Applied Soft Computing Journal 84 (2019) 105740

Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

Application of new deep genetic cascade ensemble of SVM classifiers to predict the Australian credit scoring

Paweł Pławiak [a,*], Moloud Abdar [b], U. Rajendra Acharya [c]

[a] Institute of Telecomputing, Faculty of Physics, Mathematics and Computer Science Cracow University of Technology, Warszawska 24 st., F-5, 31-155 Krakow, Poland
[b] Department of Computer Science, University of Quebec in Montreal, Montreal (QC), H2X 3Y7, Canada
[c] Department of Electronics and Computer Engineering, Ngee Ann Polytechnic, Singapore

ARTICLE INFO

Article history:
Received 5 December 2018
Received in revised form 12 July 2019
Accepted 22 August 2019
Available online 29 August 2019

Keywords:
Credit scoring
Credit risk
Data mining
Machine learning
Ensemble learning
Deep learning
Genetic algorithms
Feature selection and extraction

ABSTRACT

In the recent decades, credit scoring has become a very important analytical resource for researchers and financial institutions around the world. It helps to boost both profitability and risk control since bank credits plays a significant role in the banking industry.

In this study, a novel approach based on deep genetic cascade ensemble of different support vector machine (SVM) classifiers (called Deep Genetic Cascade Ensembles of Classifiers (DGCEC)) is applied to the Statlog Australian data. The proposed approach is a hybrid model which merges the benefits of: (a) evolutionary computation, (b) ensemble learning, and (c) deep learning. The proposed approach comprises of a novel 16-layer genetic cascade ensemble of classifiers, having: two types of SVM classifiers, normalization techniques, feature extraction methods, three types of kernel functions, parameter optimizations, and stratified 10-fold cross-validation method. The general architecture of the proposed approach consists of ensemble learning, deep learning, layered learning, supervised training, feature (attributes) selection using genetic algorithm, optimization of parameters for all classifiers by using genetic algorithm, and a new genetic layered training technique (for selection of classifiers).

Our developed model achieved the highest prediction accuracy of 97.39%. Hence, our proposed approach can be employed in the banking system to evaluate the bank credits of the applicants and aid the bank managers in making correct decisions.

1. Introduction

1.1. Credit scoring

Credit scoring is a method of assessing the credibility of the entity (person or company) applying for a bank credit. The result of a credit score is usually presented in a point form — usually the more points, the greater the creditworthiness of the potential borrower. Higher assessment scores will receive the borrower whose profile will be as close as possible to the profile of borrowers who timely repay their credits in the past. The evaluation shall take into account attributes such as: (1) profession, (2) education, (3) housing status, (3) district of residence, (4) monthly income, (5) age and marital status, (6) number of dependents, (7) bank accounts, (8) life insurance, (9) period of employment at the current position, etc. Bank on the basis of the above-mentioned attributes, establishes a cut-off threshold that, decides to grant or refuse the credit.

Tasks of credit scoring can be divided into two distinct types [1–3]: (1) Application scoring — the first type, whose task is to classify credit applicants to two risk groups: "good" and "bad". In this case, the data used for modeling usually consist of demographic and financial information about the credit applicant; (2) Behavioral scoring — in addition to financial and demographic information, data on customer payment history are also available for the current customers [4]. In this paper, we execute the first task — application scoring.

Analysis of credit risk is an important subject in the field of financial risk management and has recently become the primary objective of the banking and financial sector [5]. Nowadays, it is very common to apply the credit risk analysis for bank credits either private or business purposes. From the perspective of the banking sector, the proper assessment of applicants for bank credits plays a major role. Even small mistakes (1%) in credit score accuracy may prove to be very painful for the banks and may cause huge losses due to the insolvency of the borrowers. On the other hand, excessive criteria may deter potential customers and, consequently may reduce the turnover of banks. This problem intensified particularly after the financial crisis in 2007. As a

* Corresponding author.
E-mail addresses: plawiak@pk.edu.pl (P. Pławiak), m.abdar1987@gmail.com (M. Abdar), aru@np.edu.sg (U. Rajendra Acharya).

https://doi.org/10.1016/j.asoc.2019.105740
1568-4946/

result, banks started looking for new solutions to minimize their risks and maximize their profits. This led to development of research in credit scoring, and emergence of many new tools to assist financial experts in assessing the borrowers.

The following attention need to be paid to the following key issues while assessing the creditworthiness:

- Feature (attributes) extraction [4,6,7] and feature (attributes) selection [8–10].
- Selection of appropriate classifiers or their combination [11,12].
- Optimization of classifiers parameters [13].
- Training of classifiers using appropriate cross-validation (CV) technique to obtain and reliable results [14].

The problem of credit scoring is difficult to solve in a classic manner, due to the uncertain nature of analyzed data (financial and demographic information). The analyzed data does not determine unambiguously the credibility of the borrowers. Usually scoring is calculated by using statistical methods (e.g. discriminant analysis, linear regression) [7,15], but however, these methods have many drawbacks like assumption of linear relationship between the variables, [7,16]. Therefore, we can also find other methods in the scientific literature to calculate credit scoring such as machine learning algorithms which are popular in recent years [17–19]. Methods of data mining, in particular pattern recognition, based on real historical data, are of great importance in building the predictive models [4]. To obtain accurate results using machine learning algorithms we can include: fuzzy systems (FS) [20], neural networks (NN) [21–23], support vector machine (SVM) [13], k-nearest neighbors algorithm (kNN) [24], genetic programming (GP) [25], decision trees (DT) [26,27], and genetic algorithms (GA) [8]. In scientific literature, for most of the scoring problems, machine learning algorithms have performed better than statistical methods [7,13,28]. However, these methods suffer from following drawbacks: (1) large number of parameters to optimize, (2) tendency to fall in a local minimum, (3) likely to over-fit, and (4) high computational complexity required to learn systems [16].

Accuracy of credit assessment is crucial for the profitability of financial institutions. The motivation for doing research in this area is to increase the accuracy of credit scoring, e.g. by minimizing the percentage of bad credits classified as good credits. Increasing the accuracy of even a fraction of a percentage is considered as significant and extremely important achievement for economic reasons because it can mean huge savings for the bank.

These approaches described in the scientific literature [29–33] do not guarantee high accuracy (Table 6). Hence, it is desirable to design dedicated automated system to support the bankers to more accurately predict the credit scoring. To address this issue, we have presented a new model to achieve better performance than the existing models.

1.2. Machine learning techniques

Nowadays, different ensemble techniques are very popular approaches to combine various machine learning algorithms [34–38], which can be used to enhance the obtained outcomes. By using this approach the performance of the entire ensemble can be increased compared to individual component algorithms. In order to find out the efficiency of ensemble learning techniques, there are a variety of ensemble techniques in the literature [34,39–44]. However, some of the most ensembles of classifiers are more popular which are listed as follows: (a) Bagging (Bootstrap aggregation) [45,46], (b) Boosting (AdaBoost) [47,48], (c) Random Forest (RF) [45], (d) Stacking (Stacked Generalization) [49] as well as (e) Mixtures of Experts [50].

There are three important basic assumptions that should be mentioned here as follows: (a) quality, (b) statistical independence (diversity) and (c) efficiency (speed). It is very important to meet all of these assumptions at the same time. According to the practice requirements, taking attention to the statistical independence is the most priority in ensemble learning techniques followed by the quality [51]. This means that attention to the statistical independence and quality is very essential; however, the impact of efficiency (speed) should not be missed.

In the field of credit scoring prediction, ensemble learning methods are also popular and widely used with success: [8,11,26,28,31,32,52–56].

In the last decade, deep learning as one of the machine learning techniques [57–61] has attracted numerous attention from researchers worldwide which has a hierarchical architecture including several layers in which the subsequent steps of information processing take place. To perform the analysis and classification of patterns from raw data using the output layers, the input layers should be used in order to extract features. In other words, there are many hidden layers in deep learning architecture that the input of one layer is given by the output of previous layer expect first input layer. The input of first layer comes from the data.

Based on training methods, deep learning methods can be categorized as follows: (a) Deep discriminatory models such as: Deep Neural Networks (DNNs) [60], Recurrent Neural Networks (RNNs) [62] as well as Convolutional Neural Networks (CNNs) [61]; (b) Unsupervised (generative models) such as: Restricted Boltzmann Machines (RBMs) [63], Deep Belief Networks (DBNs) [64], Deep Boltzmann Machines (DBMs) [65] as well as Regularized Autoencoders [66].

Recently deep learning [66] has become a quite hot topic in computational intelligence research [67,68]. By using a deep structure which includes multiple layers of non-linear operations allow to learn high-level abstraction needed to solve challenging vision, language, and other AI-level tasks. Ensemble of deep learners can boost the performance of traditional methods [69].

In this article, we have applied deep learning technique in credit scoring field. Despite the tremendous potential, the current deep learning methods have disadvantages: (1) computationally complex training, (2) long and inefficient training, and (3) the effect of over fitting, which hinders their effective use in practice. The proposed solution provides fast and effective training approach and thus achieved the highest accuracy.

Nature and universe include a variety of mechanisms and patterns which can be inspired in the machine learning area to solve complex tasks, optimize and also help for training from raw data. The evolutionary computation (EC) [70] is an imitation of nature which represents how its mechanism deals with natural selection, inheritance and functioning. Generally, the EC is trained using on species, but not from an individuals, which allows going through the life of the next generations of people. Therefore, the best generation of solutions can be generated which progressively confront the conditions of the task that they have better adaptation to the nature.

The evolutionary algorithms (EAs) [71] are member of EC which some of well-known algorithms are: (a) evolutionary programming (EP) [72], (b) evolution strategy (ES) [73], (c) genetic programming (GP) [74], (d) genetic algorithm (GA) [75], (e) differential evolution (DE) [76], and (f) learning classifier system (LCS) [77,78] etc. Related techniques, also belonging to EC, are: (g) self-organization (e.g. self-organizing maps [79]), (h) swarm intelligence (SI) [80], (i) artificial immune systems (AIS) [81], (j) particle swarm optimization (PSO) [82], (k) ant colony optimization (ACO) [83], and (l) artificial bee colony algorithm (ABC) [84].

In recent years, GA shows a very good performance for optimizing a variety of tasks. In other words, GA is a popular and effective approach to optimize different ensemble learning of classifiers [75,85,86]. The EC methods are used with success in the field of credit scoring prediction: GA [8,87,88], and GP [13,25,89].

1.3. Goals

Goal 1: Develop new ensemble of classifiers based on fusion of evolutionary computation technique and ensemble learning and deep learning.

Goal 2: Develop a new, fast, efficient and effective training approach for classifiers ensemble of deep multilayer structure.

Goal 3: Verification of proposed solution based on the credit scoring prediction.

1.4. Novelty

Deep genetic cascade ensembles of classifiers (DGCEC) — designed a 16-layer ensemble of SVM classifiers, based on fusion of stratified 10-fold CV method, ensemble learning, deep learning, layered learning, supervised training, feature selection using genetic algorithm (attributes), optimization of classifiers parameters by using genetic algorithm, and also a new genetic layered training technique (for selection of classifiers).

Based on the literature review [29–34,57], the novel features of the study includes the following items:

Genetic layered training — applied new training to optimize the structure of the DGCEC system by combining the classifiers in ensemble (2nd–15th layer). The GA is used to classifiers selection, consisting in the elimination of "bad experts" (reject classifiers with incorrect responses from the 1st–14th layer).

Boost diversity — designed 72 types of classifier models including: 2 types of SVMs, 2 types of normalizations, 2 types of feature selections/extractions, 3 types of kernel functions, and 3 types of parameter optimizations based on 3 types of error calculations.

Cascade structure — proposed new, multilayer, cascade type, structure of deep ensemble, in which the nodes are classifiers and which provides appropriate information flow and fusion and mimicking mechanism of tutoring (Fig. 4).

Deep learning — pioneering use of the DGCEC method based on deep learning to credit scoring prediction.

The motivation to undertake this work is as follows: (a) existing methods obtained accuracies, (b) creating a novel method by combining 3 machine learning techniques (evolutionary computation, ensemble learning and deep learning), and (c) imitate the working of human brain (fusion and flow of information).

2. Materials and methods

2.1. Main assumptions

A1 Conducted an experiment using one of most popular data set: Australian, in order to objectively compare the proposed method to the current results.

A2 Applied only one type of classifier — SVM.

A3 By using the Winner-Takes-All (WTA) rule, the classification of data was carried out.

2.2. Data set

For research purposes, one of most popular data set was used. The data were obtained from the UCI Machine Learning Repository [90]. The real data sets with information about borrowers are described below.

- Statlog Australian Credit Approval
  - The data includes an information about 690 borrowers/instances.
  - The data contains 2 classes: accepted/good (307 instances) or rejected/bad (383 instances) credit application.
  - The data instances contains 14 attributes: 8 categorical and 6 numerical (inputs), and 1 class attribute (output: accepted or rejected). To protect the confidentiality of data, the attributes, names and values have been changed to meaningless symbolic data [4].

A comparison between both classes in the data set and description of division of data into training and testing sets for stratified 10-fold cross-validation technique is given in Table 1, and the description of decision attributes is given in Table 2.

2.3. Methods

In this stage of the study, we have applied different algorithms that were used subsequent steps of the processing and analysis of the borrowers' data. Because of to limitation of space, we do not explain them in detail. Fig. 1 represents the most effective combination of methods (framework) which is chosen based on the obtained outcomes for applied algorithms. The major criteria for selection of the best techniques are: (a) evaluation of the sum of errors relevant to all classification methods, and (b) the accuracy coefficient (ACC) 2.5. As a result, the best algorithms were chosen which the best outcomes were obtained by using them.

2.3.1. Phase I — Normalization

Three normalization types were tested which are as follows: (a) lack of normalization, (b) rescaling of data to the specific range [0, 1], and (c) standardization of signal (where 0 stands for mean signal value whereas 1 represents signal standard deviation). The best outcomes were achieved for **rescaling** and **standardization**, which were applied to normalize each of the borrowers' attributes.

2.3.2. Phase II — Feature extraction

Three feature extraction methods were tested: (a) no extraction, (b) principal component analysis (PCA) [91], and (c) self-organizing map (SOM). The best results were obtained for **no extraction** and **PCA** methods. PCA has been applied to strengthen the distinctive features/attributes of the analyzed borrowers' data.

2.3.3. Phase III — Feature selection

In this study, the feature selection approach was used to choose the most important features. To this end, we applied the proposed methodology with and without feature selection. Hence, we applied our methods with: (a) lack of selection, and (b) GA as our feature selection technique. Our outcome showed that **genetic selection** algorithm had the best performance compared to the first step (lack of selection). As mentioned earlier, a genetic algorithm (GA) [70,92] was applied as feature selection approach for selection of attributes of the borrowers' data. In this study, the genes of the population of individuals represented subsequent

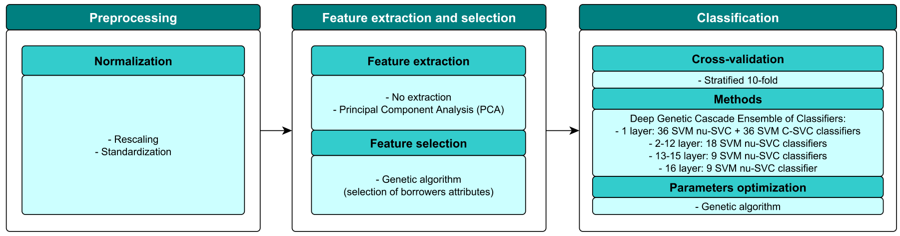

Fig. 1. Illustration of steps involved in the automated analysis of the borrowers' data.

Table 1
A description of division of data into training and testing sets for stratified 10-fold CV.

		Australian Credit Data — 14 attributes				
			stratified 10-fold CV			
			Groups 1–9		Group 10	
Class	Description	Instances number	Training set	Testing set	Training set	Testing set
C1	Accepted credit	307 (45.5%)	276	31	279	28
C2	Rejected credit	383 (55.5%)	345	38	342	41
Sum		690 (100.0%)	621 (90%)	69 (10%)	621 (90%)	69 (10%)

Table 2
Decision attributes used to evaluate the credit risk in the Australian credit data set [4,90].

Attribute	Type	Value
Inputs		
A1	Categorical	0–1
A2	Numerical	13.75–80.25
A3	Numerical	0–28
A4	Categorical	1–3
A5	Categorical	1–14
A6	Categorical	1–9
A7	Numerical	0–28.5
A8	Categorical	0–1
A9	Categorical	0–1
A10	Numerical	0–67
A11	Categorical	0–1
A12	Categorical	1–3
A13	Numerical	0–2000
A14	Numerical	1–100001
Output		
A15	Categorical	0–1

single attributes/features of borrowers' data given as input for the methods (classifiers). Genes takes the values of 0 to reject feature, or 1 to accept feature. The parameters of GA are indicated in Table 3.

2.3.4. Phase IV — Cross-validation

The **stratified 10-fold CV** [34] is employed. As a result, we created randomly 10 combinations of testing and training data sets. Division of borrowers' data into training and testing sets is illustrated in Table 1.

2.3.5. Phase V — Machine learning methods

Four types of one learner, support vector machine (SVM) [93, 94], were tested including: (a) nu-SVC (classifier), (b) C-SVC (classifier), (c) epsilon-SVR (predictor), and (e) nu-SVR (predictor). This article presents the results of **2 learners: SVM (nu-SVC) classifier, and SVM (C-SVC) classifier** (Table 4), which combination obtained the best result. As part of the research, for each of the 4 learners, four types of kernel functions were also tested: (a) linear, (b) polynomial, (c) radial basis function (RBF), and (d) sigmoid. The best results were obtained for **polynomial**, **RBF**, and **sigmoid kernel functions**. Parameters of classifiers: SVM (nu-SVC), and SVM (C-SVC) were presented in Table 3.

2.3.6. Phase VI — Optimization of parameters

One parameter optimization method — **genetic algorithm** (GA) [70,92], was tested. Moreover, **three types of error calculation** were also tested: (1) sum of errors in test data sets, (2) sum of percentage errors in testing and training data sets, and (3) sum of errors in testing and training data sets along with the acceptance features coefficient (Section 2.5). The article presents the results for all types of error calculation. Genetic algorithm parameters were described in Table 3.

2.4. Deep genetic cascade ensemble of classifiers (DGCEC)

Deep Genetic Cascade Ensemble of Classifiers (DGCEC) is a 16-layer system. In the DGCEC method, each classifier from the 1st layer is trained to increase recognition performance of accepted or rejected borrowers based on preprocessed data of borrowers. In the 2nd–16th layers, based on the preprocessed data of borrowers, and responses of classifiers from the first and previous layers, and on the basis of **deep learning** and **genetic feature selection**, process of knowledge extraction take place leading to the final result by the DGCEC method (Fig. 4).

The proposed DGCEC method is a continuation of the research described in papers [95,96].

2.4.1. Philosophy

The motivation to design the DGCEC method was fascination with the processes taking place in the neocortex of the brain. The DGCEC is a hybrid system that comprises the benefits of: (1) evolutionary computation, (2) ensemble learning, and (3) deep learning.

Characteristic features of system:

- **Classifiers (nodes) performing the functions of nerve cells**, combined in the network.
- **Layered learning** — analogously to deep learning, learning process is progressing in stages.
- **Cascade structure** — a new, multi-layered, cascade, structure of ensemble that provides appropriate flow and fusion of information through the mechanism of tutoring.
- **Tutoring** — based on a cascade structure, classifiers (neurons) from subsequent layers can be trained by propagating responses from previous layers and additionally from the first layer.

Table 3
Detailed information about DGCEC system.

Selection of features and classifier parameters optimization The *GA* combined with the stratified 10-fold CV was applied for selection of features and classifier parameters optimization	
Genetic algorithm	• Number of individuals in population: 500; • Type of gene representation: vectors of floating-point; • Structure of chromosome of individual (for example, for *SVM* classifier, type: $nu-SVC$, kernel function type: *RBF*): vector of floating point as $[g_1, g_2, f_1, \ldots, f_{56}]$, where: g_1 — the 1st gene, which defines value of 1st parameter γ, g_2 — the 2nd gene, which defines value of 2nd parameter ν, and $f_1, \ldots, f_{56}$ — 56 gene values from range: [0, 1], which define feature selection, rounded to value 1 — accepted features, or 0 — rejected features. For the other classifier type: *SMV C-SVC*, and other kernel function types: *polynomial* and *sigmoid*, the chromosome consists of appropriate number of genes, g, which defines values of optimizing parameters; • Initial population: random and uniform; • Gene values range for initial population: given for each parameter of classifier in the line *Optimizing parameters* for the local range of gene values, determined experimentally based on a wider (global) range. Range equal to [0, 1] for selection of features; • Target value of fitness function: 0; • Generation maximum number: 30 for 1st layer and 50 for 2nd–16th layer; • Crossover type: intermediate for 1st and 16th layer, and scattered for 2nd–15th layer; Crossover probability equal to 0.7; • Mutation type: uniform; Mutation probability equal to 0.1; • Number of individuals which will survive, unchanged, to the next generation: 5; • Type of fitness function scaling method: ranking; • Type of parent selection: tournament; • Fitness function was computed based on the following equations (error calculations): Sum of errors in test sets: $ERR_T = \frac{err_L}{1000} + err_T$ (1) Sum of percentage errors in testing and training data sets: $ERR_{\%} = (err_{L\%} + err_{T\%})/2$ (2) Sum of errors in testing and training data sets along with the acceptance features coefficient: $ERR_{sum} = err_L + err_T + \frac{F_a}{F}$ (3) where: err_L — number of errors in 10 training data sets; err_T — number of errors in 10 testing data sets; $err_{L\%}$ — percentage error in 10 training sets (the ratio of the sum of errors to the number of all elements, in the training sets: $err_{L\%} = \frac{err_{Lsum}}{L_{sum}} \cdot 100\%$); $err_{T\%}$ — percentage error in the 10 test sets (the ratio of the sum of errors to the number of all elements, in the test sets: $err_{T\%} = \frac{err_{Tsum}}{T_{sum}} \cdot 100\%$); $\frac{F_a}{F} = C_F$ — acceptance feature coefficient (Eq. (5), Section 2.5);
Classifiers	
1st layer 72 trained, tested and optimized classifiers — *experts*: 2 types of classifiers (nu-SVC and C-SVC) · 3 kernel function types · 2 normalization types · 2 feature extraction types · 3 error calculation types	
2nd–12th layer 18 (9 for nu-SVC responses + 9 for C-SVC responses) trained, tested and optimized classifiers — *judges*: 1 type of classifier (nu-SVC) · 3 kernel function types · 3 error calculation types	
13th–16th layer 9 trained, tested and optimized classifiers — *judges*: 1 classifier type (nu-SVC) · 3 kernel function types · 3 error calculation types	
Basic parameters	
SVM	• Type: nu-SVC or C-SVC; • Kernel function type: polynomial or RBF (Gaussian type, radial basis function) or sigmoid; • Number of outputs = 1, from the set: $\{1, 2\}$;
Optimizing parameters Based on wider (global) range, the final ranges of parameters have been experimentally determined.	
SVM	• Only for nu-SVC type: ν $(-n)$ parameter defining width of margins, from the range of [0.005; 0.8] for polynomial, RBF, sigmoid, with resolution 10^{-14}, $500 \cdot 30 = 15000$ values for 1st layer and $500 \cdot 50 = 25000$ values for 2nd-16th layer; • Only for C-SVC type: *cost* $(-c)$ parameter defining width of margins, from the range of [0.1; 10] for polynomial, RBF, and [0.1; 20] for sigmoid, with resolution 10^{-14}, $500 \cdot 30 = 15000$ values for 1st layer and $500 \cdot 50 = 25000$ values for 2nd-16th layer; • *degree* $(-d)$ parameter defining spread of kernel function, from the range of [0.01; 100] for polynomial, with resolution 10^{-14}, $500 \cdot 30 = 15000$ values for 1st layer and $500 \cdot 50 = 25000$ values for 2nd-16th layer; • γ $(-g)$ parameter defining spread of kernel function, from the range of [0.001; 0.1] for polynomial, [0.01; 100] for RBF, and [0.001; 1] for sigmoid, with resolution 10^{-14}, $500 \cdot 30 = 15000$ values for 1st layer and $500 \cdot 50 = 25000$ values for 2nd–16th layer; • *coef* 0 $(-r)$ parameter defining spread of kernel function, from the range of [0.01; 5] for polynomial, and [0.01; 10] for sigmoid, with resolution 10^{-14}, $500 \cdot 30 = 15000$ values for 1st layer and $500 \cdot 50 = 25000$ values for 2nd–16th layer;

- **Genetic layered training** — optimization of the system structure by eliminating bad experts (incorrect responses of classifiers):
  - **Connection optimization** between classifiers (nodes) from adjacent layers (selection of features) realized by a genetic algorithm as an

Table 4
A comparison of the best obtained results, for single classifiers from first layer, for 6 types of SVM.

	Classifiers					
Coefficients	**SVM**	**SVM**	**SVM**	**SVM**	**SVM**	**SVM**
Type	*C − SVC*	*nu − SVC*	*C − SVC*	*C − SVC*	*nu − SVC*	*nu − SVC*
Normalization	*Standardization*	*Rescaling*	*Standardization*	*Standardization*	*Standardization*	*Standardization*
Feature extraction	*PCA*	*None*	*None*	*None*	*None*	*PCA*
Errors	err_t	err_t	err_t	err_t	err_t	err_t
Kernel function	*RBF*	*RBF*	*Polynomial*	*Sigmoid*	*Polynomial*	*Sigmoid*
	Results for training sets					
ERR_{sum}	**562**	**585**	**535**	**751**	**742**	**757**
ACC	90.95%	90.58%	91.39%	87.91%	88.05%	87.81%
	Results for test sets					
ERR_{sum}	**83**	**82**	**81**	**80**	**80**	**76**
ACC	87.97%	88.12%	88.26%	88.41%	88.41%	88.99%

Table 5
Summary of the best results obtained, for subsequent DGCEC system layers (best classifier from first layer and best meta-classifiers from 2nd–16th layer, see Fig. 4).

	Layer							
Coefficients	**1**	**2**	**3**	**7**	**12**	**13**	**15**	**16**
Type	*nu − SVC*	*nu − SVC*	*nu − SVC*	*nu − SVC*	*nu − SVC*	*nu − SVC*	*nu − SVC*	*nu − SVC*
Errors	err_t	err_t	err_{pro}	err_t	err_t	err_{sum}	err_{sum}	err_{sum}
Kernel function	*Sigmoid*	*RBF*	*RBF*	*Polynomial*	*RBF*	*RBF*	*RBF*	*Sigmoid*
	Results for training sets							
ERR_{sum}	**757**	**165**	**0**	**0**	**0**	**0**	**0**	**47**
ACC	87.81%	97.34%	100%	100%	100%	100%	100%	99.24%
	Results for test sets							
ERR_{sum}	**76**	**55**	**47**	**36**	**29**	**25**	**20**	**18**
ACC	88.99%	92.03%	93.19%	94.78%	95.80%	96.38%	97.10%	97.39%

analogy of removing connections between nerve cells in the brain. Unconnected classifiers (neurons) are eliminated (selection of classifiers).
  - **Feedback** take place during training as GA (genetic optimization) and as cross-validation (training) analogous to back connections between nerve cells in the brain.

- **Diversity** of data pre-processing, classifiers (nodes), feature extraction and connections as an analogy to the different types of signal processing, nerve cells, and irregular connections between nerve cells in the neocortex of brain.
  - Diversity of **component classifiers** (nodes) — 72 models of classifiers in first layer of system (2 types of SVMs, 2 types of normalizations, 2 types of feature extractions, 3 types of kernel functions and 3 types of parameter optimizations based on 3 types of error calculations), and 18 models of classifiers in 2nd–12th layers of system (2 types of SVM, 3 types of kernel functions, 3 types of error calculations), and 9 models of classifiers in 13–16th layer (3 types of kernel function, 3 types of error calculation).
  - Diversity of **data pre-processing and feature extraction** in the 1st layer (2 normalization types and 2 feature extraction types);
  - Diversity of **connections** — between layers do not take place all possible connections between nodes (classifiers);
- **Bipolarity** — values of transferred signals from the set {0; 1} analogous to the values of action potentials of neurons;

networks consisting of more than 2 layers are considered as deep. This is similar to the structure of the brain, which consists of 7 layers.
- **Deep multilayered structure (deep learning)** — based on the definition of deep learning, networks consisting of more than 2 layers are considered as deep. This is similar to 7-layer structure of the neocortex of brain.
- **Abstract learning** — internal feature extraction, in the DGCEC system, information flows to successive layers of the structure creating more and more complex features (abstract concepts analogous to brain function).

Name — **deep** — because the DGCEC system topology is deep as it consists of 16 layers, **genetic** — because GA has in the study a key role, **cascade** — because the structure of the system looks like a cascade, and **ensemble of classifiers** — because the developed method comprise of many classifiers.

Layered learning — at the beginning, supervised training applied for 72 classifiers from the 1st layer. Then, supervised genetic training of 18 classifiers from the 2nd layer were performed based on the responses achieved from 72 classifier models from the 1st layer of system. And then, analogously, this training goes through all layers up to 16.

Cross-validation — 10-fold CV was combined with GA, which means that each individual (vector of features) from whole population was tested on all 10 training and test data sets. The effect of overfitting is reduced due to this approach.

1st and 16th layer

- **Genetic selection of features** — was used in the first and 16th layer for **selection** of features (borrower's attributes) and classifiers parameters **optimization** of 72 classifiers.
- **Optimization** — was carried out applying GA (Table 3), which simultaneously optimizing classifiers parameters and choosing input features.

2nd-15th layers

- **Genetic layered training** — was used to optimizing structure for 2nd–15th layer of DGCEC system, based on reference responses and selected features (*experts or judges votes*) from 1st to 14th layer. Genetic algorithm was used to discard the wrong responses (*votes*) of classifiers from 1st to 14th layer, based on the errors in all training nad test data sets, and accept only reliable responses/*votes* (Fig. 5).
- **Votes** — each classifier (*expert*) has 1 output with two values (classes) which are "*1*" or "*2*".

2.4.2. First layer

The first layer of DGCEC comprises 72 classifiers (*experts*): 36 SVM nu-SVC (2 types of normalization, 2 types of feature extraction, 3 types of kernel function and 3 types of error calculation) + 36 SVM C-SVC optimized to minimize errors in recognition of good or bad borrower's. At the inputs of each of the 72 classifiers, are given selected attributes of borrowers, comprise the most characteristic features — selected by the genetic algorithm. For each of the 72 SVM classifiers, the following parameters have been optimized: gamma in kernel function (γ, *-g*), nu of nu-SVC (ν, *-n*) or cost of C-SVC (C, *-c*) [94].

2.4.3. 2nd–12th layers

The 2nd–12th layers of the system consist of 18 classifiers *judges*: 9 SVM nu-SVC + 9 SVM nu-SVC (3 types of kernel functions and 3 types of error calculations, Fig. 4). These 18 meta-classifiers were developed to assessment *votes of experts* from the 1st layer and prior layers. *Judges* were assigned to the two groups of classifiers from first layer, the first 9 *judges* evaluated the responses/*votes* from 36–45 SVM nu-SVC classifiers, next 9 *judges* evaluated the responses/*votes* from 36–45 SVM C-SVC classifiers.

2.4.4. 13th–15th layers

The 13th–15th layer of the system consists of 9 classifiers *judges*: 9 SVM nu-SVC (3 types of kernel functions and 3 types of error calculations, Fig. 4). These 9 meta-classifiers were developed to assessment *votes of experts* from the 1st layer and previous layers. 9 *judges* evaluated the responses/*votes* from 90–99 SVM nu-SVC classifiers.

2.4.5. 16th layer

The 16th layer of DGCEC system consist of 9 classifiers *judges*: 9 SVM nu-SVC (3 types of kernel functions and 3 types of error calculations, Fig. 4). These 9 meta-classifiers were designed to evaluate *the vote of expert* from one previous layer (best response from 1 best SVM nu-SVC classifier from 15th later) and selection of 56 pre-processed attributes of borrower's. In this layer, finally, one best response from 9 meta-classifiers is derived.

Optimal parameter values were selected based on testing many configuration of parameters for the genetic algorithm and the SVM classifier (Table 3).

Figs. 2 and 3 presents the diagram of successive stages of information processing in *deep genetic cascade ensemble of classifiers (DGCEC)*. In Fig. 4, diagram of connections between layers, and

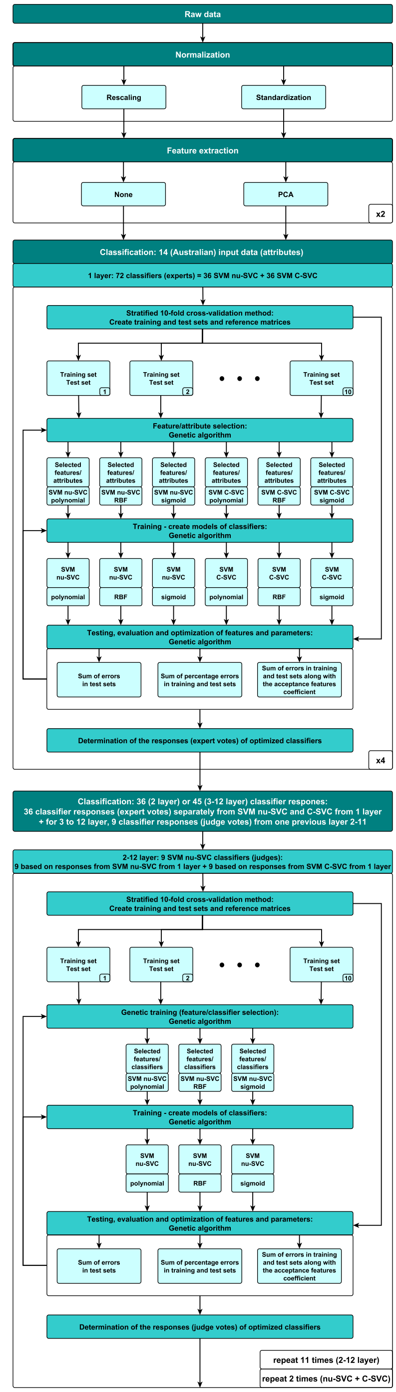

Fig. 2. Part I of *DGCEC system* indicating the information processing.

information fusion and flow in DGCEC system is shown. In Fig. 5, diagram of genetic layered training (for exemplary chromosomes of individuals and a data from single borrower) is presented.

The proposed method (DGCEC system) can combine several classical classifiers to enhance the performance of prediction. This method showed significant performance as compared to the single methods [95,97–117].

2.5. Evaluation criteria

In order to evaluate the performance of chosen algorithms, the following coefficients were computed [118,119]: (a) sum of

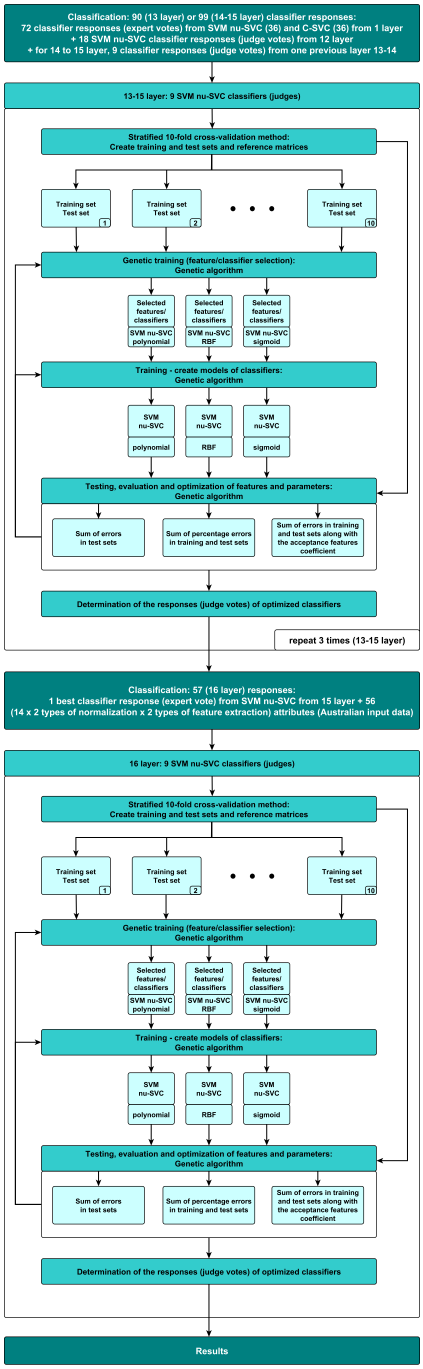

Fig. 3. Part II of *DGCEC system* illustrating the flow of information.

errors (ERR_{sum}), and (b) accuracy (ACC). These coefficients were evaluated by using the generated confusion matrices. Additionally was calculated (c) acceptance feature coefficient (C_F).

The equations of mentioned coefficients are as follows:

- **Sum of errors** (ERR_{sum}) is equal to the total number of incorrect classifications (for 690 classifications from test sets).
- **Accuracy**

$$ACC = \left(\sum_{i=1}^{N} \frac{TP + TN}{TP + FP + TN + FN} \right) \cdot 100\%/N \tag{4}$$

where:

N equal to 10, is the sets number used in the stratified 10-fold cross-validation,
TP represents the True Positive,
TN represents the True Negative,
FP represents the False Positive, and
FN represents the False Negative.

- **Acceptance feature coefficient** (C_F) — total number of accepted features F_a to total number of all features F (ratio in percent). Measure of the performance of genetic selection of features. C_F is computed using following equation:

$$C_F = \frac{F_a}{F} \cdot 100\% \tag{5}$$

3. Results

The proposed methodology of this study was implemented in the MATLAB R2014b environment and used LIBSVM library [94]. The calculations were carried out on an Intel Core i7-6700K 4.0 GHz computer with 32 GB of RAM (only a single core was used).

Table 4 shows the comparison of the best performances using a single classifiers from first layer (6 types of SVM: 2 types and 3 kernel functions).

Table 5 shows the comparison of the best performances for subsequent layers of DGCEC system (best classifier from first layer and best meta-classifiers from 2nd to 16th layer).

The Fig. 6 shows the confusion matrix of the proposed DGCEC system for the testing sets.

The summary of outcomes, obtained in predicting accurately the Australian credit scoring using the same database is shown in Table 6. All compared results were obtained using the same database — Statlog Australian Credit Approval. The summary also contains information about the data analysis methods used.

4. Discussion

4.1. Hypothesis

The obtained results proves the hypothesis that, our proposed novel deep genetic cascade ensemble of SVM classifiers (which called DGCEC system) will enable the automatic, effective, and fast prediction of credit scoring based on using Australian borrowers data.

The results, summarized in Table 5 indicates the superiority of developed method. The presented results shows that the binary classification **accuracy** of proposed DGCEC method is **97.39**%. The obtained result out performed all the published works so far (see Table 6). It can be noted from Table 6 that, the best accuracy in the previous studies was 91.97% whereas our proposed methodology achieved better performance of 97.39%. This helps the organizations to have a higher prediction accuracy (approximately 5.42%). The methods described in the discussion (Table 6) are less complex and less specialized to predict creditworthiness. Hence, the results obtained by these methods are lower than the results obtained by our DGCEC system.

4.2. Single machine learning methods

By analyzing only single machine learning methods (Table 4), it can be seen that the following results were achieved using different classifiers. To the best to the worst ACC: SVM (nu-SVC, Sigmoid) = 88.99%, SVM (nu-SVC, Polynomial) = 88.41%, SVM (C-SVC, Sigmoid) = 88.41%, SCM (C-SVC, Polynomial) = 88.26%, SVM (nu-SVC, RBF) = 88.12% and SVM (C-SVC, RBF) = 87.97%.

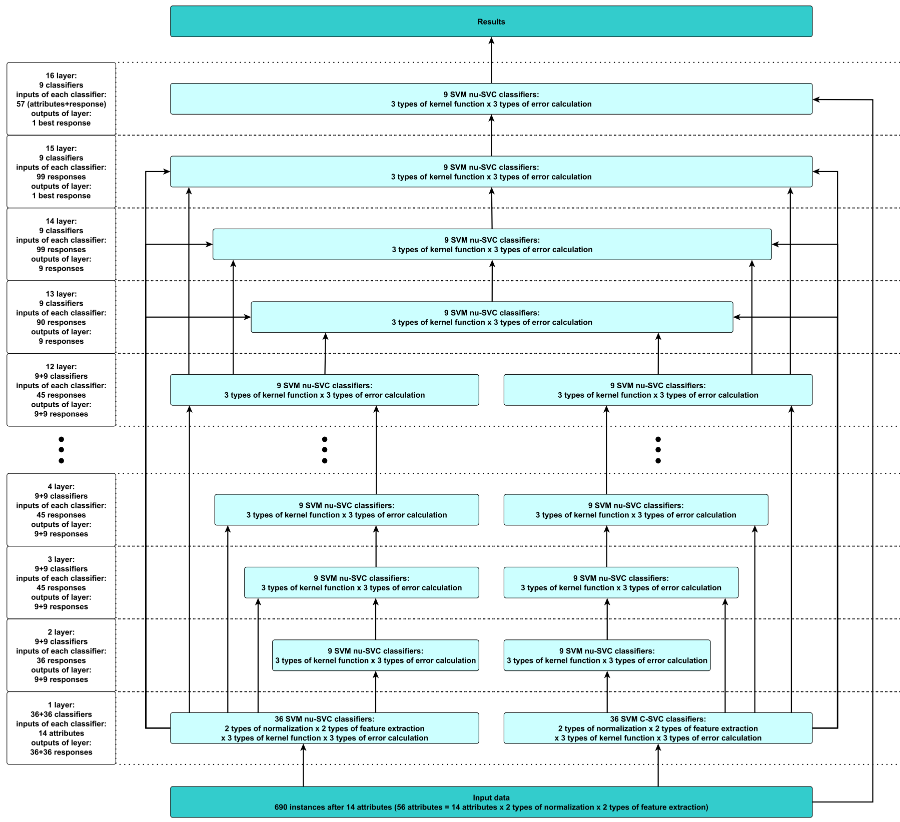

Fig. 4. Diagram of combinations between layers, information fusion and flow in *DGCEC method*.

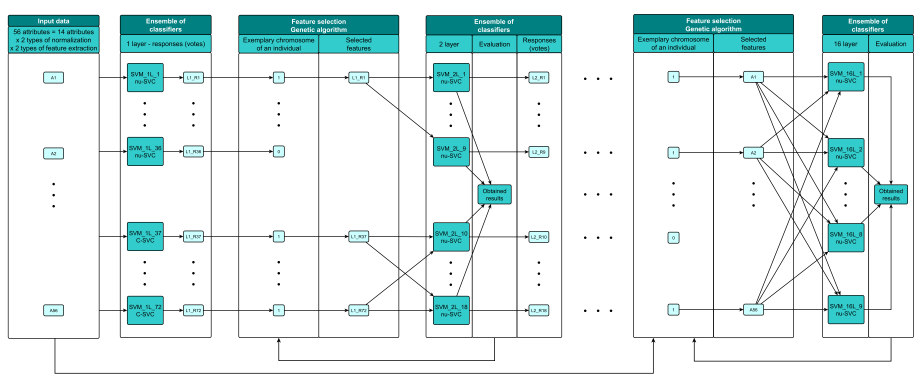

Fig. 5. Scheme of genetic layered training (feature selection by genetic algorithm) used to connecting classifiers of individuals and a single borrower data.

4.3. *Deep genetic cascade ensemble of SVM classifiers*

As we can see in Table 5, the performance of the innovative DGCEC system increased significantly with subsequent layers. This is due to the tutoring mechanism that occurred within the structure of the system (layers 2nd–15th, Fig. 4). This mechanism is analogous to human memory, resulted in memorizing and transferring the most important information to the next layer of the system.

The tutoring effect was obtained through the diversity of: (a) component classifiers, (b) optimization of component classifiers (various error calculations — fitness functions) and (c) system structure (connections between classifiers/nodes).

We have employed two mechanisms to prevent over-fitting: (a) stratified 10-fold crossed cross-validation, and (b) 3 types of error calculations (GA fitness function, Table 3) with more emphasis on evaluating errors in testing sets than in training sets.

Based on the structure of the system (Fig. 4) and comparing the obtained results (Table 5), we can conclude that the use of single machine learning method (only the first layer of the system) achieved an accuracy of 88.99%, after adding the second layer to the system (creating ensemble of classifiers), the accuracy increased by more than 3% to 92.03%, finally after applying deep learning by adding 14 subsequent layers, the accuracy increased again by above 5% to ACC = 97.39%.

It is worth mentioning that the tutoring mechanism (multi-layer structure of the system, 2nd–15th layer) also influenced in the increase of the classification accuracy of training sets (Table 5), which is similar to the processes occurring in the human brain.

A very important aspect is also the genetic layer training (Fig. 5) that affects the optimization of the DGCEC system structure. Genetic training consists of two aspects: (1) elimination of unnecessary attributes of borrowers in the first and 16th layer of

Table 6
Summary of results obtained in predicting the Australian credit scoring automatically using the same database — Statlog Australian Credit Approval.

No.	Work	Year	CV	Feature extraction	Classifier	ACC
1.	Zhang, Zhou, Leung, Zheng	2010	10-fold	–	Vertical bagging decision trees	91.97%
2.	Tsai	2014	5-fold	SOM (Self-organizing map)	MLP lub Homo. Classifier (MLP or CART or LR) Ensembles (weighted voting)	91.61%
3.	Tsai	2009	5-fold	PCA (principle component analysis)	MLP	89.93%
4.	Xu, Zhou, Wang	2009	10-fold	–	HARA (hub authority ranking applicants)	89.28%
5.	Beque, Lessman	2017	10-fold	–	Boosting — Ensemble of SVM (linear)	89.20%
6.	Gorzaczany, Rudziski	2016	10-fold	–	Fuzzy rule-based classifier with multi-objective evolutionary optimization	89.10%
7.	Zhang, Gao, Shi	2014	10-fold	–	Multi-Criteria Optimization Classifier (MCOC)	88.84%
8.	Chang, Yeh	2012	10-fold	–	Artificial immune classifier based on the artificial immune network (AINE-based classifier)	88.70%
9.	Vukovic, Delibasic, Uzelac, Suknovic	2012	10-fold	–	Hybrid model: CBRPGA (combination of CBR and	88.55%
10.	Alaraj, Abbod	2016	5-fold	Multivariate Adaptive Regression Splines (MARS)	Heterogeneous hybrid ensemble with consensus approach combination rule	88.10%
	Proposed method — DGCEC	**2019**	**10-fold**	**None + PCA**	**Deep Genetic Cascade Ensemble of SVM Classifiers (16-layer system)**	**97.39%**

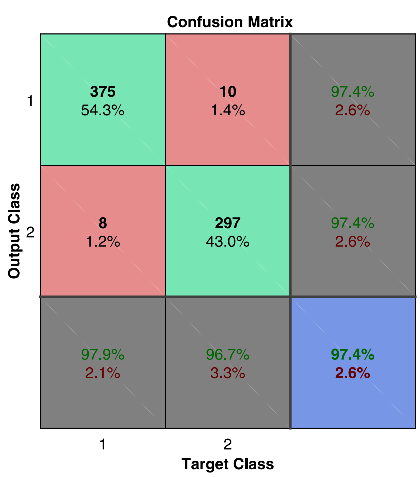

Fig. 6. Confusion matrix for DGCEC system, for test sets.

DGCEC system and (2) selection of nodes (classifiers) in 2nd-15th layer of DGCEC system.

We have obtained the highest performance for proposed DGCEC system with the following salient features: (1) diversity and quality of classifiers (nodes) included in the system, (2) cascade system structure (tutoring mechanism and appropriate genetic layered training), and (3) extraction and selection of features occurring in the subsequent layers of system.

The advantages of the proposed solution are: (1) designed novel machine learning method — deep genetic cascade ensemble of SVM classifiers, (2) novel genetic layered training is applied to the combined classifiers, and (3) obtained highest prediction performance. But, our developed model is a complex structure, requiring long-term training and optimization (similar to deep learning technique).

4.4. Australian data set

The proposed system has been tested on the Australian data set. However, DGCEC system is not limited to this data set only. DGCEC system with a similar data processing, structure, and training mechanism, after training and optimization on another data sets achieved high performance [95,96].

5. Conclusion

The goal of the study was to design novel deep genetic cascade ensemble of SVM classifiers (16-layer system), based on evolutionary computation, ensemble learning and deep learning techniques, that enables to present the effective binary classification of accepted or rejected borrowers. The research used the Statlog Australian (690 instances) credit approval which can be accessed from public machine learning repository data sets.

The novel method was based on the combination of methods including: 2 types of SVM classifiers, 2 types of normalization, 2 types of feature extraction, 3 types of kernel functions, 3 types of parameter optimization, stratified 10-fold CV, ensemble learning, deep learning, layered learning, supervised training, feature selection using genetic algorithm (attributes), optimization of classifiers parameters by using genetic algorithm, and also a new genetic layered training technique (for selection of classifiers).

Deep genetic cascade ensemble of SVM classifiers (DGCEC) achieved a binary classification accuracy of **97.39%** (**18 errors** per 690 classifications). Compared to the current studies, our results are the best to date. The obtained outcomes indicated the efficiency and effectiveness of the proposed methodology for credit scoring prediction which is based on: (1) a genetic layered training which applied to DGCEC system in order to optimize its performance, (2) diversity of component classifiers, and finally (3) cascade structure of DGCEC method.

The proposed method is worth developing because of the very promising outcomes achieved. The future works include: (1) modification of the system structure through the use of various single machine learning methods (classifiers, e.g. kNN, PNN, fuzzy system etc.), (2) testing the performance of DGCEC system using other data sets. Moreover, (3) other types of evolutionary computation techniques such as particle swarm optimization (PSO) can be used instead of GA, and (4) testing the DGCEC system using blind-fold cross validation technique to obtain more generalized model.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2019.105740.

References

[1] S.-T. Li, W. Shiue, M.-H. Huang, The evaluation of consumer loans using support vector machines, Expert Syst. Appl. 30 (4) (2006) 772–782, http://dx.doi.org/10.1016/j.eswa.2005.07.041, http://www.sciencedirect.com/science/article/pii/S0957417405001739.

[2] A. Laha, Building contextual classifiers by integrating fuzzy rule based classification technique and k-nn method for credit scoring, Adv. Eng. Inform. 21 (3) (2007) 281–291, http://dx.doi.org/10.1016/j.aei.2006.12.004, http://www.sciencedirect.com/science/article/pii/S1474034607000067, Applications Eligible for Data Mining.

[3] A. Vellido, P. Lisboa, J. Vaughan, Neural networks in business: a survey of applications (1992–1998), Expert Syst. Appl. 17 (1) (1999) 51–70, http://dx.doi.org/10.1016/S0957-4174(99)00016-0, http://www.sciencedirect.com/science/article/pii/S0957417499000160.

[4] A. Khashman, Credit risk evaluation using neural networks: Emotional versus conventional models, Appl. Soft Comput. 11 (8) (2011) 5477–5484, http://dx.doi.org/10.1016/j.asoc.2011.05.011, http://www.sciencedirect.com/science/article/pii/S1568494611001700.

[5] K.K. Lai, L. Yu, S. Wang, L. Zhou, Credit risk analysis using a reliability-based neural network ensemble model, in: S. Kollias, A. Stafylopatis, W. Duch, E. Oja (Eds.), Artificial Neural Networks – ICANN 2006: 16th International Conference, Athens, Greece, September 10-14, 2006. Proceedings, Part II, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 682–690, http://dx.doi.org/10.1007/11840930_71.

[6] N.-C. Hsieh, Hybrid mining approach in the design of credit scoring models, Expert Syst. Appl. 28 (4) (2005) 655–665, http://dx.doi.org/10.1016/j.eswa.2004.12.022, http://www.sciencedirect.com/science/article/pii/S0957417404001691.

[7] M. uteri, D. Mramor, J. Zupan, Consumer credit scoring models with limited data, Expert Syst. Appl. 36 (3, Part 1) (2009) 4736–4744, http://dx.doi.org/10.1016/j.eswa.2008.06.016, http://www.sciencedirect.com/science/article/pii/S0957417408002996.

[8] S. Oreski, D. Oreski, G. Oreski, Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment, Expert Syst. Appl. 39 (16) (2012) 12605–12617, http://dx.doi.org/10.1016/j.eswa.2012.05.023, http://www.sciencedirect.com/science/article/pii/S095741741200721X.

[9] C.-M. Wang, Y.-F. Huang, Evolutionary-based feature selection approaches with new criteria for data mining: A case study of credit approval data, Expert Syst. Appl. 36 (3, Part 2) (2009) 5900–5908, http://dx.doi.org/10.1016/j.eswa.2008.07.026, http://www.sciencedirect.com/science/article/pii/S0957417408004454.

[10] C.-F. Tsai, Feature selection in bankruptcy prediction, Knowl.-Based Syst. 22 (2) (2009) 120–127, http://dx.doi.org/10.1016/j.knosys.2008.08.002, http://www.sciencedirect.com/science/article/pii/S0950705108001536.

[11] S. Finlay, Multiple classifier architectures and their application to credit risk assessment, European J. Oper. Res. 210 (2) (2011) 368–378, http://dx.doi.org/10.1016/j.ejor.2010.09.029, http://www.sciencedirect.com/science/article/pii/S0377221710006272.

[12] S.-Y. Chang, T.-Y. Yeh, An artificial immune classifier for credit scoring analysis, Appl. Soft Comput. 12 (2) (2012) 611–618, http://dx.doi.org/10.1016/j.asoc.2011.11.002, http://www.sciencedirect.com/science/article/pii/S1568494611004261.

[13] C.-L. Huang, M.-C. Chen, C.-J. Wang, Credit scoring with a data mining approach based on support vector machines, Expert Syst. Appl. 33 (4) (2007) 847–856, http://dx.doi.org/10.1016/j.eswa.2006.07.007, http://www.sciencedirect.com/science/article/pii/S095741740600217X.

[14] G. Zhang, M.Y. Hu, B.E. Patuwo, D.C. Indro, Artificial neural networks in bankruptcy prediction: General framework and cross-validation analysis, European J. Oper. Res. 116 (1) (1999) 16–32, http://dx.doi.org/10.1016/S0377-2217(98)00051-4, http://www.sciencedirect.com/science/article/pii/S0377221798000514.

[15] S.F. Crone, S. Finlay, Instance sampling in credit scoring: An empirical study of sample size and balancing, Int. J. Forecast. 28 (1) (2012) 224–238, http://dx.doi.org/10.1016/j.ijforecast.2011.07.006, http://www.sciencedirect.com/science/article/pii/S0169207011001403, Special Section 1: The Predictability of Financial Markets Special Section 2: Credit Risk Modelling and Forecasting.

[16] R. Malhotra, D. Malhotra, Evaluating consumer loans using neural networks, Omega 31 (2) (2003) 83–96, http://dx.doi.org/10.1016/S0305-0483(03)00016-1, http://www.sciencedirect.com/science/article/pii/S0305048303000161.

[17] E. Alpaydin, Introduction to Machine Learning, MIT Press, 2014.

[18] C. Bishop, Pattern Recognition and Machine Learning, first ed., Springer, 2006.

[19] A.P. Engelbrecht, Computational Intelligence: An Introduction, second ed., Wiley, 2007.

[20] S. Akko, An empirical comparison of conventional techniques, neural networks and the three stage hybrid adaptive neuro fuzzy inference system (ANFIS) model for credit scoring analysis: The case of turkish credit card data, European J. Oper. Res. 222 (1) (2012) 168–178, http://dx.doi.org/10.1016/j.ejor.2012.04.009, http://www.sciencedirect.com/science/article/pii/S0377221712002858.

[21] D. West, Neural network credit scoring models, Comput. Oper. Res. 27 (11) (2000) 1131–1152, http://dx.doi.org/10.1016/S0305-0548(99)00149-5, http://www.sciencedirect.com/science/article/pii/S0305054899001495.

[22] A. Khashman, Neural networks for credit risk evaluation: Investigation of different neural models and learning schemes, Expert Syst. Appl. 37 (9) (2010) 6233–6239, http://dx.doi.org/10.1016/j.eswa.2010.02.101, http://www.sciencedirect.com/science/article/pii/S0957417410001405.

[23] M. Sasiada, A. Fraczek-Szczypta, R. Tadeusiewicz, Efficiency testing of artificial neural networks in predicting the properties of carbon nanomaterials as potential systems for nervous tissue stimulation and regeneration, Bio-Alg. Med-Syst. 13 (1) (2017) 25–35, http://dx.doi.org/10.1515/bams-2016-0025.

[24] I. Brown, C. Mues, An experimental comparison of classification algorithms for imbalanced credit scoring data sets, Expert Syst. Appl. 39 (3) (2012) 3446–3453, http://dx.doi.org/10.1016/j.eswa.2011.09.033, http://www.sciencedirect.com/science/article/pii/S095741741101342X.

[25] C.-S. Ong, J.-J. Huang, G.-H. Tzeng, Building credit scoring models using genetic programming, Expert Syst. Appl. 29 (1) (2005) 41–47, http://dx.doi.org/10.1016/j.eswa.2005.01.003, http://www.sciencedirect.com/science/article/pii/S0957417405000059.

[26] C. Hung, J.-H. Chen, A selective ensemble based on expected probabilities for bankruptcy prediction, Expert Syst. Appl. 36 (3, Part 1) (2009) 5297–5303, http://dx.doi.org/10.1016/j.eswa.2008.06.068, http://www.sciencedirect.com/science/article/pii/S0957417408003862.

[27] G. Wang, J. Ma, L. Huang, K. Xu, Two credit scoring models based on dual strategy ensemble trees, Knowl.-Based Syst. 26 (Supplement C) (2012) 61–68, http://dx.doi.org/10.1016/j.knosys.2011.06.020, http://www.sciencedirect.com/science/article/pii/S0950705111001353.

[28] G. Wang, J. Hao, J. Ma, H. Jiang, A comparative assessment of ensemble learning for credit scoring, Expert Syst. Appl. 38 (1) (2011) 223–230, http://dx.doi.org/10.1016/j.eswa.2010.06.048, http://www.sciencedirect.com/science/article/pii/S095741741000552X.

[29] F. Louzada, A. Ara, G.B. Fernandes, Classification methods applied to credit scoring: Systematic review and overall comparison, Surv. Oper. Res. Manag. Sci. 21 (2) (2016) 117–134, http://dx.doi.org/10.1016/j.sorms.2016.10.001, http://www.sciencedirect.com/science/article/pii/S1876735416300101.

[30] S. Lessmann, B. Baesens, H.-V. Seow, L.C. Thomas, Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research, European J. Oper. Res. 247 (1) (2015) 124–136, http://dx.doi.org/10.1016/j.ejor.2015.05.030, http://www.sciencedirect.com/science/article/pii/S0377221715004208.

[31] A. Bequ, S. Lessmann, Extreme learning machines for credit scoring: An empirical evaluation, Expert Syst. Appl. 86 (Supplement C) (2017) 42–53, http://dx.doi.org/10.1016/j.eswa.2017.05.050, http://www.sciencedirect.com/science/article/pii/S0957417417303718.

[32] C.-F. Tsai, Combining cluster analysis with classifier ensembles to predict financial distress, Inf. Fusion 16 (Supplement C) (2014) 46–58, http://dx.doi.org/10.1016/j.inffus.2011.12.001, http://www.sciencedirect.com/science/article/pii/S1566253511000807, Special Issue on Information Fusion in Hybrid Intelligent Fusion Systems.

[33] W.-Y. Lin, Y.-H. Hu, C.-F. Tsai, Machine learning in financial crisis prediction: A survey, Trans. Sys. Man Cybern. C 42 (4) (2012) 421–436, http://dx.doi.org/10.1109/TSMCC.2011.2170420.

[34] L.I. Kuncheva, Combining Pattern Classifiers: Methods and Algorithms, Wiley-Interscience, 2004.

[35] A. Rahman, B. Verma, Novel layered clustering-based approach for generating ensemble of classifiers, IEEE Trans. Neural Netw. 22 (5) (2011) 781–792, http://dx.doi.org/10.1109/TNN.2011.2118765.

[36] M.D. Muhlbaier, A. Topalis, R. Polikar, Learn^{++} .NC: combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes, IEEE Trans. Neural Netw. 20 (1) (2009) 152–168, http://dx.doi.org/10.1109/TNN.2008.2008326.

[37] N. Garcia-Pedrajas, Constructing ensembles of classifiers by means of weighted instance selection, IEEE Trans. Neural Netw. 20 (2) (2009) 258–277, http://dx.doi.org/10.1109/TNN.2008.2005496.

[38] J.J. Rodriguez, L.I. Kuncheva, C.J. Alonso, Rotation forest: a new classifier ensemble method, IEEE Trans. Pattern Anal. Mach. Intell. 28 (10) (2006) 1619–1630, http://dx.doi.org/10.1109/TPAMI.2006.211.

[39] R. Polikar, Ensemble based systems in decision making, IEEE Circuits Syst. Mag. 6 (3) (2006) 21–45, http://dx.doi.org/10.1109/MCAS.2006.1688199.

[40] L. Rokach, Ensemble-based classifiers, Artif. Intell. Rev. 33 (1–2) (2010) 1–39, http://dx.doi.org/10.1007/s10462-009-9124-7.

[41] R. Polikar, Bootstrap - inspired techniques in computation intelligence, IEEE Signal Process. Mag. 24 (4) (2007) 59–72, http://dx.doi.org/10.1109/MSP.2007.4286565.

[42] D. Opitz, R. Maclin, Popular ensemble methods: an empirical study, J. Artificial Intelligence Res. 11 (1999) 169–198.

[43] T.G. Dietterich, Ensemble methods in machine learning, in: Proceedings of the First International Workshop on Multiple Classifier Systems, in: MCS '00, Springer-Verlag, London, UK, UK, 2000, pp. 1–15, http://dl.acm.org/citation.cfm?id=648054.743935.

[44] Y. Baram, Partial classification: the benefit of deferred decision, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 769–776, http://dx.doi.org/10.1109/34.709564.

[45] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140, http://dx.doi.org/10.1023/A:1018054314350.

[46] B. Efron, R. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, New York, 1993.

[47] M.J. Kearns, R.E. Schapire, L.M. Sellie, Toward efficient agnostic learning, Mach. Learn. 17 (2) (1994) 115–141, http://dx.doi.org/10.1023/A:1022615600103.

[48] L.G. Valiant, A theory of the learnable, Commun. ACM 27 (11) (1984) 1134–1142, http://dx.doi.org/10.1145/1968.1972, http://doi.acm.org/10.1145/1968.1972.

[49] D.H. Wolpert, Stacked generalization, Neural Netw. 5 (1992) 241–259.

[50] R.A. Jacobs, M.I. Jordan, S.J. Nowlan, G.E. Hinton, Adaptive mixtures of local experts, Neural Comput. 3 (1) (1991) 79–87, http://dx.doi.org/10.1162/neco.1991.3.1.79.

[51] R. Duda, P. Hart, D. Stork, Pattern Classification, second ed., Wiley-Interscience, 2000.

[52] B. Twala, Multiple classifier application to credit risk assessment, Expert Syst. Appl. 37 (4) (2010) 3326–3336, http://dx.doi.org/10.1016/j.eswa.2009.10.018, http://www.sciencedirect.com/science/article/pii/S0957417409008847.

[53] D. Zhang, X. Zhou, S.C. Leung, J. Zheng, Vertical bagging decision trees model for credit scoring, Expert Syst. Appl. 37 (12) (2010) 7838–7843, http://dx.doi.org/10.1016/j.eswa.2010.04.054, http://www.sciencedirect.com/science/article/pii/S0957417410003544.

[54] M. Ala'raj, M.F. Abbod, A new hybrid ensemble credit scoring model based on classifiers consensus system approach, Expert Syst. Appl. 64 (Supplement C) (2016) 36–55, http://dx.doi.org/10.1016/j.eswa.2016.07.017, http://www.sciencedirect.com/science/article/pii/S0957417416303621.

[55] L. Nanni, A. Lumini, An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring, Expert Syst. Appl. 36 (2, Part 2) (2009) 3028–3033, http://dx.doi.org/10.1016/j.eswa.2008.01.018, http://www.sciencedirect.com/science/article/pii/S0957417408000249.

[56] C.-F. Tsai, Y.-F. Hsu, D.C. Yen, A comparative study of classifier ensembles for bankruptcy prediction, Appl. Soft Comput. 24 (Supplement C) (2014) 977–984, http://dx.doi.org/10.1016/j.asoc.2014.08.047, http://www.sciencedirect.com/science/article/pii/S1568494614004128.

[57] L. Deng, D. Yu, Deep learning: methods and applications, Found. Trends Signal Process. 7 (2014) 197–387, http://dx.doi.org/10.1561/2000000039.

[58] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, 2016, http://www.deeplearningbook.org, Book in preparation for MIT Press.

[59] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828, http://dx.doi.org/10.1109/TPAMI.2013.50.

[60] J. Schmidhuber, Deep learning in neural networks: An overview, Neural Netw. 61 (2015) 85–117, http://dx.doi.org/10.1016/j.neunet.2014.09.003, http://www.sciencedirect.com/science/article/pii/S0893608014002135.

[61] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.

[62] C. Goller, A. Kuchler, Learning task-dependent distributed representations by backpropagation through structure, in: Neural Networks, 1996., IEEE International Conference on, vol. 1, 1996, pp. 347–352, http://dx.doi.org/10.1109/ICNN.1996.548916.

[63] G.E. Hinton, A practical guide to training restricted Boltzmann machines, in: G. Montavon, G.B. Orr, K.-R. Müller (Eds.), Neural Networks: Tricks of the Trade, second ed., Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 599–619, http://dx.doi.org/10.1007/978-3-642-35289-8_32.

[64] G.E. Hinton, S. Osindero, Y.-W. Teh, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554, http://dx.doi.org/10.1162/neco.2006.18.7.1527.

[65] R. Salakhutdinov, G. Hinton, Deep Boltzmann machines, in: D. van Dyk, M. Welling (Eds.), Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics, in: Proceedings of Machine Learning Research, vol. 5, PMLR, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 2009, pp. 448–455, http://proceedings.mlr.press/v5/salakhutdinov09a.html.

[66] Y. Bengio, Learning deep architectures for AI, Found. Trends Mach. Learn. 2 (1) (2009) 1–127, http://dx.doi.org/10.1561/2200000006, http://dx.doi.org/10.1561/2200000006.

[67] S.M. Erfani, S. Rajasegarar, S. Karunasekera, C. Leckie, High-dimensional and large-scale anomaly detection using a linear one-class {SVM} with deep learning, Pattern Recognit. 58 (2016) 121–134, http://dx.doi.org/10.1016/j.patcog.2016.03.028, http://www.sciencedirect.com/science/article/pii/S0031320316300267.

[68] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, M.S. Lew, Deep learning for visual understanding: A review, Neurocomputing 187 (2016) 27–48, http://dx.doi.org/10.1016/j.neucom.2015.09.116, http://www.sciencedirect.com/science/article/pii/S0925231215017634, Recent Developments on Deep Big Vision.

[69] Y. Ren, L. Zhang, P.N. Suganthan, Ensemble classification and regression-recent developments, applications and future directions [review article], IEEE Comput. Intell. Mag. 11 (1) (2016) 41–53, http://dx.doi.org/10.1109/MCI.2015.2471235.

[70] T. Back, U. Hammel, H.P. Schwefel, Evolutionary computation: comments on the history and current state, IEEE Trans. Evol. Comput. 1 (1) (1997) 3–17, http://dx.doi.org/10.1109/4235.585888.

[71] D. Ashlock, Evolutionary Computation for Modeling and Optimization, first ed., Springer Publishing Company, Incorporated, 2010.

[72] L.J. Fogel, Autonomous automata, Ind. Res. 4 (1962) 14–19.

[73] I. Rechenberg, Evolutionsstrategie: optimierung technischer systeme nach prinzipien der biologischen evolution, (Ph.D. thesis), Frommann-Holzboog, Stuttgart, Germany, 1973.

[74] N.L. Cramer, A representation for the adaptive generation of simple sequential programs, in: Proceedings of the 1st International Conference on Genetic Algorithms, L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 1985, pp. 183–187, http://dl.acm.org/citation.cfm?id=645511.657085.

[75] J.H. Holland, Adaptation in Natural and Artificial Systems, MIT Press, Cambridge, MA, USA, 1992.

[76] R. Storn, K. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. 11 (4) (1997) 341–359, http://dx.doi.org/10.1023/A:1008202821328.

[77] P.L. Lanzi, Learning classifier systems: then and now, Evol. Intell. 1 (1) (2008) 63–82, http://dx.doi.org/10.1007/s12065-007-0003-3.

[78] R.J. Urbanowicz, J.H. Moore, Learning classifier systems: A complete introduction, review, and roadmap, Journal of Artificial Evolution and Applications 2009 (2009) http://dx.doi.org/10.1155/2009/736398.

[79] T. Kohonen, Self-organized formation of topologically correct feature maps, Biol. Cybernet. 43 (1) (1982) 59–69, http://dx.doi.org/10.1007/BF00337288.

[80] G. Beni, J. Wang, Swarm intelligence in cellular robotic systems, in: P. Dario, G. Sandini, P. Aebischer (Eds.), Robots and Biological Systems: Towards a New Bionics? Springer Berlin Heidelberg, Berlin, Heidelberg, 1993, pp. 703–712, http://dx.doi.org/10.1007/978-3-642-58069-7_38.

[81] J.O. Kephart, A biologically inspired immune system for computers, in: In Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems, MIT Press, 1994, pp. 130–139.

[82] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Neural Networks, 1995. Proceedings., IEEE International Conference on, vol. 4, 1995, pp. 1942–1948, http://dx.doi.org/10.1109/ICNN.1995.488968.

[83] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, IEEE Trans. Syst. Man Cybern. B 26 (1) (1996) 29–41, http://dx.doi.org/10.1109/3477.484436.

[84] D. Karaboga, An Idea Based on Honey Bee Swarm For Numerical Optimization, 2005.

[85] L.I. Kuncheva, L.C. Jain, Designing classifier fusion systems by genetic algorithms, IEEE Trans. Evol. Comput. 4 (4) (2000) 327–336, http://dx.doi.org/10.1109/4235.887233.

[86] M.L. Raymer, W.F. Punch, E.D. Goodman, L.A. Kuhn, A.K. Jain, Dimensionality reduction using genetic algorithms, IEEE Trans. Evol. Comput. 4 (2) (2000) 164–171, http://dx.doi.org/10.1109/4235.850656.

[87] M.B. Gorzaczany, F. Rudziski, A multi-objective genetic optimization for fast, fuzzy rule-based credit classification with balanced accuracy and interpretability, Appl. Soft Comput. 40 (Supplement C) (2016) 206–220, http://dx.doi.org/10.1016/j.asoc.2015.11.037, http://www.sciencedirect.com/science/article/pii/S1568494615007553.

[88] S. Vukovic, B. Delibasic, A. Uzelac, M. Suknovic, A case-based reasoning model that uses preference theory functions for credit scoring, Expert Syst. Appl. 39 (9) (2012) 8389–8395, http://dx.doi.org/10.1016/j.eswa.2012.01.181, http://www.sciencedirect.com/science/article/pii/S0957417412002096.

[89] J.-J. Huang, G.-H. Tzeng, C.-S. Ong, Two-stage genetic programming (2sgp) for the credit scoring model, Appl. Math. Comput. 174 (2) (2006) 1039–1053, http://dx.doi.org/10.1016/j.amc.2005.05.027, http://www.sciencedirect.com/science/article/pii/S009630030500531X.

[90] M. Lichman, UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, 2013, http://archive.ics.uci.edu/ml.

[91] W.J. Krzanowski, Principles of Multivariate Analysis: A User's Perspective, Oxford University Press, 2000,

[92] L. Rutkowski, Computational Intelligence: Methods and Techniques, Springer, 2008.

[93] C. Cortes, V. Vapnik, Support-vector networks, Mach. Learn. 20 (3) (1995) 273–297, http://dx.doi.org/10.1007/BF00994018.

[94] C.-C. Chang, C.-J. Lin, LIBSVM: A library for support vector machines, ACM Trans. Intell. Syst. Technol. (TIST) 2 (2011) 27:1–27:27, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[95] P. Pławiak, Novel genetic ensembles of classifiers applied to myocardium dysfunction recognition based on ECG signals, Swarm Evol. Comput. 39 (2018) 192–208, http://dx.doi.org/10.1016/j.swevo.2017.10.002, http://www.sciencedirect.com/science/article/pii/S2210650217306053.

[96] P. Pławiak, U.R. Acharya, Novel deep genetic ensemble of classifiers for arrhythmia detection using ECG signals, Neural Comput. Appl. (2019) http://dx.doi.org/10.1007/s00521-018-03980-2.

[97] M. Abdar, M. Zomorodi-Moghadam, R. Das, I.-H. Ting, Performance analysis of classification algorithms on early detection of liver disease, Expert Syst. Appl. 67 (2017) 239–251, http://dx.doi.org/10.1016/j.eswa.2016.08.065, http://www.sciencedirect.com/science/article/pii/S095741741630464X.

[98] M. Abdar, N.Y. Yen, J.C.-S. Hung, Improving the diagnosis of liver disease using multilayer perceptron neural network and boosted decision trees, J. Med. Biol. Eng. 38 (6) (2018) 953–965, http://dx.doi.org/10.1007/s40846-017-0360-z.

[99] P. Chowriappa, S. Dua, U.R. Acharya, M.M.R. Krishnan, Ensemble selection for feature-based classification of diabetic maculopathy images, Comput. Biol. Med. 43 (12) (2013) 2156–2162, http://dx.doi.org/10.1016/j.compbiomed.2013.10.003, http://www.sciencedirect.com/science/article/pii/S0010482513002813.

[100] U. Desai, R.J. Martis, U.R. Acharya, C.G. Nayak, G. Seshikala, R. Shetty K, Diagnosis of multiclass tachycardia beats using recurrence quantification analysis and ensemble classifiers, J. Mech. Med. Biol. 16 (01) (2016) 1640005, http://dx.doi.org/10.1142/S0219519416400054.

[101] P. Pławiak, Novel methodology of cardiac health recognition based on ECG signals and evolutionary-neural system, Expert Syst. Appl. 92 (2018) 334–349, http://dx.doi.org/10.1016/j.eswa.2017.09.022, https://www.sciencedirect.com/science/article/pii/S0957417417306292.

[102] K. Rzecki, T. Sośnicki, M. Baran, M. Niedźwiecki, M. Król, T. Łojewski, U. Acharya, O. Yildirim, P. Pławiak, Application of computational intelligence methods for the automated identification of paper-ink samples based on LIBS, Sensors 18 (11) (2018) 3670, http://dx.doi.org/10.3390/s18113670.

[103] K. Rzecki, P. Pławiak, M. Niedźwiecki, T. Sośnicki, J. Leśkow, M. Ciesielski, Person recognition based on touch screen gestures using computational intelligence methods, Inform. Sci. 415-416 (2017) 70–84, http://dx.doi.org/10.1016/j.ins.2017.05.041, http://www.sciencedirect.com/science/article/pii/S002002551730751X.

[104] P. Pławiak, W. Maziarz, Classification of tea specimens using novel hybrid artificial intelligence methods, Sensors Actuators B 192 (2014) 117–125, http://dx.doi.org/10.1016/j.snb.2013.10.065, http://www.sciencedirect.com/science/article/pii/S0925400513012513.

[105] P. Pławiak, An estimation of the state of consumption of a positive displacement pump based on dynamic pressure or vibrations using neural networks, Neurocomputing 144 (2014) 471–483, http://dx.doi.org/10.1016/j.neucom.2014.04.026, http://www.sciencedirect.com/science/article/pii/S0925231214006110.

[106] P. Pławiak, K. Rzecki, Approximation of phenol concentration using computational intelligence methods based on signals from the metal-oxide sensor array, IEEE Sens. J. 15 (3) (2015) 1770–1783, http://dx.doi.org/10.1109/JSEN.2014.2366432.

[107] W. Ksiażek, M. Abdar, U.R. Acharya, P. Pławiak, A novel machine learning approach for early detection of hepatocellular carcinoma patients, Cognitive Systems Research 54 (2019) 116–127, http://dx.doi.org/10.1016/j.cogsys.2018.12.001, http://www.sciencedirect.com/science/article/pii/S1389041718308714.

[108] U.R. Acharya, H. Fujita, O.S. Lih, Y. Hagiwara, J.H. Tan, M. Adam, Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network, Inform. Sci. 405 (2017) 81–90, http://dx.doi.org/10.1016/j.ins.2017.04.012, http://www.sciencedirect.com/science/article/pii/S0020025517306539.

[109] U.R. Acharya, H. Fujita, S.L. Oh, Y. Hagiwara, J.H. Tan, M. Adam, Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals, Inform. Sci. 415–416 (2017) 190–198, http://dx.doi.org/10.1016/j.ins.2017.06.027, http://www.sciencedirect.com/science/article/pii/S0020025517308009.

[110] U.R. Acharya, S.L. Oh, Y. Hagiwara, J.H. Tan, M. Adam, A. Gertych, R.S. Tan, A deep convolutional neural network model to classify heartbeats, Comput. Biol. Med. 89 (2017) 389–396, http://dx.doi.org/10.1016/j.compbiomed.2017.08.022, http://www.sciencedirect.com/science/article/pii/S0010482517302810.

[111] O. Yildirim, P. Pławiak, R.-S. Tan, U.R. Acharya, Arrhythmia detection using deep convolutional neural network with long duration ECG signals, Comput. Biol. Med. 102 (2018) 411–420, http://dx.doi.org/10.1016/j.compbiomed.2018.09.009, http://www.sciencedirect.com/science/article/pii/S0010482518302713.

[112] R.J. Martis, U.R. Acharya, L.C. Min, ECG beat classification using PCA, LDA, ICA and discrete wavelet transform, Biomed. Signal Process. Control 8 (5) (2013) 437–448, http://dx.doi.org/10.1016/j.bspc.2013.01.005, http://www.sciencedirect.com/science/article/pii/S1746809413000062.

[113] L. Ogiela, R. Tadeusiewicz, M.R. Ogiela, Cognitive analysis in diagnostic DSS-type IT systems, in: L. Rutkowski, R. Tadeusiewicz, L.A. Zadeh, J.M. Żurada (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2006, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 962–971.

[114] M.R. Ogiela, R. Tadeusiewicz, Nonlinear processing and semantic content analysis in medical imaging-a cognitive approach, IEEE Trans. Instrum. Meas. 54 (6) (2005) 2149–2155, http://dx.doi.org/10.1109/TIM.2005.858566.

[115] D. Panek, A. Skalski, J. Gajda, R. Tadeusiewicz, Acoustic analysis assessment in speech pathology detection, Int. J. Appl. Math. Comput. Sci. 25 (3) (2015) 631–643, https://content.sciendo.com/view/journals/amcs/25/3/article-p631.xml.

[116] R. Tadeusiewicz, L. Ogiela, M.R. Ogiela, Cognitive analysis techniques in business planning and decision support systems, in: L. Rutkowski, R. Tadeusiewicz, L.A. Zadeh, J.M. Żurada (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2006, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 1027–1039.

[117] K. Sun, S. Mou, J. Qiu, T. Wang, H. Gao, Adaptive fuzzy control for non-triangular structural stochastic switched nonlinear systems with full state constraints, IEEE Trans. Fuzzy Syst. (2018) 1, http://dx.doi.org/10.1109/TFUZZ.2018.2883374.

[118] T. Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. 27 (8) (2006) 861–874, http://dx.doi.org/10.1016/j.patrec.2005.10.010.

[119] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, Inf. Process. Manage. 45 (4) (2009) 427–437, http://dx.doi.org/10.1016/j.ipm.2009.03.002, http://www.sciencedirect.com/science/article/pii/S0306457309000259.

P. Pławiak, B.Eng., M.Sc., Ph.D., was born in Ostrowiec, Poland, in 1984. He obtained his B.Eng. and M.Sc. degree in Electronics and Telecommunications and his Ph.D. degree with honors in Biocybernetics and Biomedical Engineering at the AGH University of Science and Technology, Cracow, Poland, in 2012 and 2016, respectively. He is an Assistant Professor of the Institute of Teleinformatics, Cracow University of Technology, Cracow, Poland. He has published 15 papers in refereed international SCI-IF journals. He is a reviewer of many prestigious and reputed journals. His research interests include machine learning and computational intelligence (e.g., artificial neural networks, genetic algorithms, fuzzy systems, support vector machines, k-nearest neighbors, and hybrid systems), ensemble learning, deep learning, evolutionary computation, classification, pattern recognition, signal processing and analysis, data analysis and data mining, sensor techniques, medicine, biocybernetics, and biomedical engineering. Detail and most up-to-date information is available at ResearchGate and Google Scholar.

M. Abdar received the bachelor's degree in computer engineering from Damghan University, Iran, in 2015. He also received the masters degree in computer science and engineering from the University of Aizu, Aizu, Japan, in 2018. Currently, he is working toward his Ph.D. degree at the Department of Computer Science, University of Quebec in Montreal, Montreal (QC), H2X 3Y7, Canada. He has several papers about data mining, machine learning, and user modeling in some refereed international journals and conferences. He is a member of International Association of Engineers. He is also very active in five international conferences (e.g., TCP in IEEE AINA 2018, IEEE AINA 2019 and IEEE AINA 2020) and some referred international journals, such as the IEEE Access, the Future Generation Computer Systems (outstanding reviewer in October 2017), the Neurocomputing (outstanding reviewer in January 2017), the Physica A: Statistical Mechanics and its Applications, the Computers in Human Behavior, the International Journal of Applied Mathematics & Computer Science, the Expert Systems, the Journal of Internet Technology, the Health Information Management Journal, the Neural Computing and Applications, the Journal of Medical Systems, the Computer Methods and Programs in Biomedicine and the Web Intelligence as a Reviewer. His research interests include data mining, machine learning, ensemble learning, evolutionary algorithms and user modeling. He is a recipient of the Fonds de Recherche du QuebecNature et Technologies Award (ranked 5th among 20 candidates in the second round of selection process) in 2019.

U. Rajendra Acharya, M Tech, Ph.D., DEng is a senior faculty member at Ngee Ann Polytechnic, Singapore. He is also (i) Adjunct Professor at Taylors University, Malaysia, (ii) Adjunct Faculty at Singapore Institute of Technology-University of Glasgow, Singapore, and (iii) Associate faculty at Singapore University of Social Sciences, Singapore. He received his Ph.D. from National Institute of Technology Karnataka (Surathkal, India) and DEng from Chiba University (Japan). He has published more than 400 papers, in refereed international SCI-IF journals (345), international conference proceedings (42), books (17) with more than 21,500 citations in Google Scholar (with h-index of 76). He is ranked in the top 1% of the Highly Cited Researchers (2016,2017 and 2018) in Computer Science according to the Essential Science Indicators of Thomson. He has worked on various funded projects, with grants worth more than 2 million SGD. He has three patents and in the editorial board of many journals. He has served as guest editor for many journals. His major academic interests are in biomedical signal processing, biomedical imaging, data mining, visualization and biophysics for better healthcare design, delivery and therapy. Please visit https://scholar.google.com.sg/citations?user=8FjY99sAAAAJ&hl=en for more details.