Journal of Biomedical Informatics 115 (2021) 103690

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

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

Original Research

Random forest swarm optimization-based for heart diseases diagnosis

Shahrokh Asadi [a,*], SeyedEhsan Roshan [a], Michael W. Kattan [b]

[a] Data Mining Laboratory, Department of Engineering, College of Farabi, University of Tehran, Tehran, Iran
[b] Department of Quantitative Health Sciences, Cleveland Cancer Foundation, Cleveland, OH, United States

ARTICLE INFO

Keywords:
Data mining
Ensemble learning
Random forest
Diversity
Heart disease

ABSTRACT

Heart disease has been one of the leading causes of death worldwide in recent years. Among diagnostic methods for heart disease, angiography is one of the most common methods, but it is costly and has side effects. Given the difficulty of heart disease prediction, data mining can play an important role in predicting heart disease accurately. In this paper, by combining the multi-objective particle swarm optimization (MOPSO) and Random Forest, a new approach is proposed to predict heart disease. The main goal is to produce diverse and accurate decision trees and determine the (near) optimal number of them simultaneously. In this method, an evolutionary multi-objective approach is used instead of employing a commonly used approach, i.e., bootstrap, feature selection in the Random Forest, and random number selection of training sets. By doing so, different training sets with different samples and features for training each tree are generated. Also, the obtained solutions in Pareto-optimal fronts determine the required number of training sets to build the random forest. By doing so, the random forest's performance can be enhanced, and consequently, the prediction accuracy will be improved. The proposed method's effectiveness is investigated by comparing its performance over six heart datasets with individual and ensemble classifiers. The results suggest that the proposed method with the (near) optimal number of classifiers outperforms the random forest algorithm with different classifiers.

1. Introduction and literature review

Data mining refers to knowledge extraction from a large volume of data [1]. Data mining is one of the most popular sciences due to its appropriate performance and predictive power. It is used in different areas such as financial forecasting, health, weather forecasting, fraud detection, and many other fields [2–7]. Nowadays, a lot of medical data is being collected by health institutions, and the ability to extract knowledge from a large volume of raw data can be useful in making effective health decisions. Data mining in health is an emerging field and has great importance in providing the prognosis and deep understanding of medical data [8]. Due to the potential to extract knowledge and hidden patterns from a massive volume of raw data, researchers today employ data mining techniques in predicting diseases such as diabetes [9], cancer [10], and heart disease [11].

Cardiovascular disease is the main cause of heart and blood vessel disorders [12] and is a major contributor to death and disability. Coronary Artery Disease (CAD), also called "ischemic heart disease," is very common and one of the top 10 causes of death worldwide.

Due to the existing limitations in the diagnosing methods of heart disease and the difficulty of heart disease prediction, researchers seek for more appropriate methods. Because of its ability to extract knowledge and discover hidden patterns, data mining can be useful in predicting and diagnosing heart disease. Different efforts have been made to predict heart disease using data mining techniques with the goal of improving prediction accuracy.

Bashir et al. [13] (Bashir et al., 2015) suggested an ensemble classification method called BagMoov to predict heart disease. Their proposed method is based on bagging and multi-objective optimized weighted voting in a way that Naïve Bays (NB), linear regression (LR), quadratic discriminant analysis (QDA), instance-based learner, and support vector machine (SVM) algorithms were employed as base classifiers.

Arabasadi et al. [14] proposed an accurate combined algorithm to predict CAD. Their proposed algorithm increased the accuracy of the neural network (NN) by identifying the initial weights of NN and using a genetic algorithm (GA) up to 10%.

Jabbar [15] improved the K-nearest neighbor (KNN)'s performance to enhance heart disease prediction. In this method, in the preprocessing step, Particle Swarm Optimization (PSO) is used to select the feature,

* Corresponding author.
E-mail address: shahrokh.asadi@ut.ac.ir (S. Asadi).

https://doi.org/10.1016/j.jbi.2021.103690
Received 27 August 2020; Received in revised form 21 January 2021; Accepted 23 January 2021
Available online 1 February 2021

and then KNN is trained over the obtained dataset from the pre-processing step.

Samuel et al. [16] used an Artificial Neural Network (ANN) to predict the risk of heart failure. In their method, the fuzzy analytic hierarchy process (AHP) was used to compute global weights of features based on their individual contributions. Then, the global weights, which represent the feature contributions, are applied to train an ANN classifier to predict the risk of heart failure in patients. Yekkala et al. [17] analyzed bagging ensemble methods, random forest, and Adaboost accompanied by PSO (as feature selection to predict heart disease). The results showed that hybridizing bagging with PSO yields high accuracy.

Mustafa et al. [18] proposed an ensemble approach to combine prediction ability of multi-classifiers' prediction ability for better prediction. In their study, ensemble learning combined five classifiers, including SVM, ANN, Naïve Bays, regression analysis, and random forest, to predict and diagnose of the recurrence of cardiovascular disease.

Liu et al. [19] developed a hybridizing classifier system based on ReliefF and rough set theory (RFRS). The proposed method contains two subsystems: a RFRS feature selection system and a classification system with an ensemble classifier. The first system includes three steps: (1) discretization, (2) feature extraction using the ReliefF algorithm, and (3) feature reduction using a heuristic rough set reduction algorithm. In the second system, an ensemble classifier is presented based on C4.5.

Nguyen et al. [20] presented a medical classification model based on a fuzzy standard additive model and a genetic algorithm. In this method, rule initialization is handled by the adaptive vector quantization clustering. For rule optimization and parameter tuning, a genetic algorithm and gradient descent algorithm were applied, respectively. Wavelet transformation is employed to extract discriminative features for high-dimensional datasets.

Manogaran et al. [21] presented a predictive system through integrating Multi Kernel Learning (MKL) and Adaptive Neuro-Fuzzy Inference System (ANFIS) for heart disease prediction. The proposed system applied a two-step approach in which MKL was utilized to divide parameters between heart disease patients and healthy individuals. ANFIS then employs the results obtained from MKL to classify heart disease and healthy patients. The performance of the proposed method was compared with deep learning techniques such as Least Square with Support Vector Machine (LS with SVM), General Discriminant Analysis and Least Square Support Vector Machine (GDA with LS-SVM), Principal Component Analysis with Adaptive Neuro-Fuzzy Inference System (PCA with ANFIS) and Latent Dirichlet Allocation with Adaptive Neuro-Fuzzy Inference System (LDA with ANFIS).

Dolatabadi et al. [22] proposed a methodology for the automatic diagnosis of CAD was proposed by extracting HRV signals from ECG in time, frequency, and nonlinear domains. The method used PCA for dimensional reduction and, for classification, SVM was used.

During recent years, ensemble learning methods have received much attention among researchers owing to their potential in improving the accuracy of training and generalizing learning [23,24]. In fact, the main idea of ensemble learning is to weight multiple classifiers and combine them to obtain a unified classifier that outperforms the other classifiers [25]. It has been proved theoretically and experimentally that ensemble classifiers can perform better than individual ones [26,27].

Among the most popular ensemble learning methods, one can mention bagging [28], boosting [29], and random forest [30]. In ensemble learning methods, diversity has an essential role in improving performance [31,32]. The ensemble diversity means the difference between individual classifiers, which is a fundamental topic in ensemble methods [33,34]. In order to obtain a gain by combining different classifiers, individual classifiers should be different. Otherwise, no gain is obtained by doing so [33,35]. So, improving the ensemble performance depends on the diversity of base classifiers, provided that diversity does not reduce the accuracy of ensemble members [36,37].

There are three general ways to create diversity in ensemble methods: data diversity, parameter diversity, and structural diversity [38]. In data diversity, several datasets are created from the main dataset to train base classifiers, and this approach is used in Adaboost, bagging, and random forest ensemble methods. In parametric diversity, different parameters are set to produce diverse base classifiers, and finally, in structural diversity, several classifiers are used that can be different in size, parameter, and architecture.

Most medical datasets have large sizes [39,40], dimensional reduction methods, which somewhat reduce the complexity of datasets, can be useful in improving the classification of these problems. The random forest classifier is one of the most successful ensemble learning techniques used for large-sized classification and skewed problems [30]. The random forest is an ensemble of decision trees, and each tree returns a classification result. The purpose of random forest is to create binary subtrees using training samples generated through bootstrap [41] and to select a subset of features in each node at random. In other words, the random forest is a combination of bagging algorithm and random subspace suggested by HO [42].

Generally, the random forest method utilizes the implicit and explicit diversities together, and its performance depends on both the accuracy and diversity of classifiers [43]. In addition, studies have shown that determining the number of decision trees in a random forest is important, and adding a high number of classifiers to the random forest does not improve its accuracy [44–46]. Therefore, researches in the literature are being carried out to limit the number of classifiers and find the optimal subset of classifiers [47].

The performance of ensemble methods can be improved by enhancing the diversity and accuracy of classifiers and selecting the number of classifiers, and the random forest is no exception. However, there is a trade-off between diversity and accuracy of classifiers.

In the random forest, the selection of subsets of features, samples, and the number of decision trees are randomly selected. In this paper a random forest swarm optimization-based for heart disease diagnosis is proposed that utilized an evolutionary approach to select a subset of features and instances. In this method, considering the trade-off between diversity and accuracy, the Multi-Objective Particle Swarm Optimization (MOPSO) algorithm is used to simultaneously optimize these two goals to produce a diverse and appropriate subset of features and samples so that in addition to the accuracy, the diversity between trees can be improved. In this method, the set of optimal solutions obtained from MOPSO, which are diverse training sets, are used to train decision trees. Moreover, the obtained number of solutions determines the number of trees required to build a random forest; to put it in other words, no pruning approach is needed.

Some evolutionary approaches have been done to improve the random forest's performances. Kaur et al. [48] proposed a multi-objective differential evolution-based random forest technique. This method is able to tune the random forest parameters efficiently. Bursa et al. [49] proposed a hybrid method combining ant-inspired and evolution-inspired approach, which combines random forest approach, where trees are generated using a stochastic swarm metaheuristics inspired by natural processes. The method uses a tree-like structure (similar to classification or decision tree structure) to partition the provided dataset and assigns a classifier to each item of the dataset. Jabbar et al. [50] used a genetic algorithm and chi-square to extract the important features. Then, they applied the random forest algorithm on the remained features to improve the classification accuracy. Qi and Chen [51] employed a combination of random forest and PSO algorithms, where PSO was used to tune the parameters of random forest.

Xia-an et al. [52] proposed an evolutionary clustering approach to improve random forest performance. In this method, after constructing decision trees, decision trees with high similarity and decision trees with better performance are clustered in an evolutionary approach. Finally, the clusters with similar classifiers are removed. Adnan and Islam [53] used genetic algorithm to find the random forest algorithm's optimized number of trees.

A review of the literature shows that few evolutionary methods have

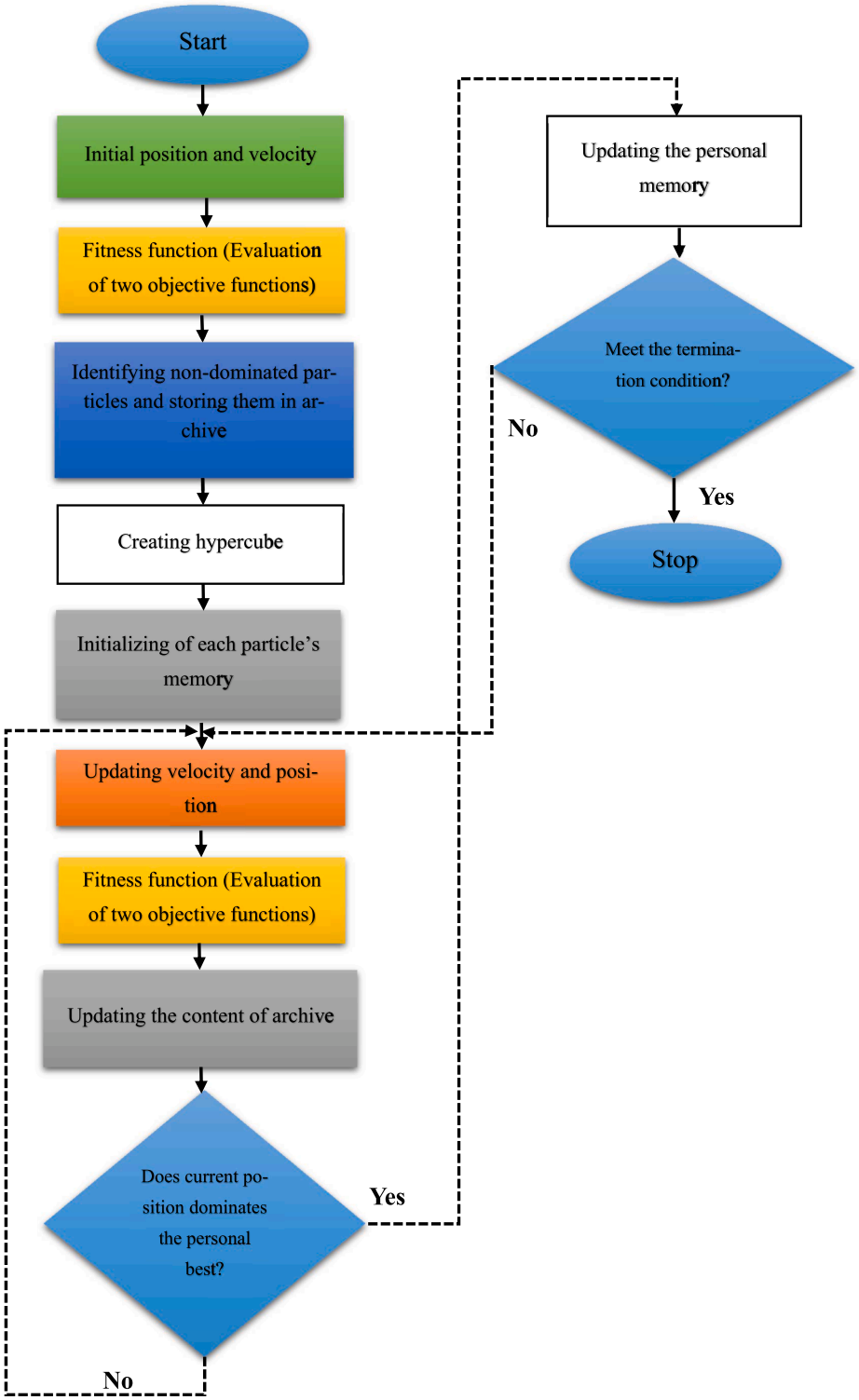


Fig. 1. The flowchart of the proposed method.

been proposed to improve the performance of random forests. To the best of our knowledge, the proposed method in this research is a new technique that uses an evolutionary approach, so that in addition to producing diverse training sets (which is carried out through producing different subsets of features and samples), it determines the (near) optimal number of decision trees.

The rest of this paper is organized as follows. Section 2 describes the proposed method. In Section 3, the conducted experiments are explained and discussed. Finally, Section 4 includes conclusions.

2. Proposed method

In this paper, the main goal is to improve the performance of the random forest algorithm through creating diverse and well-performing training sets in order to generate accurate and diverse classifiers. Since there is a trade-off between diversity and accuracy, these two criteria are simultaneously optimized to generate accurate and diverse decisions trees and determine the (near) optimal number of trees.

The proposed algorithm includes two steps. In the first step, MOPSO

Samples					features			
1	0	0	1	1	0	0	1	1

Fig. 2. A given problem with 5 samples and 4 features, in which i[th] particle shows that that samples 2, 3, and features 1 and 2 are not selected.

is employed to generate diverse training sets. The process of generating training sets is presented in the Fig. 1. In the second step, the number of generated training sets is used to train and determine the number of decision trees in the random forest. The steps are explained in the following.

2.1. The first step: MOPSO for training set generation

PSO [54] is a population-based search algorithm proposed by Kennedy and Eberhart in 1995. PSO is a simple metaheuristic algorithm that mimics swarm treatment of birds, fishes, etc. This algorithm has been widely used in the literature in different areas such as feature selection [55], medical [56], and many other optimization problems, owing to easy implementation, fewer parameters to tune, capable of parallel computing, high efficiency, and probability in finding the global optimum, fast convergence, short computational time, no overlapped computing and mutation [57]. In PSO, the particles are considered, which seek the optimal solutions through exploring the search space. Each particle has a position and velocity which position is used to evaluate the particle's quality, and velocity specifies direction and magnitude of the particle's movement.

PSO can directly be employed in single-objective problems. However, it cannot be useful in multi-objective problems in which contradictory objectives are presented. To handle such difficulty, MOPSO was developed. Similar to PSO, personal best and global best play important roles in guiding the particles and are considered as two key factors. Fig. 1 shows a flowchart of the employed MOPSO to generate optimal bags. The steps are explained as follows.

2.1.1 Initial position and velocity

In the proposed method, each particle, which is the representation of a solution (or training set), has two properties named "position" and "velocity", shown by x and v, respectively.

$$x_i = (x_{i1}, x_{i2}, \cdots, x_{iN}) \tag{1}$$

$$v_i = (v_{i1}, v_{i2}, \cdots, v_{iN}) \tag{2}$$

where N in Eqs. (1) and (2) signifies the length of each particle, which in our method consists of two portions, equals the sum of the number of samples and features, and i indicates the index of each particle. Also, x_{ij} means that whether element j exists or not. For example, a problem with 5 samples and 4 features in Fig. 2, i^{th} particle shows that samples 2, 3, and features 1 and 2 are not selected. In other words, the digits 1 and 0 respectively signify 'existence' and 'non-existence' of training samples and features.

Eq. (3) shows how one can initialize of position of each particle.

$$x_{ij} = \begin{cases} 1, if \;\; rand() < P \\ 0, if \;\; rand() \geq P \end{cases} \tag{3}$$

Where P is obtained through trial-and-error.

The initial value of velocity for each particle can be obtained using Eq. (4), where v_{max} is determined so as to keep the search space within a significant space to avoid violating the speed of particles from this value.

$$v_{ij} = -v_{max} + 2 \times rand() \times v_{max} \tag{4}$$

2.1.2 Fitness function

The fitness function measures the solution's quality. In order to select the best solution, the fitness function considers two objectives, i.e., accuracy and diversity.

$$minmize(f(s)) = min(f_1(s), f_2(s)) \tag{5}$$

The first objective function: In the proposed method, a solution (particle) is considered as a well-performing solution, if a classifier has an appropriate performance on it. Since the considered problem in this study is "minimization," classification error must be minimized. The measurement error (performance of C4.5 [58] classifier on each particle) can be computed as follows:

$$f_1(s) = Error = \frac{FP + FN}{TP + TN + FN + FP} \tag{6}$$

The second objective function: One way to develop diversity in ensemble methods is to use different training subsets and/or subsets of different features called feature selection [59]. In this method, the second objective function is used to measure the diversity of solutions. The function's value includes the degree of diversity of samples and features, where the diversity of the samples is measured using the compactness [60] criterion. The degree of diversity of the subset of features is measured using the cardinality. In the following, each one is described separately.

Compactness: In this criterion, the compactness of the training set is measured, and for more dispersion, the amount of compactness must be minimized. The Mahalanobis distance proposed by P. C. Mahalanobis [61] is used to measure the data compression, or in other words, the diversity of the generated datasets. This criterion is adopted because it does not have limitations of the Euclidean metric. The training sets may contain noise or replication. In this case, the use of Euclidean distance measurement in identifying highly correlated or repetitive data samples that do not provide any new information during the classification training will fail. The Mahalanobis criterion is preferred because of the multivariate effect size, which considers the correlation of the dataset and is therefore not dependent on scale. Covariance between data samples is also taken into account when calculating the distance. Therefore, it overcomes the intrinsic scale and correlation problems associated with measuring the Euclidean distance.

It is supposed that $G_{n\times m}$is a dataset in which m is the number of features and n is the number of points of the dataset. In order to calculate the diversity of training sets, one should first compute Mahalanobis distance of data from the average, shown by D^2, and calculated as Eq. (7).

$$D^2 = (g_i - mean)^T . C^{-1} . (g_i - mean) \tag{7}$$

In Eq. (7), D^2 is the square of Mahalanobis distance, g_i is the i^{th} row of training set G, *mean* is the average vector, and C^{-1} is the covariance inverse matrix. After calculating the distance of each point from the mean points, the inverse of the average data distance in the G is calculated according to Eq. (8). The value obtained from Eq. (8) indicates the compaction rate of the samples. Therefore, the lower this value for the dataset, the higher the data scatter.

$$compactness = \frac{1}{\left(\frac{\sum_{i=1}^{n} D_i^2}{n}\right)} (n = Number \;\; of \;\; point \in each \;\; dataset) \tag{8}$$

Cardinality: In the random forest, decision trees are trained on training sets with different features. The proposed method uses the cardinality criterion, equal to the number of features selected, and the goal is to minimize this value to reduce the complexity of the depth of the trees. Hence, we consider the inverse cardinality (C) for this criterion. Also, since the adopted approach is evolutionary, different features are selected for each training set; in other words, decision trees are

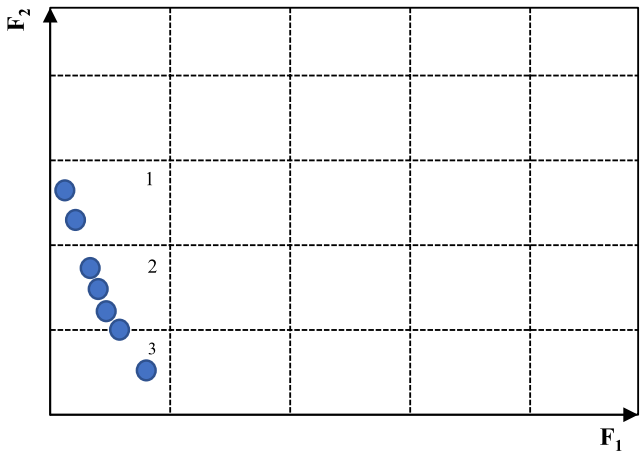


Fig. 3. Hypercube generation over archive.

trained on training sets with different features.

In general, according to the compaction and cardinality criteria, the value of the second objective function is calculated as follows:

$$f_2(s) = diversity = compactness + 1/\mathrm{C} \tag{9}$$

Identifying non-dominated particles and storing them in the archive

In multi-objective methods, there is a set of solutions that contain non-dominated particles. The x_1 particle is called non-dominated if a particle such as x_2 cannot be found to have a better fitness value in all objectives than x_1.

For example, in Fig. A1 in Appendix A, sample solutions including dominated and non-dominated ones considering two objectives F_1 and F_2 are depicted. In this figure, the goal is to minimize the values of the objective functions F_1 and F_2. According to this figure, the red circles dominate the blue circles; however, they do not dominate each other. Mathematically, the concept of domination is defined as follows [62]:

x dominates $y(x \le y)$ *if and only if* :

$$\forall i : x_i \le y_i \tag{10}$$

and

$$\exists i_0 : x_{i_0} < y_{i_0} \tag{11}$$

Eq. (10) means that y is not better than x under any circumstances, and Eq. (11) means that x is better than y at least in one aspect.

In 2002, Coello and Lechuga introduced the concept of external archiving for the information storage of non-dominated particles, and the leader is selected for each particle from this external archive.

2.1.3 Creating a hypercube

In [63], to select a leader for each particle, a hypercube is created on archived particles. By using these hypercubes, the particles are determined as a coordination system in which the coordination of each particle is defined according to the values of its objective functions. Fig. 3 presents a sample of tabulation by considering two objectives F_1 and F_2. In generating hypercubes, their number and size are arbitrary. Selecting the leader among non-dominated particles depends on a hypercube. In other words, one should first select one of the hypercubes at first, and then a particle is chosen from inside of the hypercube. In order to select a hypercube, a fitness value is assigned to each hypercube depending on the number of elite particles in which lie. The more the elite particles are in a hypercube, the less is its fitness value. The roulette wheel is employed, and one of them is selected. Once a hypercube is selected, one of its particles is randomly chosen. Generally, the selection probability of hypercubes is inverse proportional to its population. For example, the second hypercube in Fig. is not appropriate since it has high crowding.

2.1.4 Initializing of each particle's memory and updating velocity and position

The best-found position of each particle is saved in personal memory. In this step, the memory of each particle is initialized, which is equal to the initial position of that particle. Each particle's memory is equivalent to a guide for movement in search space.

PSO was originally developed for continuous problems, so velocity and position of each particle is updated according to Eqs. (12) and (13). Since the proposed method falls within discrete problems, updating of velocity and position should be carried out according to Eq. (12) and Eq. (13).

$$v_i^{t+1} = w \times v_i^t + c_1 \times r_1 \times \left(pbest[i] - x_i^t\right) + c_2 \times r_2 \times \left(leader[i] - x_i^t\right) \tag{12}$$

$$x_i^{t+1} = x_i^{t+1} + v_i^{t+1} \tag{13}$$

where, *pbest* is the personal best (the best found solution), leader is selected from archive, as referred in Section 2.1.4, c_1 and c_2 are acceleration multipliers, r_1 and r_2 are random numbers and w is inertia weight (the inertia weight is employed to control the impact of the previous history of velocities on the current velocity of a given particle [64]).

$$x_{ij} = \begin{cases} 1, if rand() < S(v_{ij}) \\ 0, if rand() \ge S(v_{ij}) \end{cases} \tag{14}$$

Where $S(.)$ represents logistic function and is employed as probability distribution for x_{ij} calculated as follows:

$$S(v_{ij}) = \frac{1}{1 + exp(v_{ij})} \tag{15}$$

2.1.5 Updating the content of archive

In this step, the content of the archive (consists of adding all current non-dominated particles into archive) and created tabulation in the previous step are updated. Then, dominated particles are removed. If the size of archive is more than a predetermined limitation, some of the particles are removed. According to [63], since the size of the repository is limited, whenever it gets full a secondary criterion is applied for retention: those particles located in less crowded areas of the objective space are given priority over those lying in highly crowded regions.

2.1.6 Updating the personal memory

When the current position of a particle is better than the saved position in memory, the personal memory will be updated. In other words, if the current position dominates the saved position in personal memory, it will replace it. If no one dominates each other, one of them is chosen randomly.

2.1.7 Termination

The stop criterion is important to terminate the evolutionary algorithms. In the proposed method, the algorithm stops when the execution number reaches the predefined value, which is determined through trial-and-error.

2.2. The second step: Using generated particles in the random forest

In the random forest, training features and samples are generated randomly and using bootstrap. Also, the number of training sets is determined randomly. Finally, these different training sets are used to train decision trees. In contrast, in the proposed method, the output of the first step, i.e., the obtained solutions, are used in the second step to train the decision trees. According to the number of training sets produced, decision trees are considered, and according to the nature of the training sets produced, i.e., diversity and well-performing, diverse, and accurate decision trees are generated.

In the end, the performance evaluation of the proposed method is carried out using "classification accuracy," "specificity," and "sensitivity" criteria over heart datasets. The classification accuracy is

Table 1
The information of used datasets.

i	Dataset	# of Attributes	# of Instance	# of Classes
1	Cleveland	22	303	binary class
2	Statlog	14	270	binary class
3	SPECT	23	267	binary class
4	SPECTF	45	267	binary class
5	VA Long Beach	14	201	binary class
6	Eric	8	209	binary class

Table 2
Tuned parameters.

Proposed Method	**MOPSO**	Existing probability of each Sample = 0.65 Number of Particles = 100, Number of divisions = 50, Personal Best Coefficient = 2, Social Best Coefficient = 2, Iteration = 100, Repository Size = 20, Particle size = training size, $v_{max} = 5$, inertia weight = 0.5
	C4.5	Prune = True, Confidence level = 0.25, Minimum number of item-sets per leaf = 2

computed using Eq. (16) and "specificity" and "sensitivity" criteria are computed as Eqs. (17) and (18), respectively.

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

$$specificity = \frac{TN}{TN + FP} \tag{17}$$

$$sensitivity = \frac{TP}{TP + FN} \tag{18}$$

Where TP, TN, FP, and FN signify True Positive, True Negative, False Positive, and False Negative, respectively. In better words, TP is those samples which are correctly predicted as positive, TN represents those samples which are correctly predicted as negative, FP refers to those instances that are wrongly predicted as positive, and FN indicates those cases which are wrongly predicted as negative.

3. Experimental results

To demonstrate the performance of the proposed algorithm, this algorithm is first implemented on six heart datasets. Then the results are compared with several state-of-the-art algorithms. In the following section, first information about the datasets and the tuned parameters, then the test results, and finally, the discussion are presented.

3.1. Datasets

In order to evaluate the proposed method's performance, six heart datasets named Statlog, Cleveland, SPECT, SPECTF, VA Long Beach, and Eric are used, which are accessible through UCI [65]. Table 1 shows the features of each dataset. In each row, details of each dataset, including name, number of features, samples, and classes, are presented. In experiments, the accuracy estimation is obtained using 10-fold cross-validation; namely, each dataset is divided into ten folds with the same number of samples. Then, for each fold, learning algorithms are trained over 9 folds and tested on one fold.

3.2. Tuning of experiments

In order to compare the proposed method's performance with other methods, six ensemble methods, i.e., BagMoov, random forest, bagging, Adaboost, stacking [66], neural network ensemble (NNE), and six individual classifiers named Naïve Bayes (NB), linear regression (LR), C4.5, Support Vector Machine, Quadratic discriminant analysis (QDA) and KNN are used.

The values of parameters may affect the algorithm's performance, and different combinations of parameters can result in different consequences. In Table 2, the related parameters with MOPSO and C4.5, which are obtained through trial-and-error, are presented. Also, in Table 2, "existing probability of each sample" which signifies the existing probability of each sample in each particle, is tested with

Table 3
The individual classifiers' performance and the proposed method over heart datasets.

	Cleveland			**Statlog**		
Method	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
KNN	81.71	71.94	77.23	68.67	61.67	65.56
SVM	93.90	65.47	80.86	94.67	65.83	81.85
NB	81.71	71.94	77.23	82.00	74.17	78.52
LR	88.41	77.7	*83.5*	87.33	76.67	*82.59*
QDA	68.29	62.59	65.68	64.00	73.33	68.15
C4.5	73.21	39.14	75.08	78.12	77.27	77.77
Proposed Method	77.40	90.80	**85.21**	89.74	86.66	**88.26**
	SPECT			**SPECTF**		
Method	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
KNN	7.27	98.11	79.40	36.36	81.13	71.91
SVM	85.45	62.26	67.04	0.00	100.00	*79.40*
NB	76.36	81.60	80.52	23.64	92.45	78.28
LR	38.18	94.81	83.15	9.09	96.23	78.28
QDA	36.36	95.75	83.52	100.00	0.00	20.60
C4.5	56.23	87.12	81.64	37.73	84.56	74.15
Proposed Method	94.20	56.80	**87.65**	92.90	60.50	**86.70**
	VA Long Beach			**Eric**		
Method	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
KNN	42.30	80.40	70.5	61.38	61.96	65.55
SVM	36.84	85.71	62.5	89.74	64.13	78.47
NB	52.5	81.25	75.50	77.78	57.61	68.90
LR	52.94	77.04	75.00	88.89	64.16	77.99
QDA	60.00	80.00	77.50	10.26	92.39	46.41
C4.5	42.8	78.18	72.00	76.86	81.00	78.46
Proposed Method	20.00	97.14	**87.50**	83.33	77.77	**80.95**

Table 4
Performance of the ensemble learning methods and the proposed method over heart datasets.

	Cleveland			**Statlog**		
Method	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
Bagging	72.72	81.08	77.96	71.42	82.60	75.95
AdaBoost	68.90	58.99	64.36	69.33	61.67	65.93
Stacking	88.41	75.54	82.51	90.00	73.33	82.59
BagMoov	93.29	73.38	84.16	92.00	74.17	84.07
NNE	82.93	78.42	80.86	77.33	79.17	78.15
Random Forest	72.00	85.29	74.57	80.00	81.25	80.74
Proposed Method	77.40	90.80	**85.21**	89.74	86.66	**88.26**
	SPECT			**SPECTF**		
Method	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
Bagging	0.00	82.35	79.24	81.16	78.44	81.13
AdaBoost	7.27	96.70	78.28	36.36	81.13	71.91
Stacking	0.00	100.00	79.40	90.91	65.09	70.41
BagMoov	27.27	96.23	80.02	7.27	96.70	78.28
NNE	47.27	87.26	79.03	47.27	85.38	77.53
Random Forest	0.00	81.25	73.58	85.00	76.15	81.13
Proposed Method	94.20	56.80	**87.65**	92.90	60.50	**86.70**
	VA Long Beach			**Eric**		
Method	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity	Accuracy
Bagging	37.50	87.50	67.5	76.92	68.48	73.21
AdaBoost	52.63	77.34	75.00	63.38	60.87	65.07
Stacking	0.00	74.50	74.50	87.18	69.57	79.43
BagMoov	54.66	80.21	81.54	86.32	73.91	80.86
NNE	68.65	59.88	68.00	79.49	73.91	77.03
Random Forest	50.00	89.28	77.50	82.05	77.17	71.42
Proposed Method	20.00	97.14	**87.50**	83.33	77.77	**80.95**

Table 5
General comparison of the classification accuracy results and the ranking of methods.

Method	**Cleveland**		**Statlog**		**SPECT**		**SPECTF**		**VA Long Beach**		**Eric**	
KNN	74.59	9	65.56	12	79.4	7	71.91	7	70.5	7	65.55	11
SVM	80.86	5	81.85	4	79.4	7	79.4	3	62.5	10	78.47	4
NB	77.23	7	78.52	6	80.52	5	78.28	4	75.50	3	68.90	10
LR	83.5	3	82.59	3	83.15	3	78.28	4	75.00	4	77.99	6
QDA	65.68	11	68.15	10	83.52	2	20.6	9	77.50	3	46.41	13
C4.5	75.08	8	77.77	8	81.64	4	74.15	6	72.00	6	78.46	5
Bagging	77.96	6	75.95	9	79.24	8	81.13	2	67.50	9	73.80	8
AdaBoost	64.36	12	65.93	11	78.28	10	71.91	7	75.00	4	65.07	12
Stacking	82.51	4	82.59	3	79.4	7	70.41	8	74.50	5	79.43	3
BagMoov	84.16	2	84.07	2	80.02	6	78.28	4	80.68	2	80.86	2
NNE	80.86	5	78.15	7	79.03	9	77.53	5	68.00	8	77.03	7
Random Forest	74.57	10	80.74	5	73.58	11	81.13	2	77.50	3	71.42	9
Proposed Method	**85.21**	**1**	**88.26**	**1**	**87.65**	**1**	**86.70**	**1**	**87.50**	**1**	**80.95**	**1**

different values. Finally, in initializing of each particle, the existing probability of each sample is considered as 0.65. Generally, all used parameters adjusted through trial-and-error method, and the pseudo code is presented in Algorithm B in Appendix B.

Also, to have a fair comparison, in implementing the employed methods, the used parameters in the aforementioned methods have been used exactly. The classifier used in the bagging methods, AdaBoost, and the meta classifier in stacking is the C4.5 decision tree. Weka and KEEL have been used to implement the methods, and the proposed technique has been implemented using MATLAB 2015a.

3.3. Results of experiments

In each row of Tables 3 and 4, the mean of accuracy, sensitivity and specificity of the proposed method, the individual classifiers KNN, SVM, NB, C4.5, LR, and QDA, in addition to ensemble methods, i.e., bagging, Adaboost, stacking, BagMoov, NNE, and random forest on Cleveland, Statlog, SPECT, SPECTF, VA Long Beach, and Eric datasets are presented. The highest rank is bolded.

According to Table 3, the proposed method outperformed all individual classifiers on the heart datasets and the second-best performance on the Cleveland and Statlog datasets belonging to the LR, on the SPECTF and the Eric datasets belonging to the SVM, and on SPECT and VA Long Beach datasets belonging to the QDA. Moreover, the comparison of the results in Table 4 shows that the proposed technique outperforms all of the other ensemble methods on all datasets except Eric, in which the proposed method performs slightly better than BagMoov. The second-best performance among the ensemble methods on Cleveland, Statlog, VA Long Beach, and Eric belongs to BagMoov, while on SPECT and SPECTF belongs to the random forest.

In Table 5, the accuracy and rank of each method applied to each dataset are shown. As it can be seen, the proposed method ranks first in all six datasets. BagMoov could gain the second rank on Cleveland, Statlog, VA Long Beach, and Eric while QDA and random forest attained the second rank on SPECT and SPECTF datasets, respectively. Also, AdaBoost on Cleveland and SPECT datasets, KNN on Statlog dataset, QDA on SPECTF and Eric datasets, and SVM on VA Long Beach dataset had the worst performance.

In order to demonstrate the performance of the proposed method, the Pareto Front for six heart datasets with a different number of samples and features is presented in Fig. 4. The graphs have two dimensions: error and diversity, which are calculated separately for each dataset, and

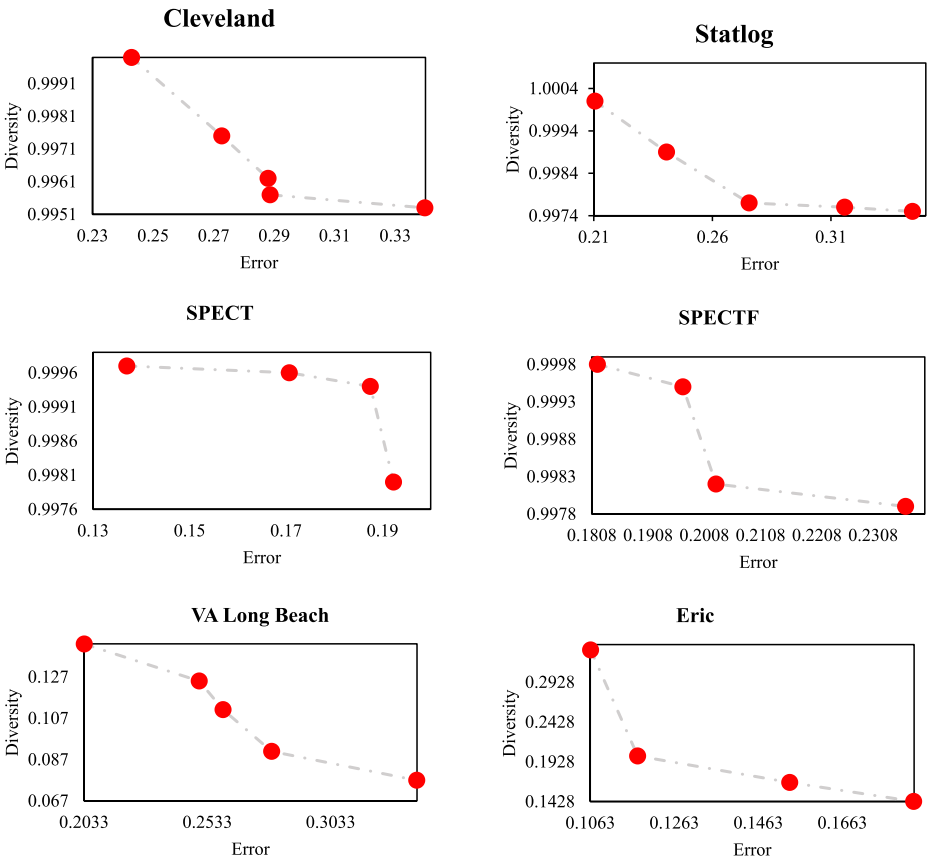


Fig. 4. The obtained near-optimal training sets using proposed methods for six datasets.

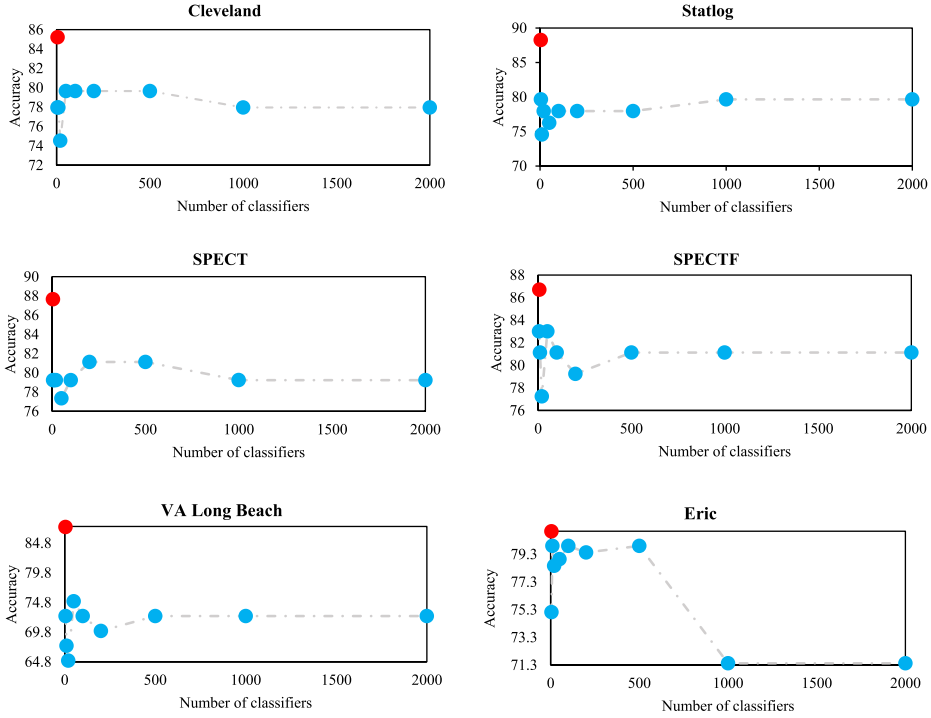


Fig. 5. Comparing the accuracy obtained from different number of classifiers in bagging algorithm with the proposed method (the blue and red points signify the performance of bagging and proposed method, respectively). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

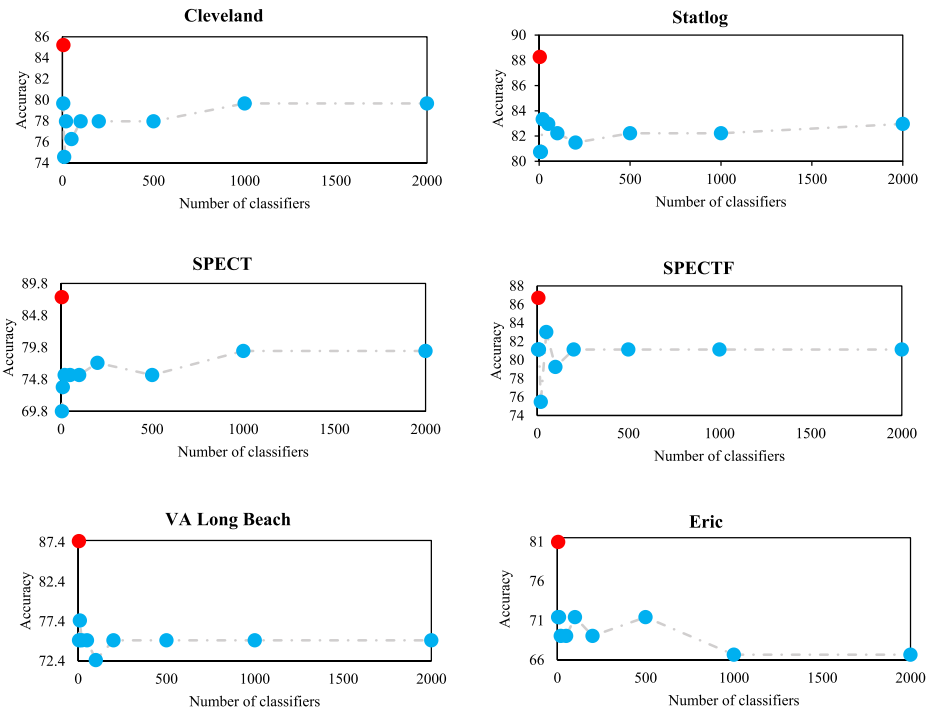


Fig. 6. Comparing the accuracy obtained from different number of classifiers in random forest with the proposed method (the blue and red points signify the performance of random forest and proposed method, respectively).

Table 6
The obtained results of bagging algorithm and random forest (RF) (with different number of classifiers) over six datasets.

	Cleveland		Statlog		SPECT		SPECTF		VA Long Beach		Eric	
# of classifiers	Bagging	RF	Bagging	RF	Bagging	RF	Bagging	RF	Bagging	RF	Bagging	RF
5	77.96	79.66	79.62	80.74	79.24	69.81	83.01	81.13	72.5	75.00	71.42	71.42
10	77.96	74.57	75.95	80.74	79.24	73.58	81.13	81.13	67.50	77.50	73.80	71.42
20	74.53	77.96	75.95	83.33	79.24	75.47	77.25	75.47	65.00	75.00	69.04	69.04
50	79.66	76.27	77.17	82.96	77.35	75.47	83.01	83.01	75.00	75.00	66.66	69.04
100	79.66	77.96	77.17	82.22	79.24	75.47	81.13	79.24	72.50	72.50	69.04	71.42
200	79.66	77.96	77.17	81.48	81.13	77.38	79.24	81.13	70.00	75.00	69.04	69.04
500	79.66	77.96	77.17	82.22	81.13	75.47	81.13	81.13	72.50	75.00	69.04	71.42
1000	77.96	79.66	78.26	82.22	79.24	79.24	81.13	81.13	72.50	75.00	71.42	66.66
2000	77.96	79.66	78.26	82.96	79.24	79.24	81.13	81.13	72.50	75.00	71.42	66.66

Table 7
Accuracy and average number of classifiers obtained by the proposed method.

Number of classifiers	Datasets	Accuracy
5	Cleveland	85.21
4	Statlog	88.26
6	SPECT	87.65
5	SPECTF	86.70
5	VA Long Beach	87.50
5	Eric	80.95

the results are related to the single execution. Furthermore, the proposed method's performance is compared with the bagging algorithm and random forest in terms of the impact of the number of classifiers over the performance. In Figs. 5 and 6 the performance of bagging and random forest are shown considering different number of classifiers, respectively, and the graphs contain two dimensions; accuracy and number of classifiers. In Figs. 5 and 6, red circles demonstrate the average performance of the proposed algorithm, and the blue one shows the bagging and random forest's performances.

According to Figs. 5 and 6, the common characteristic of bagging and the random forest is that with the increase of classifiers, the performance reaches almost a constant level, and no increase in their performance is observed. In contrast, the proposed method with a lower number of classifiers has been able to achieve higher accuracy compared to these two methods. Table 6 presents the performance of bagging and random forest methods with a different number of decision trees, and Table 7 presents the average number of classifiers and the performance of the proposed method.

3.4. Nonparametric statistical tests

In order to establish statistical significance, statistical tests are applied to comparatively evaluate forecasting accuracy between the models used in this paper. To meet this purpose, one powerful rank-based non-parametric test is initially applied called the Friedman test to rank the models and determine the existence of differences among the performance of the models [34,67,68]. In the case of difference existences the first-placed model will be chosen as the control model for being compared with the rest of the models and then five post hoc tests will be applied called Rom [69], Holm [70], Hochberg [70], Li [71], and Finner [72] in order to find whether the control model presents statistical differences among algorithms under the null hypothesis, the Friedman test states that the models are equivalent, so a rejection of this

Table 8
The average obtained ranks by each method in Friedman test (P-value computed by Friedman Test = 0.0003, Friedman statistic = 36.203297).

i	Algorithm	Ranking
1	KNN	10.41
2	SVM	6.58
3	NB	6.83
4	LR	4.66
5	QDA	9.08
6	C4.5	7.5
7	Proposed Method	1
8	Bagging	8.25
9	Adaboost	11
10	Stacking	6.41
11	BagMoov	8.33
12	NNE	8.41
13	Random Forest	7.5

hypothesis implies the existence of differences among the performance of the models studied. In this study, the number of models is 13. The experiment is designed in such a way that statistical significance between the accuracy of the models will be examined.

In Table 8, the results of applying the Friedman test are shown in order to detect whether differences in the results exist. This test is applied with a level of confidence = 0.05. Friedman's tests indicate that significant differences in the results are found. Regarding these results, a post-hoc statistical analysis is required. So the best performing model will be chosen as the control model for being compared with the rest of the models. The best performing model which is associated with the lowest rank is the proposed method. So it will be considered as the control model. Table 9 shows all the adjusted p-values for each comparison that involves the control model. The P-value is indicated in each comparison. Considering a level of significance = 0.05, Table 9 indicates that: regarding the Rom, Holm, Hochberg, Li, and Finner tests, the control model (proposed method) is statistically better than the others.

3.5. Discussion

In this paper, a new approach to improve the performance of the random forest method is presented. In the proposed method, using the MOPSO optimization algorithm and considering the error and diversity objective functions, the training sets are generated to train the classifiers, resulting in the production of diverse and accurate decision trees. By considering the effect of the number of decision trees on the performance of random forest, the proposed method determines the near-optimal number of decision trees for random forest formation. In the proposed method, this is done by considering the trade-off between diversity and classification accuracy, using a multi-objective optimization approach based on Mahalanobis distance and cardinality, and classification error during an evolutionary process.

The results obtained based on the accuracy criterion show that the proposed method outperforms random forest and bagging algorithms with a different number of classifiers. This is confirmed based on the results of Tables 6 and 7. Moreover, comparing the proposed method with individual classifiers and ensemble methods based on non-parametric statistical tests shows that the proposed method is ranked first. One of the disadvantages of the proposed technique is the production time of training sets for training decision trees, which can be increased by considering the high number of iteration. Therefore, the run time complexity of proposed method on six heart disease datasets is compared with random forest, and the result is presented in appendix. In Fig. C1 in Appendix C shows that the random forest performs better in terms of execution time than the proposed method.

Table 9
The post-hoc comparison for $\alpha = 0.05$.

i	Algorithm	Holm/Hochberg	Rom	Finner	Li
1	KNN	0.004545	0.004782	0.008512	0.036874
2	SVM	0.0125	0.013109	0.037739	0.036874
3	NB	0.01	0.010515	0.033617	0.036874
4	LR	0.025	0.025	0.045931	0.036874
5	QDA	0.005	0.00526	0.012741	0.036874
6	C4.5	0.007143	0.007513	0.025321	0.036874
7	Bagging	0.00625	0.006574	0.021145	0.036874
8	Adaboost	0.004167	0.004383	0.004265	0.036874
9	Stacking	0.016667	0.016667	0.041844	0.036874
10	BagMoov	0.05	0.05	0.05	0.05
11	NNE	0.005556	0.005844	0.016952	0.03674
12	Random Forest	0.008333	0.008764	0.029478	0.036874

4. Conclusions and future studies

Heart disease is dangerous disease and threatens the life of millions of people all around the world. Detection is costly and has side effects. Hence, heart disease prediction can play an important role in saving patients' lives.

In this paper, a bi-objective evolutionary approach with the goal of improving random forest's performance in heart disease detection was presented. The proposed method makes it possible to generate diverse and accurate classifiers over training sets by training the classifiers through producing the (near) optimal number of classifiers and diverse training sets. Moreover, training redundant and similar classifiers are prevented through determining the specific number of training sets.

According to literature, ensemble methods outperform individual classifiers, and the obtained experimental results in this study demonstrated that the proposed method had higher performance compared to the five individual classifiers, i.e., KNN, SVM, NB, C4.5, LR and QDA. Furthermore, the six ensemble methods used in the experiments had weak performances compared to the proposed method. The latter consequence can be proved through comparing the obtained results and conducted nonparametric statistical tests.

Comparing random forest with the proposed method, it can be concluded that the training sets generation method and determining the appropriate number of classifiers have important roles on random forest's performance.

Several improvements have been developed for MOPSO which can be used in future studies. Meanwhile, many other multi-objective optimization methods appear in the literature such as non-dominated sorting genetic algorithm II (NSGA-II) [73], strength Pareto EA2 (SPEA2) [74], etc., which can be employed instead of MOPSO. Finally, the proposed method can be applied over other disease datasets.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

(See Fig. A1).

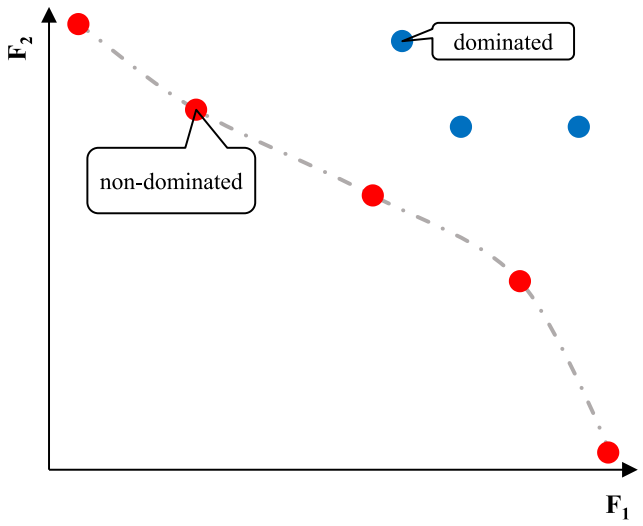


Fig. A1. Sample dominated, non-dominated solutions, and minimization problems with two objectives.

Appendix B. The pseudo-code of parameter tuning

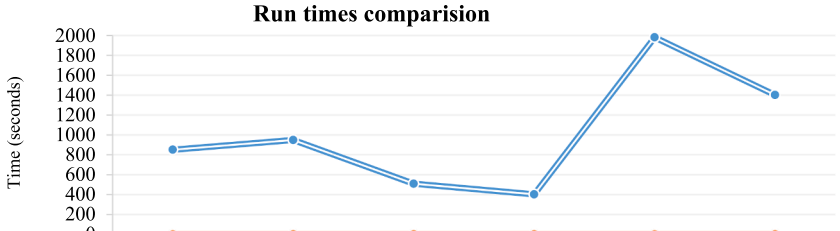
This paper used 10-fold-cross-validation that one fold is used as a test and 9 fold for training. To tune parameters, in the training phase, one fold of 9 folds is used for validation, and this process is iterated 9 times. As a result, throughout this iterative procedure, the parameters were tuned. The pseudo-code of this procedure is presented in Algorithm B.

Algorithm B

```
# set parameters randomly
for i → 1:K-1 # K: number of folds
    choose fold i^th for validation
    train model over K-1 folds except fold i^th according to predefined parameters
    evaluate model on fold i^th
    if the accuracy is satisfying
        use the defined parameters
    Else
        change parameters
        continue
```

Appendix C

(See: Fig. C1).



	Cleveland	Statlog	SPECT	SPECTF	VA Long Beach	Eric
Proposed Method	850.956245	948.421421	509.245	402.3254	1981.270881	1401.150872
Random Forest	0.627024	0.9236	0.83019	0.852001	0.66571	0.8966501

Fig. C1. Computational time of generation of training sets and formation of random forest in the proposed method compared to the random forest.

References

[1] J. Han, J. Pei, M. Kamber, Data mining: concepts and techniques, Elsevier, 2011.
[2] S. Ronoud, S. Asadi, An evolutionary deep belief network extreme learning-based for breast cancer diagnosis, Soft Comput 23 (24) (2019) 13139–13159.
[3] M.H. Tahan, S. Asadi, MEMOD: a novel multivariate evolutionary multi-objective discretization, Soft Comput 22 (1) (2018) 301–323.
[4] F. Mansourypoor, S. Asadi, Development of a Reinforcement Learning-based Evolutionary Fuzzy Rule-Based System for diabetes diagnosis, Comput. Biol. Med. 91 (2017) 337–352.
[5] M.H. Tahan, S. Asadi, EMDID: Evolutionary multi-objective discretization for imbalanced datasets, Inform. Sci. 432 (2018) 442–461, https://doi.org/10.1016/j.ins.2017.12.023.
[6] S.M.R. Kazemi, E. Hadavandi, S. Shamshirband, S. Asadi, A novel evolutionary-negative correlated mixture of experts model in tourism demand estimation, Comput. Hum. Behav. 64 (2016) 641–655.
[7] P. Abbaszadeh, A. Alipour, S. Asadi, Development of a coupled wavelet transform and evolutionary Levenberg-Marquardt neural networks for hydrological process modeling, Comput. Intell. 34 (1) (2018) 175–199.
[8] M. Shouman, T. Turner, R. Stocker, Using data mining techniques in heart disease diagnosis and treatment, in: Proc. 2012 Japan-Egypt Conf. Electron. Commun. Comput. JEC-ECC 2012, 2012: pp. 173–177. https://doi.org/10.1109/JEC-ECC.2012.6186978.
[9] D. Sisodia, D.S. Sisodia, Prediction of Diabetes using Classification Algorithms, Proc. Comput. Sci. 132 (2018) 1578–1585.
[10] V. Chaurasia, S. Pal, B.B. Tiwari, Prediction of benign and malignant breast cancer using data mining techniques, J. Algorithms Comput. Technology 12 (2) (2018) 119–126.
[11] R. Das, I. Turkoglu, A. Sengur, Effective diagnosis of heart disease through neural networks ensembles, Expert Syst. Appl. 36 (4) (2009) 7675–7680.
[12] N.D. Wong, Epidemiological studies of CHD and the evolution of preventive cardiology, Nat. Rev. Cardiol. 11 (5) (2014) 276–289.
[13] S. Bashir, U. Qamar, F.H. Khan, BagMOOV: A novel ensemble for heart disease prediction bootstrap aggregation with multi-objective optimized voting, Australas Phys. Eng. Sci. Med. 38 (2) (2015) 305–323.
[14] Z. Arabasadi, R. Alizadehsani, M. Roshanzamir, H. Moosaei, A.A. Yarifard, Computer aided decision making for heart disease detection using hybrid neural network-Genetic algorithm, Comput. Methods Programs Biomed. 141 (2017) 19–26.
[15] S. Ahmed, A. Bey, S. Hashmi, S. Yadav, Prevalence and clinical aspects of drug-induced gingival enlargement, Biomed. Res 20 (3) (2009) 212, https://doi.org/10.4103/0970-938X.54860.
[16] O.W. Samuel, G.M. Asogbon, A.K. Sangaiah, P. Fang, G. Li, An integrated decision support system based on ANN and Fuzzy_AHP for heart failure risk prediction, Expert Syst. Appl. 68 (2017) 163–172.
[17] I. Yekkala, S. Dixit, M.A. Jabbar, Prediction of heart disease using ensemble learning and Particle Swarm Optimization, in: Proc. 2017 Int. Conf. Smart Technol. Smart Nation, SmartTechCon 2017, Institute of Electrical and Electronics Engineers Inc., 2018: pp. 691–698. https://doi.org/10.1109/SmartTechCon.2017.8358460.
[18] M. Jan, A.A. Awan, M.S. Khalid, S. Nisar, Ensemble approach for developing a smart heart disease prediction system using classification algorithms, Res. Reports Clin. Cardiol. 9 (2018) 33–45, https://doi.org/10.2147/rrcc.s172035.
[19] X. Liu, X. Wang, Q. Su, M.o. Zhang, Y. Zhu, Q. Wang, Q. Wang, A Hybrid Classification System for Heart Disease Diagnosis Based on the RFRS Method, Comput. Math. Methods Med. 2017 (2017) 1–11.
[20] T. Nguyen, A. Khosravi, D. Creighton, S. Nahavandi, Classification of healthcare data using genetic fuzzy logic system and wavelets, Exp. Syst. Appl. 42 (4) (2015) 2184–2197.
[21] G. Manogaran, R. Varatharajan, M.K. Priyan, Hybrid Recommendation System for Heart Disease Diagnosis based on Multiple Kernel Learning with Adaptive Neuro-Fuzzy Inference System, Multimed. Tools Appl. 77 (4) (2018) 4379–4399.
[22] A. Davari Dolatabadi, S.E.Z. Khadem, B.M. Asl, Automated diagnosis of coronary artery disease (CAD) patients using optimized SVM, Comput. Methods Programs Biomed. 138 (2017) 117–126.
[23] R. Polikar, Ensemble based systems in decision making, IEEE Circ. Syst. Mag. 6 (3) (2006) 21–45.
[24] S.E. Roshan, S. Asadi, Improvement of Bagging performance for classification of imbalanced datasets using evolutionary multi-objective optimization, Eng. Appl. Artif. Intell. 87 (2020) 103319, https://doi.org/10.1016/j.engappai.2019.103319.
[25] L. Rokach, Ensemble-based classifiers, Artif. Intell. Rev. 33 (1-2) (2010) 1–39.
[26] Kagan Tumer, Joydeep Ghosh, Error Correlation and Error Reduction in Ensemble Classifiers, Connect. Sci. 8 (3-4) (1996) 385–404.
[27] G. Brown, J. Wyatt, R. Harris, X. Yao, Diversity creation methods: a survey and categorisation, Inform. Fusion 6 (1) (2005) 5–20.
[28] L. Breiman, Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.
[29] Y. Freund, R.E. Schapire, et al., Experiments with a new boosting algorithm, Icml (1996) 148–156.
[30] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32, https://doi.org/10.1023/A:1010933404324.
[31] L.I. Kuncheva, C.J. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Mach. Learn. 51 (2003) 181–207, https://doi.org/10.1023/A:1022859003006.
[32] Z. Donyavi, S. Asadi, Diverse training dataset generation based on a multi-objective optimization for semi-Supervised classification, Pattern Recogn. 108 (2020) 107543, https://doi.org/10.1016/j.patcog.2020.107543.
[33] Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms, 1st ed., Chapman & Hall/CRC, 2012.
[34] S. Asadi, Evolutionary fuzzification of RIPPER for regression: Case study of stock prediction, Neurocomputing 331 (2019) 121–137.
[35] D.S.C. Nascimento, A.L.V. Coelho, A.M.P. Canuto, Integrating complementary techniques for promoting diversity in classifier ensembles: A systematic study, Neurocomputing 138 (2014) 347–357.
[36] J. Abellán, C.J. Mantas, J.G. Castellano, S. Moral-García, Increasing diversity in random forest learning algorithm via imprecise probabilities, Expert Syst. Appl. 97 (2018) 228–243.
[37] S. Asadi, S.E. Roshan, A bi-objective optimization method to produce a near-optimal number of classifiers and increase diversity in Bagging, Knowl.-Based Syst. 213 (2021) 106656, https://doi.org/10.1016/j.knosys.2020.106656.
[38] Y.e. Ren, L.e. 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.
[39] Z. Cselényi, Mapping the dimensionality, density and topology of data: The growing adaptive neural gas, Comput. Methods Prog. Biomed. 78 (2) (2005) 141–156.
[40] S.H. Huang, L.R. Wulsin, H. Li, J. Guo, Dimensionality reduction for knowledge discovery in medical claims database: Application to antidepressant medication utilization study, Comput. Methods Programs Biomed. 93 (2) (2009) 115–123.
[41] R.W. Johnson, An Introduction to the Bootstrap, Teach. Stat. 23 (2001) 49–54. https://doi.org/10.1111/1467-9639.00050.
[42] T.K. Ho, Random decision forests, in: Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, IEEE Computer Society, 1995: pp. 278–282. https://doi.org/10.1109/ICDAR.1995.598994.
[43] C. Yang, X.-C. Yin, Diversity-Based Random Forests with Sample Weight Learning, Cogn. Comput. 11 (5) (2019) 685–696.
[44] P. Latinne, O. Debeir, C. Decaestecker, Limiting the number of trees in random forests, in: Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), Springer Verlag, 2001: pp. 178–187. https://doi.org/10.1007/3-540-48219-9_18.
[45] S. Bernard, L. Heutte, S. Adam, Using random forests for handwritten digit recognition, in: Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, 2007: pp. 1043–1047. https://doi.org/10.1109/ICDAR.2007.4377074.
[46] S. Bernard, L. Heutte, S. Adam, On the selection of decision trees in Random forests, in: Proc. Int. Jt. Conf. Neural Networks, 2009: pp. 302–307. https://doi.org/10.1109/IJCNN.2009.5178693.
[47] V.Y. Kulkarni, P.K. Sinha, Pruning of random forest classifiers: A survey and future directions, in: Proc. - 2012 Int. Conf. Data Sci. Eng. ICDSE 2012, 2012: pp. 64–68. https://doi.org/10.1109/ICDSE.2012.6282329.
[48] M. Kaur, H.K. Gianey, D. Singh, M. Sabharwal, Multi-objective differential evolution based random forest for e-health applications, Mod. Phys. Lett. B 33 (05) (2019) 1950022, https://doi.org/10.1142/S0217984919500222.
[49] M. Bursa, L. Lhotska, M. Macas, Hybridized Swarm Metaheuristics for Evolutionary Random Forest Generation, in: Institute of Electrical and Electronics Engineers (IEEE), 2008: pp. 150–155. https://doi.org/10.1109/his.2007.9.
[50] M.A. Jabbar, B.L. Deekshatulu, P. Chandra, Intelligent heart disease prediction system using random forest and evolutionary approach, J. Netw. Innov. Comput. 4 (2016) 175–184.
[51] C. Qi, Q. Chen, Evolutionary Random Forest Algorithms for Predicting the Maximum Failure Depth of Open Stope Hangingwalls, IEEE Access 6 (2018) 72808–72813.
[52] X.-a. Bi, X.i. Hu, H. Wu, Y. Wang, Multimodal Data Analysis of Alzheimer's Disease Based on Clustering Evolutionary Random Forest, IEEE J. Biomed. Health Inform. 24 (10) (2020) 2973–2983.
[53] M.N. Adnan, M.Z. Islam, Optimizing the number of trees in a decision forest to discover a subforest with high ensemble accuracy using a genetic algorithm, Knowl.-Based Syst. 110 (2016) 86–97.
[54] J. Kennedy, R. Eberhart, Particle swarm optimization (PSO), in: Proc. IEEE Int. Conf. Neural Networks, Perth, Aust., 1995: pp. 1942–1948.
[55] Y. Zhang, S. Wang, P. Phillips, G. Ji, Binary PSO with mutation operator for feature selection using decision tree applied to spam detection, Knowl.-Based Syst. 64 (2014) 22–31.
[56] S. Chandra, R. Bhat, H. Singh, A PSO based method for detection of brain tumors from MRI, in: 2009 World Congr. Nat. Biol. Inspired Comput. NABIC 2009 - Proc., 2009: pp. 666–671. https://doi.org/10.1109/NABIC.2009.5393455.
[57] Z. Abdmouleh, A. Gastli, L. Ben-Brahim, M. Haouari, N.A. Al-Emadi, Review of optimization techniques applied for the integration of distributed generation from renewable energy sources, Renew. Energy 113 (2017) 266–280.
[58] S.L. Salzberg, C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993, Mach. Learn. 16 (1994) 235–240. https://doi.org/10.1007/bf00993309.
[59] Alexey Tsymbal, Mykola Pechenizkiy, Pádraig Cunningham, Diversity in search strategies for ensemble feature selection, Inform. Fusion 6 (1) (2005) 83–98.
[60] Zahra Donyavi, Shahrokh Asadi, Using decomposition-based multi-objective evolutionary algorithm as synthetic example optimization for self-labeling, Swarm Evol. Comput. 58 (2020) 100736, https://doi.org/10.1016/j.swevo.2020.100736.
[61] P.C. Mahalanobis, On the generalized distance in statistics, in (1936).
[62] Mohammad Karim Sohrabi, Alireza Tajik, Multi-objective feature selection for warfarin dose prediction, Comput. Biol. Chem. 69 (2017) 126–133.
[63] C.A. Coello Coello, M.S. Lechuga, MOPSO: A proposal for multiple objective particle swarm optimization, in: Proc. 2002 Congr. Evol. Comput. CEC 2002, IEEE Computer Society, 2002: pp. 1051–1056. https://doi.org/10.1109/CEC.2002.1004388.

[64] M. Reyes-Sierra, C.A.C. Coello, others, Multi-objective particle swarm optimizers: A survey of the state-of-the-art, Int. J. Comput. Intell. Res. 2 (2006) 287–308.
[65] D. Dua, C. Graff, {UCI} Machine Learning Repository, (2017). http://archive.ics.uci.edu/ml.
[66] David H. Wolpert, Stacked generalization, Neural Networks 5 (2) (1992) 241–259.
[67] D.J. Sheskin, Handbook of Parametric and Nonparametric Statistical Procedures, Chapman and Hall/CRC (2003), https://doi.org/10.1201/9781420036268.
[68] Shahrokh Asadi, Jamal Shahrabi, Complexity-based parallel rule induction for multiclass classification, Inf. Sci. 380 (2017) 53–73.
[69] D.M. ROM, A sequentially rejective test procedure based on a modified Bonferroni inequality, Biometrika 77 (3) (1990) 663–665.
[70] S. Holm, A simple sequentially rejective multiple test procedure, Scand. J. Stat. 65–70 (1979).
[71] Jianjun (David) Li, A two-step rejection procedure for testing multiple hypotheses, J. Stat. Plan. Inference 138 (6) (2008) 1521–1527.
[72] H. Finner, On a Monotonicity Problem in Step-Down Multiple Test Procedures, J. Am. Stat. Assoc. 88 (423) (1993) 920–923.
[73] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2002) 182–197. https://doi.org/10.1109/4235.996017.
[74] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the strength Pareto evolutionary algorithm, TIK-Report. 103 (2001).