Credit Risk Scoring in Entrepreneurship: Feature Selection

Mirjana Pejić Bach
Ekonomski fakultet Zagreb, Croatia
mpejic@efzg.hr

Nataša Šarlija
Ekonomski fakultet Zagreb, Croatia
natasa@efos.hr

Jovana Zoroja
Ekonomski fakultet Zagreb, Croatia
jzoroja@efzg.hr

Božidar Jaković
Ekonomski fakultet Zagreb, Croatia
bjakovic@efzg.hr

Dijana Ćosić
Wealthengine, Washington DC, USA
dijana.cosic@gmail.com

The goal of this research is to investigate the impact of different algorithms for the feature selection for the purpose of credit risk scoring for the entrepreneurial funding by the Croatian financial institution. We use demographic and behavioral data, and apply various algorithms for the development of classification model. In addition, we evaluate several algorithms for the variable selection, which are additionally based on the classification accuracy. Sequential Minimal Optimization algorithm in combination with the Class CfcSubsetEval and ConsistencySubsetEval algorithms for variable selection was the most accurate in predicting credit default, and therefore the most useful for the credit risk scoring.

Key Words: data mining, credit scoring, variable selection, decision tress, classification
JEL Classification: C61, E51
https://doi.org/10.26493/1854-6935.17.265-287

Introduction

Data mining methods are used to find undiscovered valuable information from large databases. In other words, the main goal of data mining

techniques is to extract knowledge in order to make successful management decisions (Wu et al. 2012). Applications of data mining methods are used in almost every industry: banking, marketing, finance, manufacturing, medicine, education, trade, supply (Wei et al. 2013; Lejeune 2001; Choudhary, Harding, and Tiwari 2008). Each industry has its own characteristics, which implies the usage of different data mining methodologies. Therefore, in the banking industry, characterized with a high level of fraud and risks, which requires successful prediction of credit default, scoring, and applicants, usage of data mining techniques is very common (Ngai et al. 2011).

Data mining is one of the most common techniques used in financial analysis, especially in the banking industry. Prediction of credit risk, mostly prediction of credit default, presents an important activity of the banking industry (Thomas et al. 2005). There are several different data mining techniques that can be used for financial data analysis because of their high level of success. However, their success also depends on the data available, its cleaning, and transformation. Decision trees are one of the most commonly used methods (Quinlan 1992; Breiman et al. 1984). Decision trees are one of the classification methods which group variables into one or more categories of the target variables (Yap, Ong, and Husain 2011). When using the decision trees process it is important to follow three main steps: (i) determine the sample, (ii) choose variables, and (iii) select an appropriate algorithm.

In this paper we use ten data mining algorithms in order to develop a credit scoring system for the classification of banking clients according to the credit default, using different sets of the variables: Entrepreneurial idea; Growth plan; Marketing plan, Personal characteristics of entrepreneurs, Characteristics of SME, Characteristics of credit program, and Relationship between the entrepreneur and a financial institution.

The variables are selected by the usage of three different algorithms, provided in the Weka software: Class CfsSubsetEva algorithm, ChiSquaredVariableEval algorithm, and ConsistencySubsetEval algorithm. Previous research that tested the efficiency of algorithms for the selection of variables was mostly conducted on the retail credit risk datasets (Oreski, Oreski, and Oreski 2012). The scientific contribution of our paper is that the algorithms for the selection of variables are tested on the real-world dataset of Croatian financial institution's business clients (entrepreneurs from Eastern Croatia).

The paper consists of six sections. After the Introduction, as the first section, there is a literature review. In Literature review, data mining methodology and its usage for predicting credit default are presented. Decision trees, as one of the data mining methods, are described as well as variables and techniques selection approaches used in this research. In the third section named Methodology, data, decision trees techniques and the variable selection process are discussed. Research results are provided in the fourth and fifth section. The fourth section elaborates on results of the different variable selection strategies, while the fifth section of the paper discusses results regarding classification efficiency measures, classification matrices and falsely predicted good and bad debtors with different variable selection approaches. The last section is Conclusion.

This work has been fully supported by the Croatian Science Foundation under the project 'Process and Business Intelligence for Business Performance' – PROSPER (IP-2014-09-3729). The first version of the paper has been presented at the MIPRO conference, 2017 in Opatija, Croatia, under the title 'Selection of Variables for Credit Risk Data Mining Models: Preliminary research' (Pejić Bach et al., 2017). We thank the conference participants for their valuable comments, which helped us improving the paper.

Literature Review

DATA MINING METHODOLOGY

The amount of data has been constantly increasing, which creates difficulties for managers and successful decision making. Fast increase of valuable as well as invaluable data in databases has created a need for the use of different methodologies which can help to find, extract and analyze the data important for decision makers (Priya and Ghosh 2013).

Data mining technology combines different approaches, e.g. machine learning, statistics and database management, which are used for finding valuable patterns in data for further prediction and decision making. In addition, data mining techniques can also be used for determining relationships among data in order to create knowledge (Ngai, Xiu, and Chau 2009). The main purpose of data mining is to find and analyze disorganized information with the goal of improving business knowledge and activities.

The most commonly used data mining methods are classification, regression, clustering, visualization, decision trees, association rules, neu-

ral networks, support vector machine (Ngai et al. 2011; Strohmeier and Piazza 2013; Patel and Sarvakar 2014).

PREDICTING CREDIT DEFAULT WITH THE DATA MINING APPROACH

Countries, especially their economies and financial institutions, have been facing a strong financial crisis in the last years. Therefore, in many countries, governments have brought many saving measures in order to decrease costs and to restart economy development. In addition, credit default has increased, and nowadays banks pay much more attention to credit risk assessment and to prediction of credit default with the goal to reduce risk (Marinakis et al. 2008).

Financial institutions and banks are using different intelligent techniques, e.g. mathematical models, statistics analysis, data mining methods with the goal to make efficient credit decisions. A detailed analysis of data on the characteristics of current and previous credit users plays an important factor in forecasting the future credit default of new clients (Thomas 2000).

VARIABLES AND TECHNIQUES SELECTION APPROACHES

In order to predict credit default, financial institutions and banks mostly use behavioral and demographic variables of previous and current clients, e. g. monthly income, marital status, real-estate owner, employment, age, gender (Lucas 2001).

The main purpose of our research is to classify banking clients regarding credit default with a decision tree analysis, using different variables related to entrepreneurship activity. In addition, financial institutions and banks, when approving credits to clients, strive to select those clients who will be able to repay it in the given period of time (Wu et al. 2008). In other words, they are focused on good clients.

There are also studies about methods used in credit scoring. One of the examples is the research which used demographic and behavioral data and three data mining methods: credit scorecard, logistic regression and a decision tree model (Yap, Ong, and Husain 2011). The results of the research showed that all three methods are appropriate for use, but the scorecards method is the easiest to apply.

There is also a study in which authors have investigated recent researches conducted in the field of credit risk assessment regarding clients and their ability to repay credits (Crook, Edelman, and Thomas 2007).

Research results showed that logistic regression is the most commonly used method to group clients into good or bad debtors.

Recent studies showed that intelligence methods used for discovering credit scoring are mostly non-parametric methods and computational intelligence techniques, e.g. decision trees, artificial neural networks, support vector machines and evolutionary algorithms (Zhang, Leung, and Ye 2008; Pourzandi and Babaei 2010; Lucas 2001).

Methodology

DATA

Data used in this research was collected from an entrepreneurship credit dataset. Data were collected randomly from the database of clients (entrepreneurs from Eastern Croatia) of the financial institution that is focused on financing small and medium enterprises, mostly start-ups. There are 200 applicants in the sample.

There are two main reasons for a small dataset: (i) a quite low level of business activity regarding entrepreneurship in Croatia (Total Early State Entrepreneurial Activity (TEA) index for the year 2015 = 7.69; TEA index for the year 2004 = 7.97) which means that a low number of people is taking a credit to start a business, and (ii) financial institutions are rejecting too risky start-ups applications for a credit. Therefore, collecting a larger sample will be possible in the next few years, when the entrepreneurial climate and perception of entrepreneurship activity improves.

The following variables were used for the development of the credit scoring model: Entrepreneurial idea; Growth plan; Marketing plan, Personal characteristics of entrepreneurs, Characteristics of SME, Characteristics of credit program, Relationship between the entrepreneur and a financial institution, and Creditworthiness. Most of the variables are nominal, while two of them are numeric (Entrepreneurs' age, Number of employees, and Credit amount).

Variables related to the future plans for the SME were estimated by a banking clerk (table 1). First, it was estimated whether the entrepreneur has a clear vision of business development (for newly established SMES), or it is already established business (Variable Vision). Second, the variable Better estimated what the main competitive advantage of the SME (better quality, technology, price, or expertise of employees) is. Third, it was estimated what the main market for SME's products/services is: local, national, wider region, or the narrowly targeted customers (Variable Mar-

TABLE 1 Variables Related to the Future Plans For the SME

Category	Variable	Question asked	Answers
Entrepreneurial idea	Vision	Does the entrepreneur have a clear vision of the business?	1 – no clear vision (for newly established SMEs) 2 – clear vision (for newly established SMEs) 3 – established business
	Better	Advantages of products/services	1 – better quality 2 – better technology 3 – good price 4 – expertise of employees 100 – no answer
	Market	Market for products/services	1 – local 2 – narrow targeted customers 3 – wider region 4 – Croatia 100 – no answer
Growth plan	Reinvest	Projected percentage of the invested profit (reinvested profit/profit*100)	1 – 0 to 30% 2 – 30.01 to 50% 3 – 50.01 to 70% 4 – 70.01 to 100% 100 – missing value
Marketing plan	Ad	Promotion of products/services	1 – without promotion 2 – no need for promotion 3 – all media 5 – personal selling, presentation 6 – posters, leaflets, internet 100 – missing value
	Compet	Can the entrepreneur identify competition?	1 – no competition 2 – not defined 3 – defined competition 100 – no answer

ket). Entrepreneurs stated which percentage of the profit is planned to be reinvested in the business operations (Variable Reinvest), and what the plans for the promotion of products/services are (Variable Ad). Also, it was estimated whether the entrepreneur could identify who SME's main competitors are.

Table 2 presents the variables related to the characteristics of the entrepreneur and SME. Entrepreneurs' occupations are grouped into 5 main groups: 1 – farmer, veterinarian; 2 – trader, restaurateur; 3 – construction worker; 4 – engineer, physician, and pharmacist, 5 – Technologist,

TABLE 2 Variables Related to the Characteristics of Entrepreneur and SME

Category	Variable	Question asked	Answers
Personal characteristics of entrepreneurs	Occup	Entrepreneurs' occupations	1 – farmer, veterinarian 2 – trader, restaurateur 3 – construction worker 4 – engineer, physician, pharmacist 5 – technologist, chemist
	Age	Entrepreneurs' ages	numeric
	Location	Entrepreneurs' locations	1 – Baranja, Osijek 2 – S. Brod, Požega 3 – Đakovo, Našice 4 – Vinkovci, Vukovar
Characteristics of SME	Ind	Industry	1 – plastics, textiles 2 – car service 3 – food production 4 – health and intellectual services 5 – agriculture 6 – construction 13 – tourism
	Start	Is this a new business venture	1 – yes 2 – no
	Equip	Does the entrepreneur have some equipment?	1 – yes 0 – no
	Emp	No. of employees	numeric

chemist. Entrepreneurs' ages are expressed as a numeric variable. Entrepreneurs' locations refer to 4 geographic areas in Croatia. Table 3 represents the variables related to the credit program and the bank: interest repayment frequency (monthly, quarterly, half-yearly), grace period, principal repayment, repayment period (expressed in months), interest rates, and amount of credit (expressed in local currency). Also, the variable Client measures whether an entrepreneur has applied for a credit before. Table 4 represents the classification variable that was used for the credit scoring (variable Default), and groups clients as 'bad' or 'good' based on the regularity of their payment.

ALGORITHMS FOR VARIABLE SELECTION

Three approaches to the variable selection were applied: (1) selection of the variables using the Class CfsSubsetEval algorithm (searching ap-

TABLE 3 Variables Related to the Credit Program and the Bank

Category	Variable	Question asked	Answers
Characteristics of credit program	Int	Interest repayment frequency	1 – monthly 2 – quarterly 4 – half-yearly
	Grace	Grace period	1 – yes 0 – no
	Prin	Principal repayment	1 – monthly 5 – yearly
	Period	Repayment period (months)	numeric
	I_rate	Interest rate	4.9% 6.9% 8.9%
	Amount	Credit amount (local currency)	numeric
Relationship between the entrepreneur and a financial institution	Client	Is this the first time the entrepreneur is applying for a credit?	1 – yes 2 – no

TABLE 4 Goal Variable Used for the Credit Scoring

Variable	Question asked	Answers
Default	'Bad' clients are defined as those who have been late with their payments for more than 45 days at least once. Other clients who have not been late for more than 45 days are labeled as 'good.'	1 – bad 0 – good

proach BestFirst), (2) selection of the variables using the ChiSquaredVariableEval algorithm (searching approach Ranker), and (3) selection of the variables using the ConsistencySubsetEval (searching approach Greedy Stepwise).

There are differences in definition and usage of the three mentioned approaches to the variable selection, which were applied. The Class CfsSubsetEval algorithm is based on the individual estimation of the variables that are highly correlated with the class variables but are not highly mutually correlated.

The ChiSquaredVariableEval calculates the value of a variable regarding the value of the chi-squared statistic with respect to the class. The ConsistencySubsetEval calculates the value of a subset of variables by the level of reliability in the class values (Hall 1998).

ALGORITHMS FOR CREDIT SCORING

The following algorithms were used for the development of credit scoring in this paper: Bayesian Network Classifier, Sequential Minimal Optimization, Logistic regression, K-nearest neighbor's classifier (Lazy IB.K), decision trees C4.5, Gaussian radial basis function network, propositional rule learner, bootstrap aggregating algorithm, Random forest, and Adaboost. These algorithms were applied using Weka software and will be described from that perspective.

Bayesian Network Classifier is a fairly simple algorithm used in data mining for classification and prediction. The probabilities of class attribute values on all the given independent variables based on a simple normal distribution are calculated with the goal of developing a network of parenting and ancestor nodes that would be easy to understand. It assumes conditional independency as only the parent nodes can provide the information and not the ancestors, which can be written as Pr[node ancestors] = Pr[node parents] (Witten and Frank 2005). Once it learns the structure, it performs a calculation of posterior probability for each class on a vector of observed attribute values (Baesensl et al. 2003). Probabilities of discrete variables are based on their frequency and the probabilities of continuous variables are calculated by a normal or kernel density-based method (John et al. 1995). Bayes Network is implemented in software Weka under the name ByesNet (Bouckaert et al. 2013). However, Weka implementation allows only nominal variables, hence continues variables need to be discretized.

Sequential Minimal Optimization (SMO) is an algorithm for training Support Vector Machines which are a mix of linear modeling and instance-based learning. They create a discriminant that separates support vectors (a small number of critical boundary instances) from each class. The further away the data points are from the hyperplane, the better they are classified (Witten and Frank 2005). The SVM maximize the margin between support vectors which becomes a Quadriatic Programming problem (Platt 1998). SMO algorithm solves the QP by breaking it into a number of smaller QP problems which are then solved analytically, which makes it fastest for linear SVM and sparse data sets (Platt 1998).

$$\max_{a} W(\alpha) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} y_i y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \alpha_i \alpha_j,$$
$$0 \leq \alpha_i \leq C, \forall i,$$

$$\sum_{i=1}^{l} = y_i \alpha_i = 0. \tag{1}$$

The SMO solves the QR if the $Q_{ij} = y_i y_j k(\boldsymbol{x}_i; \boldsymbol{x}_i)$ positive semi-definite and Karush-Kuhn-Tucker (KKT) conditions are fulfilled for any i (Platt 1998):

$$\begin{aligned} &\alpha_i = 0 \Rightarrow y_i f(\boldsymbol{x}_i) \geq 1, \\ &0 < \alpha_i < X \Rightarrow y_i f(\boldsymbol{x}_i) = 1, \\ &\alpha_i = C \Rightarrow y_i f(\boldsymbol{x}_i) \leq 1. \end{aligned} \tag{2}$$

Logistic regression algorithm in Weka is a class for building a multinomial logistic regression model combined with ridge estimators (le Cessie and van Houwelingen 1992). Logistic regression calculates the probability that the input falls into a given class, so the probability is always between 0 and 1. The equation for a $p = P(y = 1)$ is (Baesensl et al. 2003):

$$p = \frac{e^{b_0 + b_1 x_1 + b_2 x_2 + \cdots}}{1 - e^{b_0 + b_1 x_1 + b_2 x_2 + \cdots}}, \tag{3}$$

where $b_0 + b_1 x_1 + b_2 x_2 + \cdots$ is a boundary function (logistic regression creates the discriminator as well as the SMO), x_n are the independent variables and b's are regression coefficients (b_0 is an intercept and b_n are the parameter vectors). In order to improve the parameter estimates and to reduce the error, Weka uses ridge estimators (le Cessie and van Houwelingen 1992).

Lazy.IBk is the K-nearest neighbors' classifier. It is an instant-based learning classifier, which is a type of lazy learning algorithm that is fairly simple (Aha, Kibler, and Albert 1991). It makes a construction hypothesis from the training instances. The output is a class membership. KNN considers only k-most similar data instances from the training set in order to classify an instance. Class label is classified by the majority of the k-nearest neighbors. Lazy.IBk uses Euclidean distance as the similarity measure (Baesens et al. 2003) :

$$d(x_1, x_j) = \|x_1 - x_j\| = [(x_1 - x_j)^T (x_1 - x_j)]^{1/2}, \tag{4}$$

$x_1, x_j \in R^n$ as the input vectors of data instance i and j.

Decision trees C4.5 (J48) is an algorithm which is very popular given that decision trees are very easy to understand. Like it's predecessor ID3, it uses the concept of information entropy (Baesens et al. 2003):

$$Entropy(S) = -p_1 \log_2(p_1) - p_0 \log_2(p_0), \tag{5}$$

Where $p_1)p_0$ is the proportion of examples in class 1 (0) in sample S. Expected reduction of entropy is (Baesens et al. 2003):

$$Gain(S, x_i) = Entropy(S) - \sum_{v \in values(x_i)} \frac{|S_v|}{|S|} Entropy(S_v), \tag{6}$$

where S_v is a subsample of S and attribute x_i has one specific value.

So, unlike ID3, it works with numerical attributes as well as with nominal ones. That is why it has built-in mechanisms that suggest 3 types of tests: Standard test on a discrete attribute (same as ID3); binary test where the P threshold and the values will be defined and more complex test based on a discrete attribute where the values will be gathered in a larger number of groups with one branch for each group. Another difference between ID3 and C4.5 is that ID3 uses 'information gain' to compare potential data and C4.5 uses 'Gain ratio' (Hssina et al. 2014) which is a proportion of information generated by the split that is useful in classification as a normalization (Baesens et al. 2003):

$$\begin{aligned} Gainratio(S, x_i) &= \frac{Gain(S, x_i)}{SplitInformation(S, x_i)}, \\ SplitInformation(S, x_i) &= \sum_{k \in values(x_i)} \frac{|S_k|}{|S|} \log_2 \frac{|S_k|}{|S|}, \end{aligned} \tag{7}$$

where S_k is a subsample of S and attribute x_i has one specific value and *SplitInformation* is the entropy of S regarding the values of x_i. C4.5 favors splits with the biggest gain ratio but the information gain must be as large as the average gain over all splits. As a result, the tree can be very complex so C4.5 follows postpruning method.

functions.RBFNetwork is a normalized Gaussian radial basis function network which is actually a neural network (Scholpkopf et al. 1996):

$$g(\boldsymbol{x}) = \sum_{i=1}^{K} w_i G_i(\boldsymbol{x}) + b = \sum_{i=1}^{K} w_i \frac{1}{(2\pi)^{d/2}\sigma_i^d} \exp(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_i\|^2}{2\sigma_i^d} + b, \tag{8}$$

where G_i is the ith Gaussian basis function with center $\boldsymbol{c}_i$ and variance σ_i^d. Variant coefficients are w_i and b is a bias term (Scholkopf et al. 1995).

rules.JRip is a rule builder, based on k-iterations of the optimization (Cohen 1995). In a building stage, this algorithm repeats growing and pruning phase until the discretion length of the rule set and examples is 64 bits greater than the smallest discretion length, or the error rate is ≥ 50% or there are no positive examples. In the growing phase, it grows one rule by adding conditions to the rule until the rule is 100% accurate.

meta.Bagging is a bagging predictor (bootstrap aggregating algorithm) that generates multiple versions of a predictor in order to get an aggregated predictor. If the outcome is numerical (estimation problem) – it averages over the versions of the predictor. For classification problems – it does a plurality votes. It makes bootstrap replicates of the learning set and to use them as new learning sets (Breiman 1996). The voting procedure can be explained as follows (Machová, Barčák, and Bednár 2006):

$$H(d_i) = sign\left(\sum_{m=1}^{M} \alpha_m H_m(d_i)\right), \tag{9}$$

d_i is classified to the class that has a majority of classifiers vote. $\alpha_m, m = 1, \ldots, M$ are set so that more precise classifiers have stronger influence on the final prediction. H_m are weak classifiers as their precision as a base classifiers can only be a little bit higher than the precision of random classification (Machová, Barčák, and Bednár 2006). Bagging was formulated by Leo Breiman in 1994 for improving classification accuracy.

trees.RandomForest is a class for constructing a forest of random trees. It was introduced by Leo Breiman in 2001. As the name says, randomness is applied both at row and at column level in order to generate many trees. As in bagging, for the classification problems – voting is used and for regression – averaging is used to declare the final result. The margin function can be formulated as (Breiman 2001):

$$mg(X, Y) = av_k I(h_k(X) = Y) - \max_{j \neq Y} av_k I(h_k(X) = j), \tag{10}$$

where $h_1(x), h_2(x), \ldots, h_k(x)$ are a group of classifiers that together with a training set (which is drawn from the distribution of random vector Y, X) define the function, and where $I(.)$ is the indicator function. So the margin calculates how much the average number of votes at X, Y for the given class exceeds the average vote for any other class. The generalization error is as follows (Breiman 2001):

$$\text{PE}^{\star} = P_{X,Y}(mg(X, Y) < 0). \tag{11}$$

Random forests produce a limiting value of the generalization error which depends on the strength of the individual trees and the correlation between them (Breiman 2001).

meta.AdaBoostM1 is a class for boosting a nominal class classifier. It uses Adaboost M1 method. Just as bagging, boosting is repeatedly running a given weak learning algorithm and combining the classifiers, but it constructs a distribution in a different way. There are two major ef-

TABLE 5 Confusion Matrix

Data class	Classified as positive	Classified as negative
Positive	True positive (TP)	False negative (FN)
Negative	False positive (FP)	True negative (TN)

fects of boosting. First one is that it generates a hypothesis whose error on the training set is small by combining many hypotheses whose error may be large. The second one is a variance reduction (taking a weighted majority over many hypotheses, which were trained on different samples from the same training set – it reduces the random variability of the combined hypothesis). Adaboost M1 calls the WeakLearn learning algorithm repeatedly in a serious of rounds. The goal is to find a hypothesis which minimizes the training error (Freund and Schapire 1996):

$$e_t = Pr_{iD_t}[h_t(x_i) \neq y_i], \tag{12}$$

where x_i is an instance, y_i is the class label associated with x_i, D_t is the distribution provided by weak learner in a round t and h_t is the hypothesis. The combined hypothesis is a weighted linear threshold of the weak hypotheses. AdaBoostM1 gives more weight to examples which seem to be the hardest. So the upper bound on error of the final hypothesis is as follows (Freund and Schapire 1996):

$$\frac{1}{m}|\{i: h_{fin}(x_i) \neq y_i\}| \leq \prod_{t=1}^{T} \sqrt{1 - 4\gamma_t^2} \leq \exp(-2 \sum_{t=1}^{T} \gamma_t^2). \tag{13}$$

The major disadvantage of Adaboost M1 is that it cannot handle a weak hypothesis with error greater than 1/2 (Freund and Schapire 1996).

ACCURACY MEASURES

The following measures were used for the comparison of algorithms for credit scoring in combination with various variable selection approaches: % correct, Kappa, True positive rate, True negative rate, False positive rate, False negative rate, IP Precision, IP Recall, F-measure, and ROC area.

The correctness of a classification algorithm can be measured by computing the number of correctly classified class instances: TP (true positive) and TN (true negative) and a number of instances that were incorrectly assigned to the class: FP (false positive) and FN (false negative). Together they form the confusion matrix for binary classification (Sokolova and Lapalme 2009) (table 5):

Percentage of correctly classified instances is defined as a sum of TP

and TN divided by a total number of instances. It is also called accuracy.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \tag{14}$$

Precision is defined as TP divided by the number of examples classified as positive (TP + FP). It is a class agreement of the data labels with the positive labels classified.

$$Precision = \frac{TP}{TP + FP}. \tag{15}$$

True positive (TP) rate is the number of correctly classified positive divided by a positive label (a sum of TP and FN). It is also called a sensitivity measure or a recall. It shows the effectiveness of an algorithm to predict positive cases. The higher the TP rate, the better the performance of an algorithm.

$$TP\ rate = \frac{TP}{TP + FN}. \tag{16}$$

In a case of the credit risk data it is a measure that shows how well the algorithms are classifying instances into 'good' clients (variable 'Default'). Both recall and precision do not take into account the number of TN.

True negative (TN) rate is the number of correctly classified negative instances divided by negative labels (sum of TN and FP). It shows how well a classifier identifies negative labels. It is also called a measure of specificity.

$$TN\ rate = \frac{TN}{TN + FN}. \tag{17}$$

False positive (FP) rate is a number of falsely classified positive instances divided by a total number of negative instances.

$$FP\ rate = \frac{FP}{FP + TN}. \tag{18}$$

False negative (FN) rate is a number of falsely predicted negative events divided by positive instances (a total of FN and TP).

$$FM\ rate = \frac{FN}{FM + TP}. \tag{19}$$

F-measure also named balanced F-score is a measurement of precision and recall as it is the approximate average of the two (Zhang, Zhang, and Yang 2003).

$$F = 2\frac{precision \cdot recall}{precision + recall}. \tag{20}$$

It is the same as accuracy but without the effect of TN. The reason to do F-measure is that there can be a very high precision and very low recall

for example. That is why F measurement does the averaging. For ratios it is better to use harmonic mean over the geometric and arithmetic mean. Same as in recall and precision, the higher the value is, the algorithm performs better (Powers 2011). A more general form of F-measure is defined as weighted harmonic mean of recall and precision:

$$F_\beta = \frac{(\beta^2 + 1) \cdot P \cdot R}{\beta^2 P + R}, \tag{21}$$

where β is a relative importance of precision and recall (Zhang 2007).

ROC (Receiver operating characteristic) graph is a fundamental tool for diagnostic test evaluation. ROC shows the FP rate (specificity) on the x axis and TP rate (sensitivity) on the y axis. It shows relations between benefits (TP) and costs (FP). The diagonal line on ROC graph presents a 50:50 chance of guessing TP on account of FP. There are 5 discrete classifiers. D presents a perfect classifier with specificity = 0 and sensitivity = 1 and C (0,7,0,7) represents a random chance of guessing TP 70% of the time. The classifier is better the further northwest it is from the classifier it compares to. Those that have low TP rate and also make a small mistake on FP are called 'conservative.' The ones that are placed in an opposite corner on the left side of ROC are called 'liberal' as they predict more TP but also do a fair share of FP. Another extreme is a point E which represents a negation of a point B and performs worse than guessing (Fawcett 2005). That classifier has the useful information but it does not apply them correctly.

Many classifiers produce a class decision on each instance and in the end produce only one confusion matrix which gives only one point in ROC space. Such are decision trees or rule sets. But rather than having only one point, scoring and voting can be used in order to create a curve. In order to compare classifiers AUC (Area under an ROC curve) can be used. It shows 'the probability that the classifier will rank a randomly chosen positive instance higher than a negative one' (Fawcett 2005).

Cohen's kappa coefficient (κ) is a classification measurement for categorical variables which measures inter-rate agreement. In other words, it is a measure of how well the algorithm performed to how well it would have performed by chance. If κ is high – there is a big difference between the accuracy and the null error rate.

$$\kappa \equiv \frac{p_0 - p_e}{1 - p_e}, \text{ where } p_0 = p_{11} + p_{12} \text{ and } p_e = f_1 g_1 + f_2 g_2, \tag{22}$$

where p_0 is the proportion of observed agreement among the raters and

TABLE 6 Variables Selected by Different Algorithms

Category	Variable	(1)	(2)	(3)
Entrepreneurial idea	Vision	✓	✓	✓
	Better	×	✓	✓
	Market	×	✓	✓
Growth plan	Reinvest	✓	✓	✓
Marketing plan	Ad	✓	✓	✓
	Compet	×	✓	✓
Personal characteristics of entrepreneurs	Occup	×	✓	✓
	Age	×	✓	×
	Location	×	×	×
Characteristics of SME	Ind	×	✓	✓
	Start	×	✓	×
	Equip	×	✓	×
	Emp	×	✓	×
Characteristics of credit program	Int	×	✓	✓
	Grace	×	✓	✓
	Prin	×	✓	×
	Period	×	✓	×
	I_rate	×	✓	×
	Amount	×	✓	×
Relationship between the entrepreneur and a financial institution	Client	×	✓	×

NOTES Column headings are as follows: (1) Class CfsSubsetEval, (2) ChiSquaredVariableEval, (3) ConsistencySubsetEval.

p_e is the hypothetical probability of a chance agreement (Byrt, Bishop, and Carlin 1993). Kappa statistic varies from 0 to 1, with the 1 as perfect agreement, and 0 as the 0 = agreement equivalent to chance.

Results

Table 6 presents the variables used for a different approach to the variable selection. The Class CfsSubsetEval algorithm selected only three variables: Variable Vision – does the entrepreneur have a clear vision of the business? Variable Ad – promotion of products/services. Variable Reinvest – projected percentage of the invested profit.

The ChiSquaredVariableEval algorithm selected all of the variables ex-

cept the Variable Location. The ConsistencySubsetEval selected all of the variables related to the Entrepreneurial idea, Growth plan, and Marketing plan. However, only a few algorithms were selected that were related to the Personal characteristics of entrepreneurs, Characteristics of SME, Characteristics of a credit program, and Relationship between the entrepreneur and a financial institution.

Research results showed that variables selected by the algorithm Class CfsSubsetEval have the best results regarding the percentage of correctly classified instances. On the other hand, according to the percentage of bad debtors falsely predicted as the good ones, the decision tree generated using the variables selected by the ChiSquaredVariableEval is the worse. According to the criteria of the minimal percentage of falsely predicted bad debtors as good, the best approach was to use the decision tree generated using the variables selected by the Class CfsSubsetEval or the decision tree generated using the variables selected by the ConsistencySubsetEval. In addition, for financial institutions, especially for banks, the most valuable data are the data on prediction of bad debtors, and in our case two mentioned algorithms should be used. However, since the Class CfsSubsetEval generates a decision tree that is based only on the Variable Vision, it is prone to subjective mistakes, since this variable was estimated by a banking clerk. The ConsistencySubsetEval could be considered as more reliable, since it produces similar results as the Class CfsSubsetEval, and it is based on a larger number of variables. Most of the variables are related to 'what has been done' instead of 'who is doing it.' In other words, variables related to Entrepreneurial idea, Growth plan, and Marketing plan were more relevant than variables related to Personal characteristics of entrepreneurs, Characteristics of SME, Characteristics of credit program, and Relationship between the entrepreneur and a financial institution.

Table 7 presents the impact of different variable selections to performance of classification algorithms using selected accuracy measures. According to the IP Recall, the most superior algorithm is C4.5 in combination of Class CfcSubsetEval, with IP Recall of 0,9829. However, according to other accuracy measures, the most superior algorithm is SVO, in combination with the Class CfcSubsetEval and ConsistencySubsetEval.

Table 8 presents the performance of various algorithms with the usage of various approaches for the variable selection, namely true positive and true negative rate, as well as false positive and false negative rate. Again, according to these accuracy measures, the most superior algorithm is

TABLE 7 Performance of Various Algorithms with the usage of Various Approaches for the Variable Selection, According to Accuracy Measures

Measure	Dataset	(1)	(2)	(3)	(4)	(5)	(6)
AdaBoost	Original dataset	68.5763	0.1778	0.7046	0.9120	0.7924	0.6016
	Class CfcSubsetEval	70.7526	0.2148	0.7100	0.9494	0.8107	0.6768
	ChiSquaredAttributeEval	70.4605	0.2182	0.7139	0.9326	0.8068	0.6512
	ConsistencySubsetEval	68.8842	0.1844	0.7060	0.9158	0.7948	0.6049
Bagging	Original dataset	64.8237	0.1079	0.6904	0.8576	0.7622	0.6406
	Class CfcSubsetEval	68.4632	0.1816	0.7064	0.9038	0.7906	0.7125
	ChiSquaredAttributeEval	66.2947	0.1421	0.6958	0.8761	0.7727	0.6575
	ConsistencySubsetEval	64.8289	0.1121	0.6918	0.8537	0.7613	0.6384
Bayesian Network	Original dataset	67.5658	0.2641	0.7539	0.7682	0.7556	0.6977
	Class CfcSubsetEval	72.9289	0.2764	0.7243	0.9612	0.8249	0.7474
	ChiSquaredAttributeEval	69.7526	0.3011	0.7580	0.8054	0.7762	0.7142
	ConsistencySubsetEval	68.2974	0.2794	0.7561	0.7775	0.7617	0.7024
C4.5	Original dataset	67.6079	0.1588	0.7010	0.8995	0.7839	0.6039
	Class CfcSubsetEval	71.1421	0.1960	0.7029	0.9829	0.8188	0.6183
	ChiSquaredAttributeEval	69.0921	0.1765	0.7015	0.9335	0.7985	0.6136
	ConsistencySubsetEval	67.3421	0.1521	0.6990	0.8979	0.7820	0.5990
K-nearest neighbor	Original dataset	64.3211	0.1466	0.7041	0.7986	0.7454	0.5994
	Class CfcSubsetEval	70.1974	0.2630	0.7344	0.8699	0.7935	0.6863
	ChiSquaredAttributeEval	65.9026	0.1545	0.7035	0.8432	0.7641	0.6394
	ConsistencySubsetEval	62.7711	0.0924	0.6863	0.8071	0.7386	0.6074
Linear logistic	Original dataset	69.6263	0.3189	0.7795	0.7705	0.7674	0.7281
	Class CfcSubsetEval	71.8921	0.2924	0.7379	0.9023	0.8089	0.7460
	ChiSquaredAttributeEval	72.2211	0.3672	0.7855	0.8092	0.7922	0.7212
	ConsistencySubsetEval	71.7263	0.3621	0.7916	0.7912	0.7848	0.7416
Random Forest	Original dataset	66.1263	0.1399	0.6974	0.8683	0.7712	0.6833
	Class CfcSubsetEval	70.6184	0.2786	0.7405	0.8660	0.7953	0.7077
	ChiSquaredAttributeEval	66.8105	0.1872	0.7099	0.8453	0.7688	0.6832
	ConsistencySubsetEval	66.6342	0.1509	0.7000	0.8746	0.7754	0.6710
RBF Network	Original dataset	66.7658	0.2297	0.7358	0.7860	0.7550	0.6693
	Class CfcSubsetEval	72.6474	0.2995	0.7372	0.9209	0.8166	0.7179
	ChiSquaredAttributeEval	69.9711	0.2952	0.7532	0.8226	0.7818	0.6963
	ConsistencySubsetEval	67.5368	0.2513	0.7418	0.7878	0.7600	0.6723
Ripper rule induction	Original dataset	65.8237	0.1393	0.6969	0.8582	0.7673	0.5673
	Class CfcSubsetEval	68.9105	0.2058	0.7140	0.8939	0.7904	0.6004
	ChiSquaredAttributeEval	68.6105	0.2129	0.7168	0.8751	0.7852	0.6017
	ConsistencySubsetEval	68.6368	0.2090	0.7152	0.8807	0.7864	0.5976
SVO	Original dataset	74.4132	0.4168	0.8043	0.8209	0.8088	0.7081
	Class CfcSubsetEval	73.1289	0.2824	0.7259	0.9612	0.8259	0.6217
	ChiSquaredAttributeEval	73.7237	0.3815	0.7789	0.8521	0.8102	0.6835
	ConsistencySubsetEval	75.7921	0.4445	0.8095	0.8403	0.8204	0.7195

NOTES Column headings are as follows: (1) % correct, (2) kappa, (3) IP precission, (4) IP recall, (5) F-measure, (6) ROC area.

TABLE 8 Performance of Various Algorithms with the Usage of Various Approaches for the Variable Selection, According to Forecasting Rate

Measure	Dataset	(1)	(2)	(3)	(4)
AdaBoost	Original dataset	0.9120	0.2448	0.7552	0.0880
	Class CfcSubsetEval	0.9494	0.2345	0.7655	0.0506
	ChiSquaredAttributeEval	0.9326	0.2590	0.7410	0.0674
	ConsistencySubsetEval	0.9158	0.2464	0.7536	0.0842
Bagging	Original dataset	0.8576	0.2407	0.7593	0.1424
	Class CfcSubsetEval	0.9038	0.2571	0.7429	0.0962
	ChiSquaredAttributeEval	0.8761	0.2488	0.7512	0.1239
	ConsistencySubsetEval	0.8537	0.2488	0.7512	0.1463
Bayesian Network	Original dataset	0.7682	0.4957	0.5043	0.2318
	Class CfcSubsetEval	0.9612	0.2767	0.7233	0.0388
	ChiSquaredAttributeEval	0.8054	0.4871	0.5129	0.1946
	ConsistencySubsetEval	0.7775	0.4993	0.5007	0.2225
C4.5	Original dataset	0.8995	0.2407	0.7593	0.1005
	Class CfcSubsetEval	0.9829	0.1805	0.8195	0.0171
	ChiSquaredAttributeEval	0.9335	0.2176	0.7824	0.0665
	ConsistencySubsetEval	0.8979	0.2357	0.7643	0.1021
K-nearest neighbor	Original dataset	0.7986	0.3393	0.6607	0.2014
	Class CfcSubsetEval	0.8699	0.3738	0.6262	0.1301
	ChiSquaredAttributeEval	0.8432	0.2981	0.7019	0.1568
	ConsistencySubsetEval	0.8071	0.2774	0.7226	0.1929
Linear logistic	Original dataset	0.7705	0.5521	0.4479	0.2295
	Class CfcSubsetEval	0.9023	0.3631	0.6369	0.0977
	ChiSquaredAttributeEval	0.8092	0.5543	0.4457	0.1908
	ConsistencySubsetEval	0.7912	0.5726	0.4274	0.2088
Random Forest	Original dataset	0.8683	0.2581	0.7419	0.1317
	Class CfcSubsetEval	0.8660	0.3945	0.6055	0.1340
	ChiSquaredAttributeEval	0.8453	0.3238	0.6762	0.1547
	ConsistencySubsetEval	0.8746	0.2610	0.7390	0.1254
RBF Network	Original dataset	0.7860	0.4383	0.5617	0.2140
	Class CfcSubsetEval	0.9209	0.3476	0.6524	0.0791
	ChiSquaredAttributeEval	0.8226	0.4617	0.5383	0.1774
	ConsistencySubsetEval	0.7878	0.4567	0.5433	0.2122
Ripper rule induction	Original dataset	0.8582	0.2669	0.7331	0.1418
	Class CfcSubsetEval	0.8939	0.2905	0.7095	0.1061
	ChiSquaredAttributeEval	0.8751	0.3174	0.6826	0.1249
	ConsistencySubsetEval	0.8807	0.3074	0.6926	0.1193
SVO	Original dataset	0.8209	0.5952	0.4048	0.1791
	Class CfcSubsetEval	0.9612	0.2821	0.7179	0.0388
	ChiSquaredAttributeEval	0.8521	0.5150	0.4850	0.1479
	ConsistencySubsetEval	0.8403	0.5988	0.4012	0.1597

NOTES Column headings are as follows: (1) true positive rate, (2) true negative rate, (3) false positive rate, (4) false negative rate.

SVO, in combination with the Class CfcSubsetEval and ConsistencySubsetEval.

Conclusion

The goal of the research was to investigate which data mining algorithm would reveal the best accuracy results for the purpose of developing credit scoring for entrepreneurs financing in one Croatian banking institution. The novelty of this work is based on the data as well as algorithm selection process. We use the data from one Croatian banking institution, which provided the dataset of entrepreneurial credit, with non-standard data, such as behavioral and demographic data, as well as data which are specific for entrepreneurial projects as well as entrepreneurial attitudes. On this dataset, we tested the accuracy of various data mining algorithms, combined with various algorithms for variable selection. Based on most of the accuracy measures, the most accurate algorithm was SVO in combination with the Class CfcSubsetEval and ConsistencySubsetEval algorithms for variable selection.

References

Aha, D., D. Kibler, and M. Albert. 1991. 'Instance-Based Learning Algorithms.' *Machine Learning* 6 (1): 37–66.

Baesensl, B., T. Van Gestel, S. Viaene, M. Stepanova, J. Suykens, and J. Vanthienen. 2003. 'Benchmarking State-of-the-Art Classification Algorithms for Credit Scoring.' *Journal of the Operational Research Society* 54 (6): 627–35.

Bouckaert, R. R., E. Frank, M. Hall, R. Kirkby, P. Reutemann, A. Seewald, A., and D. Scuse. 2013. *WEKA Manual for Version 3-7-8.* Hamilton: University of Waikato.

Breiman, L. 1996. 'Bagging Predictors.' *Machine Learning* 24 (2): 123–40.

———. 2001. 'Random Forests.' *Machine Learning* 45 (1): 5–32.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees.* Belmont, CA: Wadsworth.

Byrt, T., J. Bishop, and J. B. Carlin. 1993. 'Bias, Prevalence and Kappa.' *Journal of Clinical Epidemiology* 46 (5): 423–29.

Choudhary, A. K., J. A. Harding, and M. K. Tiwari. 2008. 'Data Mining in Manufacturing: A Review Based on the Kind of Knowledge.' *Journal of Intelligent Manufacturing* 20 (5): 501–21.

Cohen, W. W. 1995. 'Fast Effective Rule Induction.' In *ICML '95: Proceedings of the Twelfth International Conference on Machine Learning,* 115–23. San Mateo, CA: Morgan Kaufmann.

Crook, J. N., D. B. Edelman, and L. C. Thomas. 2007. 'Recent Developments in Consumer Credit Risk Assessment.' *European Journal of Operational Research* 183 (3): 1447–65.

Fawcett, T. 2006. 'An Introduction to ROC Analysis.' *Pattern Recognition Letters* 27 (8): 861–74.

Freund, Y., and R. E. Schapire. 1996. 'Experiments with a New Boosting Algorithm.' In *Machine Learning: Proceedings of the Thirteenth International Conference on Machine Learning,* 148–56. Bari: The International Machine Learning Society.

Hall, M. A. 1998. *Correlation-Based Feature Subset Selection for Machine Learning.* Hamilton: The University of Waikato.

Hssina, B., A. Merbouha, H. Ezzikouri, and M. Erritali. 2014. 'A Comparative Study of Decision Tree ID3 and C4.5.' *International Journal of Advanced Computer Science and Applications* 10 (Special Issue): 13–19.

John, G. H., and P. Langley. 1995. 'Estimating Continuous Distributions in Bayesian Classifiers.' In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence,* 338–45. San Mateo, CA: Morgan Kaufmann.

le Cessie, S., and J. C. van Houwelingen. 1992. 'Ridge Estimators in Logistic Regression.' *Applied Statistics* 41 (1): 191–201.

Lejeune, M. A. P. M. 2001. 'Measuring the Impact of Data Mining on Churn Management.' *Internet Research: Electronic Networking Applications and Policy* 11 (5): 375–87.

Lucas, A. 2001 'Statistical Challenges in Credit Card Issuing.' *Applied Stochastic Models in Business and Industry* 17 (1): 83–92.

Machová, K., F. Barčák, and P. Bednár. 2006. 'A Bagging Method Using Decision Trees in the Role of Base Classifiers.' *Acta Polytechnica Hungarica* 3 (2): 121–32.

Marinakis, Y., M. Marinaki, M. Doumpos, N. Matsatsinis, and C. Zopounidis. 2008. 'Optimization of Nearest Neighbor Classifiers via Metaheuristic Algorithms for Credit Risk Assessment.' *Journal of Global Optimization* 42 (2): 279–93.

Ngai, E. W. T., L. Xiu, and D. C. K. Chau. 2009. 'Application of Data Mining Techniques in Customer Relationship Management: A Literature Review and Classification.' *Expert Systems with Applications* 36 (10): 2592–602.

Ngai, E. W. T., Y. Hu, Y. H. Wong, Y. Chen, and X. Sun. 2011. 'The Application of Data Mining Techniques in Financial Fraud Detection: A Classification Framework and an Academic Review of Literature.' *Decision Support Systems* 50 (3): 559–69.

Oreski, S., D. Oreski, and G., Oreski. 2012. 'Hybrid System with Genetic Algorithm and Artificial Neural Networks and Its Application to Re-

tail Credit Risk Assessment.' *Expert Systems with Applications* 39 (16): 12605–17.

Patel, H. G., and K. Sarvakar. 2014. 'Research Challenges and Comparative Study of Various Classification Technique Using Data Mining.' *International Journal of Latest Technology in Engineering, Management & Applied Science* 3 (9): 170–76.

Platt, J. C. 1998. 'Fast Training of Support Vector Machines Using Sequential Minimal Optimization.' In *Advances in Kernel Methods: Support Vector Learning,* edited by A. J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, 185–208. Cambridge, MA: MIT Press.

Pejić Bach, M., J. Zoroja, B. Jaković, and N. Šarlija. 2017. 'Selection of Variables for Credit Risk Data Mining Models: Preliminary research.' In *40th Jubilee International Convention on Information and Communication Technology, Electronics and Microelectronics,* edited by P. Biljanović, 1599–604. Rijeka: Croatian Society for Information and Communication Technology, Electronics and Microelectronics.

Pourzandi, M. M., and K. Babaei. 2010. 'Using Genetic Algorithm in Optimizing Decision Trees for Credit Scoring of Banks Customers.' *Journal of Information Technology Management* 2 (4): 23–38.

Powers, D. 2011. 'Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation.' *Journal of Machine Learning Technologies* 2 (1): 37–63.

Priya, P. I., and D. K. Ghosh. 2013. 'A Survey on Different Clustering Algorithms in Data Mining Technique.' *International Journal of Modern Engineering Research* 3 (1): 267–74.

Quinlan, J. 1992. *C4.5: Programs for Machine Learning.* San Francisco, CA: Morgan Kaufmann.

Scholkopf, B., K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik. 1996. *Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers.* Cambridge, MA: Massachusetts Institute of Technology.

Sokolova, M., and G. Lapalme. 2009. 'A Systematic Analysis of Performance Measures for Classification Tasks.' *Information Processing and Management* 45 (4): 427–37.

Strohmeier, S., and F. Piazza. 2013. 'Domain Driven Data Mining in Human Resources Management: A Review of Current Research.' *Expert Systems with Applications* 40 (7): 2410–20.

Thomas, L. C. 2000. 'A Survey of Credit and Behavioral Scoring.' *International Journal of Forecasting* 16 (2): 149–72.

Thomas, L. C., R. W. Oliver, and D. J. Hand. 2005. 'A Survey of the Issues in Consumer Credit Modelling Research.' *Journal of the Operational Research Society* 56 (9): 1006–15.

Wei, J.-T., M.-C. Lee, H.-K. Chen, and H.-H. Wu. 2013. 'Customer Relationship Management in the Hairdressing Industry: An Application of Data Mining Techniques.' *Expert Systems with Applications* 40 (18): 7513–18.

Witten, I. H., and E. Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques.* San Francisco, CA: Morgan Kaufmann.

Wu, R.-S., C. S. Ou, H.-J. Lin, S.-I. Chang, and D. C. Yen. 2012. 'Using Data Mining Technique to Enhance Tax Evasion Detection Performance.' *Expert Systems with Applications* 39 (10): 8769–77.

Wu, X., V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. F. M. Ng, B. Liu, P. S. Yu, Z. H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg. 2008. 'Top 10 Algorithms in Data Mining.' *Knowledge and Information Systems* 14 (1): 1–37.

Yap, B. W., S. H. Ong, and N. H. M. Husain. 2011. 'Using Data Mining to Improve Assessment of Credit Worthiness via Credit Scoring Models.' *Expert Systems with Applications* 38 (10): 13274–83.

Zhang, S., C. Zhang, and Q. Yang. 2003. 'Data Preparation for Data Mining.' *Applied Artificial Intelligence* 17 (5–6): 375–81.

Zhang, H. 2007. 'Comments on Data Mining Static Code Attributes to Learn Defect Predictors.' *IEEE Transactions on Software Engineering* 33 (9): 635–37.

Zhang, D. F., S. Leung, and Z. M. Ye. 2008. 'A Decision Tree Scoring Model Based on Genetic Algorithm and K-Means Algorithm.' In *Proceedings of the 3rd International Conference on Convergence and Hybrid Information Technology,* 1043–47. Busan: IEEE.