Development of a Hybrid Credit Scoring Model for the Banking System

Mohamad Ali Rastegar[a,*], Mitra Faraji[b]

[a] Department of Financial Engineering, Faculty of Industrial Engineering & Systems, Tarbiat Modares University, Jalal AleAhmad Ave., Tehran, Iran. P.O.Box: 14115-111.
[b] Department of Financial Engineering, Faculty of Financial Science, Khatam University, Hakim Azam Street, Tehran, Iran.

Abstract

The banking system tend to internalize scoring according to Basel II & III and their Central Bank regulations. Consequently, these banking systems are in dire need of credit scoring models. In this study, first, we present a probabilistic neural network (PNN) algorithm for credit scoring of bank customers optimized by means of a genetic algorithm. Based on data from legal customers of one Iranian bank, its performance is compared with seven common machine-learning algorithms. Then we developed a new hybrid performance metric, called probabilities of credit scoring correctness, by combining several performance metrics. The banking system has proposed several credit-scoring models. Models such as single classifiers, hybrid models, and ensemble models determine the class of customers (good or bad). In order to calculate the expected loss and unexpected loss, banks need the probability of default. In general, the proposed model can utilize m performance metrics and n classifiers; the larger *m* and *n*, the more reliable the customer class estimates will be. In fact, the purpose of this paper is to create a hybrid approach for credit scoring Iranian banks' clients, thus obtaining the probability of default and credit risk models for the banking system, especially the weak banking system.

Keywords: Credit Scoring, Machine Learning, Classification, Probabilistic Neural Networks (PNN), Optimization

1. Introduction

As banks are known to act as intermediaries of funds, one of their most important activities is lending. Getting bank receivables paid is key to obtaining a bank's financial resources, and a failure to do so results in both asset and equity loss. These activities are exposed to credit risk, so the credit standing of bank borrowers should be assessed [1]. The Basel II agreement states that capital storage requirements for banks should be based on the bank's credit default risk, as well as their credit-scoring mechanism. Credit scoring agencies have been included in this agreement as one of the main methods of assessing and measuring credit risks [2].
Poor and developing countries such as Iran do not have reputable credit rating agencies and their banking system utilizes internal credit scoring based on Basel II & III and their Central Bank regulations. As a result, banks in such countries proceed to evaluate the credit risk of their customers according to models developed in the bank.

* Corresponding Author
Tel. and Fax: +98 21 82884387;
Cell: +989127307267
E-mail addresses: ma_rastegar@modares.ac.ir (M.A. Rastegar), mitra_faraji1991@yahoo.com (M. Faraji)

Since the banking system is facing a dire need for credit scoring models, machine learning can be of great help to this industry in poor or developing countries. The credit models contain different statistical and non-statistical credit risk models based on classifications for scoring purposes. Some of the various classification models we can mention include the k-nearest neighbors ([3];[4];[5];[6];[7]), support vector machine ([8];[9];[10];[11]), logit regression ([12];[13];[14];[15];[16];[17]), various neural networks such as feed-forward neural networks, back propagation and multilayer perceptron ([18];[19];[20];[21];[22];[23];[24];[25];[26]), decision tree ([27];[28];[29]), Boosting ([30];[31];[32];[33];[34]) and the dynamic ensemble classification - soft probability technique [35]. Additionally, optimization-based methods and metaheuristic algorithms are presented separately ([36];[37];[38];[39];[40]). The metaheuristic algorithms such as genetic algorithms (GA), and particle swarm optimization (PSO), are the algorithms which, inspired by nature, physics, and humans, are capable of performing optimization operations in the search space with high accuracy.

The efficiency and accuracy of prediction classification models in credit scoring can be improved through a variety of hybrid approaches. A hybrid NN-LR credit scoring model was proposed by [25]. A neural network model is trained in the first stage and then combined with logistic regression in the second stage. According to the H-measure, the AUC, and the accuracy of three benchmark datasets, the hybrid model performed exceptionally well. [41] propose a combination entropy-based approach for proactive credit scoring that overcomes cold-start and data imbalance issues that prevent canonical approaches from being effective. In addition to being able to overcome the cold-start issue, this approach also allows for the management of unbalanced data distribution, allowing it to be used in conjunction with existing approaches. Based on voting-based outlier detection and balanced sampling, [42] proposed a hybrid ensemble model for credit scoring. The new hybrid ensemble model achieves superior predictive performance by adjusting to outliers and imbalances in the data.

The probabilistic neural network is another classification method that has been used by various researchers ([43];[44];) and has been shown to be extremely effective in the credit scoring model. Based on the Bayesian probability density function, the output of the pattern layer of this neural network performs better or worse in classification based on a smoothing parameter whose value impacts the performance in classification. In previous studies, the value of this parameter was thought to be equal to the standard deviation. However, when applying the GA-PSO algorithm in the present study, we get this parameter by combining Genetic Algorithm with Particle Swarm Optimization. There are two different scenarios considering the optimization of the fitness function for this purpose. In the first scenario, ACC is set at its maximum value through adjusting the smoothing parameter. However, in the second scenario, the goal is to maximize the value of the Score function.

In this study, first, a probabilistic neural network (PNN) algorithm is presented for credit scoring of bank customers, and it is optimized using a genetic algorithm. Data from legal customers of one Iranian bank is used to compare its performance with seven common machine-learning algorithms. Our next step was to combine several performance metrics into a new hybrid metric called probabilities of credit scoring correctness. The banking system has proposed several credit-scoring models. Models such as single classifiers, hybrid models, and ensemble models determine the class of customers (good or bad). In order to calculate the expected loss and unexpected loss, banks need the probability of default. In general, the proposed model can utilize m performance metrics and n classifiers; the larger m and n, the more reliable the customer class estimates will be. In fact, the purpose of this paper is to create a hybrid approach for credit-scoring Iranian banks' clients, thus obtaining the probability of default and credit risk models for the banking system, especially the weak banking system.

The rest of this paper is organized as follows: Section 2 addresses research methodology and data sources. Section 3 discusses the proposed algorithm results. Section 4 concludes and discusses further research opportunities.

2. Research methodology

For customer credit scoring, different classification methods and performance metrics are available. As mentioned, there are *n* possible classifiers and *m* performance metrics to use for credit scoring in the proposed method. We used 8 common classifiers, namely KNN, AdaBoost, D-Tree, Logit, SVM, ANN, and DECSP in this paper.

A KNN algorithm falls into the nonparametric classification method category, which means that the algorithms does not rely on assumptions about the underlying distribution of data. It is one of the primary techniques in creating classifiers that are often a benchmark to apply to more complex algorithms [6].

For binary classification, AdaBoost is one of the most popular boosting algorithms. It constructs high quality competitive classifiers by sequentially combining training weak classifiers; it also shows special attention to weak classifiers accompanied by good performance [31].

Decision trees (DTs) are used for structuring and displaying knowledge from a large number of samples. The DT structure is like a flowchart, where nodes represent test inputs, branches mark test outputs, and leaf nodes indicate class labels [29]. The decision tree approach makes it possible to classify and interpret complex decisions easily.

Logistic regression is the regression analysis that is conducted when the dependent variable is binary. Like all regression analyses, logistic regression is predictive. It describes and explains relationships between one dependent binary variable and at least one nominal, ordinal, interval, or ratio-level independent variable [13].

SVM is a relatively new learning algorithm used to categorize binary data. This technique simply sub-divides binary data by a hyperplane while maximizing the margin between the hyperplane and the examples [8].

Neuronal networks refer to organic or artificial systems of neurons that will recognize underlying relationships in a piece of information by replicating the way the human brain works [19].

DECSP is a new classification method proposed by [35]. This method combines different classifiers for the samples in the testing set based on their classification results, as well as the relative costs of Type I error and Type II error in the validation set, to get an interval probability of default by using soft probability.

Another method for credit scoring that was shown to have high performance is the Probabilistic Neural Network (PNN), which has been proven by [44]. As the PNN technique has been optimized using the GA-PSO algorithm, this algorithm and its optimization process are described in the following. It should be noted that the parameters from 7 other methods were selected to avoid over-fitting of the out-of-sample data.

2.1. Probabilistic Neural Network

In the theory of statistical pattern classification, PNNs were introduced in 1990 by [45]. PNNs provide a scalable alternative to conventional back-propagation neural networks in classification problems without the need for massive backward and forward calculations inherent in ordinary neural networks. Additionally, they can handle smaller sets of training data [46].

PNN consists of four layers: input, pattern, summation and output. When an input is present, the first layer computes distances between that input vector and the training input vector to produce a vector whose elements indicate how close the input is to a training input. Each input vector is added to the second layer, which produces a vector of probabilities as the net output. Finally, a compete transfer function picks the maximum probability from all the classes on the output of the second layer, making it a 1 in the case of that class and 0 in the case of the other classes.

Following is the Bayesian probability density function corresponding to each output of the PNN pattern node [44]:

$$P\left(X \mid C_i\right) = \frac{1}{\left(2\pi\sigma^2\right)^{m/2} n_i} \sum_{k=1}^{n} \exp\left[\frac{-\left\|X - X_{ik}\right\|^2}{2\sigma^2}\right] \tag{1}$$

where, X is the vector of observed input, n_i is the number of training pattern for class C_i, m is the vector dimension, X_{ik} is the kth training vector for class C_i, and σ is the smoothing parameter. In this research, σ is obtained from the optimization process with GA-PSO algorithm.

2.2. Hybrid GA-PSO Algorithm

Several natural phenomena, such as genetic inheritance and Darwinian struggles for survival, have been expressed mathematically in terms of evolutionary algorithms (EAs). These represent an interesting category of heuristic search [47]. Due to the increased availability of computational power, availability of robust open-source software libraries, and increasing demand for artificial intelligence techniques, evolutionary algorithms are likely to see increased development and use much like other artificial intelligence techniques [48]. An EA consists of the following four steps: initialization, selection, operators, and termination. Each of these steps corresponds to the nature of natural selection in general, enabling implementations of this algorithm category to be easily modularized. As in natural selection, fitter members will proliferate in an EA while less fit members will die off and not contribute to the next generation [49].

In the context of metaheuristic algorithms, both GA and PSO algorithms are widely used. GA is derived from natural selection and the genetic principle. It involves intelligent exploitation of random searches made to discover solution space areas with better performance.

A population of randomly generated individuals forms the base of the evolution, and each iteration is named a generation. The fitness of every individual of the population is evaluated in each generation. The fittest individuals are stochastically selected from the population, and each individual's genome undergoes modification to form a new generation. These new candidate solutions are then used in the next algorithm iteration. The algorithm usually terminates when the population reaches a certain fitness level or a maximum number of generations [50].

Generally, Particle Swarm Optimization (PSO), proposed by [51], is a population-based algorithm that is optimized using the intelligence of certain animals, such as flocks of birds or schools of fish [52]. The algorithm quantifies each particle based on its position and velocity. With the best previously and current positions of each comparison particle stored in memory, the best particle motions are chosen in the next step, and then their speed and position are updated according to the best local and global answers. The position of a particle is determined by adding the velocity (v) of the particle (p) with its current position:

$$p_{ij}\left(t+1\right)=p_{ij}\left(t\right)+v_{ij}\left(t+1\right) \tag{2}$$

The velocity of each particle is also obtained as:

$$v_{ij}\left(t+1\right)=wv_{ij}\left(t\right)+c_1r_{1j}\left(t\right)\left[p_{ij}^{best}-p_{ij}\left(t\right)\right]+c_2r_{2j}\left(t\right)\left[p_{j}^{gbest}-p_{ij}\left(t\right)\right] \tag{3}$$

where, p_{ij}^{best} is the ith particle's own best position, p_{j}^{gbest} is the global best position, w is the inertia weight ($0\le w<1$), c_1 and c_2 are the personal and global learning coefficients, respectively ($0\le c_1,c_2\le 2$), r_1 and r_2 are uniform distributed random numbers ($0\le r_1,r_2\le 1$).

The PSO algorithm has a high convergence rate, but compared to the genetic algorithm, it has lower accuracy and is likely to remain in local best points. On the other hand, the genetic algorithm has high accuracy but has a relatively slow convergence rate. A hybrid GA-PSO algorithm has been proposed in order to overcome the shortcomings of PSO and GA ([53];[54];[55]). Fig. 1 shows the flowchart of the GA-PSO hybrid algorithm.

2.3. Classification performance criteria

In this study, the following criteria will be used for measuring and assessing the classification algorithm:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{4}$$

$$\text{ACC} = \frac{\text{TN} + \text{TP}}{\text{TP+TN} + \text{FP} + \text{FN}} \tag{5}$$

$$\text{Recall or Sensitivity} = \frac{\text{TP}}{\text{TP+FN}} \tag{6}$$

$$\text{F-Score} = \left(2 \times \text{Recall} \times \text{Precision}\right) \ / \left(\text{Recall} + \text{Precision}\right) \tag{7}$$

$$\text{Error}_{\text{Type I}} = \frac{\text{FP}}{\text{TN} + \text{FP}} \tag{8}$$

$$\text{Error}_{\text{Type II}} = \frac{\text{FN}}{\text{FN} + \text{TP}} \tag{9}$$

$$\text{TE} = \frac{\text{FN} + \text{FP}}{\text{TN} + \text{FP} + \text{FN} + \text{TP}} \tag{10}$$

Where, TP stands for true positive, TN for true negative, and FP and FN stand for false positive and false negative, respectively. The classification score functions is also used to rank the classifiers:

$$\text{Score} = \text{ACC} + \text{Precision} - \frac{5}{6}\text{Error}_{\text{Type I}} - \frac{1}{6}\text{Error}_{\text{Type II}} \tag{11}$$

2.4. Optimization scenarios

a. ACC as a fitness

We consider the fitness function as the value of ACC,

$$Z = \max \ \text{ACC}(\sigma) \tag{12}$$

Therefore, in this scenario, the goal is to find the optimal value of σ to maximize the ACC value.

b. Score as a fitness

In this scenario, we consider the fitness function as follows:

$$Z = \max \ \text{Score}(\sigma) \tag{13}$$

In this scenario, the goal is to find the optimal value of σ to maximize the Score value.

3. Data and Variables

Predictor variables are related to the information of legal customers of one of the Iranian banks. A total of 3028 bank loans have been received by these legal customers, according to the available information, 1432 cases have been labeled as good customers and 1596 cases have been labeled as bad customers by bank experts.

In this study, 34 predictor variables have been used, 6 of which are qualitative variables (x1, x30, x31, x32, x33, x34) and the other 28 variables are quantitative, most of which are financial ratios (Table 1). The descriptive statistics of the quantitative variables are shown in Table 2.

4. Results and analysis

In this research, we first normalize the predictor variables using the min-max method. Then we classify the data using the 10-fold cross-validation.

5. Sensitivity Analysis

First, it is necessary to examine the effect of smoothing parameter (σ) on the classification performance of the PNN by considering the performance criteria. For this purpose, we changed σ between 0 and 1 and obtained the sensitivity of this parameter on the performance criteria. Fig. 2 shows the results of this sensitivity analysis. According to Fig. 2, the performance criteria take different values for different σ values. This result shows that deciding on the appropriate value of σ requires compromises between these criteria and this is the reason for using two different scenarios to get its value.

GA-PSO-PNN Algorithm

The parameters of GA-PSO algorithm are given in Table 3. Also, the convergence diagram of the algorithm for proposed scenarios illustrated in Fig. 3 shows that the algorithm has reached convergence with fewer iterations. The optimal value of σ and the fitness for these two scenarios are shown in the Table 4. In order to evaluate the performance of these two scenarios, we perform the classification using the PNN model with optimal σ values.

Table 5 and Table 6 show the results of performance of the optimal PNN classification method for the first and second scenarios, respectively. According to these tables, except for ACC, the type II error, and the total error values, the second scenario performs better than the first scenario. Of course, it should be noted that the difference in ACC values of these two scenarios is not large, and therefore in general we can say that the second scenario has a better performance and is selected as the superior scenario.

6. Comparison of different scoring models

Comparing different models where the same 10-folds, KNN (K = 10), AdaBoost (10 learning cycle), SVM (RBF kernel function), Logit regression, DT, feed-forward ANN (10 hidden layer), DECSP (With DT, AdaBoost and ANN as three classifiers with totally 165 classifiers), and PNN ($\sigma = 0.0017$), are used in this study. The average performance of each model is given in Table 7. The ROC diagram of the classification methods is also shown in Fig. 4. The results show that in terms of precision criterion, GA-PSO-PNN method has better performance with the value of 94.2%. Among them, the DECSP method is in the second place (86.4%) and the decision tree method is in the third place (84.1%). Also in terms of this criterion, the Logit method has the worst performance with the value of 55.9%. Moreover, GA-PSO-PNN has a very low type I error. But its type II error is relatively high, roughly equal to that of the SVM method.

In terms of ACC, the DECSP outperforms the others, with a slight difference, DT is in the second place and also GA-PSO-PNN is in the third place with ACC equal 80.1%.

In terms of F-score, DECSP performed better than others, DT is in the second place, and GA-PSO-PNN is in the third place.

The score criterion is used to rank classification methods. As shown in Table 7, GA-PSO-PNN has a higher score than the other methods and is in the first place, also DECSP method is in the second place and DT is in the third place.

Fig. 5 and Fig. 6 show a comparison of the performance criteria of the studied methods. In general, it can be concluded that the GA-PSO-PNN method is superior to other studied methods in two ways: First, this algorithm is an optimization method and its classification parameters are obtained through optimization, and this is a major advantage over other methods. On the other hand, the structure of the studied neural network (e.g. PNN) is probabilistic and, like other methods except DECSP, is not definite and its classification is done by considering the probabilistic layers. Therefore, this method can be well used for credit scoring of legal customers of banks in order to grant banking facilities and greatly reduce the bank’s credit risk.

Moreover, the results of how customers are categorized by different techniques are shown statistically in Table 8. For example, in this table, there was a customer labeled 1 (bad) that was not properly categorized by any algorithm. There were also 29 customers labeled 1, which were properly categorized by all techniques. Among the customers with label 0 (good), there were 2 customers who were not properly categorized by any of the techniques, and 15 customers were properly categorized by all techniques.

The Friedman test was used to compare the classifiers. The average ranks, Chi-square values, and asymptotic significance calculated are shown in Table 9. If the asymptotic significance does not exceed the significance level equal to 0.05, we can conclude that there is a significant statistical difference in the results of the classifiers. The comparison shows that the proposed hybrid GA-PSO-PNN algorithm ranks first in terms of Score.

To obtain the probability of categorizing a customer with the 8 studied methods, we consider criteria Precision, ACC, Recall, F-Score, TE, Type I error, and Type II error. Given that the classification performance of the different methods is not the same, therefore, we consider the Score value of each method as a weight. Multiplying this weight by the values of each of the 7 criteria mentioned, and then dividing the result by its maximum value, we normalize the values of the criteria as follows:

$$A_{i,j} = \left(C_{i,j} \times \mathrm{Score}_i\right) / \max\left(C_{i,j} \times \mathrm{Score}_i\right) \tag{14}$$

where $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, m$ are the number of methods and criteria, respectively (here $n = 8$ and $m = 7$), $C_{i,j}$ is the value of the *j*th criterion for the *i*th method, and $A_{i,j}$ is its normalized value. The probability of categorizing a customer with these 8 methods for each of the criteria can be calculated as follows:

$$P_j = \frac{\sum_{i=1}^{n} W_{i,j}}{\sum_{i=1}^{n} C_{i,j}}, \qquad j = 1, 2, \ldots, m \tag{15}$$

where

$$\begin{cases} W_{i,j} = C_{i,j} \times \mathrm{Label}_i & \text{if Actual Class} = 1 \\ W_{i,j} = C_{i,j} \times \left(1 - \mathrm{Label}_i\right) & \text{otherwise} \end{cases}$$

Label_i in the above equation means a customer class for the *i*th method. Therefore, the total probability of categorizing a customer with *n* methods and *m* criteria is as follows.

$$P = \frac{1}{m} \sum_{j=1}^{m} P_j \tag{16}$$

Consider Table 10, which shows the classification of two good and two bad customers using different methods. The results of probability evaluation for these 4 customers are shown in Table 11. According to this table, it is clear that different probabilities are obtained for different criteria. The overall probability of a customer's class can also be considered as the final decision criterion for labeling a customer as a good or bad customer.

7. Conclusions

The presence of credit scoring agencies in developed countries helps to reduce the credit risk of banks in these countries through proper credit scoring using various machine learning and artificial intelligence techniques. However predominantly poor and developing countries banking system suffers from a lack of reputable agencies for credit scoring of costumers. For this reason, their banks tend to internal scoring according to the Basel II & III and the regulations of the Central Bank.

Given that different learning machine techniques show some of the available facts according to the evaluation criteria, providing a combined model of these techniques can be a great help in making the right decision to credit a customer. For this purpose, in this research, 8 different techniques were used to conduct credit scoring of legal customers of an Iranian bank. The objects of using these 8 different techniques are: 1) to compare the performance of the techniques, and 2) to determine the category of customers (good or bad) of the test data for these techniques, so that banking system specially, weak banking systems, can make the right decision in credit scoring of their customers by using these techniques. Moreover, in this paper an optimized PNN algorithm with GA-PSO algorithm is presented and its performance is compared with 7 other methods (KNN, SVM, DT, Logit, AdaBoost, ANN, and DECSP).

Through sensitivity analysis, it was shown that the smoothing parameter of PNN algorithm has a significant effect on the classification performance of this algorithm. Therefore, to accurately evaluate the value of this parameter, the optimization process with GA-PSO hybrid algorithm was used. The reason for using this hybrid algorithm is its high accuracy and convergence speed, which enables it to converge to the global optimal point with a suitable convergence speed. For the optimization process, two different scenarios were used: 1) in the first scenario, the goal is to maximize the ACC value, and 2) in the second scenario, the goal is to maximize the Score function. The results of GA-PSO-PNN optimization based on these two scenarios showed that the performance of the second scenario is much better than the first scenario according to the classification performance criteria and therefore this scenario was selected as the superior scenario. The value of σ in this scenario is equal to 0.0017. The results of this study showed that the GA-PSO-PNN algorithm has a very high efficiency in credit scoring of bank customers.

In general, the results of this study showed that the banking system, in terms of a comparative approach, can take into account the results of Fig. 5 and Fig. 6 to decide on the credit scoring of its customers. Finally, considering all the results of this research, it can provide a combined interpretive model for the banking system, especially in poor and developing countries that suffer from a lack of credit rating agencies, for correct credit scoring of customers and evaluating the probability of customer default with high accuracy.

In future work, it is recommended to reduce the predictor variables using PCA reduction methods based on metaheuristic algorithms and analyze the performance of the proposed method according to this dimensional reduction. A hybrid fitness function also can be used to reduce the total error criterion (TE).

References

1. Kruck, A. "The Regulatory Use of Credit Ratings in Germany and the US: A Resource Dependence View on the Transfer of (Quasi-) Regulatory Authority" , *German Policy Studies/Politikfeldanalyse*, **9**(1), (2013).
2. Yalta, A. T., & Yalta, A. Y. "Are credit rating agencies regionally biased?" , *Economic Systems*, **42**(4), 682-694 (2018).
3. Li, F.-C. "The hybrid credit scoring strategies based on knn classifier." , *Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, **1**, IEEE. 330–334 (2009).
4. Zhou, H., Wang, J., Wu, J., et al. "Application of the hybrid SVM-KNN model for credit scoring." , *Ninth International Conference on Computational Intelligence and Security*, 174–177. IEEE (2013).
5. Abdelmoula, A. K." Bank Credit Risk Analysis with K-Nearest-Neighbor Classifier: Case of Tunisian Banks." , *Journal of Accounting and Management Information Systems*, **14**(1), 79–106 (2015).
6. Mukid, M. A., Widiharih, T., Rusgiyono, A., et al. "Credit scoring analysis using

weighted k nearest neighbor. " , *Journal of Physics: Conference Series*, **1025**, 12114 (2018).

7. Pratiwi, H., Mukid, M. A., Hoyyi, A., et al . "Credit scoring analysis using pseudo nearest neighbor." , *Journal of Physics: Conference Series*, **1217**(1), 12100. IOP Publishing (2019).
8. Bellotti, T., & Crook, J. "Support vector machines for credit scoring and discovery of significant features." , *Expert Systems with Applications*, **36**(2 PART 2), 3302–3308 (2009). https://doi.org/10.1016/j.eswa.2008.01.005
9. Shi, J., Zhang, S., & Qiu, L. "Credit scoring by feature-weighted support vector machines." , *Journal of Zhejiang University SCIENCE C*, **14**(3), 197–204 (2013).
10. Yang, S., & Zhang, Z. "A novel fuzzy SVM based on fuzzy c-means for credit scoring." ,*2nd IEEE International Conference on Computer and Communications (ICCC)*, 1349–1353. IEEE (2016).
11. Goh, R. Y., & Lee, L. S. "Credit Scoring: A Review on Support Vector Machines and Metaheuristic Approaches." , *Advances in Operations Research*, (2019). https://doi.org/10.1155/2019/1974794
12. Joanes, D. N. "Reject inference applied to logistic regression for credit scoring." , *IMA Journal of Management Mathematics*, **5**(1), 35–43 (1993). https://doi.org/10.1093/imaman/5.1.35
13. Bolton, C. "Logistic regression and its application in credit scoring." , *University of Pretoria* (2010).
14. Bahnsen, A. C., Aouada, D., & Ottersten, B. "Example-dependent cost-sensitive logistic regression for credit scoring." *Proceedings - 13th International Conference on Machine Learning and Applications, ICMLA*, 263–269 (2014). https://doi.org/10.1109/ICMLA.2014.48
15. Sohn, S. Y., Kim, D. H., & Yoon, J. H. "Technology credit scoring model with fuzzy logistic regression." *Applied Soft Computing Journal*, **43**, 150–158 (2016). https://doi.org/10.1016/j.asoc.2016.02.025
16. Dumitrescu, E., Hue, S., Hurlin, C, et al. " Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects." *European Journal of Operational Research*, **297**(3), 1178-1192 (2022).
17. Shen, F., Yang, Z., Zhao, X., et al. "Reject inference in credit scoring using a three-way decision and safe semi-supervised support vector machine." *Information Sciences* (2022).
18. West, D. " Neural network credit scoring models." *Computers & Operations Research*, **27**(11–12), 1131–1152 (2000).
19. Marcano-Cedeno, A., Marin-De-La-Barcena, A., Jiménez-Trillo, J., et al. " Artificial metaplasticity neural network applied to credit scoring." *International Journal of Neural Systems*, **21**(04), 311–317 (2011).
20. Tsai, C.-F., & Hung, C. "Modeling credit scoring using neural network ensembles." *Kybernetes* (2014).
21. Zhao, Z., Xu, S., Kang, B. H., et al. "Investigation and improvement of multi-layer perceptron neural networks for credit scoring." *Expert Systems with Applications*, **42**(7), 3508–3516 (2015).
22. AghaeiRad, A., Chen, N., & Ribeiro, B. "Improve credit scoring using transfer of learned knowledge from self-organizing map." *Neural Computing and Applications*, **28**(6), 1329–1342 (2017). https://doi.org/10.1007/s00521-016-2567-2
23. Ayouche, S., Aboulaich, R., & Ellaia, R. "Partnership credit scoring classification probem: A neural network approach." *International Journal of Applied Engineering Research*, **12**(5), 693–704 (2017).
24. Munkhdalai, L., Wang, L., Park, H. W., et al. "Advanced Neural Network Approach, Its Explanation with LIME for Credit Scoring Application." *Asian Conference on Intelligent Information and Database Systems*, 407–419. Springer (2019).
25. Munkhdalai, L., Lee, J. Y., & Ryu, K. H. "A Hybrid Credit Scoring Model Using Neural Networks and Logistic Regression." In *Advances in Intelligent Information Hiding and Multimedia Signal Processing* (pp. 251–258). Springer (2020).

26. Fonseca, D. P., Wanke, P. F., & Correa, H. L. “A two-stage fuzzy neural approach for credit risk assessment in a Brazilian credit card company.” *Applied Soft Computing*, **92**, 106329 (2020).

27. Kim, S. Y., & Upneja, A. “Predicting restaurant financial distress using decision tree and AdaBoosted decision tree models.” *Economic Modelling*, **36**, 354–362 (2014).

28. Xia, Y., Liu, C., Li, Y., & Liu, N. “A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring.” *Expert Systems with Applications*, **78**, 225–241(2017).

29. Zhang, D., Zhou, X., Leung, S. C. H., et al. “Vertical bagging decision trees model for credit scoring.” *Expert Systems with Applications*, **37**(12), 7838–7843 (2010).

30. Chen, X., Zhou, C., Wang, X., et al. “The Credit Scoring Model Based on Logistic-BP-AdaBoost Algorithm and its Application in P2P Credit Platform.” 119–130 (2017). https://doi.org/10.1007/978-981-10-2920-2_11

31. Shen, F., Zhao, X., Lan, D., et al. “A Hybrid Model of AdaBoost and Back-Propagation Neural Network for Credit Scoring.” *International Conference on Management Science and Engineering Management*, 78–90. Springer (2017).

32. Damrongsakmethee, T., & Neagoe, V. E. “C4.5 Decision Tree Enhanced with AdaBoost Versus Multilayer Perceptron for Credit Scoring Modeling.” *Advances in Intelligent Systems and Computing*, **1047**, 216–226 (2019). https://doi.org/10.1007/978-3-030-31362-3_22

33. Ozgur, O., Karagol, E. T., & Ozbugday, F. C. “Machine learning approach to drivers of bank lending: evidence from an emerging economy.” *Financial Innovation*, **7**(1), 1-29 (2021).

34. Liu, W., Fan, H., & Xia, M. “Credit scoring based on tree-enhanced gradient boosting decision trees.” *Expert Systems with Applications*, **189**, 116034 (2022).

35. Feng, X., Xiao, Z., Zhong, B., et al. “Dynamic ensemble classification for credit scoring using soft probability.” *Applied Soft Computing Journal*, **65**, 139–151 (2018). https://doi.org/10.1016/j.asoc.2018.01.021

36. Chacko, A., Antonidoss, A., & Sebastain, A. “Optimized algorithm for credit scoring.” *International Journal of Advanced Trends in Computer Science and Engineering*, **9**(1.3 Special Issue), 361–365 (2020). https://doi.org/10.30534/ijatcse/2020/5691.32020

37. Jadhav, S., He, H., & Jenkins, K. “Information gain directed genetic algorithm wrapper feature selection for credit rating.” *Applied Soft Computing Journal*, **69**, 541–553 (2018). https://doi.org/10.1016/j.asoc.2018.04.033

38. Pławiak, P., Abdar, M., Pławiak, J., et al. “DGHNL: A new deep genetic hierarchical network of learners for prediction of credit scoring.” *Information Sciences*, **516**, 401–418 (2020).

39. Zhang R, Qiu Z (2020) “Optimizing hyper-parameters of neural networks with swarm intelligence: A novel framework for credit scoring.” *PLoS ONE* **15**(6): e0234254. https://doi.org/10.1371/journal.pone.0234254 (2020).

40. Lubis, H., Sirait, P., & Halim, A. “KNN Method on Credit Risk Classification with Binary Particle Swarm Optimization Based Feature Selection.” *INFOKUM*, **9**(2, June), 211-218 (2021).

41. Carta, S., Ferreira, A., Recupero, D. R., et al. “A combined entropy-based approach for a proactive credit scoring.” *Engineering Applications of Artificial Intelligence*, **87**, 103292 (2020).

42. Zhang, W., Yang, D., & Zhang, S. “A new hybrid ensemble model with voting-based outlier detection and balanced sampling for credit scoring.” *Expert Systems with Applications*, **174**, 114744 (2021).

43. Abedin, M. Z., Guotai, C., Colombage, S., et al. “Credit default prediction using a support vector machine and a probabilistic neural network.” *Journal of Credit Risk*, **14**(2), 1–27 (2018). https://doi.org/10.21314/JCR.2017.233

44. Abdou, H. A., Mitra, S., Fry, J., et al. “Would two-stage scoring models alleviate bank exposure to bad debt? ” *Expert Systems with Applications*, **128**, 1–13 (2019). https://doi.org/10.1016/j.eswa.2019.03.028

45. Specht, D. F. “Probabilistic neural networks.” *Neural Networks*, **3**(1), 109–118 (1990).
46. Mohebali, B., Tahmassebi, A., Meyer-Baese, A., & Gandomi, A. H. “Probabilistic neural networks: a brief overview of theory, implementation, and application.” *In Handbook of Probabilistic Models* (pp. 347–367). Elsevier (2020).
47. Dasgupta, D., & Michalewicz, Z. “Evolutionary algorithms—an overview.” *In Evolutionary Algorithms in Engineering Applications* (pp. 3–28). Springer (1997).
48. Sloss, A. N., & Gustafson, S. 2019 Dasgupta, D., & Michalewicz, Z. “Evolutionary algorithms—an overview.” *In Evolutionary Algorithms in Engineering Applications* (pp. 3–28). Springer (1997).
49. Evolutionary algorithms review. Dasgupta, D., & Michalewicz, Z. “Evolutionary algorithms—an overview.” *In Evolutionary Algorithms in Engineering Applications* (pp. 3–28). Springer (1997).
50. In Genetic Programming Theory and Practice XVII (pp. 307–344). Springer (2020).
51. Yu, X., & Gen, M. “Introduction to evolutionary algorithms.” *Springer Science & Business Media* (2010).
52. Goldberg, D. E. “Genetic algorithms.” *Pearson Education India* (2006).
53. Kennedy, J., & Eberhart, R. “Particle swarm optimization.” *Proceedings of ICNN’95-International Conference on Neural Networks*, **4**, 1942–1948. IEEE (1995).
54. Wang, D., Tan, D., & Liu, L. “Particle swarm optimization algorithm: an overview.” *Soft Computing*, **22**(2), 387–408 (2018).
55. Premalatha, K., & Natarajan, A. M. “Hybrid PSO and GA for global maximization.” *Int. J. Open Problems Compt. Math*, **2**(4), 597–608 (2009).
56. Yu, M., Zhang, Y., Chen, K., et al. “Integration of process planning and scheduling using a hybrid GA/PSO algorithm.” *The International Journal of Advanced Manufacturing Technology*, **78**(1–4), 583–592 (2015).
57. Manasrah, A. M., & Ba Ali, H. “Workflow scheduling using hybrid GA-PSO algorithm in cloud computing.” *Wireless Communications and Mobile Computing*, (2018).

List of Tables:

List of Figures:

Fig. 6. Comparison of Precision, type I error, type II error, and total error criteria of the studied methods.

Tables:

Table 1 Predictor variables.

x1: Loan Status (1 = Settled, 0.8 = deferred, 0.6 = past due, 0.4 = extended, 0.2 = respited)	x18: Working Capital to Current Liabilities Ratio
x2: Loan Amount	x19: Accounts Receivable to Net Sales Ratio
x3: Cash Balance in the Bank	x20: Accounts Receivable to Liabilities Ratio
x4: Debt to Asset Ratio	x21: Accounts Payable to Net Sales
x5: Equity to Assets Ratio	x22: Sales to Assets Ratio
x6: Long-term Debt to Total Assets Ratio	x23: Sales to Fixed Assets Ratio
x7: Net Profit to Financial Cost Ratio	x24: Net Profit to Assets Ratio
x8: Current Ratio	x25: Net Profit to Net Sales Ratio
x9: Cash Ratio	x26: Net Profit to Fixed Assets Ratio
x10: Working Capital Ratio	x27: Net Profit to Equity Ratio
x11: Current Assets to Total Liabilities Ratio	x28: Cost to Net Sales Ratio
x12: Current Debt to Total Assets Ratio	x29: Firm Size
x13: Cash to Total Assets Ratio	x30: Property Ownership Status (0 = lease, 0.5 = with non-collateral ownership document, 1 = goodwill)
x14: Cash to Net Sales Ratio	x31: Relationship Between Activity and License (0 = Unrelated, 1= Related)
x15: Working Capital to Net Sales Ratio	x32: Type of collateral (1 = property (real estate), 0.75 = long-term deposits, 0.5 = bonds, 0.25 = cashier's check and more)
x16: Current Assets to Net Sales Ratio	x33: Repayment Period (years)
x17: Cash to Current Debt Ratio	x34: Interest Rate

Table 2 The descriptive statistics of the quantitative variables.

Variable	**Min**	**Max**	**Mean**	**std**	**skewness**	**kurtosis**
x2	2150000.00	45402746410.00	948656751.38	2704257619.61	7.90	90.17
x3	0.00	1867893000000.00	23078475008.25	116216413084.08	5.95	45.44
x4	0.00	6.74	0.83	0.43	4.36	47.64
x5	-5.74	1.55	0.16	0.41	-5.36	62.50
x6	-0.08	0.96	0.04	0.12	3.95	19.32
x7	-36291.25	1202.75	-40.10	788.18	-35.23	1513.54
x8	0.00	156.48	2.19	9.25	13.21	202.07
x9	0.00	24.85	0.60	1.91	9.88	109.31
x10	0.00	156.48	1.99	8.86	14.31	233.33
x11	-5.81	1.00	0.01	0.47	-3.19	30.25
x12	0.00	6.74	0.79	0.44	4.11	44.01
x13	0.00	1.00	0.10	0.13	2.79	13.62
x14	-0.28	373.57	0.50	10.40	33.32	1152.09
x15	-578.57	143.28	-0.98	19.23	-26.86	788.98
x16	-578.57	143.28	-0.98	19.23	-26.86	788.98
x17	0.00	24.85	0.25	1.01	11.12	173.12
x18	-1.00	155.48	1.20	9.24	13.21	202.19
x19	-5.41	140.91	0.34	2.96	39.23	1762.75
x20	0.00	16.72	0.38	1.29	11.16	138.42
x21	-0.95	47.57	0.30	1.87	15.81	299.18
x22	-0.27	626.59	1.78	11.85	48.93	2565.37
x23	-327.87	95922.44	351.47	3830.43	21.38	517.23
x24	-1.19	16.67	0.40	1.73	7.75	65.44
x25	-0.71	8.73	0.18	0.33	7.38	159.89
x26	-327.87	1792.53	19.26	117.95	9.34	106.19
x27	-122.48	5923.68	9.40	148.00	28.96	985.82
x28	-413.94	29.83	-0.18	7.58	-53.90	2943.56
x29	13.49	31.90	22.86	2.58	0.88	6.50

Table 3 GA-PSO algorithm parameters

GA		PSO	
Parameter	Value	Parameter	Value
Population size	10	Inertia weight	1
Crossover (%)	70	Inertia weight damping ratio	0.99
Mutation (%)	30	c_1	1.5
Mutation rate	0.1	c_2	2
Max Iter	30		

Table 4 Optimal values of σ and finesse for the two proposed scenarios.

Scenario	**Parameters**	**Value**
1	σ	0.0228
	ACC	0.8444
2	σ	0.0017
	Score	1.6846

Table 5 Performance of the optimal PNN classification for the first scenario ($\sigma = 0.0228$).

Fold	**Precision**	**ACC**	**Recall**	**F-Score**	**TE**	**E-Type I**	**E-Type II**	**Score**
1	0.860	0.848	0.826	0.842	0.152	0.131	0.174	1.570
2	0.825	0.855	0.861	0.843	0.145	0.151	0.139	1.531
3	0.825	0.835	0.825	0.825	0.165	0.156	0.175	1.501
4	0.853	0.848	0.797	0.824	0.172	0.140	0.203	1.531
5	0.769	0.825	0.791	0.780	0.205	0.201	0.209	1.362
6	0.769	0.845	0.827	0.797	0.185	0.194	0.173	1.394
7	0.825	0.845	0.843	0.834	0.155	0.153	0.157	1.516
8	0.750	0.859	0.812	0.780	0.201	0.212	0.188	1.341
9	0.792	0.842	0.826	0.809	0.178	0.182	0.174	1.433
10	0.832	0.838	0.826	0.829	0.162	0.152	0.174	1.514
Mean	**0.810**	**0.844**	**0.824**	**0.816**	**0.172**	**0.167**	**0.176**	**1.469**

Table 6 Performance of the optimal PNN classification for the second scenario ($\sigma = 0.0017$).

Fold	**Precision**	**ACC**	**Recall**	**F-Score**	**TE**	**E-Type I**	**E-Type II**	**Score**
1	0.930	0.788	0.711	0.806	0.212	0.087	0.289	1.656
2	0.930	0.851	0.792	0.855	0.149	0.074	0.208	1.685
3	0.958	0.809	0.725	0.825	0.191	0.053	0.275	1.684
4	0.958	0.818	0.737	0.833	0.182	0.051	0.263	1.693
5	0.930	0.769	0.689	0.792	0.231	0.091	0.311	1.692
6	0.923	0.769	0.691	0.790	0.231	0.098	0.309	1.589
7	0.965	0.832	0.750	0.844	0.168	0.042	0.250	1.742
8	0.944	0.802	0.723	0.819	0.198	0.070	0.277	1.696
9	0.965	0.848	0.772	0.858	0.152	0.041	0.228	1.742
10	0.916	0.798	0.728	0.811	0.202	0.098	0.272	1.659
Mean	**0.942**	**0.808**	**0.732**	**0.823**	**0.192**	**0.070**	**0.268**	**1.685**

Table 7 Comparison of the average performance of all methods.

Method	Precision	ACC	Recall	F-Score	TE	E-Type I	E-Type II	Score
KNN	0.691	0.775	0.806	0.744	0.225	0.245	0.194	1.231
Adaboost	0.630	0.676	0.669	0.647	0.324	0.316	0.331	0.988
D-Tree	0.841	0.852	0.845	0.843	0.148	0.142	0.155	**1.548**
Logit	0.559	0.645	0.645	0.598	0.355	0.354	0.355	0.850
SVM	0.634	0.703	0.708	0.668	0.297	0.300	0.292	1.038
ANN	0.716	0.750	0.746	0.729	0.250	0.245	0.254	1.219
GA-PSO-PNN	0.942	0.808	0.732	0.823	0.192	0.070	0.268	**1.685**
DECSP	0.864	0.868	0.890	0.876	0.132	0.156	0.110	**1.583**

Table 8 Statistics on how to categorize the bank's legal costumers with 8 different techniques**.**

No. of True Labels	**No. of Algorithms**	**Actual Label (Class)**	**No. of True Labels**	**No. of Algorithms**	**Actual Label (Class)**
2	0		1	0	
5	1		9	1	
8	2		11	2	
8	3		13	3	
16	4	0	22	4	1
23	5		24	5	
27	6		23	6	
31	7		35	7	
18	8		29	8	

Table 9 Comparison of classification results by the Friedman test.

Method	Average Rank
KNN	3.97
Adaboost	3.18
D-Tree	6.02
Logit	1.73
SVM	2.48
ANN	5.51
GA-PSO-PNN	7.58
DECSP	6.39
χ^2	36.72
Asymptotic significance	0.016

Table 10 Classification of good and bad customers according to different methods.

Method	**Customer1 Class for Actual Label 1**	**Customer 2 Class for Actual Label 1**	**Customer 3 Class for Actual Label 0**	**Customer 4 Class for Actual Label 0**
KNN	1	1	1	0
AdaBoost	1	1	1	1
D-Tree	0	1	0	0
Logit	0	0	1	1
SVM	0	1	1	0
ANN	1	1	0	0
GA-PSO-PNN	1	1	0	0
DECSP	1	1	0	0

Table 11 Probability of correctness of each customer's class (%).

Probability	**Precision**	**ACC**	**Recall**	**F-Score**	**TE**	**E-Type I**	**E-Type II**	**Total**
Costumer1: Class 1	68.52	66.99	66.76	67.55	63.09	60.55	63.99	66.97
Costumer2: Class 1	93.86	93.03	92.97	93.42	86.73	85.74	87.14	90.41
Costumer3: Class 0	66.31	63.15	62.34	64.31	46.90	41.87	50.10	56.43
Costumer4: Class 0	85.81	84.54	84.49	85.14	72.65	70.94	73.20	79.54

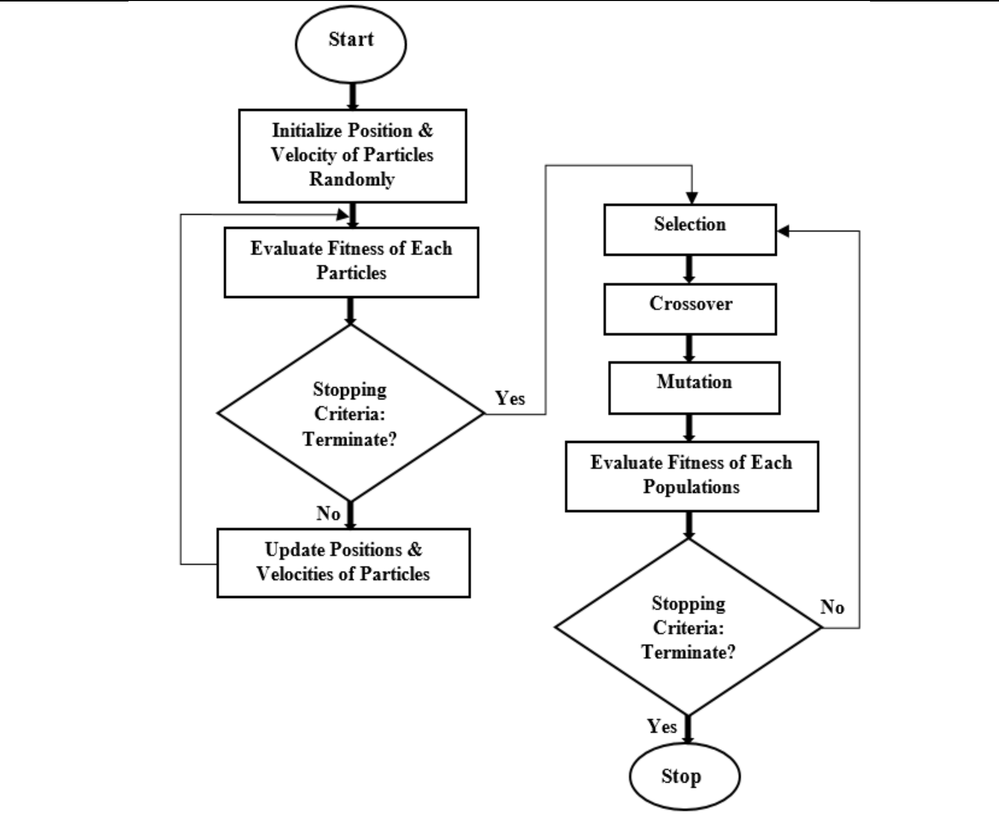


Fig 1. Flowchart of GA-PSO hybrid algorithm.

Figures:

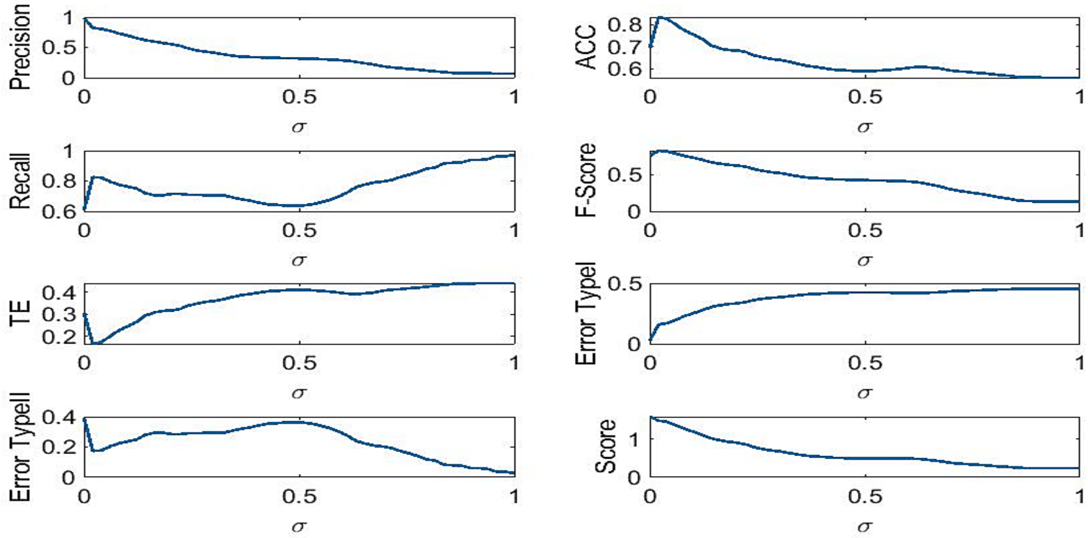
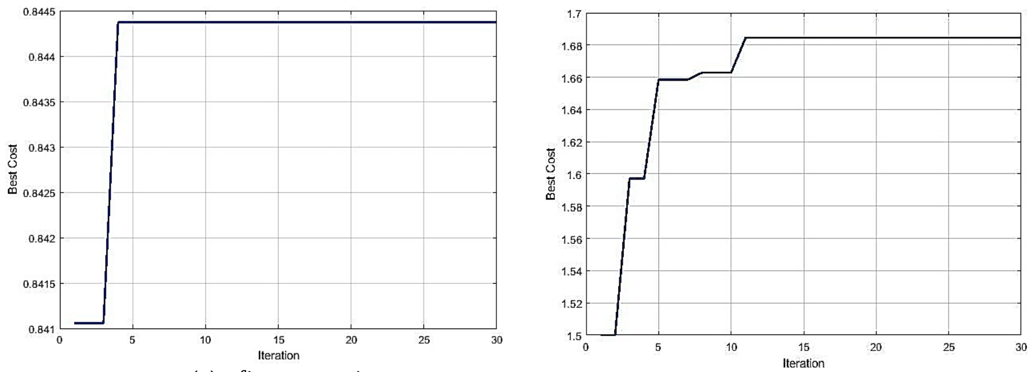


Fig. 2. Sensitivity of classification performance criteria of PNN method to σ .



(a) first scenario

(b) second scenario

Fig. 3. Convergence diagram of the proposed scenarios.

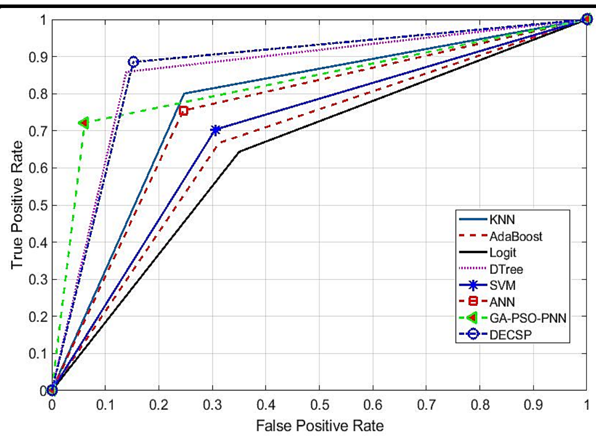


Fig 4. Comparison of ROC diagram of the proposed method with other classification methods.

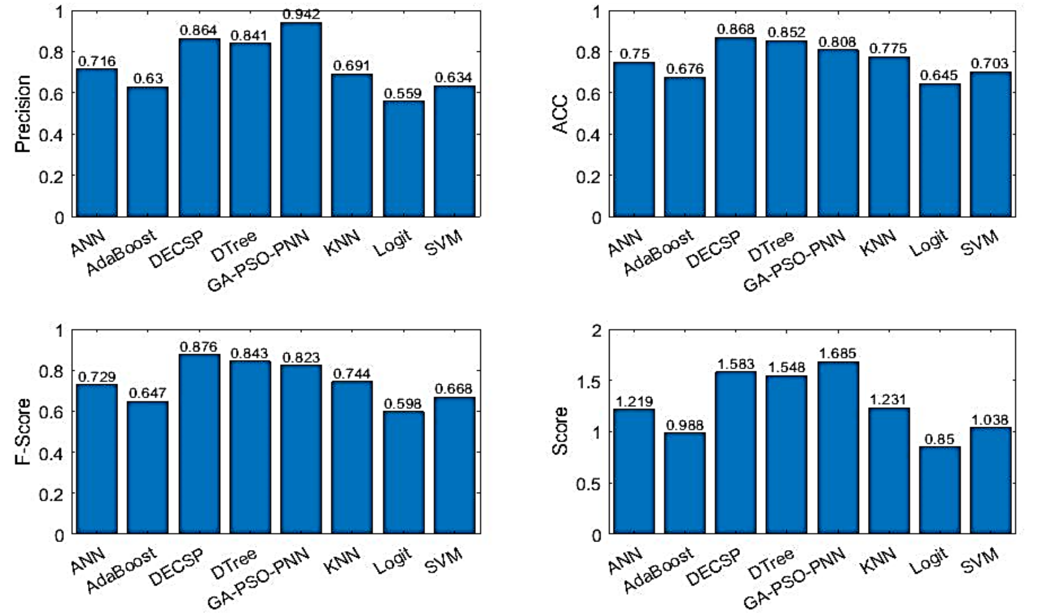


Fig. 5. Comparison of Precision, ACC, F-Score, and Score criteria of the studied methods.

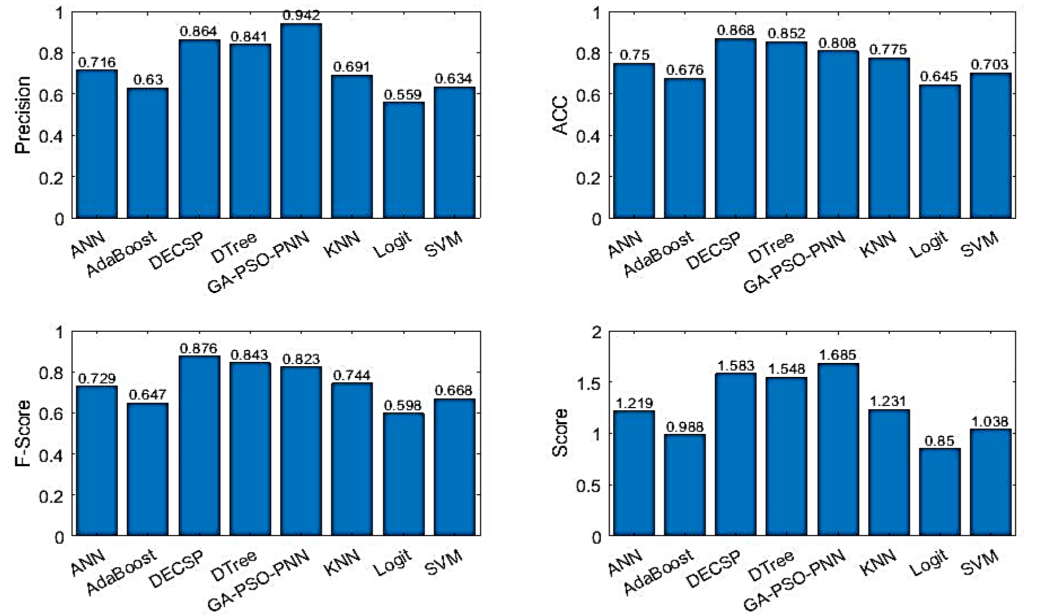


Fig. 6. Comparison of Precision, type I error, type II error, and total error criteria of the studied methods.

Biographies

Mohammad Ali Rastegar is a Faculty member of Industrial Engineering and Systems at Tarbiat Modares University, Tehran, Iran. His research interests include Financial and Banking risks modelling, Algorithmic Trading, Asset-liabilities management, agent-based simulation, FSCM and Investment management. He has published papers in some reputable journals such as Handbook of Power Systems, Solar Energy, Expert Systems with Applications, etc.

Mitra Faraji is an MSc graduate student from Department of Finance & Risk Management, Khatam University, Tehran, Iran. Her research interests include Credit Scoring, Banking and Machine Learning.