of the value of K depends upon the data. Generally, larger values of K reduce the effect of noise on the classification, but make boundaries between classes less distinct. Proper value of K is usually selected experimentally by trying different values and applying this one which provides the best results of classification in cross-validation experiments on the validation data set.

4. Data base used in numerical experiments

In the numerical experiments we have used the credit reports data of 2217 cases corresponding to companies of known destinations (the class of insolvency risk) assessed by the human experts. The profile of activity of companies was very different (commerce, production, banking and other services). Generally the credit report data contains the most important information such as following.

1. The identification of the company.
2. Registry data — date established, legal form, registered status, registered authority, registration number.
3. Legal filings — bankruptcy filings, court judgments, tax, liens, etc.
4. Management and staff — key managers, staff data.
5. Board of directors — appointments (name, board function, ID, address, biography)
6. Share capital — authorized capital, type of share, number of shares, paid-up/issued capital, etc.
7. Shareholders — composition (name, percentage of shares, address), how listed, etc.
8. Corporate affiliation — structure (name, affiliation type, address).
9. Financial accounts containing such positions as
   - date of account
   - consolidation
   - period
   - sales turnover
   - gross profit
   - operating profit
   - profit before tax
   - profit after tax
   - current assets
   - non-current assets
   - total assets
   - current liabilities
   - long-term liabilities
   - total liabilities
   - shareholders' equity
10. Payment behavior regarding the payment experience of the subject, assessed by the experts on the basis of the record of his payments from the past.

Additionally the reports contain also information on different types of consolidation (non-consolidated, consolidated, group consolidated). The quantity of companies belonging to different groups of consolidation is presented in Table 2.

Taking into consideration the final expert assessment of the insolvency risk associated with each company, the structure of the whole data set was as presented in Table 3. As it is seen the representation of different classes was not equal. The highest number of companies in the base was classified as good or satisfactory. The smallest representation (91 out of 2217) corresponds to the class named *poor*.

From this very rich data base we have preselected some introductory set of features, which according to the experience of the human expert has the biggest impact on the decision of classifying the company to the proper class. In Table 4 we list the set of these attributes taken directly from the data base.

Coding of the *class of the sales turnover* is made at assumption of five classes. Assignment to one of these 5 classes is based on the following rules, showing how big the company is:

if sales is more than 1 billion USD — class 1
if sales belongs to <100 million–1 billion USD — class 2
if sales belongs to <20 million–100 million USD — class 3

Table 2
The quantity of data of different types of consolidation in the considered database.

Group consolidated	720
Consolidated	637
Non-consolidated	860
Total	2217

Table 3
The quantity of cases belonging to different groups of insolvency risk.

Excellent	211
Good	800
Satisfactory	783
Passable	332
Poor	91
Total:	2217

if sales belongs to <2 million–20 million USD — class 4
if sales is less than 2 million USD — class 5

The feature concerned with the profit after tax ('Is profit after tax < 0') is coded as zero, when the profit is below zero and as one in other case. The feature 'class of positive profit after tax' has been coded by us in 4 classes, depending on the volume of the profit. We have assumed 13 classes of the feature 'class of payment behavior', assigned to the company on the basis to the experience how promptly the payment has been made in the past (class 1 — all payments made very promptly, class 12 — serious payment delays reported and class 13 — refuse to pay). We have considered four classes representing the feature 'legal form class': private limited liability company, joined stock company, sole proprietorship and the fourth class composed of other companies. The feature 'is group employment' has been coded by zero when there is no group employment and as one when such type of employment exists.

On the basis of the presented above attributes we have generated additional features being the ratios of some of them. The most important are financial ratios which show the level of financial condition of company, although they don't take into account the size of the company (for example the level of sales turnover). The financial ratios used by us are calculated using the following formulas [11]

- Gross margin (GM)

$$GM = \frac{Gross\ \ profit}{Sales\ \ turnover}$$

- Operating margin (OM)

$$OM = \frac{Operating\ \ profit}{Sales\ \ turnover}$$

- Net profit margin (NPM)

$$NPM = \frac{Profit\ \ after\ \ tax}{Sales\ \ turnover}$$

- Return on equity (ROE)

$$ROE = \frac{Profit\ \ after\ \ tax}{Shareholders's\ \ equity}$$

- Return on assets (ROA)

$$ROA = \frac{Profit\ \ after\ \ tax}{Total\ \ assets}$$

- Equity slope (ES)

$$ES = \frac{Equity(\mathrm{t}+\Delta\mathrm{t}) - Equity(\mathrm{t})}{\Delta t}$$

- Current Ratio (CR)

$$CR = \frac{Total\ \ assets}{Total\ \ liabilities}$$

- Debt Ratio (DR)

$$DR = \frac{Total\ \ liabilities}{Total\ \ assets}$$

- Long Term Ratio (LTR)

$$LTR = \frac{Long\ term\ \ liabilities}{Share\ holders'\ \ equity}$$

The last part of generated features is connected with Altman's variables, known as z-score. The Z-score formula for predicting bankruptcy was published in 1968 by Edward I. Altman [1]. The formula may be used to predict the probability that a company will go into bankruptcy within two years. Taking them into account we have added the following variables derived from z-score.

$$T_1 = \frac{Current\ \ assets\ \ -\ \ Current\ \ liabilities}{Total\ \ assets}$$

$$T_2 = \frac{Operating\ \ profit}{Total\ \ assets}$$

$$T_3 = \frac{Shareholders'\ \ equity}{Total\ \ liabilities}$$

$$T_4 = \frac{Sale\ \ turnover}{Total\ \ assets}$$

All different currencies used in financial reports of international companies existing in our data base have been converted to common currency of US dollars on the basis of the actual currency rate. Taking into account the original features and their numerical descriptors defined above we got the set of 40 potential features that were used in classification of the insolvency risk assessment.

Table 4
The set of attributes taken from the credit reports as the potential features.

Consolidation class (three classes)	Total assets of the last year
Class of sales turnover of the last year	Current liabilities of the last year
Profit before tax of the last year	Total liabilities at the end of the last year
Current assets of the last year	Trend of sales turnover (the slope of linear model)
Shareholders' equity	Trend of profit after tax
Availability of company by phone	Trend of shareholders' equity
Availability of company by fax	Age of company (in years)
Legal form class	Number of employees
Is group employment	Number of employees for group
Class of payment behavior	Class of sales turnover
Is profit after tax <0	Class of positive profit after tax
Sum of profit and equity (to cover losses)	Registry status
Credit limit	Current ratio
Share capital (authorized or issued/paid-up)	

5. Numerical results of experiments

In the first step of procedure we have to map the input financial data of company organized in the form of vector $\mathbf{x}$ composed of WOE entries into 7 binary models of classes, from M_1 to M_7. Each model is then represented by the value of $y_{Mi}(\mathbf{x})$ defined by the logistic regression Eq. (3). The important advantage of this method is the individual approach to each model, especially the selection of input attributes. Observe that each model of classes may be individually tuned with respect to the optimal set of input attributes. In fact, the application of stepwise fit of feature selection in Matlab (the function *sequentialfs*) [15] for each model has delivered different number of the most important features. In our experiments we have used the following control parameters of Matlab: *penter* = 0.03 and *premove* = 0.07. The variable (feature) was added to the selected feature set when its significance level of hypothesis, that this attribute is not important, was smaller than the value 0.03 and was removed from the set, when this level was higher than 0.07. The procedure of feature selection has been repeated 50 times at random split of the data into learning (90%) and testing (10%) parts. Application of this procedure has generated the set of the most important features for each run of the cross validation. Below we list the contents of the features for particular models selected most often in all cross validation runs.

- Model M_1 (nine features): class of consolidation, credit limit, ROE, long term debt ratio, availability of company by phone, age of company, number of employees, share capital, class of payment behavior.
- Model M_2 (four features): credit limit, operating margin, equity slope, share capital.
- Model M_3 (six features): credit limit, operating margin, long term debt ratio, is group employment, share capital, class of payment behavior.
- Model M_4 (six features): class of consolidation, credit limit, current ratio, is group employment, class of sales turnover of the last year, class of the profit after tax.
- Model M_5 (five features): credit limit, operating margin, equity slope, share capital, number of employees.
- Model M_6 (four features): credit limit, group employment, share capital, class of payment behavior.
- Model M_7 (eight features): credit limit, net profit margin, current ratio, long term debt ratio, profit slope, availability of company by phone, is group employment, share capital.

These elements formed the input vector $\mathbf{x}$ for each model of logistic regression, on the basis of which the regression defined by (3) was performed. As a result of this operation the probability value $y_{Mi}(\mathbf{x})$ associated with each model (M_1, M_2, …,M_7) was calculated. To assess the quality of these results we have prepared the receiver operating characteristic curve (ROC) for each model. The ROC displays the trade off between true positive rate (TPR) and false positive rate (FPR) of a classifier. Good classification model should be located as close as possible to the upper left corner of the diagram, while model that makes random guesses resides along the main diagonal connecting the points (TPR = 0, FPR = 0) and (TPR = 1, FPR = 1). Fig. 2 presents these characteristics for all created models. The horizontal axis represents the false positive rate and vertical one the true positive rate of all classification models M_1, M_2,…, M_7. We present the average and two extreme (maximum and minimum) characteristics obtained in 50 cross validation trials. As it is seen the ROC characteristics of all classifiers prove good classification properties of the created classifying models.

In Table 5 we summarize the performance of the created classifying models by pointing the mean, median and standard deviation (std) of areas under curves (AUC) of ROC, as a result of cross validation of the models (50 repetitions of the procedure of creation of the models using randomly selected data from the data base). For all classifiers we have got very good values of AUC. Both mean and median values are very close to each other and all are higher than 0.9. The remarkable is very low values of standard deviations observed in all cases.

Seven outputs of binomial logistic regression representing the first stage models (from M_1 to M_7) were used as inputs to the second stage classifier. The basic classifier was the Support Vector Machine. The SVM network with Gaussian kernel was applied in one against one mode. The hyperparameters: σ of the Gaussian function and the regularization constant C were set to 2.5 and 100 respectively. They have been selected after the introductory experiments performed on the validation data set (10% of the learning data set).

The experiments of learning and testing have been repeated also 50 times at the random choice of the data forming both sets with proportion: 90% of data used in learning and 10% for testing only. The final mean accuracy of classification was equal 82% at standard deviation of 3%. To illustrate the performance of the SVM classifier system we have created the confusion matrices on the basis of the numerical results of all 50 experiments. Three matrices have been created: one representing the mean values of the class memberships (Fig. 3a), the second to the median values of class membership (Fig. 3b) and the last one the for standard deviations (Fig. 4).

The confusion matrix illustrates how the cases belonging to different classes have been classified by our system. The rows represent the actual outputs of our system and the columns — the targets. The upper number in each entry of the matrix is the average number of the actually recognized classes in testing mode, calculated for all 50 experiments and the bottom one — its percentage, referred to the total number of all cases used in testing (221). The diagonal entries of this matrix represent the mean (or median) quantity of the properly recognized cases (the upper value) and also its ratio with respect to the total representation of all testing data (lower value expressed as percentage). Each entry outside the diagonal means error (the number of misclassifications and its relative value). The entry in the (i,j)th position (for $i \neq j$) of the matrix means false assignment of the case of *j*th class to the *i*th one.

The last column of the matrix represents the total percentage measure of accuracy of actual recognition for the class pointed by the classifier. The upper (green) number represents the ratio of the number of the properly recognized cases to the total number of cases pointed by this particular output. This is the specificity defined as the ratio of the true positive cases to the sum of true positive and false positive cases. The bottom numbers in the last column denoted in red color represent the false alarm ratios (the complement of specificity to one). The last row of the matrix represents the ratios of the number of the properly recognized cases to the total number of true cases (targets). This is the sensitivity of the system, defined as the ratio of the true positive cases to the sum of true positive and false negative cases. The bottom numbers in this row denoted in red color are the misclassification ratios (the complement of sensitivity to one).

Both results (mean and median) are very close to each other. Note that practically all misclassifications of classes are grouped along the main diagonal and the observed difference of estimated class membership was not higher than one in both directions. The only exception is the positions (1,3) and (4,2) of the confusion matrix for mean values. We should also note small values of standard deviations (Fig. 4). It means good stability of the applied classifier, quite insensitive to the choice of the learning data set.

To assess properly the quality of SVM classifier we have repeated the final classification experiments by applying three other classification systems: fuzzy KNN (at $K = 20$), the decision tree and ordinal regression. The comparative statistical results observed on the same data sets are depicted in Table 6. Bold numbers indicate the best results with respect to the mean and median in all 50 cross validation trials.

As it is seen the application of SVM as the final classification system results in the best outcome. Both mean and median results belong to the best, although it should be noted that the difference between the SVM and the other methods is not very large. Note also small values of standard deviations in all cases. It means that our approach produces stable results, rather insensitive to the composition of the learning and testing data.

The next question that should be answered is how the logistic regression and two stage classification applied in our solution was important for obtaining the presented results. In the next experiments we have repeated the classification procedure for the same set of data by using one step approach at application of SVM, decision tree, ordinal regression and fuzzy KNN as the classifiers (without logistic regression). However, this time we used the raw original numerical values of features. This approach needed also repetition of

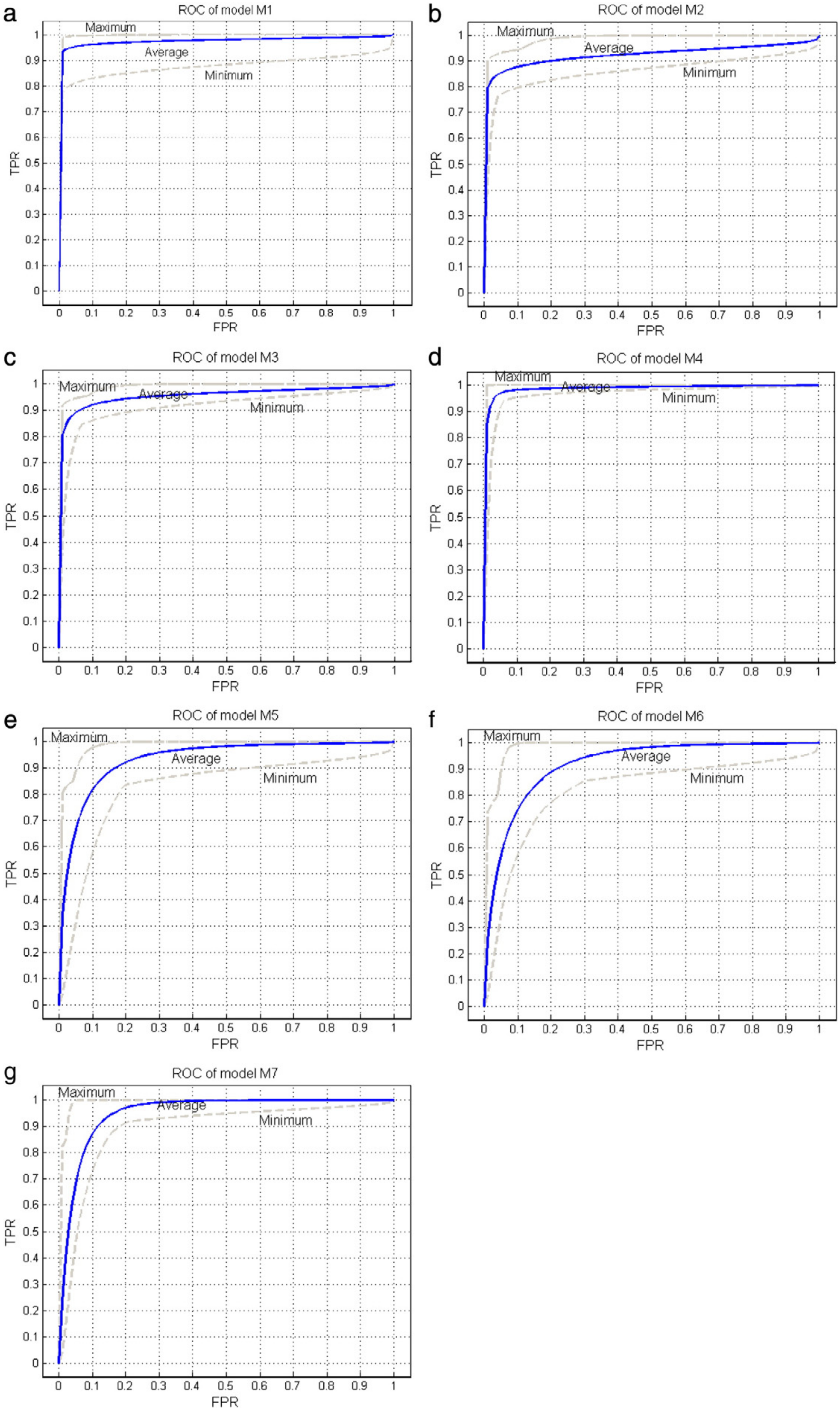


Fig. 2. The ROC curves (the average, maximum and minimum) for 7 classifying models: a) M1, b) M2, c) M3, d) M4, e) M5, f) M6, g) M7.

Table 5
The AUC values of ROC corresponding to 7 models $M_1, M_2, \ldots, M_7$.

Model	M_1	M_2	M_3	M_4	M_5	M_6	M_7
Mean (AUC)	0.990	0.969	0.980	0.993	0.933	0.913	0.952
Median (AUC)	0.993	0.971	0.980	0.994	0.034	0.915	0.955
Std (AUC)	0.009	0.010	0.008	0.004	0.018	0.023	0.013

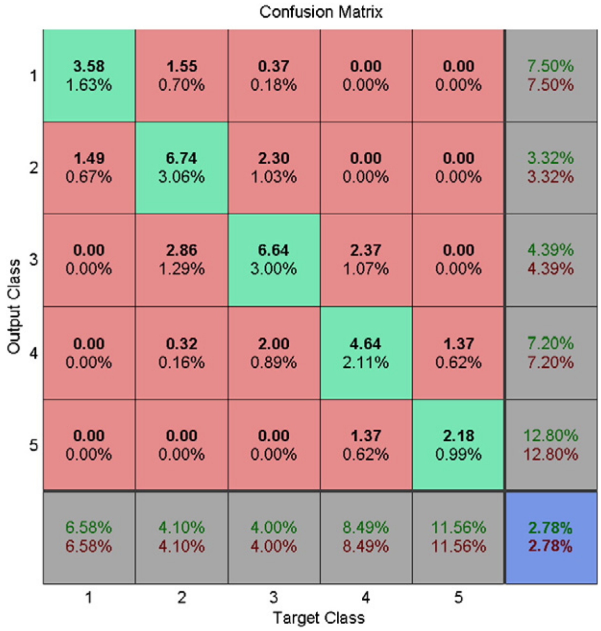
the selection procedure of the input data to all models. Once again the Matlab function *sequentialfs* implementing stepwise feature selection [15] was performed on the original data without converting them to WOE representation. We have done it by using the same (as in the previous case) control parameters of the stepwise method. The selection procedure has been repeated 50 times for each classifier and the final results are the mean of all 50 trials on the testing data. As a result of these experiments we have got different number of the most discriminating features for each classification method (most often selected in all 50 runs of the cross validation). They are summarized below:

- SVM classifier (5 features): existence of phone connection to the company, profit after tax, sales turnover class, payment behavior class and maximum credit limit.
- Decision tree (6 features): debt ratio, payment behavior class, ROE, sales turnover class, consolidation class and maximum credit limit.
- Ordinal regression (7 features): existence of phone connection to the company, profit after tax, ROE, maximum credit limit, net profit margin, share capital.
- Fuzzy KNN (5 features): existence of phone connection to the company, profit after tax, sales turnover class, consolidation class, credit limit.

The selected features have been normalized by applying the formula

$$x := \frac{x}{\max(abs(x))} \tag{9}$$

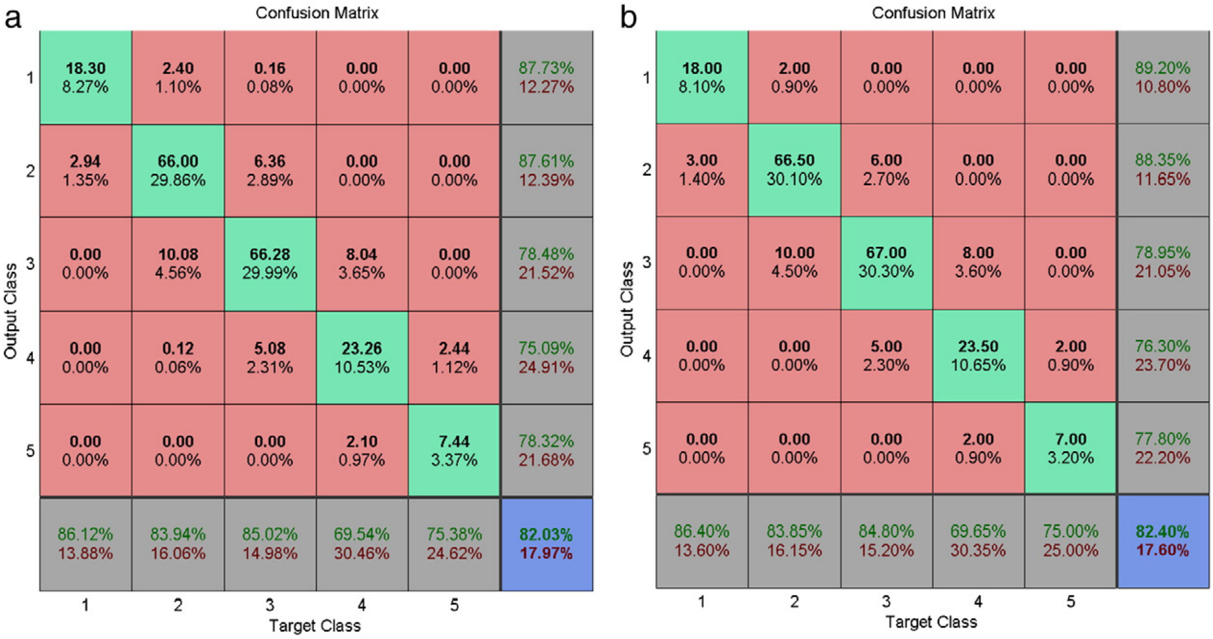
where the maximum is calculated for each feature independently. The normalized features have been applied directly to the input of the classifiers. The whole procedure of classification has been repeated 50 times at random division of the data for learning set (90% of data) and testing set (10% of data). Table 7 depicts the obtained results compared to the presented approach. Similarly as before the bold numbers indicate the best results obtained in all 50 cross validation experiments.


Confusion Matrix
Output Class
Target Class

1	**3.58** 1.63%	**1.55** 0.70%	**0.37** 0.18%	**0.00** 0.00%	**0.00** 0.00%	7.50% 7.50%
2	**1.49** 0.67%	**6.74** 3.06%	**2.30** 1.03%	**0.00** 0.00%	**0.00** 0.00%	3.32% 3.32%
3	**0.00** 0.00%	**2.86** 1.29%	**6.64** 3.00%	**2.37** 1.07%	**0.00** 0.00%	4.39% 4.39%
4	**0.00** 0.00%	**0.32** 0.16%	**2.00** 0.89%	**4.64** 2.11%	**1.37** 0.62%	7.20% 7.20%
5	**0.00** 0.00%	**0.00** 0.00%	**0.00** 0.00%	**1.37** 0.62%	**2.18** 0.99%	12.80% 12.80%
	6.58% 6.58%	4.10% 4.10%	4.00% 4.00%	8.49% 8.49%	11.56% 11.56%	**2.78%** **2.78%**
	1	2	3	4	5	

Fig. 4. The confusion matrix representing standard deviation of mean results of classification.

The results confirm, that application of logistic regression in combination with SVM classifier belongs to the best. We observe its advantage over simple (one-step) application of SVM, decision tree, ordinal regression and fuzzy KNN in terms of accuracy (average increase of accuracy around 3% with respect to the best single step classification). We have also noted some superiority of the proposed approach when analyzing the misclassified cases. In the proposed approach practically all misclassifications happened at the neighboring


a
Confusion Matrix
Output Class
Target Class

1	**18.30** 8.27%	**2.40** 1.10%	**0.16** 0.08%	**0.00** 0.00%	**0.00** 0.00%	87.73% 12.27%
2	**2.94** 1.35%	**66.00** 29.86%	**6.36** 2.89%	**0.00** 0.00%	**0.00** 0.00%	87.61% 12.39%
3	**0.00** 0.00%	**10.08** 4.56%	**66.28** 29.99%	**8.04** 3.65%	**0.00** 0.00%	78.48% 21.52%
4	**0.00** 0.00%	**0.12** 0.06%	**5.08** 2.31%	**23.26** 10.53%	**2.44** 1.12%	75.09% 24.91%
5	**0.00** 0.00%	**0.00** 0.00%	**0.00** 0.00%	**2.10** 0.97%	**7.44** 3.37%	78.32% 21.68%
	86.12% 13.88%	83.94% 16.06%	85.02% 14.98%	69.54% 30.46%	75.38% 24.62%	**82.03%** **17.97%**
	1	2	3	4	5	

b
Confusion Matrix
Output Class
Target Class

1	**18.00** 8.10%	**2.00** 0.90%	**0.00** 0.00%	**0.00** 0.00%	**0.00** 0.00%	89.20% 10.80%
2	**3.00** 1.40%	**66.50** 30.10%	**6.00** 2.70%	**0.00** 0.00%	**0.00** 0.00%	88.35% 11.65%
3	**0.00** 0.00%	**10.00** 4.50%	**67.00** 30.30%	**8.00** 3.60%	**0.00** 0.00%	78.95% 21.05%
4	**0.00** 0.00%	**0.00** 0.00%	**5.00** 2.30%	**23.50** 10.65%	**2.00** 0.90%	76.30% 23.70%
5	**0.00** 0.00%	**0.00** 0.00%	**0.00** 0.00%	**2.00** 0.90%	**7.00** 3.20%	77.80% 22.20%
	86.40% 13.60%	83.85% 16.15%	84.80% 15.20%	69.65% 30.35%	75.00% 25.00%	**82.40%** **17.60%**
	1	2	3	4	5	

Fig. 3. The confusion matrices obtained at 50 cross validation experiments: a) the mean, b) the median values.

Table 6
The comparative statistical results concerning the accuracy at application of different final classification systems.

	SVM	Decision tree	Ordinal regression	Fuzzy KNN
Mean	**82.03%**	81.70%	81.69%	80.74%
Median	**82.40%**	81.67%	81.90%	80.00%
Std	**2.78%**	2.61%	2.62%	2.85%

Table 7
Comparison of the accuracy of classification of data at the proposed approach and the application of one stage classifiers.

	Logistic regression + SVM	SVM	Decision tree	Ordinal regression	Fuzzy KNN
Mean	**82.03%**	78.35%	79.07%	69.06%	78.37%
Median	**82.40%**	78.44%	79.67%	69.12%	78.45%
Std	**2.78%**	2.70%	2.81%	2.82%	2.95%

classes (with 2 exceptions presented in Fig. 3a — total mean equal 0.14%). At simple, one-step approach this ratio was higher and in the case of the best decision tree classifier rose up to 0.26%.

6. Conclusion

In this study we have proposed the novel 2-stage approach to the problem of credit rating prediction of the companies. The most important point of this approach is the combination of the logistic regression method with WOE representation of data, forming the first stage of binary classification, and application of the second stage of final classification, in which the results of the first stage form the input information for second stage classifier.

Application of WOE, reflecting the information about the proportion of the classes and their association with the input attributes, provides the natural way of implementing the human knowledge of the process into the classification methodology. The presented approach split large classification problem into few smaller models, each recognizing two classes only. Such method of solution is much easier to control. This way of data processing enables to apply at each level the dedicated feature selection methods such as stepwise or forward and backward regression, leading to the simplification of the classification decision and at the same time to better understanding the basic relationships between the attributes and the classes under recognition.

The proposed approach automates the procedure of the assessment of the financial condition of the company applying for the loan, enabling to simplify the whole process and reduce the cost of its preparation. All steps of data processing are done automatically without intervention of the human operator.

The numerical experiments have been performed using the data set of few thousand financial entries corresponding to many international companies. The results of these experiments have shown that our system provides acceptable results and is superior over the classical, one stage classification systems.

References

[1] E. Altman, Managing Credit Risk, Wiley, New York, 2008.
[2] R. Anderson, The Credit Scoring Toolkit: Theory and Practice for Retail Credit Risk Management and Decision Automation, Oxford University Press, Oxford, 2007.
[3] R. Bender, A. Benner, Calculating ordinal regression models in SAS and S-Plus, Biometrical Journal 42 (6) (2000) 677–699.
[4] G.F. Bonham-Carter, F.P. Agterberg, D.F. Wright, Weights of evidence modeling: a new approach to mapping mineral potential, in: F.P. Agterberg, G.F. Bonham-Carter (Eds.), Statistical Applications in the Earth Sciences, Geological Survey of Canada, Montreal, 1989, pp. 171–183.
[5] L. Breiman, J. Friedman, R. Olshen, C. Stone, Classification and Regression Trees, CRC Press, Boca Raton, FL, 1984.
[6] J.C. David, R. Malhorta, D.K. Malhorta, Evaluating consumer loans using neural networks, Omega 31 (2003) 83–96.
[7] A. Gelman, J. Carlin, J. Stern, D. Rubin, Bayesian Data Analysis, Chapman and Hall/CRC, Boca Raton, FL, 1995.
[8] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, Journal of Machine Learning Research 3 (2003) 1158–1182.
[9] C.W. Hsu, C.J. Lin, A comparison methods for multi class support vector machines, IEEE Trans. Neural Networks 13 (2002) 415–425.
[10] http://www.amazon.com/Credit-Scoring-Toolkit-ManagementAutomation/dp/0199226407.
[11] Z. Huang, H. Chen, C.J. Hsu, W.H. Chen, S. Wu, Credit rating analysis with SVM and neural networks: a market comparative study, Decision Support Systems 37 (2004) 543–558.
[12] J.M. Keller, M.R. Gray, J.A. Givens, A fuzzy k-nearest neighbour algorithm, IEEE Transactions Systems Man and Cybernetics 15 (4) (1985) 580–585.
[13] K.C. Lee, I. Han, Y. Kwon, Hybrid Neural Network models for bankruptcy prediction, Decision Support Systems 18 (1996) 63–72.
[14] D. MacKay, Information Theory, Inference, and Learning Algorithms, Cambridge University Press, Cambridge, 2003.
[15] Matlab User Manual, MathWorks, Natick, USA, 2010.
[16] P. McCullagh, J.A. Nelder, Generalized Linear Models, Chapman & Hall, New York, 1990.
[17] P.C. Pendharkar, Hybrid approaches for classification under information acquisition cost constraint, Decision Support Systems 41 (1) (2005) 228–241.
[18] B. Schölkopf, A. Smola, Learning with Kernels, MIT Press, Cambridge. MA, 2002.
[19] A.P. Sinha, H. Zhao, Incorporating domain knowledge into data mining classifiers: an application in indirect lending, Decision Support Systems 46 (1) (2008) 287–299.
[20] V. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.

Stanislaw Osowski was born in Poland in 1948. He received the M.Sc., Ph.D., and habilitate doctorate (Dr.Sc.) degrees from Warsaw University of Technology, Warsaw, Poland, in 1972, 1975, and 1981, respectively, all in electrical engineering. Currently he is a professor of electrical engineering at the Institute of the Theory of Electrical Engineering, Measurement and Information Systems, Warsaw University of Technology and also at Military University of Technology. His research and teaching interest are in the computational intelligence, neural networks and data mining.

Bartosz Swiderski was born in Poland in 1978. He received the M. Sc. degree from Lodz University and Ph.D. degree in 2007 from Warsaw University of Technology. Actually he is an Adjunct in Faculty of Applied Informatics and Mathematics of University of Life Sciences. His research interest is in the computational intelligence and data mining.

Jaroslaw Kurek was born in Poland in 1981. He received the M.Sc. and Ph.D. degrees in computer engineering from Warsaw University of Technology, Warsaw, Poland in 2004 and 2008, respectively. Currently he is an Adjunct in Faculty of Applied Informatics and Mathematics of University of Life Sciences. His research interest is in the computational intelligence and data mining.