Expert Systems with Applications 39 (2012) 12605–12617

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment

Stjepan Oreski [a,*], Dijana Oreski [b], Goran Oreski [a]

[a] Bank of Karlovac, I.G.Kovacica 1, 47000 Karlovac, Croatia
[b] Faculty of Organization and Informatics, University of Zagreb, Pavlinska 2, 42000 Varaždin, Croatia

ARTICLE INFO

Keywords:
Classification
Credit scoring
Neural network
Genetic algorithm
Feature selection

ABSTRACT

The databases of the banks around the world have accumulated large quantities of information about clients and their financial and payment history. These databases can be used for the credit risk assessment, but they are commonly high dimensional. Irrelevant features in a training dataset may produce less accurate results of classification analysis. Data preprocessing is required to prepare the data for classification to increase the predictive accuracy. Feature selection is a preprocessing technique commonly used on high dimensional data and its purposes include reducing dimensionality, removing irrelevant and redundant features, facilitating data understanding, reducing the amount of data needed for learning, improving predictive accuracy of algorithms, and increasing interpretability of models. In this paper we investigate the extent to which the total data, owned by a bank, can be a good basis for predicting the borrower's ability to repay the loan on time. We propose a feature selection technique for finding an optimum feature subset that enhances the classification accuracy of neural network classifiers. Experiments were conducted on the credit dataset collected at a Croatian bank to assess the accuracy of our technique. We found that the hybrid system with genetic algorithm is competitive and can be used as feature selection technique to discover the most significant features in determining risk of default.

1. Introduction

The credit crisis, which began in July 2007, has shaken financial markets, undermined consumer and investor confidence, raised serious concerns and fears of financial institutions about the stability of financial markets in general, and was threatening economies around the world. While this crisis had many causes, it is clear now that banks, governments and others institutions can do more to prevent many of these problems in the future.

In this context, Basel Committee's response to the crisis is a comprehensive set of reform measures, to strengthen the regulation, supervision and risk management of the banking sector. These measures form a new international regulatory framework for banks – "Basel III".[1] The reforms target (BIS, 2011):

- Bank-level, or microprudential, regulation, which will help raise the resilience of individual banking institutions to periods of stress.
- Macroprudential, system wide risks that can build up across the banking sector as well as the procyclical amplification of these risks over time.

Complementary measures are taken at a bank level and the system as a whole, because the greater resilience at the individual bank level reduces the risk of system wide shocks, and vice versa.

Regulation, on the one hand, competition even more so on the other, is forcing banks to apply advanced methods in risk management. Competition has reduced the interest margin to a level that the bank can be successful only if there are no unexpected losses. Similarly, the space for acquiring additional first-class collateral is increasingly narrow. The client is willing to provide additional first-class collateral only when the loan cannot be realized in other banks without such insurance. In these circumstances, bank management was forced to seek new solutions for their business, which will have, at the same time, more flexibility and sensitivity to risk. Therefore, we can observe the risk assessment is probably the most important and most difficult segment of banking operations, and bank management the most responsible in the prevention of the aforementioned problems.

Space for action at the level of banks has since Basel II allowed banks to measure credit risk using internal ratings based (IRB) approach in order to determine the capital level. In taking this step

* Corresponding author. Tel.: +385 98 246 328; fax: +385 47 614 306.
E-mail addresses: stjepan.oreski@kaba.hr (S. Oreski), dijana.oreski@foi.hr (D. Oreski), goran.oreski@kaba.hr (G. Oreski).

[1] The Basel Committee's oversight body – the Group of Central Bank Governors and Heads of Supervision (GHOS) – agreed on the broad framework of Basel III in September 2009 and the Committee set out concrete proposals in December 2009. These consultative documents formed the basis of the Committee's response to the financial crisis and are part of the global initiatives to strengthen the financial regulatory system that have been endorsed by the G20 Leaders. The GHOS subsequently agreed on key design elements of the reform package at its July 2010 meeting and on the calibration and transition to implement the measures at its September 2010 meeting (BIS, 2011). Basel III is part of the Committee's continuous effort to enhance the banking regulatory framework. It builds on the International Convergence of Capital Measurement and Capital Standards document (Basel II).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.05.023

(BIS, 2006), the Basel Committee is also putting forward a detailed set of minimum requirements designed to ensure the integrity of these internal risk assessments. In order for internal risk assessments systems to ensure the integrity, banks have to collect data from many sources on daily bases, and use it in the evaluation of loan applicants and in regular bases classification of its own clients. A regulatory requirement was made for banks to use sophisticated credit scoring models for enhancing the efficiency of capital allocation (Khashman, 2010). Consequently, a robust and systematic credit scoring model and a loan evaluation model is important in realizing the IRB approach. The use of a classification method depends on the complexity of the institution, the size and the type of the loan, and represents an important area of study credit scoring system.

Current credit scoring models can be categorized into two major approaches (Li, Shiue, & Huang, 2006): specialized judgment and statistical modelling. The former relies on the expertise and tacit knowledge of specialists, which leads to financial experts' fatigue, misjudgment and slow response since the assessing process is usually time-consuming and laborious. The latter approach, on the other hand, can reduce such overheads due to its objectivity and consistence in nature.

Nowadays, financial institutions and researchers have developed many different quantitative credit scoring models. Šušteršic, Mramor, and Zupan (2009) have classified quantitative credit scoring models as follows: based on classical statistical methods, and based on artificial intelligence. Classical statistical methods are linear discriminant analysis, linear regression, logit, probit, tobit, binary tree and minimum method. The two most commonly used are discriminant analysis (DA) and logistic regression. Malhotra and Malhotra (2003) state that discriminant analysis suffers from bias of extreme data points, multivariate normality assumption, and equal group covariance assumptions. None of these restrictions apply to neural network models. Šušteršic et al. (2009) state that the weakness of the linear discriminant analysis is the assumption of a linear relationship between variables, which is usually nonlinear and the sensitivity to deviations from the multivariate normality assumption. Logistic regression does not require the multivariate normality assumption. Because of the linear relationship between variables both DA and logistical regression are reported to have a lack of accuracy.

There are also more sophisticated models known as artificial intelligence: expert systems, fuzzy systems, neural networks and genetic algorithms. Among these the neural networks are the possible alternative to the DA and logistic regression due to the possible complex nonlinear relationship between variables. In the literature in most cases of credit scoring problems the neural networks are more accurate than DA and logistic regression (Šušteršic et al., 2009). However, a large numbers of parameters, such as network topology, learning rate and training methods, have to be fine-tuned before the neural networks can be deployed successfully. Furthermore, drawbacks like trapping into local optimum, overfitting, and requiring huge time in learning computation tend to occur (Malhotra & Malhotra, 2003).

From the available literature, Khashman (2010) deduces that using neural networks for credit scoring and a loan evaluation has been effective over the past decade. The capability of neural networks, based on the back propagation learning algorithm, in such applications is due to the way the network operates, and the availability of training data. When feeding the information of a credit applicant to the neural network, variables are taken as input to the neural network and a linear combination of them is taken with arbitrary weights. The variables are linearly combined and subject to a non-linear transformation represented by a certain activation function (most frequently sigmoid function), then fed as inputs into the next layer for similar manipulation. The final function yields values which can be compared to a desired value. Each training case is submitted to the network, the final output compared with the observed value and the difference, the error, is propagated back through the network and the weights modified at each layer according to the contribution each weight makes to the error value. In essence, the network takes data, transforms it using the weights and activation functions into hidden value space and then possibly into further hidden value space; if further layers exist, and eventually into output layer space which is linearly separable.

Within the academic community there has been a growing body of literature on the application of many different methods for credit scoring and loan evaluation. There is no consensus as to which method a model developer should adopt for a given problem. Given this uncertainty, it is not unusual for a practitioner to construct several classifiers using different techniques, and then choose the one that yields the best solution for their problem. However, when comparing classifiers, it does not necessarily follow that the best classifier overall, outperforms all others throughout the regions of the problem domain. Consequently, error rates can often be reduced by combining the output of several classifiers. The research of classifier combination is rich in much of the relevant literature (Finlay, 2011; Twala, 2010), and represents another important area of the study of the credit scoring system.

In our opinion, the single classifier represents the first, combination of classifiers represents the second and the input data represents the third important area of credit scoring system study. Researchers did not regard the selection of variables as a crucial step of model development, possibly due to the problem of data availability. Hence, the issue of variable selection is a crucial and a challenging problem to solve before different credit scoring techniques are used to develop the best performing model (Šušteršic et al., 2009). As it is known, different variable selection techniques give different results on the same dataset. In this paper, we aim to design a hybrid system with genetic algorithm and artificial neural networks (GA-NN) for finding an optimum feature subset at retail credit risk assessment that enhances the classification accuracy of neural network classifier. We examine various combinations of the input data in terms of their contribution to correct classification of the credit applicant from the aspect of credit risks.

The remaining sections of this paper are organized as follows. Section 2 describes the problem of consumer loans to be studied in the paper and reviews the previous literature related to the problem. A brief overview of techniques and concepts used in the research is given in the third section. Section 4 describes the experimental design for data collection, feature selection, classification, performance evaluation and comparison. Section 5 discusses the experimental analysis and results that focus on prediction accuracy and misclassification costs. Section 6 concludes this paper and gives some guidelines for future work.

2. Problem statement and literature review

According to BIS (2006), credit risk is most simply defined as the potential that a bank borrower or the counterparty will fail to meet their obligations in accordance with the agreed terms.

The accuracy of the forecasts of a good or bad customer in terms of credit risk can be improved by: a good selection of input data, using the best methods of classification and combining the results of different classification methods. Until a few years, the body of research on consumer credit risk measurement was quite sparse. Quantitative consumer credit scoring models were developed much later than those for business credit, mainly, due to the problem of availability of data. Data were limited to the own databases of financial institutions. Nowadays, some data are publicly avail-

able in several countries and financial institutions and researchers have developed many different quantitative credit scoring techniques (Šušteršic et al., 2009).

The primary aim of this study is to investigate the extent to which the total data on customers, owned by a bank, can be a good basis for predicting the borrower's ability to repay the loan on time. Therefore, the hypothesis H1 and H2 have been set up:

H1: From the existing data on bank customers we can choose such a set of data (indicators) that provide a good basis for predicting the credit quality of the borrower. Under a set of data, the dataset that will be considered to provide a good basis for predicting the credit quality of the borrower, will be the one based on which the estimated accuracy of the predictions will be on a level above 80%.

H2: The GA-NN (genetic algorithm with neural networks) technique, developed in this study, is statistically significantly more accurate at 95% confidence level in comparison to some commonly used feature extraction techniques such as: Information gain, Gain ratio, Gini index and Correlation.

In recent literature on the subject of feature selection technique and methods of classification we found that Malhotra and Malhotra (2003) used a pooled data set of loans made by 12 different credit unions with a total of 1078 observations with six input variables:

1. home ownership,
2. length of time at current residence (years),
3. credit card,
4. the ratio of the total payment to the total income (ratio 1),
5. the ratio of debt to the total income (ratio 2), and
6. the credit rating of the applicant

as the factors that can discriminate between a good and a bad loan. They discovered that the neural network models consistently perform better than the MDA models in identifying potential problem loans.

The modern data mining techniques, which have made a significant contribution to the field of information science, can be adopted to construct the credit scoring models. From the computational results (Huang, Chen, & Wang, 2007) made by Tam and Kiang (1992), the neural network is most accurate in bank failure prediction, followed by linear discriminant analysis, logistic regression, decision trees, and k-nearest neighbor. In comparison with other techniques, they concluded that neural network models are more accurate, adaptive and robust.

Šušteršic et al. (2009) designed the neural network consumer credit scoring models for financial institutions where data usually used in previous research are not available. They use an extensive primarily accounting data set on transactions and account balances of clients available in each financial institution. The database for this study was created by a Slovenian bank that merged all the accounting and a few other internal bank data available for 581 short term consumer loans granted to its existing and new clients in the period 1994 to 1998. From 581 loans 401 (69.0%) were performing and 180 (31.0%) were nonperforming. The performing loans in their database were randomly selected from all performing loans the bank granted in that period and the same applies for nonperforming loans, respectively. The characteristics of each client were in the original database described by 84 variables. Finally, 21 variables selected, were used for further study as they enabled the highest accuracy in pre-testing.

Tsai, Lin, Cheng, and Lin (2009) construct the consumer loan default predicting model through conducting the empirical analysis on the customers of unsecured consumer loan from a certain financial institution in Taiwan, and adopt the borrower's demographic variables and money attitude as discriminant information. That study primarily included basic demographic variables as the influencing factors in exploring the consumer loan default behavior. After including the money attitude, the predicting accuracy rate of the model relatively increased with only regarding the basic characteristics as the variables. As a result, except for considering the borrower's demographic variables, in terms of selecting the predicting variables, this study is also added borrower's money attitude to expectably make a more accurate prediction about the possibility of default. Dataset sample has 281 cases, each case with 14 predictor variables. Within the sample, there are 207 non-default borrowers and 74 default borrowers.

Khashman's (2010) paper describes a credit risk evaluation system that uses supervised neural network models based on the back propagation learning algorithm. The neural networks were trained using real world credit application cases from German credit approval dataset which has 1000 cases, each case with 24 numerical attributes, recording various financial and demographic information about the applicants. The class attribute describes people as either good (700 observations) or bad (300 observations) credits. Other attributes include status of existing checking account, credit history, credit purpose, credit amount, savings account/bonds, duration of present employment, installment rate in percentage of disposable income, marital status and gender, other debtors/guarantors, duration in current residence, property, age, number of existing credits at this bank, job, telephone ownership, whether foreign worker, and number of dependents.

Twala (2010) treats credit risk prediction as a kind of machine learning (ML) problem. Among others, he obtained two datasets from the UCI repository of ML, German and Australian. German data was described in Khashman (2010). Australian credit card applications data set has 690 observations with 15 attributes. Of the attributes, nine are discrete with 2–14 values, and six continuous attributes. There are 307 positive instances and 383 negative instances in this data set. One or more attribute values are missing from 37 instances. All attribute names and values have been changed to meaningless symbols to protect confidentiality of the data.

Finlay (2011) used two real world data sets. Data set A was supplied by Experian UK. It contained details of retail credit applications made to several lending institutions between April and June 2002. Data set A contained 88,789 observations of which 75,528 cases were classified as good and 13,261 as bad. 39 independent variables were available. Data set B was a behavioral scoring data set provided by a supplier of revolving credit. A random sample of existing customer accounts was taken from single point in time during 2002. The data set contained 120,508 goods and 18,098 bad. 54 independent variables were available, examples of which were current and historic statement balance, arrears status, payments and various ratios of these variables over 3, 6 and 12 months.

Finlay (2010) was using behavioral scoring data set supplied by a large UK catalogue retailer that provides revolving credit. The data set contained 54 predictor variables. These are typical behavioral scoring variables. The sample contained 105,134 observations classified as good and 17,109 classified as bad.

Dataset samples for all listed studies except Finlay's were no more than 1078 cases. Finlay's research stands out significantly in the number of observed cases, but at the expense of time horizon that is not longer than 12 months. Our dataset sample has 1000 cases with time horizon of 7 years. All listed studies use a minimum of 6 to maximum 81 independent variables related to demographics, finance, behavior and money attitude information about applicants. However, there is no study which compiles all types of the information about applicants. In addition, in comparison with classical statistical techniques, many authors concluded

that neural network (NN) models are more accurate, adaptive and robust; consequently we were using NN as fitness function.

3. Methodology

As the primary aim of this study is to investigate the extent to which the total data which a bank has on consumers can be a good basis for predicting the borrower's ability to repay the loan on time, central position is given to feature selection. The following techniques were used in the feature selection; genetic algorithm, Forward selection, Information gain, Gain ratio, Gini index and Correlation. Beside that, in the proposed Neural Network Generic Model (NNGM) for classification with parameters optimization, and in genetic algorithm, focal point belongs to Neural Networks. In order to achieve the purpose of this study, in this section we will briefly describe the techniques and concepts used in the research.

3.1. Genetic algorithm

Genetic algorithm (GA) is an efficient optimization procedure. The basic principle of the genetic algorithm is inspired by the mechanisms of biological evolution (Šušteršic et al., 2009). In a genetic algorithm, a population of strings (called chromosomes), which encode candidate solutions (called individuals, members, or phenotypes) to an optimization problem, evolves toward better solutions. Traditionally, solutions are represented in the binary form as strings of 0s and 1s. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population (Fig. 1).

The core of the genetic algorithm is a loop that is performed until the condition for the completion of an evolutionary process is not met. The body of this loop makes the selection and reproduction. In the reproduction stage a completely new set of members of the new population is created from the parents through the application of genetic operators. The way we will perform genetic operators are not exactly determined, because the selection of genetic operators depends on the given optimization problem. So, on the one hand is available a simple core of the genetic algorithm, on the other hand, there is the problem that we want to solve. Presentation, control mechanisms, fitness function, method of initialization and genetic operators should also be determined. Deciding about a fitness function can be the most difficult part of the algorithm. The general idea is to give a higher fitness score to the chromosome which comes closer to solving the problem, because the chance of being selected is bigger for the chromosomes with higher fitness. Basically, there are two types of selection; proportional selection and ranking selection. The roulette wheel as a proportional selection is a commonly used selection method. It does not guarantee that the fittest member goes through to the next generation merely that it has a very good chance of doing so. It works like this: We can imagine that the population's total fitness score is represented by a pie chart, or roulette wheel. Now we assign a slice of the wheel to each member of the population. The size of the slice is proportional to that chromosomes fitness score. The fitter a member gets the bigger slice of the pie. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

```
Genetic_algorithm(){
  Generate_initial_population();
  until (condition_is_not_met_for_the_completion_of_an_evolutionary_process(){
    select_better_individuals_to_reproduce ();
    reproduction_generate_new_population ();
  }
}
```

Fig. 1. Pseudocode for the Genetic algorithm according to Golub (2001).

3.2. Forward selection

This algorithm performs the weighting under the naive assumption that the features are independent from each other. Each feature is weighted with a linear search. This approach may deliver good results after a short time if the features indeed are not highly correlated. This algorithm starts with an empty selection of features and, in each round, it adds each unused feature of the given set of examples. For each added feature, the performance is estimated using a cross-validation. Only the feature giving the highest increase of performance is added to the selection. Then a new round is started with the modified selection. The iteration will be aborted when stopping criterion is met, which can be when: (1) there is no any increase in performance, (2) there is increase in performance less then specified, either relative or absolute, and (3) there is selected determinate number of features. The enhanced algorithm is described below:

1. Create an initial population with n individuals where n is the number of features of the input example set. Each individual will use exactly one of the features.
2. Evaluate the feature sets and select only the best k.
3. For each of the k feature sets do: If there are j unused features, make j copies of the feature set and add exactly one of the previously unused features to the feature set.
4. As long as the stopping criterion is not met, go to 2.

When the parameter k has default value 1 it means that the standard selection algorithms are used. Using other values increases the runtime, but might help to avoid local optimum in the search of the global optimum.

3.3. Information gain

Information gain is a feature selection technique which provides a ranking for each feature describing the given training tuples. Information gain is based on pioneering work of Claude Shannon on information theory, which studied the value or "information content" of messages. The feature with the highest information gain minimizes the information needed to classify the tuples in the resulting partitions and reflects the lowest degree of randomness or "impurity" in these partitions. Such an approach minimizes the expected number of tests needed in the classification process.

The expected information needed to classify a tuple in D is given by

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i), \tag{1}$$

where p_i is the probability that an arbitrary tuple in D belongs to class C_i and is estimated by $|C_{i,D}|/|D|$. A log function with base 2 is used, because the information is encoded in bits. $Info(D)$ is just an average amount of information needed to identify the class label of a tuple in D. Note that, at this point, the information we have

is based solely on the proportions of tuples of each class. *Info*(*D*) is also known as the entropy of *D*.

Now, suppose we were to partition the tuples in *D* on some feature A having v distinct values, $\{a_1, a_2, \ldots, a_v\}$, as observed from the training data. How much more information would still be needed after this partitioning in order to arrive at an exact classification? This amount is measured by

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j). \quad (2)$$

The term $|D_j|/|D|$ acts as the weight of the jth partition. $Info_A(D)$ is the expected information required to classify a tuple from *D* based on the partitioning on *A*. The smaller the expected information (still) required, the greater the purity of the partitions. Information gain is defined as the difference between the original information requirement (i.e., based on just the proportion of classes) and the new requirement (i.e., obtained after partitioning on *A*). That is,

$$Gain(A) = Info(D) - Info_A(D). \quad (3)$$

Gain(*A*) tells us how much would be gained by branching on *A*. It is the expected reduction in the information requirement caused by knowing the value of *A*. The feature *A* with the higher information gain, (Gain(*A*)), is better ranked at the given training tuples.

3.4. Gain ratio

The information gain technique is biased toward tests with many outcomes. An extension to information gain is known as gain ratio, which attempts to overcome this bias. It applies a kind of normalization to information gain using a "split information" value defined analogously with Info(*D*) as

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right). \quad (4)$$

This value considers the number of tuples having that outcome with respect to the total number of tuples in D. The gain ratio is defined as

$$GainRatio(A) = \frac{Gain(A)}{SplitInfo(A)}. \quad (5)$$

The feature with the higher gain ratio is better ranked at the given training tuples.

3.5. Gini index

The Gini index measures the impurity of *D*, a data partition or set of training tuples, as

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2, \quad (6)$$

where p_i is the probability that a tuple in *D* belongs to class C_i and is estimated by $|C_{i,D}|/|D|$. The sum is computed over m classes. The Gini index considers a binary split for each feature. If a binary split on A partitions *D* into D_1 and D_2, the Gini index of *D* given that partitioning is

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2). \quad (7)$$

The reduction in impurity that would be incurred by feature A is

$$\Delta Gini(A) = Gini(D) - Gini_A(D). \quad (8)$$

The feature that maximizes the reduction in impurity (or, equivalently, has the minimum Gini index) is the best ranked at the given training tuples.

3.6. Neural networks

Neural network (NN) is information processing computing system that uses an enormous amount of simple linking artificial nerves to simulate the capability of biological neural network (Tsai et al., 2009). There are many different kinds of neural networks and neural network algorithms. The most representative and popular neural network algorithm is backpropagation. Multilayer feed-forward network is the type of neural network on which the backpropagation algorithm performs.

The essential features of the artificial neural network (ANN) are processing units (the neurons or nodes) and the learning algorithm used to find values of the ANNs parameters, called weights, for a particular problem. The neurons are connected to one another so that the output from one neuron can be the input to many other neurons. Each neuron transforms a multivariate input to a single output value using a predefined simple function. In our case is used the sigmoid function:

$$f(s) = \frac{1}{1 + e^{-s}}, \quad \text{where } \boldsymbol{s} \text{ is the input and } \boldsymbol{f} \text{ is the output.} \quad (9)$$

In most cases the form of this function is identical in all neurons, however each set of parameters (weights) in this function is different for each neuron. The values of the weights are determined by a training sub-set consisting of data with known inputs and outputs. Network architecture is the organization of neurons and the type of connections permitted. The neurons are arranged in a series of layers with connections between neurons in other layers, but not between neurons in the same layer. The layer receiving the inputs is called the input or the first layer. The final layer providing the target output signal or answer is the output layer. Any layers between these two layers are called hidden layers (Šušteršic et al., 2009).

Before training can begin, we must decide on the network topology by specifying the number of units in the input layer, the number of hidden layers (if more than one), the number of units in each hidden layer, and the number of units in the output layer. Normalizing the input values for each feature measured in the training tuples will help speed up the learning phase. Typically, input values are normalized so as to fall between 0.0 and 1.0. There are no clear rules as to the "best" number of hidden layer units. Network design is a trial-and-error process and may affect the accuracy of the resulting trained network. The initial values of the weights may also affect the resulting accuracy. Once a network has been trained and its accuracy is not considered acceptable, it is common to repeat the training process with a different network topology or a different set of initial weights or a different learning rate or momentum. Different validation techniques for accuracy estimation can be used to help decide when an acceptable network has been found. A number of automated techniques have been proposed that search for "good" parameters. In the study we propose NN Generic model for parameters optimization (NNGM) based on genetic algorithm.

4. Model development

From the highest point of view, credit risk assessment process consists of: (1) data preprocessing with attribute selection as separate section, (2) classification and evaluation and (3) comparison of the results, as is shown in Fig. 2.

4.1. Data preprocessing

After data collecting from credit institution the descriptive data summarization was made. Descriptive data summarization pro-

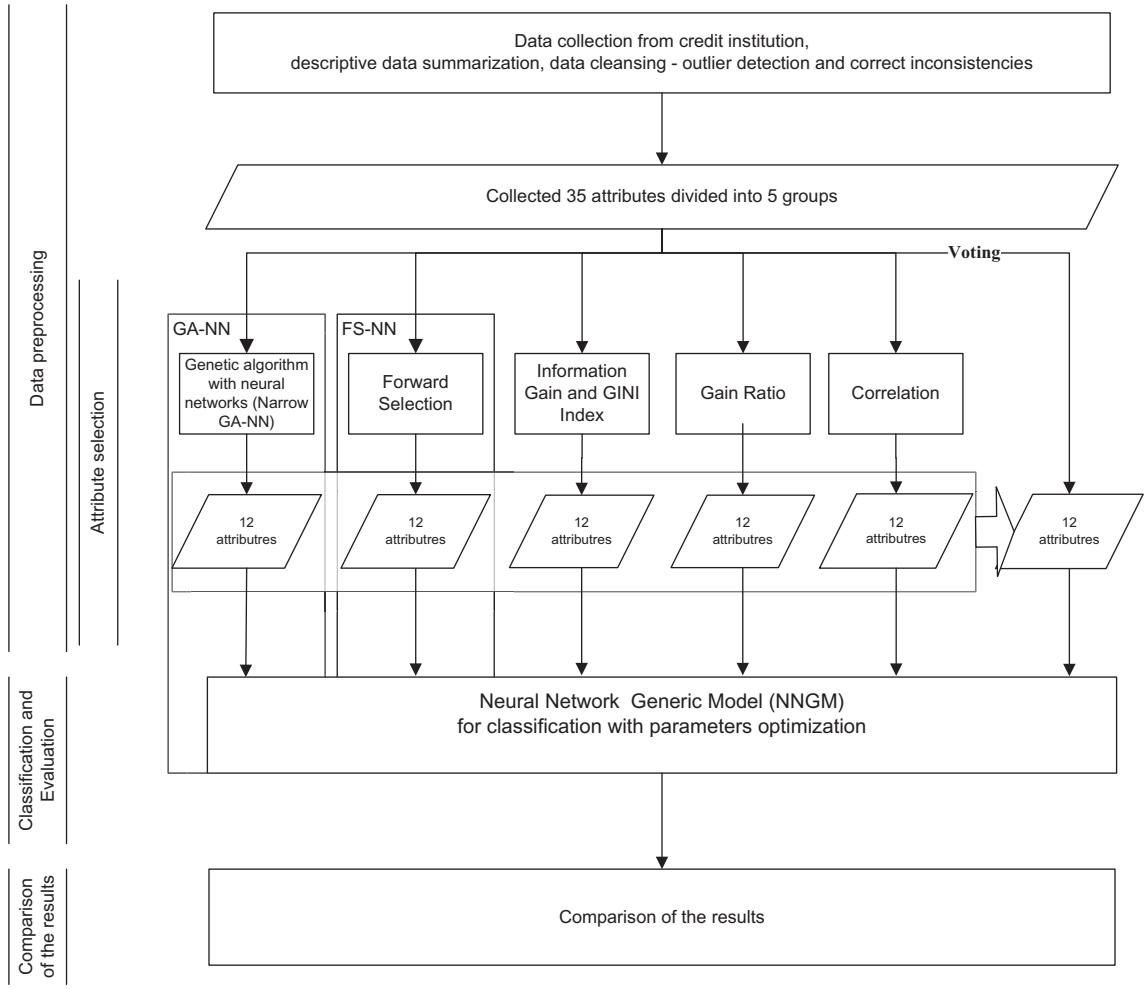

Fig. 2. The flowchart of credit risk assessment process.

vides the analytical foundation for data preprocessing. The basic statistical measures for data summarization including mean, standard deviation and range are shown in Appendix A as useful values for measuring the dispersion of data.

Incomplete, noisy, and inconsistent data are commonplace properties of large real world databases and data warehouses. There are many possible reasons for noisy data (having incorrect attribute values). Consequently, data cleaning (or data cleansing) routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data (Han & Kamber, 2006). Data cleaning is performed as an iterative process consisting of discrepancy detection and data transformation. In this step, we wrote our own scripts for finding outliers, and inconsistent values that need investigation. Values that are more than $1.5 \times \text{IQR}$ (interquartile range is defined as $\text{IQR} = Q3 - Q1$) above the third quartile or below the first quartile and inside of the 2% of all values were flagged as potential outliers. In the next step, these values were transformed to the specified bounds.

4.1.1. Feature (attribute) selection

It is possible that many of features may be irrelevant to the classification task, or redundant. The selection of features is performed in the data preprocessing phase to improve the efficiency of the classification system f as a whole, from the aspects of the accuracy, speed and scalability. The aim of it is to find a feature set obtainable from the original data that will enable an accurate classification to be performed. In performing this task, we used the following techniques: genetic algorithm, Forward selection, Information gain, Gain ratio, Gini index and Correlation, for which, the basics were described in the methodology section.

An optimal feature subset does not need to be unique because it may be possible to achieve the same accuracy using a different set of features (e.g. when two features are perfectly correlated, one can be replaced by the other). By definition, to get the highest possible accuracy, the best subset that a feature selection algorithm can select is an optimal feature subset.

Information gain, Gain ratio, Gini index and Correlation as the feature selection techniques provide a ranking for each feature describing the given training tuples. Their implementations are based on the equations described earlier in this paper. According to Kohavi and John (1997) these techniques belong to the filter approach to feature subset selection. The basic characteristic of the filter approach is that it does not take into account the biases of the classification algorithms; features are filtered independently of the classification algorithm. The filter approach is an attempt to assess merits of features from the data, ignoring the classification algorithm. A more sophisticated technique is created for the feature selection based on genetic algorithm and ANN as fitness function and belongs to wrapper approach. As it is shown in Fig. 2, and in more details in Fig. 3, combining GA with the NN classifier, we can simultaneously perform the feature selection task and classification. But for the purposes of comparison techniques, the GA-NN technique is embedded in the overall credit risk assessment process where it is tasked only with feature selection, what is

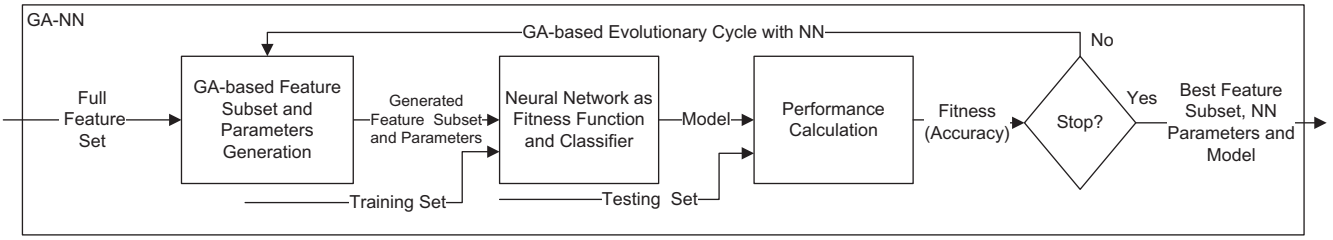

Fig. 3. The flowchart of the GA-NN technique.

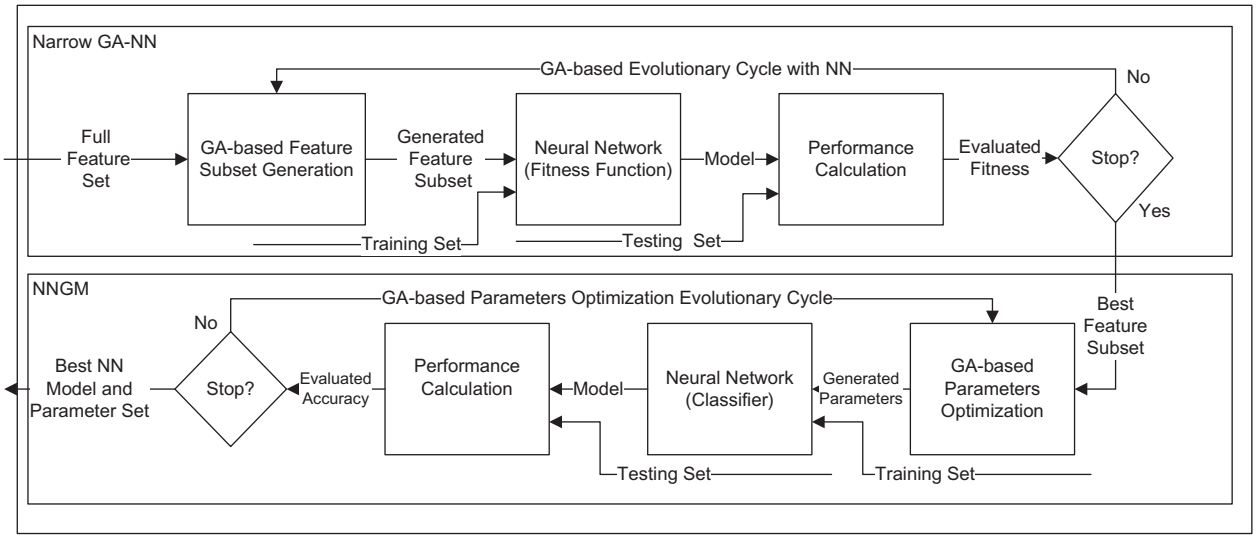

Fig. 4. GA-NN technique embedded in credit risk assessment process modified from Huang et al. (2007).

shown under the caption Narrow GA-NN in the flowchart diagram in Fig. 4. From this diagram we can see that the classification results of the GA-NN technique are not conclusive because in the further course of the process only selected attributes are taken. NNGM, which shares many common characteristics of the GA-NN technique, is responsible for optimizing the parameters of the final classifier (Fig. 4). As this paper attempts to give a general comparison of feature extraction efficiency among some commonly used techniques and the GA-NN, this way comparison is enabled. Besides, FS-NN (Forward selection with neural networks) technique was created for the selection of features that also belong to a group of wrapping techniques. The algorithm of FS-NN technique was described in methodology section.

Model shown in Fig. 3 was made using Rapid Miner 5.1.15 tool with the parameters shown in Table 1.

Table 1
Summary of the GA-NN parameters.

Parameter	Setting
Population initialization	
Population size	30
Initial probability for an feature to be switched on	0.7
Minimum number of features	10
Reproduction	
Fitness measure	Accuracy
Fitness function	Neural network
The type of neural network	Multilayer feed-forward network
Network algorithm	Backpropagation
Activation function	Sigmoid
The number of hidden layers	1
The size of the hidden layer	(Number of features + number of classes) / 2 + 1
Training cycles	[300; 600]
Learning rate	[0.3; 1.0]
Momentum	[0.2; 0.7]
Selection scheme	Tournament
Tournament size	0.25
Dynamic selection pressure	Yes
Keep best individual	Yes
Mutation probability	1/ number of features
Crossover probability	0.5
Crossover type	Shuffle
The condition for the completion	
Maximal fitness	Infinity
Maximum number of generations	6
Use early stopping	Yes
Generations without improvement	2

4.2. Classification and evaluation

We prove the efficiency of the variables selection with accuracy of the class prediction. Neural network was used as a classifier owing to many studies (Huang et al., 2007; Malhotra & Malhotra, 2003; Šušteršic et al., 2009; Zhang, Hu, Patuwo, & Indro, 1999) which conclude that neural networks are more accurate, adaptive, and robust in comparison to other classical statistical techniques. In addition, neural network models do not require multivariate normality assumption, outliers' elimination, linear dependency with class variable, discrete value characteristics, equal group covariance assumptions and other assumptions which are required by some other methods.

The performance of the artificial neural network is certainly dependent on the network topology and parameters, therefore, trial-and-error is usually proposed as the best guide in most cases (Malhotra & Malhotra, 2003). This approach is very time consuming, and, after a many trials, we propose our approach. According

to Li et al. (2006), Cybenko; Hornik, Stinchcombe, and White showed that the one-hidden-layer network is sufficient to simulate complicated systems with desired accuracy. Therefore, the one-hidden-layer structure will be applied. The number of hidden neurons was determined through the following equation (number of features + number of classes)/2 + 1.Even though, in most cases, trial and error is the best guide for parameters optimization, genetic algorithm can be applied to the problem of parameterization of the artificial neural network. This way we get a generic model for parameters optimization of the artificial neural network (NNGM). This generic model (Fig. 4) is used in estimating the effectiveness of the feature selection algorithm with parameters shown in the Table 2.

As it is shown in the Table 2 in NNGM model, based on genetic algorithm, accuracy is a fitness measure. Accuracy of the class prediction is calculated using the 10-fold cross-validation technique. This technique estimates the performance of the model and tests the effect of sampling variation on the model performance. If loan applicants are selected randomly one can lose important information about them due to the fact that the percentage of bad loans is usually small compared to the performing ones. For a smaller population the random method by the rule (Šušteršic et al., 2009) does not produce a distribution as good as for the larger group and, therefore, it is inferior. For this reason subsets of data are created by k-fold cross-validation technique which uses stratified sampling. Stratified sampling builds random subsets and ensures that the class distribution in the subsets is (almost) the same as in the whole example set.

k-Fold cross-validation technique is better than a simple validation technique, because a simple validation technique divides a data set into a training sample and a holdout sample that tests the predictive effectiveness of the fitted model. As the best model is tailored to fit only one sub-sample, a holdout sample, the model often estimates the true error rate overoptimistically (Malhotra & Malhotra, 2003). In the *k*-fold cross-validation procedure, the credit dataset was divided into *k* independent groups. A model was trained using the first $k-1$ groups of samples and the trained model was tested using the ***k***th group. This procedure was repeated until each of the groups has been used as a testing subset once. The overall scoring accuracy was reported as an average across all *k* groups. A merit of cross-validation is that the credit scorning model is developed with a large proportion of the available data and that all the data is used to test the resulting models.

Table 2
Summary of the NNGM parameters.

Parameter	Setting
Population initialization	
Population size	30
Reproduction	
Fitness measure	Accuracy
Fitness function	Neural network
The type of neural network	Multilayer feed-forward network
Network algorithm	Back-propagation
Activation function	Sigmoid
The number of hidden layers	1
The size of the hidden layer	(Number of features + number of classes) / 2 + 1
Training cycles	[300; 1000]
Learning rate	[0.3; 1.0]
Momentum	[0.2; 0.7]
Selection scheme	Tournament
Tournament size	0.25
Keep best individual	Yes
Mutation type	Gaussian mutation
Crossover probability	0.9
The condition for the completion	
Maximal fitness	Infinity
Maximum number of generations	10
Use early stopping	Yes
Generations without improvement	2

4.3. Comparison of the results

The results of the classification and validation are shown in the confusion matrix (CM), which is a useful tool for analyzing how well a classifier can recognize tuples of different classes (Han & Kamber, 2006). A confusion matrix for two classes is shown in Table 3. Given *m* classes, a confusion matrix is a table of at least size *m* by *m*. An entry, $CM_{i,j}$ in the first *m* rows and *m* columns indicates the number of tuples of class *i* that were labeled by the classifier as class *j*. A classifier has good accuracy if most of the tuples are represented along the diagonal of the confusion matrix, from entry $CM_{1,1}$ to entry $CM_{m,m}$, with the rest of the entries being close to zero. The table may has additional rows or columns to provide totals, precision or recognition rates per class.

The number of correctly classified objects, from the whole sample, divided by the total number of objects from the whole sample gives the accuracy of the model. Comparison of feature extraction efficiency among used algorithms and the GA-NN will be performed with pairwise *t*-tests. With these tests we determine if differences of all the estimated mean values of accuracy are considered as significant. In addition, we will compute the total relative cost of misclassification (RC) according to Swicegood and Clark (Sarlija, Bensic, & Zekic-Susac, 2009):

$$RC = \alpha(P_I C_I) + (1-\alpha)(P_{II} C_{II}), \tag{10}$$

where α is the probability of being a 'bad' client, P_I is the probability of a type I error, C_I is the relative cost of the type I error, P_{II} is the probability of the type II error, and C_{II} is the relative cost of the type II error. The RC of each model is computed for seven scenarios, while the best model for each scenario is the model with the lowest RC value.

5. Empirical analysis

Credit scoring tasks can be divided into two distinct types. The first type is application scoring, where the task is to classify credit applicants into "good" and "bad" risk groups. In this paper, we will focus on this type of task. The second type of task deals with the existing customers after the loans are made and is called behavioral scoring (Khashman, 2010). In consumer application credit scoring, characteristics usually used in different models include time at present address, home status, postcode, telephone, applicant's annual income, credit card ownership, type of bank account, age, type of occupation, purpose of the loan, marital status, time with the bank, time with the employer, credit bureau rating, monthly debt as a proportion of monthly income, time in the current job, number of dependents (Sarlija et al., 2009). According to Khashman (2010) the data used for modeling generally consists of financial information and demographic information about the loan applicant. In contrast, behavioral scoring tasks deals with the existing customers and along with other information, payment history information is also used.

5.1. Real world credit data set

The credit dataset for this study was collected at a Croatian bank covering the period from September 2004 to September 2011. In the sampling process data were collected on current and savings accounts of 32,000 potential applicants. From potential

Table 3
Confusion matrix.

		Predicted result		Recognition rate (%)
		Default	Non-default	
Real	Default	True positives	False negatives (Type I error)	Sensitivity
	Non-default	False positives (Type II error)	True negatives	Specificity
	Overall			Accuracy

applicants in further consideration we selected only those cases which took credit in an amount less than or equal 100,000 HRK, and who had a current account with the bank for at least 15 months on the date of loan approval. A period of 15 months is a performance period and the characteristics of the performance in this period are used in developing scoring models. From the set of candidates, 1000 cases were randomly selected, including the 750 who successfully fulfilled their credit obligations, i.e. good credit customers, and 250 who were late in performing their obligations and therefore are placed in a group of bad credit customers. The client is "bad" if they defaulted for 90 days or more than, at any time in the life of the loan, which is in accordance with the Basel New Accord definition. The Accord says that someone has defaulted if they are 90, or in some countries 180 days overdue or deemed by the lender to be unlikely to pay (De Andrade & Thomas, 2007).

Although the credit scoring of loan applications is modeled, application data is combined with current accounts' behavior data in order to accurately evaluate loan applications and enable the scoring without the presence of the client. This is the reason why a client is required to have had a current account with the bank for at least 15 months on the date of loan the approval. In addition to application data, as it has been mentioned, behavior data of a client such as payment history, financial conditions, delinquency history and past credit history is taken into account.

The characteristics of each client initially were described by 37 variables. They referred to client's gender and age, credit purpose, credit amount, number of existing credits at this bank, credit history with the bank before the loan was granted, installment rate in percentage of disposable income and detailed data on accounts balances and transactions with the bank. After reducing the initial set of variables due to the fact that some variables had identical value in all cases or extremely high correlation, the final number of variables used in the research was 33 regular features and 2 (id,label) special features.

The variables were divided into five main groups: (i) basic characteristics; (ii) payment history (monthly averages); (iii) financial conditions; (iv) delinquency history; and (v) past credit experiences. List of variables with their explanation and descriptive statistics for the development data sample is given in Appendix A.

5.2. Experimental results

Seven feature selection techniques were tested: genetic algorithm with neural networks (GA-NN), Forward selection with neural networks (FS-NN), Information gain, Gain ratio, Gini index, Correlation and Voting (the overlapped features). Since Information gain algorithm and the Gini index gave the same results, the results of Information gain are not given in table with the results. The top 12 features selected by using the aforementioned techniques are shown in Appendix B. The top 12 features were selected because the estimated accuracy began to fall after reducing the number of features below 12. Once the features for every technique have been selected, the most significant features for the Voting technique can be chosen. Features that appear in more than half of the other techniques for feature selection are relevant for the purposes of this technique. It can be seen from Appendix B that the significant features for all techniques were: RIDI, BCO and OINT. The feature significant for all techniques except one was TWB, and the features significant for 3 of 5 techniques were: AGE, LMM, TOUT, TITO, RII, BAL, INPO and CHD. These 12 features are the most significant features in other feature selection techniques and they represent the features of Voting technique. In order to estimate efficiency of the mentioned features selection techniques for each technique the classification was made using NNGM.

The number of correctly classified objects from the whole sample depends also on the chosen *cutoff* value. If the targets are described with two values – zero and one and the model turns the values between zero and one, then the accuracy should be highest at *cutoff* value 0.5 (Šušteršic et al., 2009). But at this *cutoff* value the misclassification cost of the model is not necessarily optimal. This depends on the bank's cost of granting a nonperforming loan (type I error) relative to the opportunity cost of not granting a performing loan (type II error). For this reason, two estimations for each technique are shown in this study. First, with the cutoff value (the threshold for the decision of granting or rejecting the loan) near to 0.5 which turned out, in terms of costs, to be better than 0.5, as the type I error is reduced considerably relative to increase of type II error. At the same time, the average accuracy in the prediction of the model with the increased threshold practically did not change. We made a second estimation for each selected feature set, with such cutoff value where the aim was to reduce the type I error to the value less than 25%, and these were shown in the tables with results as the second estimation for a selected feature set. As noted earlier, in some circumstances the errors of the model can be optimal at lower accuracy instead of at the highest one. It depends on the ratio between type I error and the type II error, which still depends on economic cycles, circumstances and preferences of the bank. The results are tested on the whole data sample using the described 10-fold cross-validation.

To avoid a multitude of tables, we show the CM only for results of two extreme feature selection techniques. As it is shown in Table 4, the overall average accuracy rate of classification using features selected by GA-NN with cutoff value of 0.60 is 82.30%, which is much better than the overall average accuracy rate of the classification using the same feature selection with cutoff value 0.77, which is 73.90%. The deterioration of accuracy is expected, given that there was a goal of raising the cutoff value to reduce the type I error, which happened. Type I error is reduced from 44.00% to 23.20%. The deterioration of accuracy occurs due to the

Table 4
The predicted results of NNGM using features selected by the GA-NN.

		Predicted result		Recognition rate (%)
		Default	Non-default	
Cutoff value (0.60)				
Real	Default	140	110	56.00
	Non-default	67	683	91.07
Accuracy rate (%)				82.30
Cutoff value (0.77)				
Real	Default	192	58	76.80
	Non-default	203	547	72.93
Accuracy rate (%)				73.90

fact that at the same time the type II error increased significantly from 8.93% to 27.07%. Is the price of reduction of type I error is too high? The answer to this question could be obtained by calculating the cost of misclassification. In any case it can be seen from Table 4 that the improvement of 52 cases of false negatives to true positives is paid with a worsening of the 136 cases, from true negatives to false positives. Interesting results were obtained using the Gain ratio. Compared to other techniques, while the maximum accuracy is relatively poor, only 79.20%, the technique gives very good results of 73.60% under the condition of the type I error less than 25%. This is due to the fact that the improvement of 44 cases of false negatives to true positives is paid with a worsening of a relatively smaller number of 100 cases, from true negatives to false positives, as it is shown in Table 5. Similar changes have occurred with the other techniques. The consolidated results of all the techniques are presented in Tables 6 and 7.

5.3. Comparison of the results

In our research different feature selection techniques are used, but the classification and validation method is kept the same for all feature selection techniques, to support a simple direct comparison of results. In order to estimate the efficiency of feature selection techniques among used techniques, we compare: (1) technique accuracy, as previously described, for the two cutoff values and (2) classification cost in order to find the model which reduces the cost for the bank the most.

Table 5
The predicted results of NNGM using features selected by the Gain ratio.

		Predicted result		Recognition rate (%)
		Default	Non-default	
Cutoff value (0.59)				
Real	Default	147	103	58.80
	Non-default	105	645	86.00
Accuracy rate (%)				79.20
Cutoff value (0.76)				
Real	Default	191	59	76.40
	Non-default	205	545	72.67
Accuracy rate (%)				73.60

Table 6
Comparison of the results of all techniques with maximum accuracy.

Selection technique	Recognition rate (%)		Average accuracy (%)	Std. dev. (%)
	Default	Non-default		
GA-NN	56.00	91.07	82.30	1.85
Gain ratio	58.80	86.00	79.20	3.94
Gini index	59.20	84.67	78.30	3.20
Correlation	59.20	87.07	80.10	2.55
FS-NN	56.80	88.00	80.20	4.83
Voting	60.80	86.67	80.20	2.23

Table 7
Comparison of the results of all techniques with cutoff value maximum accuracy fitted to Type I error < 25%.

Selection technique	Recognition rate (%)		Average accuracy (%)	Std. dev. (%)
	Default	Non-default		
GA-NN	76.80	72.93	73.90	3.83
Gain ratio	76.40	72.67	73.60	4.03
Gini index	76.00	70.53	71.90	4.50
Correlation	75.60	72.00	72.90	5.45
FS-NN	76.40	71.20	72.50	5.97
Voting	76.00	75.20	75.40	4.92

For the purposes of comparing technique accuracy, the prediction with the maximum overall average accuracy rate is the most accurate one. Table 6 shows that the GA-NN technique produced the most accurate prediction with the overall average accuracy rate of 82.30%, with a standard deviation of 1.85%. Based on pairwise *t*-test shown in Table 8, on average, the overall accuracy of the GA-NN technique is better than the overall average accuracy of all the other techniques except FS-NN, and the difference is statistically significant at 95% confidence level in favor of the GA-NN technique. In case of FS-NN the difference is not statistically significant.

It can be seen from Tables 6 and 8 that the GA-NN technique achieves the best results in terms of the overall average accuracy rate. Moving the cutoff point to where a Type I error < 0.25 (Table 7), we do not get such good results of the GA-NN technique compared to the other techniques. This is quite expected since the optimization in feature selection using the GA-NN technique is conducted in relation to the accuracy rate, with no additional conditions. It is a classic example in which the best overall selected feature set in one type of conditions does not outperform other feature sets under other conditions, in our example the cutoff points. If the execution of a process can find the best set of features for certain conditions, it is justified to assume that for changed conditions we would have to repeat the process of selection with changed conditions in order to get the best set of features. It is because the features, chosen as the optimal combination for specific requirements, do not necessarily meet the optimality condition for some other conditions. If that is true, and has been proven in our case, then the wrapper techniques have an advantage because they give a different combination of features for different target conditions, which is not the case with filter techniques because they provide a ranking of features regardless of the classification methods or additional terms of classification. In order for the wrapper technique to exploit its potential advantages over the filter technique, wrapper technique entails the obligation to re-find the optimal combination of the attributes for each change of the target function. This represents an additional effort and cost which will be redeemed by the lower costs of misclassification.

When we look at the results only in terms of the GA-NN technique and our example, for each comparison in which there are changed conditions, and in which we want to get the best performance from the selected features it is necessary to carry out a new optimization process using GA-NN technique. Potentially for any new conditions we can possibly get another set of features that best fit the requirements of optimization. As in this example we conducted the optimization using the GA-NN technique only from the standpoint of maximum accuracy it is logical that variables obtained by the process give the best results just from the standpoint of maximum accuracy, and not some other conditions. Overall, while GA-NN technique gives the best results for the objective function for which it is optimized, Voting (ensemble) technique has shown the best stability what is in accordance with the findings of Schowe (2011). The limitation of GA-NN as a feature selection technique is long runtime; GA-NN is a computationally intensive technique.

Table 8
Pairwise t-Test between GA-NN all other techniques.

Selection technique	Results Avg. +/- Std.	p-value[a]
GA-NN	0.823 +/- 0.018	–
Gain ratio	0.792 +/- 0.039	0.042
Gini index	0.783 +/- 0.032	0.003
Correlation	0.801 +/- 0.025	0.046
FS-NN	0.802 +/- 0.048	0.288
Voting	0.802 +/- 0.022	0.038

[a] alpha = 0.050.

Table 9
Comparison of costs.

Selection technique	Cost ratio (C_I:C_{II})						
	1:1	2:1	3:1	4:1	5:1	8:1	10:1
Results with maximum accuracy							
GA-NN[1]	0.1770[a]	0.2870[a]	0.3970	0.5070	0.6170	0.9470	1.1670
Gain ratio	0.2080	0.3110	0.4140	0.5170	0.6200	0.9290	1.1350
Gini index	0.2170	0.3190	0.4210	0.5230	0.6250	0.9310	1.1350
Correlation	0.1990	0.3010	0.4030	0.5050	0.6070	0.9130	1.1170
FS-NN	0.1980	0.3060	0.4140	0.5220	0.6300	0.9540	1.1700
Voting[1]	0.1980	0.2960	0.3940	0.4920	0.5900	0.8840	1.0800
Results with cutoff value fitted to Type I error < 25%							
GA-NN[2]	0.2610	0.3190	0.3770	0.4350	0.4930	0.6670	0.7830[a]
Gain ratio	0.2640	0.3230	0.3820	0.4410	0.5000	0.6770	0.7950
Gini index	0.2810	0.3410	0.4010	0.4610	0.5210	0.7010	0.8210
Correlation	0.2710	0.3320	0.3930	0.4540	0.5150	0.6980	0.8200
FS-NN	0.2750	0.3340	0.3930	0.4520	0.5110	0.6880	0.8060
Voting[2]	0.2460	0.3060	0.3660[a]	0.4260[a]	0.4860[a]	0.6660[a]	0.7860

[a] The best model for each ratio.

Table A.1
Input variables with descriptive statistics.

Attribute	Type	Code	Description	Statistics	Range
att1	Integer	ID	Row Id	avg = 500.500 + / − 288.819	[1; 1000]
Group1		G1	Basic characteristics		
att2	Integer	AGE	*Age*	avg = 46.198 + / − 14.097	[20; 80]
att3	Integer	GENDER	*Gender*	avg = 0.506 + / − 0.500	[0; 1]
att4	Integer	POST	*Postcode*	avg = 0.581 + / − 0.494	[0; 1]
att5	Integer	TLF	Telephone	avg = 0.906 + / − 0.292	[0; 1]
att6	Integer	TAPA	Time at present address	avg = 2.722 + / − 1.594	[0; 6]
att7	Integer	TWB	*Time with the bank*	avg = 14.039 + / − 8.347	[1; 50]
att8	Integer	ACAGE	*Age of the clients current account*	avg = 9.825 + / − 7.566	[1; 34]
att9	Integer	MOA	*Month of loan approval date*	avg = 6.400 + / − 3.265	[1; 12]
att10	Integer	LMM	*Loan maturity in months (repayment period)*	avg = 39.378 + / − 19.562	[11; 61]
att11	Integer	POL	*Purpose of loan*	avg = 0.874 + / − 0.508	[0; 5]
att12	Integer	CRAM	*Loan amount (in HRK)*	avg = 25057.85 + / − 18075.76	[2000; 100000]
Group2		G2	Payment history (monthly average)		
att13	Integer	TIN	*Total payments to the account (monthly income)*	avg = 4373.09 + / − 3351.98	[1; 32522]
att14	Real	CSHT	*Cash payments to the account/ Total payments*	avg = 0.071 + / − 0.158	[0; 1]
att15	Real	RPT	*Regular payments (salaries)/ Total payments*	avg = 0.884 + / − 0.206	[0; 1]
att16	Real	OTIN	*Contracted overdraft/ Total payments*	avg = 1.754 + / − 1.272	[0; 5]
att17	Integer	TOUT	*Total withdrawals from the account (outcome)*	avg = 4665.159 + / − 4967.562	[1; 117664]
att18	Real	TITO	*Total payments/ Total withdrawals*	avg = 0.991 + / − 0.238	[0.010; 3]
att19	Real	PSTW	*EFTPOS withdrawals/ Total withdrawals*	avg = 0.164 + / − 0.170	[0; 0.950]
att20	Real	TMTW	*ATMs withdrawals/ Total withdrawals*	avg = 0.398 + / − 0.303	[0; 1]
att21	Real	SSTW	*Self-service withdrawals/ Total withdrawals*	avg = 0.563 + / − 0.322	[0; 1]
att22	Real	COTW	*Contracted overdraft/ Total withdrawals*	avg = 1.680 + / − 1.235	[0; 5]
att23	Real	RIDI	*The ratio of installment/ to disposable income*	avg = 0.296 + / − 0.323	[0.030; 2]
att24	Real	RII	*The ratio of the income on the loan approval date / to the income year before*	avg = 1.388 + / − 1.000	[0.010; 5]
Group3		G3	Financial conditions		
att25	Integer	COD	*Contracted overdraft*	avg = 7921 + / − 7283.530	[0; 29000]
att26	Integer	TDB	*Time deposits (balance on the loan approval date)*	avg = 1248.602 + / − 7892.941	[0; 107590]
att27	Integer	BAL	*Balance of all accounts in the bank*	avg = − 5353.451 + / − 7085.791	[-30632; 31338]
att28	Real	BCO	*Balance/ Contracted overdraft*	avg = − 0.448 + / − 1.473	[-9; 9]
Group4		G4	Delinquency history		
att29	Integer	TECO	*Number of times client exceeded contracted overdraft*	avg = 2.683 + / − 3.237	[0; 12]
att30	Real	INPO	*Interest on positive balance/ Interest on overdraft*	avg = 2.157 + / − 3.092	[0; 10]
att31	Integer	OINT	*Interest on overdraft*	avg = 7.514 + / − 25.154	[0; 369]
Group5		G5	Credit history		
att32	Integer	CHG	Number of *loans with amount greater than current*	avg = 0.212 + / − 0.491	[0; 3]
att33	Integer	CHLE	Number of *loans with amount less than or equal to current*	avg = 0.504 + / − 0.750	[0; 4]
att34	Integer	CHD	Credit history – client defaulted	avg = 0.031 + / − 0.173	[0; 1]
att35	Binominal	IR	Internal rating – label (criterion variable)	mode = 1 (750), least = 0 (250)	0 (250), 1 (750)

In accordance with the Eq. (10), the total relative cost of misclassification (RC) of each model for seven scenarios is computed below, while the best model for each scenario is the model with the lowest RC value. It can be seen from Table 9 that the most accurate prediction will be the most appropriate one for the bank in case the cost of predicting a bad client as a good one (type I error) is equal to the cost of predicting a good client as a bad one (type II error) and when the cost ratio (C_I:C_{II}) is 2:1. As it can be seen, for other scenarios the most accurate prediction (GA-NN[1]) does not give the lowest RC value. Other scenarios are likely for the bank. The voting[2] selection technique gives the lowest RC values for 3.1, 4:1, 5:1 and 8:1 cost ratios (type I error/type II error) and the GA-NN[2] technique for 10:1 ratio. It is justified to expect that banks will probably want to optimize the cost function for some ratios and not the accuracy function. For the GA-NN technique this is only a modification in the fitness function. That was not done in this, because the best ratio is determined by each bank for itself.

6. Conclusions and future works

From the total data set which the bank has, the features that are useful in the classification of clients have to be chosen, and the ones that are redundant and those which enter the noise into the system have to be omitted. In theory, even if better classification accuracy is not achieved, there are many potential benefits of variable and feature selection: facilitating data visualization, data understanding, and reducing the dimension.

To select the features several standard selection techniques were used. Beside them, our own technique for the selection of features has been created, that is the hybrid of GA and NN. This features selection technique selects the features so that already perform the classification in the process of feature selection. One set of features that gave the best accuracy of classification is chosen as the optimal set of features. For the purposes of unambiguous comparison with other feature selection techniques, further procedure was performed on a selected set of features. Classification results, which are achieved by the GA-NN, are not taken as final for the underlying set of features for the purposes of further proceedings. The final classification results needed for comparison with other feature selection techniques was achieved by putting the selected features to the input of NNGM, as well as for other feature selection techniques. The classification results for all feature selection techniques were reached in the same way and their uniform comparability can thus be ensured.

From the experimental results we have concluded that our GA-NN model is significantly better in feature selection for classification compared to some other techniques used for selecting features. This proves the hypothesis H2. The same results show that based on the total data that the bank has on its clients it could classify clients in terms of their loans riskiness with maximum accuracy above 80% and as precisely as we find in the literature on this subject (Crook, Edelman, & Lyn, 2007; Šušteršic et al., 2009; Zekić-Sušac, Šarlija, & Benšić, 2004; Zekic-Sušac, Benšić, & Šarlija, 2005). This proves the H1 hypothesis.

Everything mentioned gives the bank ability to create such products, which are simultaneously in accordance with regulation, on the one hand, but even more so competitive on the other. Competition forces the bank management to seek new solutions for their business, which will have, at the same time, more flexibility and sensitivity to risk. Therefore, this paper can be observed as one step in looking for the ability to assess creditworthiness without the physical presence of the client.

It is reasonable to expect that the results that were obtained by applying neural networks in the process of classification can be further improved by using other artificial intelligence methods of classification, especially by the Support Vector Machine method, or maybe ensemble of methods. Therefore, future work should continue to compare different methods of classification on this data set, as well as on an expanded data set with the data of credit bureaus.

Table B.1
Selected features.

Attribute code	Feature selection technique					
	GA-NN	FS-NN	GINI	Gain Ratio	Correlation	Voting
G1						
AGE			√		√	2
GENDER						0
POST						0
TLF	√	√				2
TAPA					√	1
TWB	√	√	√		√	4
ACAGE		√		√	√	3
MOA		√				1
LMM	√			√	√	3
POL						0
CRAM			√			1
G2						
TIN			√	√		2
CSHT						0
RPT			√	√		2
OTIN	√					1
TOUT	√		√	√		3
TITO	√	√		√		3
PSTW	√					1
TMTW						0
SSTW						0
COTW		√				1
RIDI	√	√	√	√	√	5
RII	√		√	√		3
G3						
COD						0
TDB		√				1
BAL			√	√	√	3
BCO	√	√	√	√	√	5
G4						
TECO		√			√	2
INPO			√	√	√	3
OINT	√	√	√	√	√	5
G5						
CHG						0
CHLE						0
CHD	√	√			√	3

Appendix A

See Table A.1.

Appendix B

See Table B.1.

References

BIS (2006). International convergence of capital measurement and capital standards: A revised framework. Basel Committee of Banking Supervision, Bank for International Settlements, Basel.

Crook, J. N., Edelman, D. B., & Lyn, C. (2007). Recent developments in consumer credit risk assessment. *European Journal of Operational Research, 183*, 1447–1465.

De Andrade, F. W. M., & Thomas, L. C. (2007). Structural models in consumer credit. *European Journal of Operational Research, 183*, 1569–1581.

Finlay, S. (2010). Credit scoring for profitability objectives. *European Journal of Operational Research, 202*, 528–537.

Finlay, S. (2011). Multiple classifier architectures and their application to credit risk assessment. *European Journal of Operational Research, 210*(2), 368–378.

Golub, M. (2001). Poboljšavanje djelotvornosti paralelnih genetskih algoritama. PhD thesis. Zagreb: University of Zagreb.

Han, J., & Kamber, M. (2006). *Mining: Concepts and techniques* (2nd ed.). CA: Morgan Kaufmann Publishers.

<http://www.bis.org/bcbs/basel3.htm>. Last accessed: 21.11.11.

Huang, C.-L., Chen, M.-C., & Wang, C.-J. (2007). Credit scoring with a data mining approach based on support vector machines. *Expert Systems with Applications, 33*, 847–856.

Khashman, A. (2010). Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes. *Expert Systems with Applications, 37*, 6233–6239.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence, 97*, 273–324.

Li, S.-T., Shiue, W., & Huang, M.-H. (2006). The evaluation of consumer loans using support vector machines. *Expert Systems with Applications, 30*, 772–782.

Malhotra, R., & Malhotra, D. K. (2003). Evaluating consumer loans using neural networks. *Omega, 31*(2), 83–96.

Sarlija, N., Bensic, M., & Zekic-Susac, M. (2009). Comparison procedure of predicting the time to default in behavioural scoring. *Expert Systems with Applications, 36*, 8778–8788.

Schowe, B. (2011). *Selection for high-dimensional data with RapidMiner*. Dortmund: Technical University of Dortmund.

Šušteršic, M., Mramor, D., & Zupan, J. (2009). Consumer credit scoring models with limited data. *Expert Systems with Applications, 36*, 4736–4744.

Tsai, M.-C., Lin, S.-P., Cheng, C.-C., & Lin, Y.-P. (2009). The consumer loan default predicting model – An application of DEA–DA and neural network. *Expert Systems with Applications, 36*, 11682–11690.

Twala, B. (2010). Multiple classifier application to credit risk assessment. *Expert Systems with Applications, 37*, 3326–3336.

Zekic-Sušac, M., Benšić, M., & Šarlija, N. (2005). Selecting neural network architecture for investment profitability predictions. *Journal of Information and Organizational Sciences, 29*, 83–95.

Zekić-Sušac, M., Šarlija, N., & Benšić, M. (2004). Small business credit scoring: a comparison of logistic regression, neural network and decision tree models. In *Proceedings of the 26th international conference on information technology interfaces, June 7–10* (pp. 265–270). Croatia: Cavtat/Dubrovnik.

Zhang, G., Hu, M. Y., Patuwo, B. E., & Indro, D. C. (1999). Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis. *European Journal of Operations Research, 116*, 16–32.