



A novel bio-inspired hybrid multi-filter wrapper gene selection method with ensemble classifier for microarray data

Babak Nouri-Moghaddam¹ · Mehdi Ghazanfari¹ · Mohammad Fathian¹

Received: 10 October 2020 / Accepted: 26 August 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

Microarray technology is known as one of the most important tools for collecting DNA expression data. This technology allows researchers to investigate and examine types of diseases and their origins. However, microarray data are often associated with a small sample size, a significant number of genes, imbalanced data, etc., making classification models inefficient. Thus, a new hybrid solution based on a multi-filter and adaptive chaotic multi-objective forest optimization algorithm (AC-MOFOA) is presented to solve the gene selection problem and construct the Ensemble Classifier. In the proposed solution, a multi-filter model (i.e., ensemble filter) is proposed as preprocessing step to reduce the dataset's dimensions, using a combination of five filter methods to remove redundant and irrelevant genes. Accordingly, the results of the five filter methods are combined using a voting-based function. Additionally, the results of the proposed multi-filter indicate that it has good capability in reducing the gene subset size and selecting relevant genes. Then, an AC-MOFOA based on the concepts of non-dominated sorting, crowding distance, chaos theory, and adaptive operators is presented. AC-MOFOA as a wrapper method aimed at reducing dataset dimensions, optimizing KELM, and increasing the accuracy of the classification, simultaneously. Next, in this method, an ensemble classifier model is presented using AC-MOFOA results to classify microarray data. The performance of the proposed algorithm was evaluated on nine public microarray datasets, and its results were compared in terms of the number of selected genes, classification efficiency, execution time, time complexity, hypervolume indicator, and spacing metric with five hybrid multi-objective methods, and three hybrid single-objective methods. According to the results, the proposed hybrid method could increase the accuracy of the KELM in most datasets by reducing the dataset's dimensions and achieve similar or superior performance compared to other multi-objective methods. Furthermore, the proposed Ensemble Classifier model could provide better classification accuracy and generalizability in the seven of nine microarray datasets compared to conventional ensemble methods. Moreover, the comparison results of the Ensemble Classifier model with three state-of-the-art ensemble generation methods indicate its competitive performance in which the proposed ensemble model achieved better results in the five of nine datasets.

Keywords Gene selection · DNA microarray data · Hybrid method · Multi-filter · Multi-objective wrapper · Forest optimization algorithm · Ensemble classification

1 Introduction

One of the most important advances in medical technology in the recent decade is the development of DNA. Microarray technology makes it possible to solve the problem of gene expression profiling [1]. Cancer occurs mainly due to the changes in genes and their unwanted mutations. Identification of these genes and their effects on the development of cancer plays a major role in the diagnosis and treatment of such diseases. One of the important

✉ Mehdi Ghazanfari
mehdi@iust.ac.ir

Babak Nouri-Moghaddam
b_nourimoghaddam@ind.iust.ac.ir;
babaknouriit85@gmail.com

Mohammad Fathian
fathian@iust.ac.ir

¹ Department of Industrial Engineering, Iran University of Science and Technology, 1684613114 Tehran, Iran

applications of microarray data is its use in identifying cancer patients [2]. Accordingly, in recent years, extensive studies have been conducted by relying on machine learning and data mining methods to provide classification methods in cancer diagnosis using microarray data. However, using machine learning methods to classify such data is very challenging due to reasons such as a small number of samples, a great number of genes, imbalanced data, data complexity, and data shift [2].

The presence of a large number of genes makes it very difficult to identify and select genes that are effective in the disease. Also, it makes the classification models more complex and increases the training time of these models. Moreover, a small number of samples and the imbalanced data make classification models after training to be less generalizable to predict new data labels. To overcome the challenges of microarray data, solutions such as gene selection to reduce dimensions and ensemble learning to increase the generalizability of the classification model have been considered by many researchers [1–3]. The solutions of reducing dimension in microarray data are divided into two general classes of feature selection and feature extraction. They have positive results such as simplifying classification models, reducing learning time, and usually increasing classification accuracy. For dimension reduction, feature extraction methods map the original gene space to lower dimension space [4]. However, this approach reduces the interpretability of the original genes and the possibility of identifying the effect of the original genes on the final result. Unlike the previous methods, feature/gene selection aims at identifying a subset of prominent genes among all genes and uses statistical methods or meta-heuristic search methods to achieve this objective. In feature selection methods, subsets of selected genes are composed of genes of the main dataset. Therefore, they allow researchers to have better interpretation and analysis [5, 6]. Thus, feature/gene selection is one of the suitable solutions to reduce the dimensions of microarray data and selects genes that are effective in different types of cancers. However, the problem of selecting the subset of prominent genes in microarray data is known as an *NP-hard* problem [7].

Gene selection methods can generally be divided into five classes, namely filter, wrapper, embedded, ensemble, and hybrid [2]. Wrapper methods usually include a meta-heuristic algorithm as a search method to find sub-optimal subsets and a prediction model. Accordingly, the performance of the prediction model acts as a fitness function to evaluate the quality of the found feature subsets in the search process [8]. Filter methods rely on information theory and statistical techniques, and these approaches attempt to identify a sub-optimal subset of features such that the members of these subsets have the least internal

correlation and the highest correlation with the output [9]. Furthermore, in recent years, researchers studied the potential of using a combination of filter methods as multi-filter (i.e., ensemble) to reduce the bias effect of filter methods in gene selection and combine the strengths of different filter methods [10–13]. Accordingly, multi-filter methods usually include two or more filters and an aggregation function (e.g., union, intersect, etc.) to combine the results of the filter methods. Hybrid methods try to exploit the positive points of both filter and wrapper methods simultaneously so that the filter method, as a preprocessing step, can reduce the dimensions of the gene space and then the wrapper method can select prominent genes among the remaining genes [1, 4, 14]. Given the advantages of hybrid methods, they have been the subject of intense research in recent years. Based on the search method, hybrid methods can be divided into single-objective and multi-objective groups. Many hybrid single-objective methods such as GA [15–25], PSO [26, 27], FOA [28], differential evolution algorithm (DE) [29], ant colony optimization algorithm (ACO) [30, 31], shuffled frog algorithm (SFA) [32], teaching–learning-based optimization (TLBO), gravitational search algorithm (GSA) (TLBOGSA) [33], and cuckoo optimization algorithm (COA) [34] have been proposed so far. Additionally, hybrid multi-objective methods have been considered by many researchers in recent years for the simultaneously optimizing the objectives of minimizing the number of genes and maximizing the efficiency of the classification model. Some of these methods are non-dominated sorting GA II (NSGA-II) [35–38], MOPSO [39], MOGA [40], multi-objective spotted hyena optimizer (MOSHO) [41, 42], multi-objective simplified swarm optimization (MOSSO) [10], MOACO [43], and so on.

Imbalanced data and the small number of samples are other problems of microarray data that challenge the performance of classification models in training and coping with unseen data. One method to cope with such challenges is to use ensemble learning models, which is in good agreement with many studies that have utilized these methods on the microarray datasets [44–47]. The aim of developing such a system is to offer a trade-off solution between test error and training error in an automated classification model. Ensemble models have been used effectively so far in a range of problems like feature selection, missing features, imbalanced data, incremental learning, concept drift learning, and other applications [48]. The main difference among these methods stems from three factors: (1) the way of selecting the training data, (2) the process of creating ensemble learner members, and (3) the law of combining the output of classifiers [48]. The ability of ensemble classification models in reducing training error and increasing the generalizability of the

model has made them usable in the area of feature selection studies [49–57].

The problem of gene selection usually follows two conflicting objectives, which include reducing the number of genes and increasing the efficiency of classification. Accordingly, it can be solved as a multi-objective optimization problem (MOP). The final solution for MOPs is usually presented in the form of a set of non-dominated solutions, which is a trade-off between conflicting objectives [58]. As a result, the gene selection problem could be solved by utilizing wrapper methods with multi-objective meta-heuristic algorithms as search methods [35–38, 41, 42]. Also, given the research literature, studies on solving the problem of gene selection using the hybrid multi-objective method are more limited compared to those of the hybrid single-objective method. Besides, many studies show that solving the gene selection problem using multi-objective metaheuristic methods could improve the results [39, 42, 59, 60]. Most of the proposed hybrid solutions have solved the problem of gene selection only by considering the criterion of classification efficiency as a single objective [2, 3, 5].

Given what was stated above, to solve the problem of gene selection and to construct an ensemble classification model in microarray data, a hybrid solution based on multi-filter and novel adaptive chaotic multi-objective forest algorithm (AC-MOFOA) wrapper method is presented in this paper. In the proposed solution, multi-filter is employed as a pre-processing step to reduce the dimensions of the dataset. Due to the combination of several filter methods (i.e., IG, Minimum Redundancy Maximum Relevance (mRMR), RelifF, Correlation-Based Feature Selection (CFS), and Fisher-score), the multi-filter method has less bias in selecting distinct genes compared to single-filter methods. Also, the multi-objective forest algorithm as a wrapper minimizes the number of genes, maximizes the classification criteria, and optimizes the ELM kernel parameters, simultaneously. Accordingly, the final output of the wrapper will be a set of non-dominated solutions that has solutions with different gene numbers and classification accuracy and provides the diversity needed to construct an ensemble classification model. The results of the proposed algorithm are compared with three single-objective methods (i.e., GA, adaptive GA, and TLBOGSA) and five hybrid multi-objective methods (i.e., MOSSO, multi-objective chaotic emperor penguin optimization (MOCEPO), multi-objective spotted hyena optimizer and salp swarm algorithm (C-HMOSHSSA), non-dominated sorting PSO (NSPSO), and multi-objective binary biogeography-based optimization (MOBBBO)) on nine public microarray datasets, which have the number of different genes, classes, and samples. The main objectives of this study are as follows:

- Providing a new solution based on multi-filter and wrapper for gene selection in the microarray
- Introducing a new adaptive multi-objective forest algorithm based on chaos theory and non-dominated sorting
- Using multi-filter to pre-process data and reduce the number of data genes
- Selecting effective genes simultaneously with optimizing KELM classifier parameters
- Constructing ensemble classifier using the results of the final Pareto front

The novelty of our work can be highlighted as follows:

- A novel hybrid filter-wrapper method is proposed
- A brand new non-dominated sorting-based MOFOA is presented
- A new adaptive chaotic operators is introduced to MOFOA
- New combination of filters for multi-filter generation is introduced
- A novel approach for ensemble generation from Pareto front solutions is proposed

The rest of this article is organized as follows: In Sect. 2, we will explain the fundamental principles of the theory and previous methods of gene selection. In Sect. 3, the proposed idea for solving the gene selection problem is described. Section 4 presents the design of the experiments and their details. In Sect. 5, the results of the experiments are analyzed. Finally, in Sect. 6, conclusions and suggestions for future works are presented.

2 Background

2.1 Basic theories

The basic theories of the methods used in this manuscript have been presented in this subsection.

2.1.1 Forest optimization algorithm (FOA)

The search process in wrapper methods is usually based on a metaheuristic method such as GA and PSO. The FOA is one of the new metaheuristic algorithms that has been introduced in recent years. This algorithm tries to provide a solution for optimization problems by modeling trees' reproduction, growth, and competition in the forest.

FOA has features such as high speed, low number of function evaluations, effective global, and local search. In recent years, this algorithm has been successfully used to solve the problem of feature selection as a single-objective approach [61, 62]. The general steps of FOA have been

presented in Supplementary Information. FOA has two important parameters, i.e., local seeding change (*LSC*) and global seeding change (*GSC*). The number of children generated by this operator for each parent is *LSC*. Moreover, the number of changed variables and selected trees in the global seedling operator are some of FOA parameters denoted by global seeding *GSC* and transfer rate, respectively.

2.1.2 Extreme learning machine

The extreme learning machine (ELM) model, as one of the newest classification models based on artificial neural networks, was presented by Huang [63]. ELM is based on single-hidden layer feedforward networks (SLFNs) [64]. This neural network model has been presented based on not adjusting the parameters of the hidden-layer network. Compared to conventional neural network training methods, ELM has features such as better generalizability, faster learning, and not being trapped in local optimum [63]. For linear separation of data and improving classification efficiency, Kernel ELM, as a version of ELM, maps data from a smaller space to a larger space. Based on the details provided in the Supplementary Information, the performance of KELM depends on the values of γ of the kernel function and C (i.e., the regularization parameter). Hence, these two parameters need to be optimized.

2.1.3 Filter methods

Using statistical techniques and information theory, filter methods measure the intrinsic relationship between the genes and output and identify prominent genes. These methods have low computational overhead and high scalability. However, due to the lack of using a classification model in the gene selection process, they might compromise the accuracy of classification. Filter methods can be divided into two classes, including univariate and multivariate. Univariate methods independently examine the dependence of each gene on the target output, in which the relations between the genes are ignored. In contrast, multivariate methods try to reduce the redundancy of gene subsets by considering the dependence of each gene on the target output and relationships between the genes. Multivariate methods have a higher computational overhead compared to univariate methods. The description of the five filter method used in this paper (i.e., IG, mRMR, ReliFF, CFS, and Fisher-score) is presented in Supplementary Information.

2.1.4 Multi-objective optimization

Problems in the real world usually include the objectives that need to be optimized simultaneously. To solve such problems, a set of solutions that represents a tradeoff among different objectives is required. The set of solutions to multi-objective problems is known as Pareto optimal solutions. A multi-objective optimization (MOO) problem is defined as follows [58, 65, 66]:

$$\min \vec{G} := [G_1(\vec{v}), G_2(\vec{v}), \dots, G_k(\vec{v})] \quad (1)$$

subject to:

$$f_i(\vec{v}) \leq 0 \quad i = 1, 2, 3, \dots, m \quad (2)$$

$$h_j(\vec{v}) = 0 \quad i = 1, 2, 3, \dots, t \quad (3)$$

where $\vec{v} = [v_1, v_2, \dots, v_n]^T$ is a vector of decision variables, $G_i : R^n \rightarrow R$, $i = 1, \dots, k$ is objective functions, and $f_i, h_j : R^n \rightarrow R$, $i = 1, \dots, m, j = 1, \dots, t$ represents constraint functions. The definitions of these parameters are as follows:

Definition 1 The vector $\vec{p} = (p_1, p_2, \dots, p_k)$ dominates the vector $\vec{q} = (q_1, q_2, \dots, q_k)$ as Pareto ($\vec{p} \preceq \vec{q}$), if the following relationship is established:

$$\forall i \in \{1, \dots, k\} : p_i \leq q_i \wedge \exists i \in \{1, \dots, k\} : p_i < q_i \quad (4)$$

Definition 2 The solution $\vec{v} \in \mathcal{F}$ (\mathcal{F} represents the acceptable solution space) is an optimal Pareto solution if there is no solution $\vec{v}' \in \mathcal{F}$ in which $\vec{q} = \vec{G}(\vec{v}') = (\vec{G}_1(\vec{v}'), \dots, \vec{G}_k(\vec{v}'))$ dominates $\vec{p} = \vec{G}(\vec{v}) = (\vec{G}_1(\vec{v}), \dots, \vec{G}_k(\vec{v}))$.

Definition 3 For a given MOP, $\vec{F}(\vec{v})$, the optimal Pareto set \mathcal{P}^* is defined as follows:

$$\mathcal{P}^* := \left\{ \vec{v} \in \mathcal{F} \mid \neg \exists \vec{v}' \in \mathcal{F}, \vec{G}(\vec{v}') \preceq \vec{G}(\vec{v}) \right\} \quad (5)$$

Definition 4 For a given MOP, $\vec{G}(\vec{v})$, the optimal Pareto set \mathcal{P}^* , and optimal Pareto set, \mathcal{PF}^* is defined as follows:

$$\mathcal{PF}^* := \left\{ \vec{u} = \vec{G}(\vec{v}) \mid \vec{v} \in \mathcal{P}^* \right\} \quad (6)$$

As a result, the optimal Pareto front of the set \mathcal{F} of all decision variable vectors will include members that meet the criteria 2 and 3.

2.2 Literature review

Gene selection is one of the important methods in the preprocessing of microarray data to reduce data dimensions and simplify classification models. In general, gene selection methods can be divided into 5 groups: Filter, Wrapper, Hybrid, Ensemble, and Embedded. In Filter methods, the inherent relationship between genes and output is measured using statistical techniques and information theory, and related genes are identified. These methods include correlation-based filter [67], ReliefF [68], symmetrical uncertainty (SU) [67], information gain (IG) [4], and so on. In addition, these methods have low computational overhead and high scalability, but they might reduce the classification accuracy because of not using a classification model in the gene selection process. On the other hand, wrappers include meta-heuristic search methods (e.g., genetic algorithm (GA), forest optimization algorithm (FOA), and particle swarm optimization (PSO)) and a classification model that tries to identify a quasi-optimal subset of genes by considering the classification performance criterion. Wrappers usually have high-computational overhead, but they provide better results than the filter due to using classifiers in the search process. Hybrid methods typically use a combination of a filter and a wrapper to select genes.

Among the mentioned methods, hybrid methods have been considered by many researchers in recent years as they combine positive features of filter and wrapper methods. In hybrid methods, a filter is usually applied to the data as data preprocessing step to select the genes that have the most relationship with the output and the least relationship within the set, and to reduce the initial data set dimensions. Next, a wrapper is used to select a subset of quasi-optimal genes among the remaining gene sets. We review the hybrid methods of gene selection in this section. Depending on the type of search method, hybrid methods can be divided into single-objective and multi-objective groups. Hybrid single-objective algorithms usually follow the objectives of minimizing the number of selected genes or maximizing the efficiency of classification or a combination of these objectives.

Shreem et al. [69] proposed a hybrid solution based on Harmony search and the Markov blanket filter in which the Markov blanket was considered as a local search to improve harmony search solutions. In [31], a hybrid solution is presented based on the Fisher-score filter to reduce the initial dimensions and the ACO algorithm for selecting the optimal gene subset. Elyasigomari et al. [34] proposed a hybrid solution based on the mRMR filter and the hybrid algorithm of the COA and harmony search [34]. In this method, the Harmony search method is used as a local optimizer to improve COA solutions. In [25], a hybrid

method was proposed based on the T-test filter and the Nested-GA algorithm in which GA has two sections of outer and inner. The outer section selects genes based on the accuracy of the SVM classifier and the inner section is executed on the DNA methylation dataset. Shukla et al. [24] proposed a new idea based on conditional mutual information (MI) and adaptive GA filters. In another study, a hybrid solution based on ensemble filter and GA was presented [23]. In this method, three filter methods of ReliefF, Chi-Square, and SU were used to construct the ensemble filter. Based on the conducted study, the Union merge function outperforms the Top N Gene. In [26], a method was proposed based on correlation base filter and improved PSO. Gangavarapu et al. [70] proposed a new idea based on an ensemble filter that uses 5 filter methods of mRMR, IG, CFS, CFSS, and oneRFeatureEval to form the ensemble filter. Then, the parameters of the penalty functions are optimized using Greedy search and GA. In [71], a hybrid solution is presented based on artificial bee colony (ABC) called PrABC, which is an objective function as a combination of classification accuracy and the number of features. In this method, the dimensions of the dataset are reduced using an ensemble filter based on IG, correlation, and relief. Finally, PrABC is used to select the optimal gene subset. Bir-Jmel et al. [30] presented an innovative hybrid method using a filter based on the MWIS graph and ACO. In the proposed method, a local search is used to improve ACO solutions. In [28], a hybrid solution is proposed based on ANOVA filter and enhanced Jaya-based FOA. In this method, both parameters LSC and GSC in the FOA are considered in the range of 1%–50% and optimized using enhanced Jaya. In recent years, deep learning methods have gained great research interest in solving many real-world problems. Abo-Hammour et al. in [72] presented a new approach for solving the prediction of linear dynamical systems by GA. Accordingly, they have utilized GA to simultaneously determine the input–output data sequence of the model in the lack of any knowledge concerning the order, the correct order of the model, and the correct parameters. Furthermore, some applications of the optimization algorithms in mathematics, such as inverse kinematics problem and singular boundary value problem, have been studied in [73–75]. In [76], a novel methodology was proposed by combining deep neural network (DNN) with GA. Based on the proposed scheme, deep features have been extracted using DNN. Then, GA has been applied to the extracted features to determine their optimum combination. In the next step, several classifiers such as SVM, KNN, etc., have been utilized to classify the features. Moreover, in many studies, DNN has been used to solve disease detection and prediction [77–79]. For example, in [77], a new method for detection COVID-19 has been proposed. The proposed approach is based on the

convolutional neural network (CNN) and convolutional long short-term memory (ConvLSTM). The proposed method has been trained on two datasets, namely: CT images and X-ray images. This method has achieved 100% accuracy and F1-score in some cases. Other hybrid single-objective methods include GA [16–25], PSO [26, 27], FOA [28], DE [29], ACO [30, 31], SFA [32], COA [34], and so on.

Multi-objective metaheuristic algorithms optimize several conflicting objectives in their search process [80]. Moreover, the problem of gene selection is inherently a multi-objective optimization problem that has at least two conflicting objectives: (1) minimizing the number of genes and (2) maximizing the criterion of classification efficiency. Many researchers have recently focused on solving the problem of multi-objective gene selection. The solution of such algorithms is a set of non-dominated solutions that provide a tradeoff among different objectives for users. The set of non-dominated solutions is also known as the Pareto Front solutions.

In [40], a hybrid solution is proposed based on the relevance filter and MOGA. In this method, first, the relevance filter selects 100 genes and then MOGA continues the search by considering the objectives of the accuracy of classification and size of the subset of selected genes. Li et al. [60] proposed a new idea by combining the Fisher Score-Markov filter and the MOBBBO algorithm. In this approach, Fisher-Markov as a preprocessing step reduces dimensions of the database by removing irrelevant genes. Next, MOBBBO simultaneously uses the search to find the optimal subset of genes and optimize the parameters of the SVM kernel function. In another proposed solution, a combination of correlation coefficient and NSGA-II is presented [36]. In [39], a hybrid method is presented by considering Quartile filters and non-dominated sorting MOPSO. Lai et al. proposed a hybrid idea based on the averaged filter method (AFM) and MOSSO [10]. AFM is an Ensemble Filter created by combining several filter methods. In [81], a model is proposed by combining filter and wrapper methods in which the T-test filter is considered as one of the objectives. Baliarsingh et al. [59] applied the Fisher-score method along with the MOCEPO. In the proposed method, conventional methods of generating random numbers were replaced with chaos theory to improve efficiency. In [11], a solution called C-HMOSHSSA is presented by combining Fisher-score and MOSHO and the Salp swarm algorithm (SSA). In another study, a combination of a Multi-filter and BOFS was used to solve the gene selection problem and construct an ensemble classifier [11]. In [82], a hybrid solution was presented based on Fisher-score and MOFOA in which the concepts of repository and binary tournament selection are used to solve the problem of multi-objective gene

selection. Divya et al. [41] have proposed a hybrid solution based on IG filter and MOSHO considering the accuracy of SVM and the number of the selected genes. Some other hybrid multi-objective methods include NSGA-II [35–38], MOPSO [10, 39, 83], MOGA [40], MOSHO [41, 42], MOSSO [10], MOACO [43], MOBAT [84, 85], and so on.

Most of the studies conducted to solve the gene selection problem have focused on single-objective metaheuristic algorithms. Besides, many studies show that solving the gene selection problem using multi-objective metaheuristic methods could improve the results [39, 42, 59, 60]. In this regard, studies conducted on hybrid multi-objective gene selection methods are less than single-objective studies. Thus, our study aims to present a hybrid multi-filter and multi-objective wrapper solution based on the forest optimization algorithm to solve the gene selection problem and construct an ensemble classifier.

3 The proposed idea

Given what was stated in the previous section, it can be concluded that the problem of gene selection is a multi-objective problem that must simultaneously solve two conflicting objectives of reducing the number of genes and increasing the efficiency of the classification model. The FOA algorithm is one of the latest metaheuristic algorithms used as a single-objective and multi-objective to solve the problem of feature/gene selection [28, 62, 82, 86]. However, based on the “no free lunch” theory, no solution is optimal for all problems and a new and improved solution can always be proposed. Hence, this study presents a new solution based on multi-filter and AC-MOFOA to simultaneously solve the problem of gene selection and the construction of ensemble classifiers.

In the proposed solution, the Multi-filter first reduces the dimensions of the dataset by removing irrelevant and redundant genes. The considered multi-filter consists of 5 filter methods of IG, Fisher-score, mMRM, CFS, and ReliefF. AC-MOFOA is then presented as a new version of the FOA algorithm based on the concepts of chaos theory, adaptive operators, non-dominated sorting (NS), and crowding distance. AC-MOFOA identifies quasi-optimal gene subsets, considering the objectives of maximizing KELM classification accuracy and minimizing the number of selected genes. Based on the concepts of MOO, the output of MOP problem-solving algorithms is presented as a set of solutions that are not superior to each other. Also, the members of the solutions set have the necessary variety to construct Ensemble classifiers due to training with different subsets of datasets and different KELM classifiers. Therefore, members of the final Pareto Front set are used to construct the Ensemble classifier to obtain a final

classification model for microarray data classification. To have a good understanding, the general flowchart of the proposed method is shown in Fig. 1, followed by describing the main sections.

3.1 Multi-filter step

Filter methods have low computational load and high scalability. However, each filter approach has advantages and disadvantages and using only one filter method to identify related genes can cause bias in the final result. Thus, in the proposed method, a multi-filter is used for the initial preprocessing of the data and identification of subset of genes that have the highest correlation with the output and lowest internal correlation. The objective of multi-filter is to reduce the bias effect of filter methods in gene selection and to combine the strengths of different filter methods. For this purpose, we carried out a series of

experiments using ten well-known filter methods to determine the high performance filter methods and top gene selection threshold. The selected ten filter methods include different multivariate and univariate methods. According to the results presented in Sect. 5.1, five filter methods have been selected to form the proposed multi-filter. The desired multi-filter includes five filters, including IG, Fisher-score, mMRM, CFS, and ReliefF. In this system, Fisher-score, ReliefF and IG are univariate filter methods and mMRM, and CFS are multivariate filter methods. In this step, each of the five methods in the multi-filter is first applied to the dataset and then 30% of the superior genes are extracted from the output of each method based on the rank of the genes. Then, using the voting mechanism and “min vote” threshold, the results of five filter methods are combined and the subset of output genes of the multi-filter step is determined. In this study, the min vote threshold was considered to be 3.

3.2 Wrapper AC-MOFOA step

In the present study, a new version of the FOA called AC-MOFOA is proposed. This new algorithm is based on the concepts of chaos theory, adaptive operators, non-dominated sorting, and crowding distance. One of the important challenges in FOA is determining the optimal value of *LSC* and *GSC* parameters. Thus, adaptive local seeding and adaptive global seeding operators were proposed to automatically adapt the *LSC* and *GSC* values during the search process to solve this challenge in AC-MOFOA. Chaos theory was also used to improve AC-MOFOA diversity and faster convergence. AC-MOFOA optimizes KELM model parameters in addition to searching for quasi-optimal gene subsets. AC-MOFOA pseudo code is displayed in Algorithm 1, with its steps described in the following. Furthermore, Table 1 of the Supplementary Information file describes all the variables of Algorithm 1.

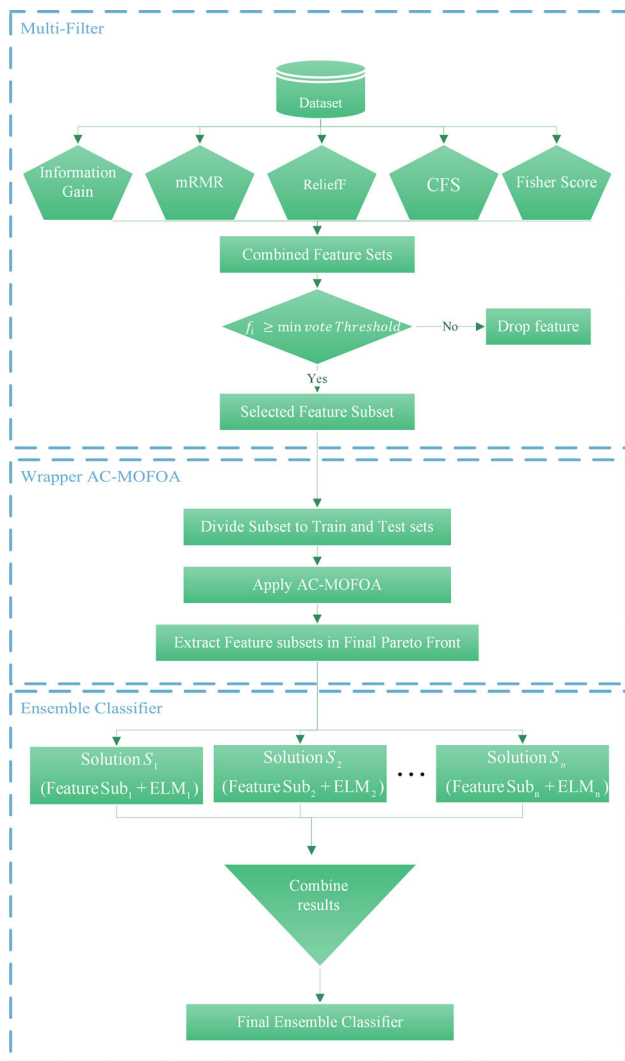


Fig. 1 The flowchart of the proposed method

Table 1 Selected datasets summary

Datasets	# of genes	# of Classes	# of Samples
SRBCT	2308	4	83
Tumors_9	5726	9	60
Leukaemia3	7129	3	72
Colon_Prostate	10,937	2	355
Lung	12,601	5	203
GCM	16,064	14	190
Breast	24,482	2	97
Rscctc_5	54,614	4	89
Rscctc_6	59,005	5	92

Algorithm 1: AC-MOFOA pseudo code

```

1: begin
2: Apply Multi-Filter
3: Get most discriminative genes from Multi-Filter
4: Divide each microarray Dataset into two sets (i.e., Training set and a Test set)
5: Initialize the Forest with Chaotic Trees
6: for  $T = 1$  to Max Number of Iterations do
7:   Assess each Tree in the Forest according to objective functions (i.e., classification accuracy and selected features ratio)
8:   Apply Non-Dominated Sorting to Forest
9:    $FI \leftarrow$  Extract Lowest Rank Pareto Front according to Non-Dominated Sort results
10:  Set Age of all trees in  $FI$  Zero
11:  Calculate crowding distance for trees in  $FI$  according to equation (9)
12:  Remove All trees with  $Age > Max-Age$  and add them to Candidate population
13:  while Forest size  $>$  Area-limit
14:    select a tree from Forest according to Pareto rank and Crowding-distance
    /* trees in Highest Pareto rank with lowest crowding distance will have high priority to remove */
15:    Remove the select tree from Forest
16:    add removed tree to Candidate population
17:  end while
18:  Calculate LSC according to equation (10)
19: for  $j = 1$  to Area-Limit
20:   Apply Roulette wheel to select Parent Tree from  $FI$ 
   /* the solution with more crowding distance have higher selection probability */
21:   Apply Adaptive Chaotic Local-Seeding(selected Parent,  $T$ , LSC)
22:   Add new Trees to the Forest
23:    $j \leftarrow j + LSC$ 
24: end for
25: For All Trees in the Forest  $Age \leftarrow Age + 1$  /*except for new trees add by local seeding operator*/
26: Calculate GSC according to equation (11)
27: Randomly select some trees of the Candidate population according to the “transfer rate”
28: for each selected tree from Candidate population
29:   Apply Adaptive Chaotic Global-Seeding(selected tree,  $T$ , GSC)
30:   Add new Trees to the Forest
31: end for
32:  $T \leftarrow T + 1$ 
33: end for
34: For the solutions in  $FI$ , calculate the classification accuracy on the Test set
35: Return the solutions in  $FI$  as final Pareto front with their Training and Test classification accuracy
36: End

```

3.2.1 Initialization

In AC-MOFOA, each tree is denoted as a vector that represents the values of tree age, C and γ parameters related to KELM, and values related to gene selection (i.e., v_1, v_2, \dots, v_n). In this method, the size of each tree is $n + 3$, which includes n number of data genes, and three variables including age, C , and γ . Figure 2 illustrates the structure of a tree or solution.

AC-MOFOA uses Chaos theory to initialize the trees. Chaos theory is one of the new techniques used to improve

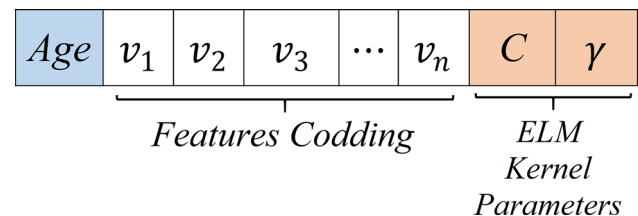


Fig. 2 Tree structure in AC-MOFOA

the search ability of metaheuristic algorithms. Chaos can be described as the behavior of a non-linear dynamic

system which is highly sensitive to the initial state and can be calculated by deterministic algorithms. In addition to having random features, chaos theory has other features such as sensitivity, stochasticity, and ergodicity to preliminary situations. Based on these features, chaos theory ensures diversity among the population-based metaheuristic algorithm solutions and enhances the convergence performance of these algorithms [59, 87, 88]. For this purpose, the logistic map function will replace the generation of random numbers. The logistic map functions have main behaviors: chaotic and convergent. The chaotic and convergent behavior leads to exploration and exploitation, respectively [89]. The logistic map function is defined as follows:

$$v_i^{t+1} = 4 \cdot v_i^t \cdot (1 - v_i^t) \tag{7}$$

where v_i^t represents the value of chaotic map for i -th variable, in which t and $t + 1$ represent the iteration number. Note that v_i^0 is randomly initiated in the interval $(0, 1)$. In the next iteration of AC-MOFOA to calculate v_i^2 using seeding operators, Eq. (7) will applied on v_i^1 . In the proposed solution, based on the KELM studies, the parameters C and γ are considered in range of $[2^{-8}, 2^8]$ [90]. Due to initializing the values of C and γ , v_i^t in Eq. (7) is replaced with C^t and γ^t where the results of this equation mapped to $[2^{-8}, 2^8]$. The value of the age variable will be zero for all new trees generated by AC-MOFOA operators.

The values of the variables v_1, v_2, \dots, v_n belong to an interval of $(0, 1)$, which are mapped to binary space based on Eq. (8). In Eq. (8), if the value of the v_i is greater than 0.5, the binary value of the gene will be equal to 1; otherwise, it will be equal to 0. A value of 1 means that the gene is selected while a value of 0 means that it is not selected.

$$f(v_i) = \begin{cases} 1 & \text{if } v_i \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

According to Algorithm 1, the trees in the Forest will be initialized in step 5. Next, the main loop of the AC-MOFOA starts in step 6. Afterward, in step 7, all trees in the forest will be evaluated according to the objective functions (i.e., classification accuracy and selected genes ratio).

3.2.2 Handling multi-objective

To solve the problem of multi-objective gene selection, some changes must be applied to FOA. For this purpose, the idea of *NS* and crowding distance presented in NSGA-II [91] was employed. *NS* is a technique for ranking Pareto optimal solutions based on the concept of dominance. M and N are the number of objectives and solutions,

respectively. *NS* first extracts the set of solutions that are not dominated by any of the solutions obtained and forms the Pareto front F_1 . Then, among the remaining solutions, the solutions that were dominated only by members of F_1 are extracted to form F_2 and this process continues accordingly. The set of solutions F_1 is considered as the Pareto optimal solutions. The *NS* step is carried out in step 8 of the AC-MOFOA. After applying *NS* on Forest population, the Pareto front F_1 will be determined in step 9. Moreover, in step 10, the age of all trees in F_1 will be set to zero. According to FOA, the local-seeding operator only applies to trees with age equal to zero. So, Step 10 allows the local-seeding operator to exploit around the found Pareto front F_1 and improve the solutions in F_1 .

Crowding-distance is used in NSGA-II to prioritize members within each F_i and direct search to less crowded areas. Crowding-distance is calculated using Eq. (9):

$$cd_i^m = cd_i^m + \frac{obj_m^{j+1} - obj_m^{j-1}}{obj_m^{\max} - obj_m^{\min}} \tag{9}$$

where $m = 1, 2, \dots, M$ is the number of objectives, i is the number of the solution in the list sorted by m -th objective, and obj_m^j represents the value of the m -th objective function for the solution i . The crowding-distance value for the boundary points is assumed to be ∞ . A smaller value of cd_i means that there are other solutions adjacent to the solution i , and the solution i is in a more crowded space, and vice versa. To direct the search to less crowded areas, sorting within each F_i will be done in a descending order from larger cd values to smaller values. In step 11 of the AC-MOFOA, the crowding distance will be calculated on the solutions in F_1 , which later be used in the parent selection phase of the local-seeding operator.

3.2.3 Population control

Forest size control is one of the most important operators of the FOA. In this process, old and improperly fitted trees are removed from the Forest and added to the candidate population. After implementing the adaptive chaotic local seeding operation, the age of all the trees in the Forest increases by one unit. In AC-MOFOA, the age of F_1 member trees is considered to be 0. To limit the population, in step 12, trees older than *Max-Age* are removed from the Forest and added to the candidate population. If the Forest size is still larger than the *Area-Limit* after removing the old trees, the trees with the highest Pareto rank will be removed from the forest and added to the candidate population (steps 13–17). Additionally, if there is a need to remove a part of F_i , trees based on cd value (from lower value to higher value) are removed from F_i and added to the candidate population.

3.2.4 Adaptive chaotic local-seeding

Determining the *LSC* value is one of the challenges of FOA and is determined by trial and error based on the dimensions of the problem. In order to overcome this challenge, we have introduced the adaptive local-seeding operator in which the *LSC* value is determined adaptively by the algorithm itself. Accordingly, *LSC* value is first set at 20% of the data set dimensions and then it is updated in each step through the following equation.

$$LSC = \left\lfloor \frac{LSC}{e^{1/T}} \right\rfloor \quad (10)$$

where *T* represents the current number of iterations of the algorithm. Based on Eq. (10), the minimum *LSC* value will be 1. The value of *LSC* will be calculated in step 18 of AC-MOFOA.

Next, the local seeding operator will be applied to the selected parents. The local seeding operator has been introduced in the FOA to generate exploitation capability. It generates an *LSC* number of new trees for each parent around trees with an age of *zero*. The local seeding operator in AC-MOFOA is a single-parent operator. Each parent is selected among the F_1 members using the roulette wheel. In this method, members with larger *cd* values are more likely to be selected to increase the diversity between solutions. This operator changes the value of a variable in the parent tree using logistic map function according to Eq. (7). An example of this operator is illustrated in Fig. 3.

Local seeding operator implemented using a For-loop in AC-MOFOA (steps 19–24). In step 20, a parent will be selected from F_1 using the roulette-wheel method. According to the above explanation, the selection probabilities will be proportionate to the crowding distance of each solution in F_1 . Moreover, in step 21, a *LSC* number of new trees will be generated using the adaptive chaotic local seeding function. Furthermore, the new trees will be added to Forest in step 22. Afterward, the counter of the loop will be increased by *LSC* in step 23.

3.2.5 Adaptive chaotic global-seeding

Another important parameter of FOA is the *GSC* parameter. Setting its optimal value is one of the challenges of FOA. In the proposed solution, the *GSC* value is set

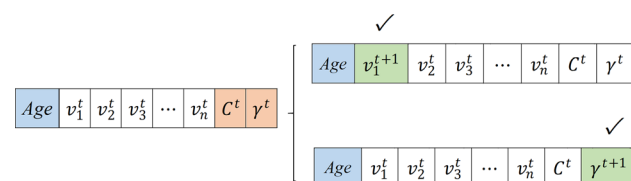


Fig. 3 Example of a local seeding operator with *LSC* = 2

adaptively by the algorithm itself to overcome this challenge. In this method, the *GSC* value is first set at 30% of the dataset dimensions and then is updated in each step through Eq. (11):

$$GSC = \left\lceil \frac{GSC}{e^{1/T}} \right\rceil + 1 \quad (11)$$

Based on Eq. (11), the minimum value of *GSC* will be 2. Accordingly, in step 26 the new value of *GSC* will be calculated.

The global seedling operator is used to create exploration capability to generate new trees in FOA. This operator selects based on the size of the transfer rate parameter from the trees in the candidate population and then changes the *GSC* number of the variables in each tree. To change variables, a chaotic number in the allowable range of the variable is generated using Eq. (7) and replaces its previous value. Figure 4 illustrates the function of this operator.

In step 27, several trees will be selected as parents from the candidate population with respect to the transfer rate parameter. Furthermore, in steps 28–31, the adaptive chaotic global seeding function will be applied to each selected tree to generate new trees.

3.2.6 Objective functions

AC-MOFOA has two objectives of maximizing the accuracy of the classification and minimizing the number of genes. The classification accuracy (*Acc*) is calculated using Eq. (12):

$$Acc = \frac{TP + TN}{FP + FN + TP + TN} \quad (12)$$

The parameters *FP*, *TP*, *FN*, and *TN* represent false positive, true positive, false negative, and true negative, respectively.

3.3 Creation of ensemble classifier

Imbalanced data and the small number of samples are other issues of microarray data that challenge the performance of classification models in training and coping with unseen data. One way to cope with such challenges is to use ensemble learning models. The aim of developing such a system is to provide a trade-off solution between bias error (training error) and variance error (test error) in an

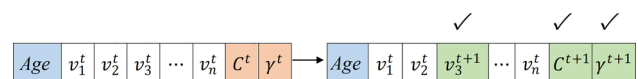


Fig. 4 Example of Global seeding operator performance in AC-MOFOA with *GSC* = 3

automated classification system. One of the important features in constructing an ensemble model is to create diversity in basic classifiers. Several methods have been proposed to create diversity in basic classifiers, e.g., the use of different classification models or training each classification model with different subsets of datasets. The final solution of AC-MOFOA is a set of non-dominated solutions, each including a different subset of genes and a KELM with different kernel parameters. Due to the optimization of C and γ parameters in AC-MOFOA, the KELM classifiers in the final Pareto set have different kernel and regularization parameters. Furthermore, each classifier has been trained on different gene combinations of the data set. Therefore, it can be stated that the members of the final Pareto front have sufficient diversity in terms of basic classifiers and training subsets to construct an ensemble classifier. Accordingly, the output of the final Pareto set (F_1) members is combined using the voting mechanism to form the final ensemble classifier.

4 Experimental design

To design the experimental scenarios, nine microarray datasets [92, 93] were used considering the diversity in the number of samples, genes, and classes (Table 1). After reducing the dimensions in the multi-filter step, each of the datasets is randomly divided into two batches (80% for training and 20% for the test) while maintaining the ratio of classes. To evaluate the subset of selected genes, the KELM classifier with RBF kernel function is used and the kernel (γ) and ELM (C) parameters are optimized by the proposed solution. KELM and tenfold cross-validation (CV) were used to evaluate the subset of selected genes [8].

In the first phase of the experiments, we compared our proposed multi-filter with three other studies (i.e., [23, 70, 71]), which have used a combination of filter methods as preprocessing step. In the second phase, we have compared the proposed hybrid multi-filter multi-objective wrapper approach with three hybrid single-objective methods, namely: GA [23], Adaptive GA [24], and TLBOGSA [33]. Afterward, five Hybrid multi-objective methods, namely C-HMOSHSSA [42], MOBBBO [60], MOCEPO [59], MOSSO [10], and NSPSO [39] were selected for comparing and evaluating the AC-MOFOA results. Moreover, to compare the results of the proposed ensemble method, three conventional and three novel multi-objective ensemble methods were selected. The selected conventional ensemble methods were Random Forest, Bagging, and Adaboost. Furthermore, NSGA-II [94], NSGA-II [95], and sp-MODE [96] were selected as novel ensemble methods.

Python 3.7 programming language is used to implement all methods. We also used the *ELMClassifier* function in the *Python-ELM v0.3* Library with default settings. A PC with 16 GB ram and Intel Core i7 6700HQ hardware was used for performing the tests.

To compare the results fairly, the parameters of the selected algorithms are obtained based on the information of reference articles or through the Taguchi experiment design method [97]. In all algorithms, the number of individuals (i.e., tree, habitat, hyena, salp, and particle) is equal to 60 and the termination condition is considered to be 40,000 evaluations. The results were also examined in 50 independent runs. In algorithms with continuous expression, the threshold for selecting or not selecting a feature is considered to be 0.5. Table 2 presents a summary of the parameters of compared algorithms.

A total of 50 executions were used for comparing the results based on the Pareto front. Accordingly, the obtained Pareto sets in 50 independent executions are first placed in a Union set, and then non-dominated solutions are extracted among them to form the final Pareto set [80]. Furthermore, the average Pareto set was extracted from the Union set. The Average Pareto set demonstrates the average behavior of a multi-objective method in different runs [80, 98–102]. Also, in the next section, Box-plot charts are used to represent some of the obtained results. These charts were plotted based on the data collected in the Union set.

4.1 Comparison criteria of the multi-objective metaheuristics

To compare the results of the multi-objective methods, three criteria have been considered: 1) Hypervolume indicator (HV), 2) Success Counting measure (SCC), and 3) Spacing metrics. The details of these criteria presented as bellow:

4.1.1 Hypervolume indicator (HV)

Diversity and convergence measures are used to evaluate the performance of multi-objective metaheuristic algorithms in identifying the optimal Pareto front. The diversity measure assesses the level of diversity of Pareto front solutions and the convergence measure assesses the degree of convergence of solutions to the main Pareto front. Hypervolume is one of the criteria used for evaluating the performance of multi-objective metaheuristic algorithms. This criterion simultaneously evaluates both diversity and convergence measures [103, 104]. Hypervolume is calculated based on the following equation:

$$HV = \text{volume} \left(\bigcup_{i=1}^{|P|} v_i \right) \quad (13)$$

Table 2 Parameters and settings of selected algorithms

Algorithms	Representation	Operators	Parameters
GA [23]	Binary	2-point Cross-over, uniform mutation	Exchange $p = 0.5$, mutation flip $q = 0.01$, Mutation percentage = 0.3
Adaptive GA [24]	Binary	2-point Cross-over, Conditional mutation	P_c and P_m calculated adaptively
TLBOGSA [33]	Binary	Presented operators	$P_c = 0.7$, $P_m = 0.4$, $\rho, \omega = 0.7$, $c_1, c_2 = 2$
MOBBBO [60]	Binary	Basic Habitat Migration and Mutation Strategy	Habitat modification probability = 1, mutation rate = 0.5
C-HMOSHSSA [42]	Continuous	Standard Spotted Hyena and salp swarm operators	c_1, c_2, c_3 are random numbers, L is max iteration allowed, $g > 1$
MOCEPO[59]	Continuous	Standard Emperor Penguin operators	$M = 2$, r_1, r_2 , are logistic map numbers
MOSSO [10]	Continuous	Standard SSO operators	$c_g < c_p < c_w$, are random numbers
NSPSO [39]	Continuous	Standard PSO operators	r_1, r_2 , are random numbers, $0 < \alpha < 1$
AC-MOFOA	Continuous	Adaptive Chaotic Local and Global seeding	transfer-rate = 30%, lifetime = 15, LSC = 20%, GSC = 30% of variables

where P is the set of non-dominated individuals of the found Pareto front, and v_i is the volume of the individual i .

4.1.2 Success counting

One of the quantified criteria for comparing the performance of multi-objective algorithms is the success counting criterion (SCC) [105]. Based on this criterion, to quantify the performance of multi-objective algorithms, the final Pareto front of each method is aggregated in set P , and then the Pareto front obtained by all methods is extracted from the members of set P . After this step, the level of contribution of each method in the formation of the optimal Pareto front is calculated using Eq. (14).

$$SCC = \sum_{i=1}^n S_i \quad (14)$$

In Eq. (14), if the solution i belongs to the studied method, $S_i = 1$; otherwise, $S_i = 0$. Also, n represents the total number of members of the optimal Pareto front. According to this criterion, a method with higher SCC values shows better performance in identifying the final Pareto front.

4.1.3 Spacing metric

To evaluate the diversity among the Pareto front solutions, the spacing metric will be used [102]. This criterion illustrates how uniformly the solutions in Pareto set have been distributed. This metric is calculated based on Eq. (15).

$$S(A) = \sqrt{\frac{\sum_{i=1}^{|A|} (\bar{d} - d_i)^2}{(|A| - 1)}} \quad (15)$$

where A is the Pareto front set, d_i is the distance of i -th member of A and the nearest solution to it, and the mean of d_i is shown by \bar{d} . The lower value of this criterion illustrates a more uniform distribution of the Pareto set solutions.

5 Experiments and results

In this section, the test results are presented in several subsections: (1) providing multi-filter step results in terms of classification accuracy and dimension reduction rate, (2) comparison of the AC-MOFOA with single-objective hybrid methods, (3) comparison of AC-MOFOA with multi-objective algorithms according to the final Pareto front, SCC, and objective function distribution, (4) AC-MOFOA performance evaluation using Student T-test on HV, and spacing criteria, (5) presenting the comparison results of the examined methods in terms of CPU execution time, and space and time complexity, and (6) analyzing the results of the ensemble classifier. Each of the subsections is described in the following.

5.1 Analyzing the multi-filter results

The multi-filter step is the first step in the proposed hybrid solution for pre-processing and reducing the data dimensions. Thus, we first analyze the results of this step. First, we examine the results of the filter methods to determine the top gene selection threshold and high-performance

Table 3 The classification accuracy results of different filter methods based on the top genes selection threshold (i.e., 25%, 30%, and 35%)

Dataset	Type	Selection threshold	Tumors_9	Leukaemia3	Colon_Prostate	Breast	Rscct_6
Information gain (IG)	Uni	35%	77.89	96.78	96.03	75.11	76.71
		30%	80.68	96.58	96.81	77.46	79.13
		25%	79.73	95.91	95.76	76.51	77.27
Gain ratio	Uni	35%	75.45	93.91	95.29	78.67	73.15
		30%	77.24	94.47	95.56	75.54	74.31
		25%	76.11	94.26	94.93	77.82	73.89
Symmetrical Uncertainty	Uni	35%	73.28	95.38	94.85	68.29	74.43
		30%	76.97	94.67	94.28	71.68	76.2
		25%	75.92	94.23	94.44	72.28	78.74
Fisher Score	Uni	35%	78.59	96.09	96.37	76.7	75.28
		30%	77.46	96.69	97.42	75.83	76.37
		25%	79.86	96.4	96.2	79.68	75.11
Chi-square	Uni	35%	74.51	94.43	96.61	69.39	75.64
		30%	78.37	95.65	96.03	72.47	74.56
		25%	75.88	95.33	95.72	71.56	76.04
ReliefF	Uni	35%	77.85	95.82	96.67	76.23	76.73
		30%	79.86	97.06	96.14	79.18	79.41
		25%	77.85	94.83	96.37	77.74	77.16
Correlation	Uni	35%	78.37	93.13	95.93	70.82	73.92
		30%	77.72	94.59	96.33	73.96	74.73
		25%	75.93	93.94	96.65	72.09	75.28
Correlation-based feature selection (CFS)	Multi	35%	81.49	96.37	97.18	77.15	75.66
		30%	78.98	97.16	97.31	81.44	80.16
		25%	75.93	96.49	96.94	78.12	77.92
Fast Correlation-based feature selection (FCBF)	Multi	35%	76.73	96.37	96.65	74.33	74.86
		30%	76.59	96.51	97.52	77.23	76.52
		25%	75.93	96.13	96.94	76.64	78.21
Minimum-Redundancy-Maximum-Relevance (mRMR)	Multi	35%	79.31	96.26	97.42	77.29	76.97
		30%	82.7	97.25	97.68	78.67	80.35
		25%	79.02	96.47	97.29	80.75	78.48

filter methods. In this experiment, we have considered three different thresholds (i.e., 25%, 30%, and 35%) to select top-ranked genes. Furthermore, ten well-known filter methods have been considered regarding their types (i.e., univariate and multivariate). Accordingly, the results of these experiments on five microarray datasets are presented in Table 3. Based on the results, in most cases, filter methods achieve better classification accuracy by selecting 30% of top-ranked genes. Afterward, we sort the filter methods based on the best result of each method on dataset i to determine the high-performance methods. According to Table 3, the five high-performance methods in most datasets are IG, Fisher-score, mMRM, CFS, and ReliefF. Thus, these five methods will be selected as members of the proposed multi-filter.

Based on Table 4, multi-filter consists of five filter methods of IG, Fisher-score, mMRM, CFS, and ReliefF. It could reduce the number of genes in all datasets by at least 73%. The highest reduction in the number of genes was observed in the Colon-Prostate dataset with 80.9% and the lowest reduction was observed in Rscct_5 dataset with 73.43%. Also, comparing KELM accuracy results in two modes of training with all genes and training with selected genes suggests an improvement in classification accuracy for selected genes (Table 4). Due to employing the combination of univariate and multivariate filter methods, Multi-Filter ability and efficiency increased in selecting the most prominent genes from the dataset; also the bias of using a single filter method can be decreased. Thus, it can be concluded that the Multi-Filter step as preprocess step

Table 4 Result of applying multi-filter step

Datasets	# of genes	KELM acc	# of selected genes by multi-filter	% gene reduction ratio	KELM acc. on selected genes
SRBCT	2308	88.23	472	79.5%	90.72
Tumors_9	5726	78.98	1209	78.8%	84.41
Leukaemia3	7129	97.22	1836	74.2%	97.66
Colon_Prostate	10,937	95.74	2081	80.9%	97.89
Lung	12,601	90.24	2745	78.21%	93.97
GCM	16,064	70.53	3653	77.25%	78.03
Breast	24,482	68.04	5992	75.52%	86.53
Rsctc_5	54,614	69.66	14,507	73.43%	72.38
Rsctc_6	59,005	81.523	15,511	73.71%	82.69

could effectively reduce dimensions of the datasets, and improve the classification accuracy.

Furthermore, we have compared the proposed multi-filter step with three other recent studies (i.e., [23, 70, 71]). The compared studies are hybrid methods that have implemented a multi/ensemble filter as a preprocessing step. According to each approach's multi/ensemble filter step, the comparison results were presented in Table 5. Based on Table 5, the proposed multi-filter achieved better classification accuracy on six of nine datasets. Moreover, the proposed approach obtained the same results with [23] and [70] in Colon-Prostate and Breast, respectively. However, in GCM, the [23] acquired better results than our approach.

From the series of experiments presented in this section, it can be seen that the proposed multi-filter is efficient as preprocessing step in most cases. The proposed approach could reduce the number of genes in the dataset by removing redundant and irrelevant genes, which improves the classification performance.

5.2 Analyzing of the AC-MOFOA results

In this section, first, we analyze the comparison results of the proposed hybrid multi-filter and AC-MOFOA wrapper with the recent hybrid single-objective approaches. Afterward, we compare the performance of the proposed approach with other hybrid multi-objective studies.

5.2.1 Comparison with hybrid single-objective approaches

To compare the results of the proposed hybrid multi-filter AC-MOFOA with hybrid single-objective studies, three hybrid single-objective methods, namely: GA [23], Adaptive GA [24], and TLBOGSA [33], were selected. The results of these experiments are illustrated in Fig. 5. Moreover, in Fig. 5, AC-MOFOA-ND represents the non-dominated solution found in all runs, and AC-MOFOA-Ave is the average Pareto front. According to Fig. 5, the proposed approach achieved better results than the single-objective approaches concerning the number of selected genes and classification accuracy. The non-dominated solution found by AC-MOFOA dominates the solutions of

Table 5 Comparison of the proposed multi-filter with other ensemble filters based on the KELM accuracy

Datasets	Multi-filter method [23]	Multi-filter method [70]	Multi-filter method [71]	Proposed multi-filter
SRBCT	89.93	86.06	88.69	90.72
Tumors_9	82.37	81.61	79.54	84.41
Leukaemia3	96.85	96.24	95.77	97.66
Colon_Prostate	97.89	95.49	96.23	97.89
Lung	92.54	91.62	89.51	93.97
GCM	78.62	77.83	76.86	78.03
Breast	85.41	86.53	82.92	86.53
Rsctc_5	72.38	69.19	70.18	72.38
Rsctc_6	82.08	80.57	81.73	82.69

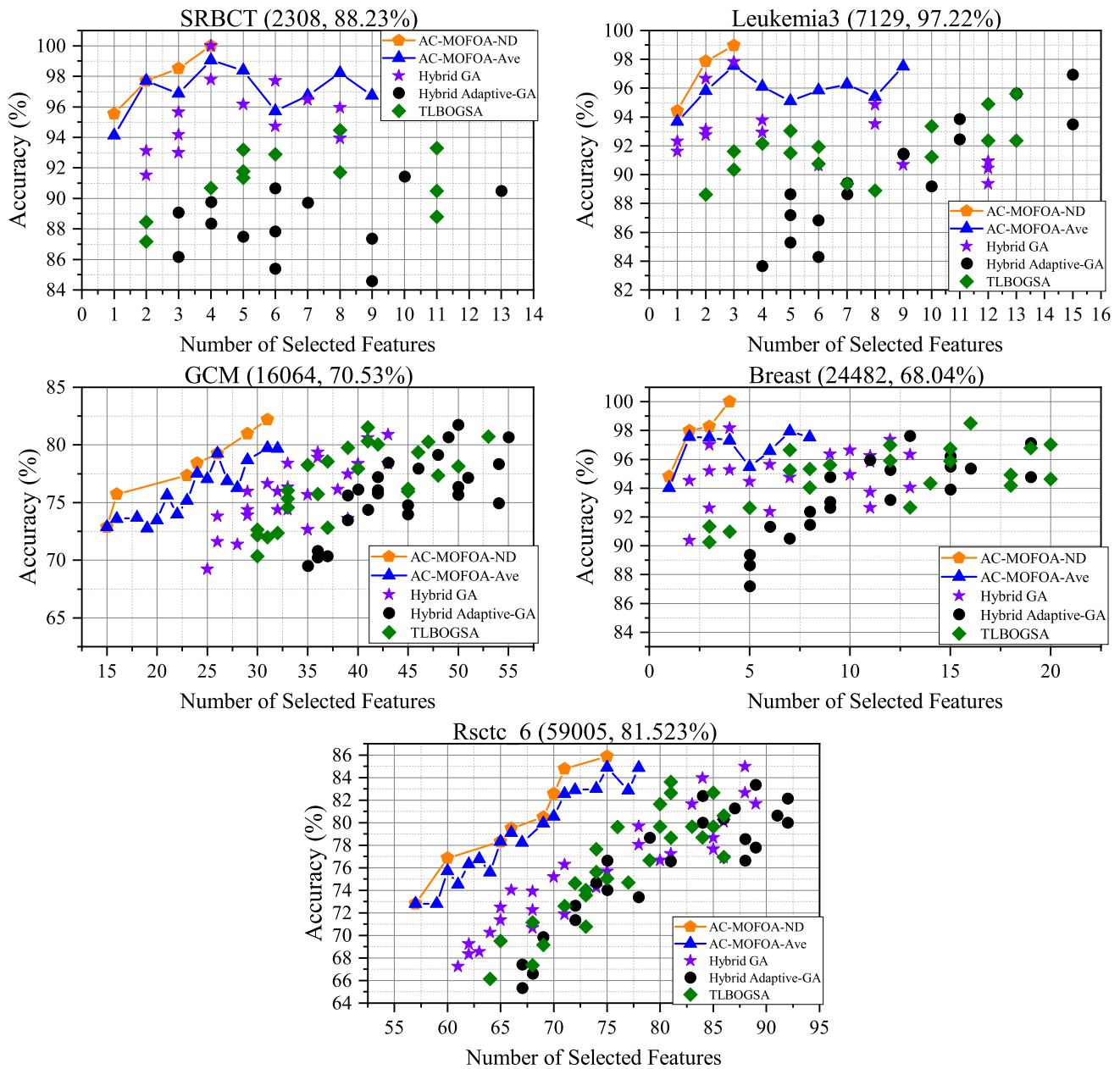


Fig. 5 Comparing AC-MOFOA with other hybrid single-objective algorithms based on non-dominated solutions and average Pareto front on the test set. AC-MOFOA-ND represents the non-dominated solutions, and AC-MOFOA-Ave is the average Pareto Front

the single-objective methods in most cases. Moreover, the average Pareto front of AC-MOFOA shows the average behavior of the AC-MOFOA in all 50 independent runs. Based on the results, in terms of the average Pareto front, AC-MOFOA achieved similar or better results than the single-objective methods. However, in some datasets like Breast, the hybrid approach [23] acquired the solutions which are better than the average Pareto front of AC-MOFOA. The experiments showed that solving the gene selection problem using multi-objective optimization approaches can improve the results.

5.2.2 Comparison with hybrid multi-objective approaches

To evaluate the performance of AC-MOFOA and compare its results with C-HMOSHSSA, MOBBBO, MOCEPO, MOSSO, and NSPSO methods, Figs. 6, 7, 8, 9, 10 were plotted based on the final Pareto front, classification accuracy distribution, and distribution of selected gene numbers in two modes of test and train. Based on Fig. 6, AC-MOFOA in 7 datasets could achieve higher classification accuracy by selecting fewer genes, and AC-MOFOA in 2 datasets (including SRBCT and Breast) showed

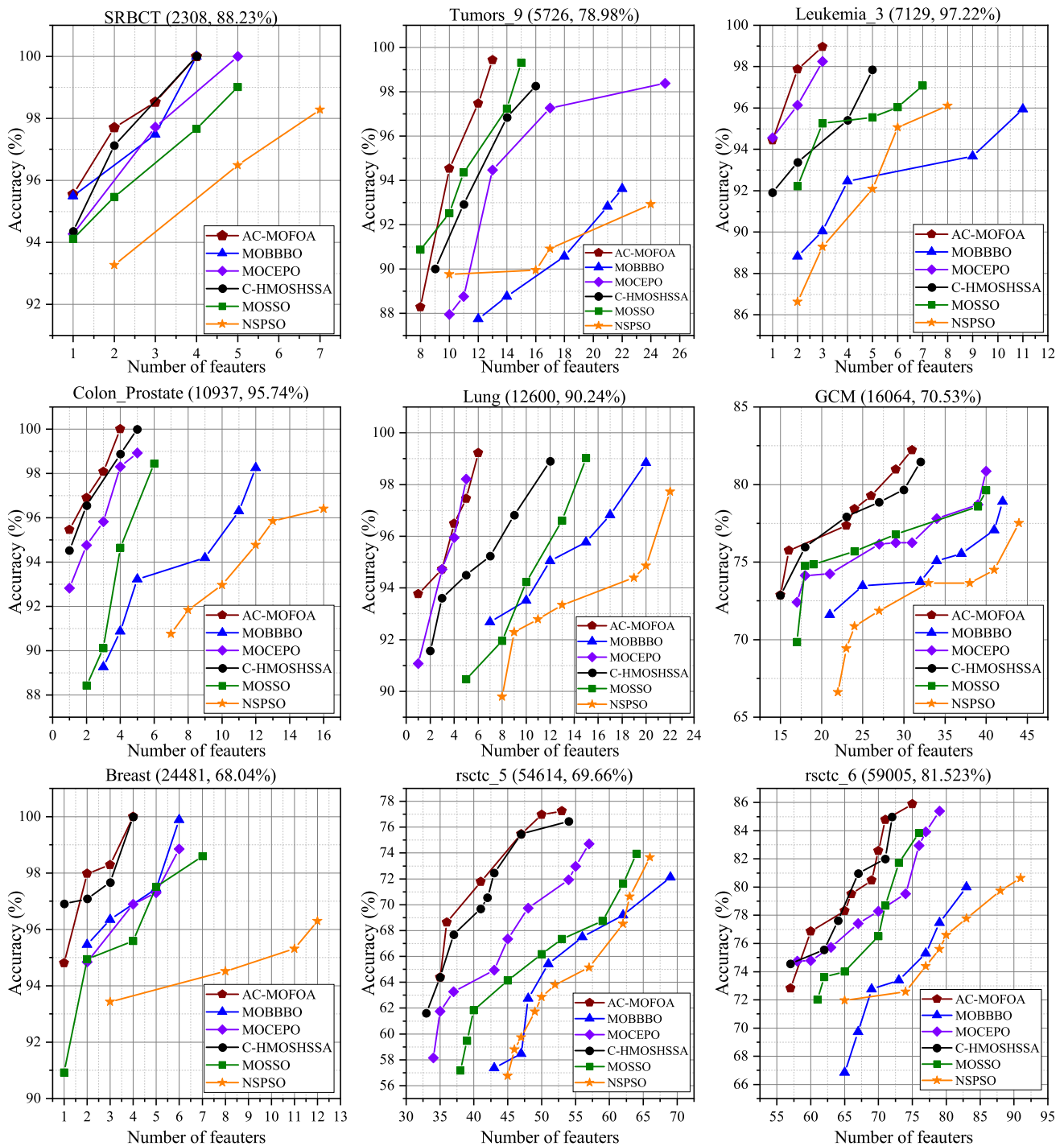


Fig. 6 Comparing AC-MOFOA with other multi-objective algorithms based on non-dominated solutions on the test set

performance similar to C-HMOSHSSA in terms of set objectives. Moreover, AC-MOFOA in all datasets could significantly increase the classification accuracy by considerably reducing the number of genes. For example, in the breast dataset, AC-MOFOA by selecting 4 of 5592 genes (a set of genes selected by the Multi-filter) could achieve a classification accuracy of 100%.

To compare the level of contribution of each method in the formation of the final Pareto front, the SCC results on the test dataset were displayed in Table 6. Based on Table 6, in most datasets AC-MOFOA could achieve higher SCC values than other methods, suggesting that AC-MOFOA has a greater contribution in identifying the final Pareto front.

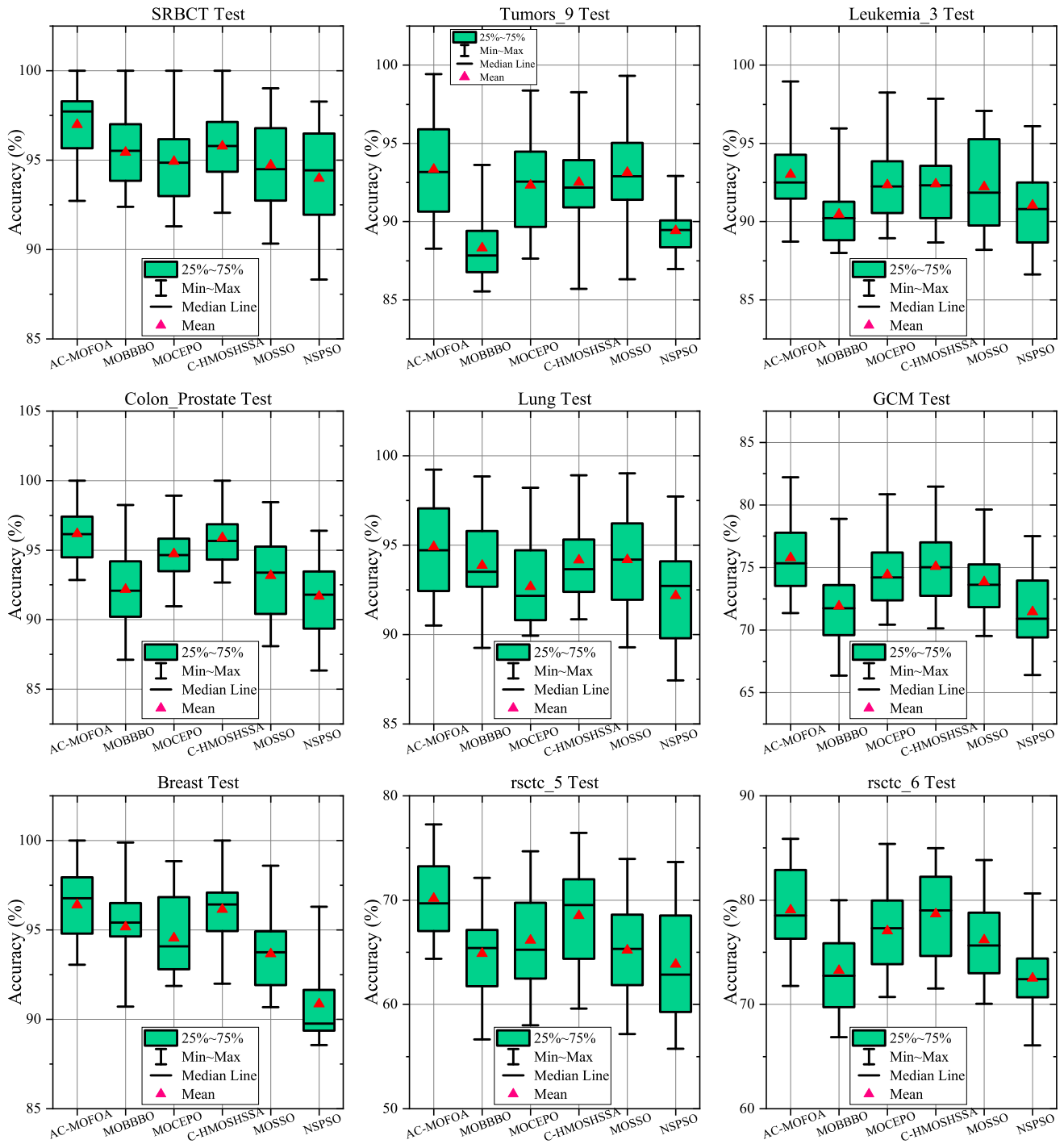


Fig. 7 Comparison of the classification accuracy distribution of AC-MOFOA solutions on the test set

Metaheuristic algorithms have stochastic nature and achieve variant results in different executions. Thus, to compare the results of these algorithms, it is better to compare the distribution of solutions obtained in all 50 executions. Figure 7 presents the distribution of the accuracy of the solutions obtained by AC-MOFOA in the test data. Based on this figure, the accuracy of AC-MOFOA

solutions in 50 independent executions in terms of statistical criteria such as mean and median in most datasets is better than that of other hybrid multi-objective methods. AC-MOFOA also outperformed other methods in terms of maximum and minimum accuracy. AC-MOFOA could increase the efficiency of KELM classification by

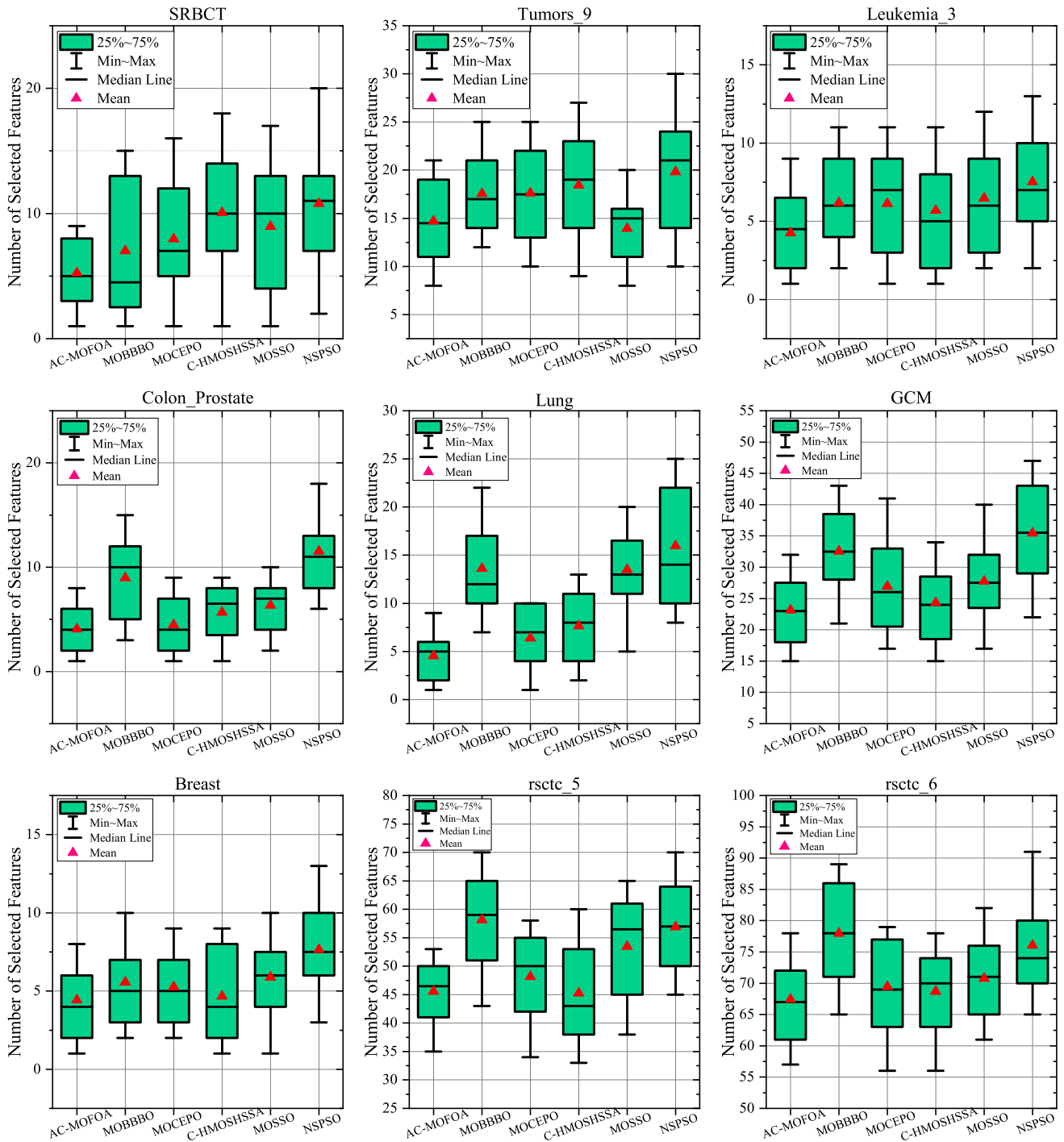


Fig. 8 Comparison of the number of selected genes distributions of AC-MOFOA solutions on the test set

optimizing the RBF kernel (γ) and ELM (C) parameters simultaneously with reducing the number of genes.

Figure 8 compares distributions of the selected genes numbers by AC-MOFOA and other multi-objective methods in 50 independent executions. Based on Fig. 8, it can be seen that AC-MOFOA outperforms other multi-objective algorithms in most datasets in terms of the maximum

and the minimum number of selected genes. Also, in 7 cases of datasets, the proposed method could provide better results than other methods in terms of the mean and median number of selected genes. Only in rsctc_5 and Tumors_9 datasets, C-SHMOSHSSA, and MOSSO methods, respectively, outperformed AC-MOFOA in the objective function of minimizing the number of genes.

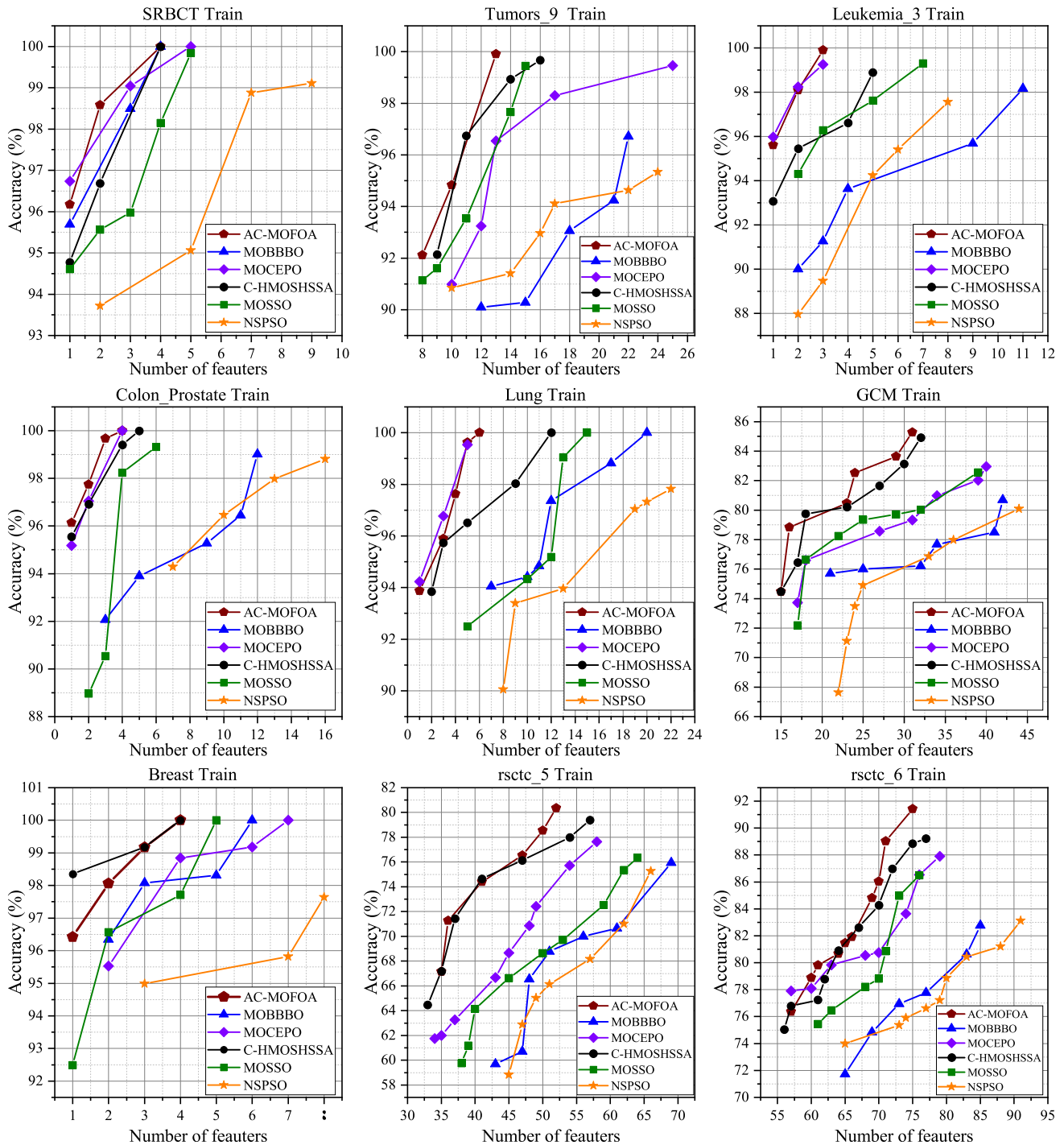


Fig. 9 Comparison of AC-MOFOA with other multi-objective algorithms based on non-dominated solutions on the train set

When comparing the distribution two objectives, they were examined independently and were different from the results in which both of these objectives were examined simultaneously using the concepts of dominance. For example, an algorithm may have achieved solutions with a higher number of genes but lower classification accuracy,

while this solution has been dominated by other solutions in that algorithm.

Figure 9 presents the results of comparing the AC-MOFOA Pareto front with the other 5 methods on the train set. According to this figure, compared to other methods, AC-MOFOA solutions in six of nine datasets achieved higher accuracy by selecting the same or fewer number of

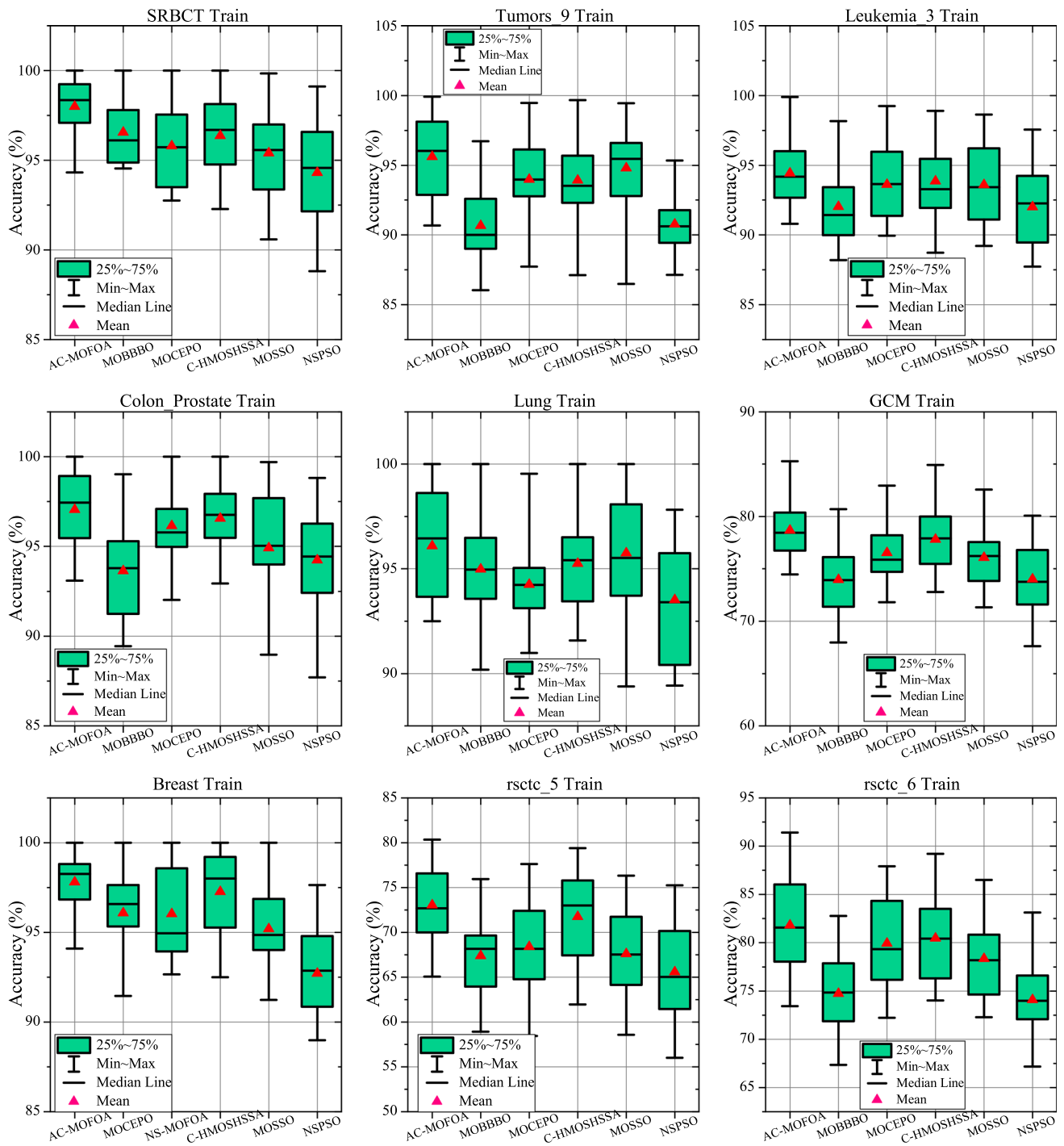


Fig. 10 Comparison of the classification accuracy distribution of AC-MOFOA solutions on the train set

genes. For example, in the rsctc_5 dataset, AC-MOFOA obtained 80.35% accuracy while selecting 52 genes, which was better than the other five methods. Furthermore, in two of nine datasets, the proposed approach acquired similar results compared to other methods. AC-MOFOA in SRBCT and Colon-Prostate reached similar results to C-HMOSHSSA and MOCEPO, accordingly. However, in

the Breast dataset, C-HMOSHSSA achieves similar or better solutions compared to AC-MOFOA. Analysis of the SCC criterion in the training data is shown in Table 7. According to this table, AC-MOFOA had more contribution in identifying the final Pareto front and could identify more than 50% of the final Pareto front in 8 datasets,

Table 6 SCC measure comparison on the test set

Datasets	AC-MOFOA	MOCEPO	MOBBBO	C-HMOSSA	MOSSO	NSPSO
SRBCT	4	0	2	1	0	0
Tumors_9	3	0	0	0	1	0
Leukaemia3	3	1	0	0	0	0
Colon_Prostate	4	0	0	0	0	0
Lung	4	2	0	0	0	0
GCM	6	0	0	3	0	0
Breast	3	0	0	2	0	0
Rstct_5	6	0	0	4	0	0
Rstct_6	6	1	0	4	0	0

Table 7 SCC measure comparison on the train set

Datasets	AC-MOFOA	MOCEPO	MOBBBO	C-HMOSSA	MOSSO	NSPSO
SRBCT	2	1	0	0	0	0
Tumors_9	3	0	0	1	0	0
Leukaemia3	2	2	0	0	0	0
Colon_Prostate	4	1	0	0	0	0
Lung	3	2	0	0	0	0
GCM	6	0	0	2	0	0
Breast	2	0	0	3	0	0
Rstct_5	6	0	0	4	0	0
Rstct_6	9	1	0	3	0	0

indicating its ability to detect the solutions with a fewer number of genes and higher classification accuracy.

Figure 10 shows the results of comparing multi-objective algorithms based on the classification accuracy distribution on the train set. As can be seen from this figure, AC-MOFOA in Tumors_9, Leukemia3, GCM, rstct_5, and rstct_6 datasets obtained higher classification accuracy compared to other methods. In Addition, in eight datasets, AC-MOFOA has acquired better accuracy with respect to the minimum value of accuracy. Moreover, AC-MOFOA outperforms other methods in terms of mean and median accuracy. However, in four datasets, AC-MOFOA has similar performance compared to other methods in term of maximum accuracy.

The obtained results indicate that AC-MOFOA outperformed other methods in terms of achieving the optimal Pareto front and involving in the formation of the final Pareto front. In detail, high initial values of *LSC* and *GSC* in the adaptive local and global seeding operators lead to enhance the exploitation and exploration capabilities of the AC-MOFOA. Local seeding operator is applied to the solution of F_1 that boosts the ability of AC-MOFOA to improve the Pareto solutions further. Moreover, employing the global seeding operator on the Candidate Population gives a chance to less fitted trees for better exploration of the problem space. In addition, AC-MOFOA uses *NS* to identify the Pareto front and rank the solutions.

Furthermore, AC-MOFOA implements the crowding distance, which increases its ability to direct the search to the less crowded areas of the Pareto front, leading to a more uniform final Pareto front. Also, a separate study on the distribution of each objective (namely, classification accuracy and the number of selected genes) shows that AC-MOFOA solutions were superior to other methods in most datasets in terms of statistical criteria such as mean, median, upper and lower bounds.

5.3 Performance evaluation based on Hypervolume and Student t-test

To evaluate the efficiency and effectiveness of AC-MOFOA compared to other multi-objective algorithms, Student T-test was performed on normalized values of hypervolume. To perform the t-test, the hypervolume values were first calculated in 50 independent executions for the MOSSO, MOCEPO, C-HMOSHSSA, MOBBBO, NSPSO, and AC-MOFOA algorithms. Next, the t-test was performed with a significance level of 0.05. Tables 8 and 9 show the t-test results in both test and train modes, respectively. Comparisons were made from left to top, respectively. Superior, similar, and worse performance of the AC-MOFOA compared to the corresponding algorithm are displayed by “+”, “=”, and “-”, respectively.

Table 8 T-test of hypervolume ratios in the *test* data

Datasets	MOCEPO	MOBBBO	C- HMOSHSSA	MOSSO	NSPSO
SRBCT	+	+	+	+	+
Tumors_9	+	+	+	+	+
Leukaemia3	+	+	+	+	+
Colon_Prostate	+	+	+	+	+
Lung	=	+	+	+	+
GCM	+	+	=	+	+
Breast	+	+	=	+	+
Rstct_5	+	+	=	+	+
Rstct_6	+	+	=	+	+

Table 9 T-test of hypervolume ratios in the *train* data

Datasets	MOCEPO	MOBBBO	C- HMOSHSSA	MOSSO	NSPSO
SRBCT	=	+	+	+	+
Tumors_9	+	+	+	+	+
Leukaemia3	=	+	+	+	+
Colon_Prostate	+	+	+	+	+
Lung	=	+	+	+	+
GCM	+	+	=	+	+
Breast	+	+	=	+	+
Rstct_5	+	+	=	+	+
Rstct_6	+	+	+	+	+

As can be seen from Table 8, the AC-MOFOA results in all test datasets are significantly superior to the three methods of MOBBBO, NSPSO, and MOSSO. Moreover, MOCEPO provided similar results to AC-MOFOA only in the Lung database and showed worse performance in other cases. AC-MOFOA could also achieve superior results over C-HMOSHSSA in 5 out of 9 datasets and had similar results in 4 datasets.

Table 9 shows the results of the t-test on the Hypervolume values of the train data. According to the results of the t-test, it can be concluded that AC-MOFOA outperformed other methods in most datasets.

AC-MOFOA showed similar results with MOCEPO and C-HMOSHSSA in only a few cases. For example, we can refer to three datasets of Lung, SRBCT, and Leukemia3 in which AC-MOFOA had a similar performance to MOCEPO.

5.4 Performance evaluation based on Spacing metric and Student t-test

Diversity is one of the important criteria in evaluating the Pareto front solution. For this purpose, we have used the spacing metric to evaluate the diversity and uniformity of the Pareto sets. The spacing metric is calculated according to Eq. (15) and indicates how uniformly solutions are

distributed. According to this criterion, a smaller $S(A)$ value infers more uniformly distributed solutions.

T-test results on the spacing metric values are presented in Tables 10 and 11. According to Table 10, AC-MOFOA has similar results in finding a uniform Pareto front compared to MOCEPO and C-HMOSHSSA on seven of the nine datasets. Furthermore, in the SRBCT and Rstct_6, AC-MOFOA performed better than C-HMOSHSSA and MOCEPO, respectively. Moreover, the proposed approach is significantly better than other methods in the lung dataset. Based on Table 10, AC-MOFOA has significantly better results compared to NSPSO, MOSSO, and MOBBBO in six of the nine cases.

Table 11 presents the results of the spacing metric on the train data. Based on the results, AC-MOFOA achieved similar or better results compared to other multi-objective approaches in most cases. For example, AC-MOFOA significantly performed better than NSPSO in all datasets. However, the only exception is the GCM, where MOSSO has acquired more uniform results compared to AC-MOFOA. The Spacing metrics results indicate that the proposed approach could perform similar or better than other methods in terms of finding a more uniform Pareto front.

Table 10 T-test of spacing metric in the *test* data

Datasets	MOCEPO	MOBBBO	C- HMOSHSSA	MOSSO	NSPSO
SRBCT	=	+	+	+	+
Tumors_9	=	=	=	=	+
Leukaemia3	=	=	=	+	+
Colon_Prostate	=	+	=	+	+
Lung	+	+	+	+	+
GCM	=	+	=	=	+
Breast	=	=	=	+	+
Rstct_5	=	+	=	=	=
Rstct_6	+	+	=	+	+

Table 11 T-test of Spacing metric in the *train* data

Datasets	MOCEPO	MOBBBO	C- HMOSHSSA	MOSSO	NSPSO
SRBCT	=	=	=	+	+
Tumors_9	=	=	=	=	+
Leukaemia3	=	+	=	=	+
Colon_Prostate	=	+	=	+	+
Lung	+	+	+	+	+
GCM	=	+	=	-	+
Breast	+	+	=	+	+
Rstct_5	=	+	=	=	+
Rstct_6	+	+	=	=	+

5.5 Investigating the time and space complexity

Time and space complexity analysis is commonly used to analyze the performance of computer algorithms. Therefore, in this section, we discuss AC-MOFOA complexity analyses. By analyzing the AC-MOFOA algorithm, it can be seen that the two steps of 8 and 11 have the highest time complexity among all the steps of the algorithm. In step 8, a non-dominated sorting operation is performed with the time order of $O(MN^2)$, assuming that M is the number of objectives and N is the number of trees. Also, based on a study conducted by Jensen [106], with an optimal implementation, the time complexity of $O(N\log^{M-1}N)$ can be achieved. In Step 11, crowding-distance is calculated based on time complexity of $O(MN\log N)$. Based on the mentioned points, the time complexity of AC-MOFOA will be from the $O(MN^2)$ in the normal implementation and $O(N\log^{M-1}N)$ in the optimal implementation. Jensen [106] reported that NS-based algorithms in the normal implementation are from $O(MN^2)$.

AC-MOFOA has two memory sections that are used to store Forest and Candidate Population, respectively. Assuming that the dimensions of each tree are k , the space complexity required to store the Forest will be from $O(N \times k)$. The space complexity of Candidate Population will also be from $O(N_{cand.} \times k)$. Therefore, it can be

concluded that the general space complexity of AC-MOFOA is from $O(\max(N, N_{cand.}) \times k)$.

Figure 11 presents the results of comparing the mean execution time of AC-MOFOA with other multi-objective algorithms. The mean execution time of algorithms is calculated based on the 50 independent executions time on each of the datasets. Analyzing the AC-MOFOA execution time shows that this method had a shorter execution time than MOCEPO, MOBBBO, C-HMOSHSSA, and NSPSO. Due to its single-parent structure, AC-MOFOA only needs to select one parent among the Pareto Front members. Thus, it spends less time in the parent-selection step. Also, AC-MOFOA only needs to make limited changes in the structure of the parent tree to generate new trees, resulting in the reduced computational overhead of global and local seeding operators. Furthermore, implementing the chaos theory on local and global seeding operators increases the AC-MOFOA convergence speed. At the same time, it maintains the necessary diversity needed for effective search of the problem space. As a result of this, the execution time of AC-MOFOA can be reduced compare to the traditional methods. Among the methods compared, the MOSSO algorithm had a shorter execution time than AC-MOFOA in some scenarios because of using the Archive structure to maintain the Pareto front and being a single parent.

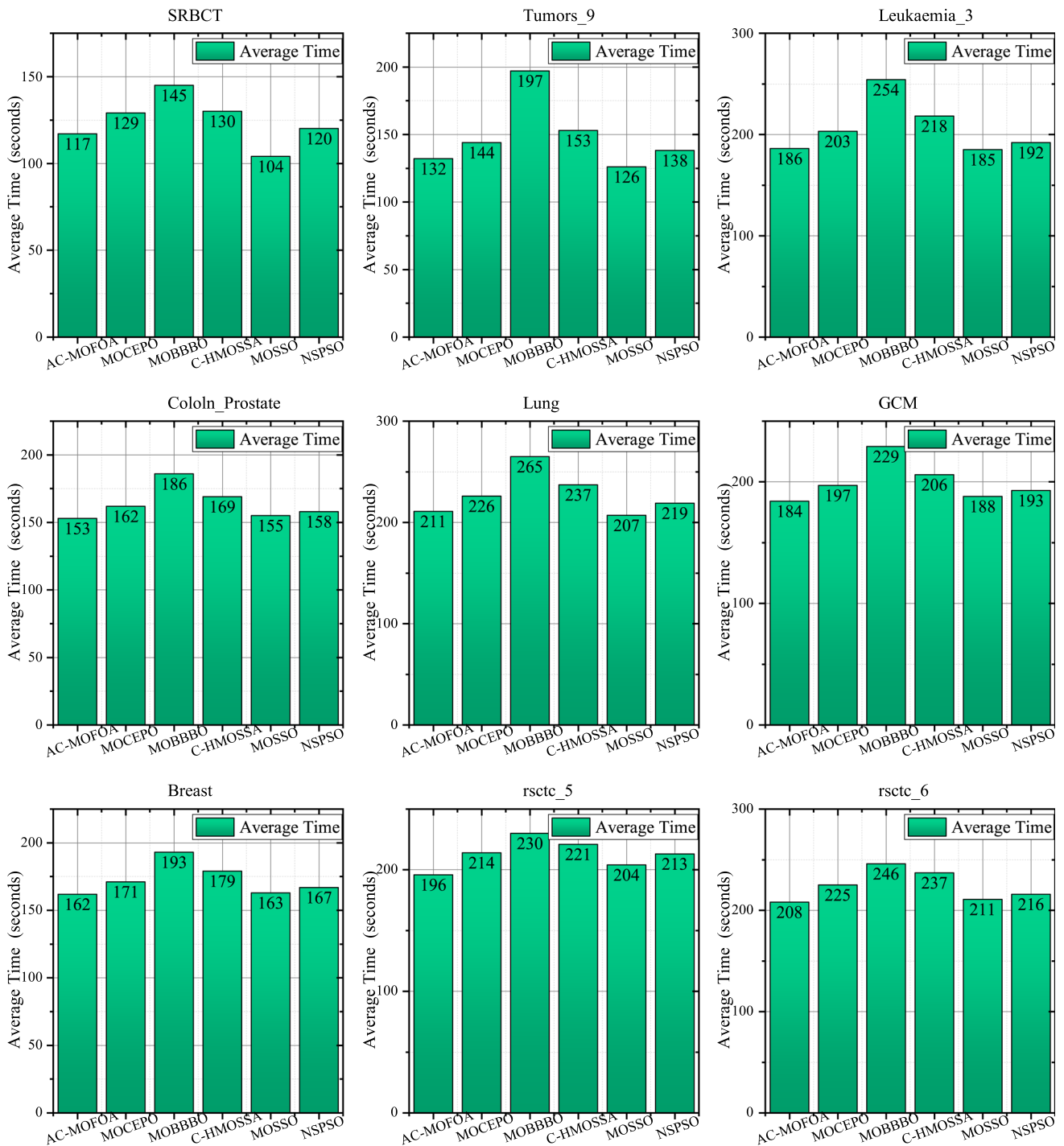


Fig.11 Comparison of the mean execution time of AC-MOFOA with other multi-objective algorithms (in seconds)

5.6 Studying the results of the ensemble classifier

In this section, we analyze the results of the Ensemble classifier made from members of the Pareto Front. The comparison results of the proposed Ensemble classifier with the three ensemble learning methods (i.e., *Adaboost*,

Bagging, and *Random Forest*) are shown in Table 12. Analysis of the results suggests that the proposed Ensemble classifier in all datasets could improve the classification accuracy. The proposed method also outperformed the other methods in 7 datasets. For example, in Breast dataset, the proposed method could improve the classification accuracy by more than 30% compared to the non-

optimized KELM. In fact, due to the optimization of the RBF kernel (γ) and ELM (C) function parameters, AC-MOFOA could improve the classification efficiency of each of the Pareto Front KELMs simultaneously with reducing the number of genes. Accordingly, the combination of improved KELMs could provide better results compared to conventional ensemble learning methods. However, the proposed method in two datasets of Leukemia3 and Lung yielded less accurate solutions than the Random Forest and Bagging methods.

For more analysis, we compared the proposed ensemble approach with three novel ensemble methods, i.e., NSGA-II [94], NSGA-II [95], and sp-MODE [96]. The results of the experiments have been illustrated in Fig. 12. According to Fig. 12, the proposed ensemble method has obtained better results than other methods in five datasets, namely: SRBCT, Leukemia3, Colon_Prostate, Breast, and Rscct_5. For example, in Leukemia3, the proposed method reached 97.86% classification accuracy, while the second-best method (i.e., NSGA-II [94]) achieved 96.49% accuracy. Moreover, in Tumors_9 and GCM, the proposed method and NSGA-II [95] acquired similar results. On the other hand, NSGA-II [94] in Rscct_6 and NSGA-II [95] in Lung obtained better classification accuracy than the proposed ensemble method. Based on the results, the proposed ensemble method is an effective approach in classifying the Microarray data and could provide similar or better results compared to other methods in most cases.

5.7 Further discussion

Based on the obtained results, the proposed method could successfully achieve the set objectives (i.e., reducing the dimensions of microarray datasets, increasing the accuracy of the classification, and constructing the Ensemble Classifier). The considered Multi-Filter includes five filter methods, including IG, Fisher-score, mRMR, CFS, and ReliefF. The selected combination includes univariate and multivariate Methods. In this system, univariate methods rank genes based on their individual relation with output.

On the other hand, multivariate methods rank genes based on their intra-relations and relation with output. By reducing the bias of single-filter methods, Multi-filter could select prominent and influential genes of the dataset and reduce dimensions of them to an acceptable level. Additionally, Multi-filter considers the dependency of each gene with output and redundancy between genes. Furthermore, AC-MOFOA simultaneously reduced the size of the gene subset and improved classification accuracy using the NS , crowding distance, chaos theory, adaptive local, and adaptive global seeding operators.

The use of adaptive local and global seeding operators enabled the AC-MOFOA to overcome the challenge of determining LSC and GSC , which leads to broader global and local search to find the optimal Pareto front in the early stages. Due to the implementation of the chaos theory concepts in local and global seeding operators, AC-MOFOA could achieve faster convergence compared to the traditional methods. Furthermore, the local seeding operator in AC-MOFOA uses the solutions of F_1 for generating new trees, which improves AC-MOFOA's search ability toward the optimal Pareto front. Besides, the use of NS and crowding distance enabled the algorithm to perform well in identifying the uniform and divers Pareto fronts. Finally, members of final Pareto front, including pairs of subset genes and KELM, were utilized to construct the Ensemble classifier. It is of note that the members of the Pareto front have the necessary diversity to construct an ensemble learner because of using different gene subsets and KELM classifiers with different optimized parameters. The results show that the proposed Ensemble classifier outperforms conventional and novel Ensemble classification methods in terms of classification accuracy and generalizability.

Moreover, by selecting the most prominent genes related to specific cancer diseases, medical scientists could focus on a small subset of genes to study the effects of each gene in causing cancer disease. Furthermore, the cost of experiments for diagnosing cancer will be reduced [22, 53, 107]. Additionally, the proposed ensemble classifier has high classification performance, which can be used

Table 12 Comparison results of proposed Ensemble Classifier with conventional ensemble methods

Datasets	KELM	Proposed ensemble classifier	Random forest	Adaboost	Bagging
SRBCT	88.23	98.65	91.97	95.18	94.59
Tumors_9	78.98	95.13	92.67	93.15	90.6
Leukaemia3	97.22	97.86	97.93	96.74	94.13
Colon_Prostate	95.74	96.92	96.08	95.27	95.78
Lung	90.24	97.6	92.14	91.61	98.04
GCM	70.53	79.11	74.24	72.05	76.19
Breast	68.04	97.45	82.51	88.36	91.31
Rscct_5	69.66	73.52	70.48	71.66	71.98
Rscct_6	81.523	83.94	79.73	80.73	82.64

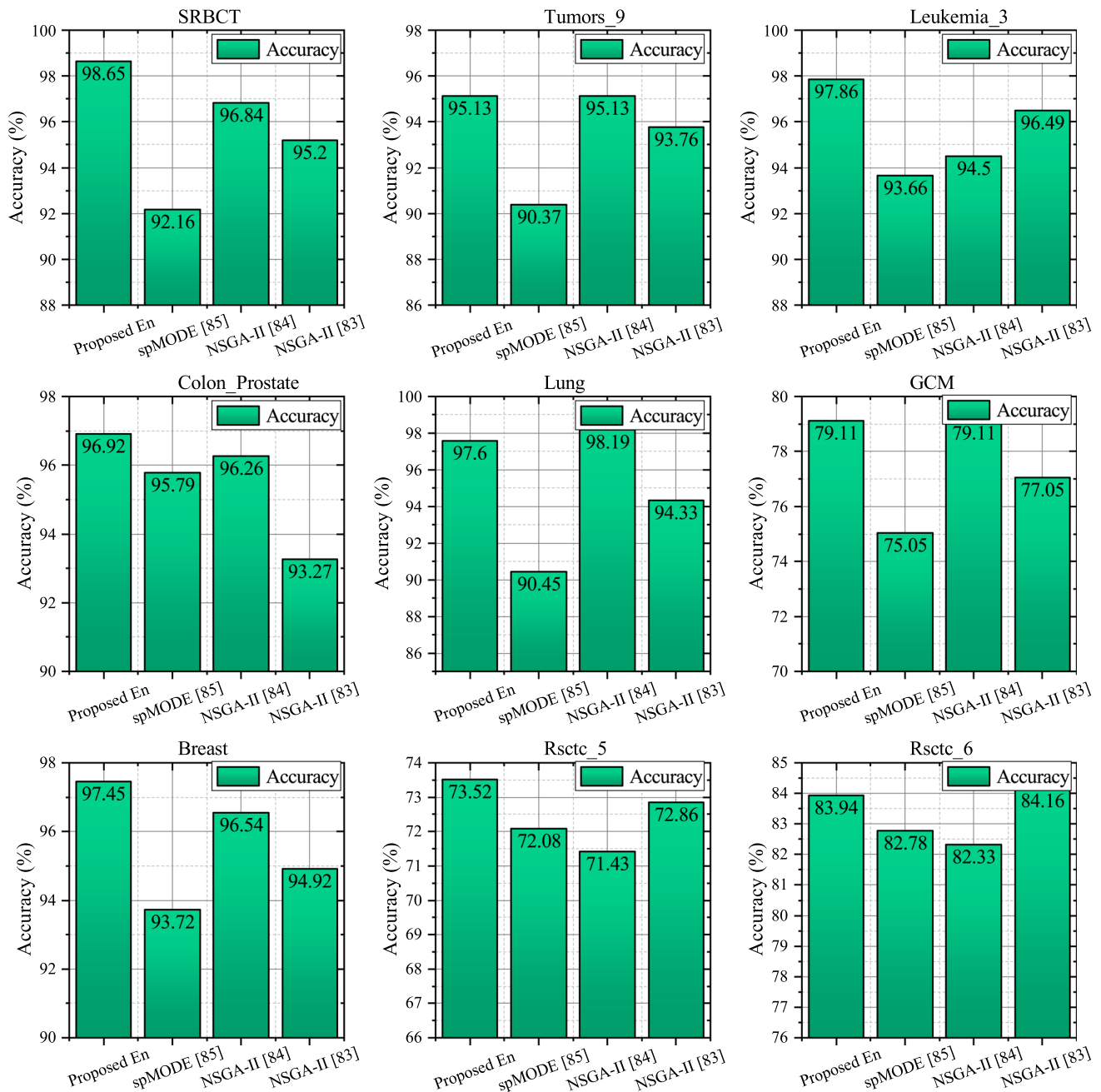


Fig.12 Comparison of the classification accuracy of the proposed ensemble approach with other recent ensemble approaches

for early diagnoses and prognosis of cancer. Accordingly, this could improve the treatment effectiveness and enhance the survival chance of the patients.

6 Conclusion

In the present study, a hybrid method of Multi-filter and AC-MOFOA was presented to solve the problem of gene selection and construct Ensemble Classifiers for the microarray datasets. Based on experiments and results, it

could successfully achieve the set objectives. The proposed multi-filter was developed by combining five filter methods, namely IG, Fisher-score, CFS, mRMR, and ReliefF, using the voting mechanism. The combination of several univariate and multivariate filter methods in constructing Multi-filter has reduced the bias effect compared to single filter mode and, as a result, the proposed Multi-filter as a pre-processing step could reduce the dimensions of the data and increase the classification accuracy. Moreover, the results of this step we compared with three multi/ensemble filter approaches. According to the results, the proposed

multi-filter could obtain better classification accuracy by selecting the prominent genes in six of nine datasets compared to other methods. Moreover, the proposed approach obtained the same results with other methods in Colon-Prostate and Breast, respectively. However, only in GCM, it achieved an accuracy worse than the method in [23]. In the second step of the proposed hybrid method, AC-MOFOA was presented using the concepts of *NS*, crowding distance, chaos theory, and adaptive operators. The second step of the proposed method's objectives was selecting the quasi-optimal subset of genes, optimizing the KELM classification parameters, and increasing the classification accuracy. AC-MOFOA uses *NS* to identify the Pareto front and rank the solutions. In addition, using the crowding distance, the ability to direct the AC-MOFOA search to the less crowded areas of the Pareto front increased, leading to a more uniform final Pareto front. Using chaos theory and adaptive local and global seeding, AC-MOFOA could improve its global and local search without needing to set the *LSC* and *GSC* parameters. Moreover, utilizing the logistic map as a chaos function in global and local seeding operators, the convergence speed and search diversity of AC-MOFOA have been improved. Finally, in the third step of the proposed method, the Ensemble Classifier model was formed using the final Pareto front KELM classifiers. To evaluate the effectiveness and efficiency of the proposed solution, the results were compared with three hybrid single-objective gene selection methods (i.e., GA, Adaptive GA, and TLBOGSA) and five hybrid multi-objective gene selection methods of MOSSO, MOCEPO, C-HMOSHSSA, NSPSO, and MOBBBO on nine microarray datasets with different dimensions. Based on the results, AC-MOFOA in most datasets could simultaneously increase classification accuracy by reducing the dimensions of the datasets. Moreover, the distribution analysis of the accuracy of solutions and the number of genes selected by AC-MOFOA based on statistical criteria confirm its effectiveness. Furthermore, the hypervolume indicator, spacing metric, and SCC analysis show that AC-MOFOA achieves better results on identifying the optimal Pareto front. For example, SCC metric analyses indicated that AC-MOFOA identifies more than 50% of the final Pareto front in test datasets. In addition, as can be seen from the Hypervolume ratio results, the AC-MOFOA results in all test datasets are significantly superior to the three methods of MOBBBO, NSPSO, and MOSSO. Moreover, MOCEPO provided similar results to AC-MOFOA only in the Lung dataset and showed worse performance in other cases.

Additionally, the proposed Ensemble Classifier model provides better performance for microarray data classification by increasing the accuracy of classification compared to conventional ensemble learning methods (i.e.,

Adaboost, *Bagging*, and *Random Forest*) and recent methods. Accordingly, the proposed ensemble method has obtained better results than other methods in five datasets: SRBCT, Leukemia3, Colon_Prostate, Breast, and Rscstc_5. For example, in Leukemia3, the proposed method reached 97.86% classification accuracy, while the second-best method achieved 96.49% accuracy. Besides, in Tumors_9 and GCM, the proposed method acquired similar results compared to other methods. In conclusion, AC-MOFOA could successfully solve the problem of gene selection and construction of the Ensemble classifier.

However, due to developments and advances in DNA microarray that have resulted in increasing dimensions and samples of such data, conventional computers will lose their ability to solve the problem of gene selection and microarray data classification. Hence, there is an urgent need to develop methods based on big data architecture. In future studies, we will review and develop scalable methods based on multi-objective metaheuristic algorithms in the big data context. The microarray data are usually highly imbalanced, which makes the classification task very challenging. To improve the classification performance, other classification metrics such as precision and specificity should be included in gene selection methods as objective functions. Due to the explosive growth of the non-dominated solution in the objective space with more than three objectives, the efficiency of multi-objective methods decreases [108, 109]. As a result, it is necessary to provide a new many-objective metaheuristic algorithm to solve the gene selection problem. Therefore, in future studies, we will focus on studying a many-objective version of the Forest optimization algorithm. From another perspective, deep learning models are proven to be very successful in solving many real-world problems. For future direction, using deep learning models such as deep ELM on microarray data classification can be considered by researchers. Optimizing the structure of deep ELM [110–112] can be done using metaheuristic algorithms, in which variable-length vectors can be utilized to present the different network structures. Solving the gene selection problem and deep ELM structure optimization simultaneously using low/high-level hybrid metaheuristics algorithms is an interesting research idea.

Declarations

Conflict of interest The authors declare no conflicts of interest.

References

1. Tyagi V, Mishra A (2013) A survey on different feature selection methods for microarray data analysis. *Int J Comput Appl* 67:36–40. <https://doi.org/10.5120/11482-7181>
2. Bolón-Canedo V, Sánchez-Marroño N, Alonso-Betanzos A et al (2014) A review of microarray datasets and applied feature selection methods. *Inf Sci (Ny)* 282:111–135. <https://doi.org/10.1016/j.ins.2014.05.042>
3. Almugren N, Alshamlan H (2019) A survey on hybrid feature selection methods in microarray gene expression data for cancer classification. *IEEE Access* 7:78533–78548. <https://doi.org/10.1109/ACCESS.2019.2922987>
4. Khalid S, Khalil T, Nasreen S (2014) A survey of feature selection and feature extraction techniques in machine learning. *Proc 2014 Sci Inf Conf SAI 2014* 372–378. <https://doi.org/10.1109/SAI.2014.6918213>
5. Remeseiro B, Bolon-Canedo V (2019) A review of feature selection methods in medical applications. *Comput Biol Med* 112:103375. <https://doi.org/10.1016/j.compbimed.2019.103375>
6. Nguyen BH, Xue B, Zhang M (2020) A survey on swarm intelligence approaches to feature selection in data mining. *Swarm Evol Comput* 54:100663. <https://doi.org/10.1016/j.swevo.2020.100663>
7. Amaldi E, Kann V (1998) On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theor Comput Sci* 209:237–260. [https://doi.org/10.1016/S0304-3975\(97\)00115-1](https://doi.org/10.1016/S0304-3975(97)00115-1)
8. Kohavi R, John GH (1997) Wrappers for feature subset selection. *Artif Intell* 97:273–324. [https://doi.org/10.1016/S0004-3702\(97\)00043-X](https://doi.org/10.1016/S0004-3702(97)00043-X)
9. Chandrashekar G, Sahin F (2014) A survey on feature selection methods. *Comput Electr Eng* 40:16–28. <https://doi.org/10.1016/j.compeleceng.2013.11.024>
10. Lai CM (2018) Multi-objective simplified swarm optimization with weighting scheme for gene selection. *Appl Soft Comput J* 65:58–68. <https://doi.org/10.1016/j.asoc.2017.12.049>
11. Dash R, Misra BB (2018) A multi-objective feature selection and classifier ensemble technique for microarray data analysis. *Int J Data Min Bioinform* 20:123–160. <https://doi.org/10.1504/IJDMB.2018.093683>
12. Osanaiye O, Cai H, Choo K-KR et al (2016) Ensemble-based multi-filter feature selection method for DDoS detection in cloud computing. *EURASIP J Wirel Commun Netw* 2016:130. <https://doi.org/10.1186/s13638-016-0623-3>
13. Balogun AO, Basri S, Abdulkadir SJ, Sobri AH (2019) A hybrid multi-filter wrapper feature selection method for software defect predictors
14. Miao J, Niu L (2016) A survey on feature selection. *Procedia Comput Sci* 91:919–926. <https://doi.org/10.1016/j.procs.2016.07.111>
15. Lu H, Chen J, Yan K et al (2017) A hybrid feature selection algorithm for gene expression data classification. *Neurocomputing* 256:56–62. <https://doi.org/10.1016/j.neucom.2016.07.080>
16. Chuang LY, Yang CH, Wu KC, Yang CH (2011) A hybrid feature selection method for DNA microarray data. *Comput Biol Med* 41:228–237. <https://doi.org/10.1016/j.compbimed.2011.02.004>
17. Yang CH, Chuang LY, Yang CH (2010) IG-GA: a hybrid filter/wrapper method for feature selection of microarray data. *J Med Biol Eng* 30:23–28
18. Shukla AK, Singh P, Vardhan M (2018) A hybrid gene selection method for microarray recognition. *Biocybern Biomed Eng* 38:975–991. <https://doi.org/10.1016/j.bbe.2018.08.004>
19. Nguyen T, Khosravi A, Creighton D, Nahavandi S (2015) A novel aggregate gene selection method for microarray data classification. *Pattern Recognit Lett* 60–61:16–23. <https://doi.org/10.1016/j.patrec.2015.03.018>
20. Dashtban M, Balafar M (2017) Gene selection for microarray cancer classification using a new evolutionary method employing artificial intelligence concepts. *Genomics* 109:91–107. <https://doi.org/10.1016/j.ygeno.2017.01.004>
21. Lee CP, Leu Y (2011) A novel hybrid feature selection method for microarray data analysis. *Appl Soft Comput J* 11:208–213. <https://doi.org/10.1016/j.asoc.2009.11.010>
22. Jansi Rani M, Devaraj D (2019) Two-stage hybrid gene selection using mutual information and genetic algorithm for cancer data classification. *J Med Syst.* <https://doi.org/10.1007/s10916-019-1372-8>
23. Ghosh M, Adhikary S, Ghosh KK et al (2019) Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods. *Med Biol Eng Comput* 57:159–176. <https://doi.org/10.1007/s11517-018-1874-4>
24. Shukla AK, Singh P, Vardhan M (2018) A two-stage gene selection method for biomarker discovery from microarray data for cancer classification. *Chemom Intell Lab Syst* 183:47–58. <https://doi.org/10.1016/j.chemolab.2018.10.009>
25. Sayed S, Nassef M, Badr A, Farag I (2019) A nested genetic algorithm for feature selection in high-dimensional cancer microarray datasets. *Expert Syst Appl* 121:233–243. <https://doi.org/10.1016/j.eswa.2018.12.022>
26. Jain I, Jain VK, Jain R (2018) Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. *Appl Soft Comput J* 62:203–215. <https://doi.org/10.1016/j.asoc.2017.09.038>
27. Shen Q, Shi WM, Kong W (2008) Hybrid particle swarm optimization and tabu search approach for selecting genes for tumor classification using gene expression data. *Comput Biol Chem* 32:53–60. <https://doi.org/10.1016/j.compbiolchem.2007.10.001>
28. Baliarsingh SK, Vipsita S, Dash B (2020) A new optimal gene selection approach for cancer classification using enhanced Jaya-based forest optimization algorithm. *Neural Comput Appl* 32:8599–8616. <https://doi.org/10.1007/s00521-019-04355-x>
29. Apolloni J, Leguizamón G, Alba E (2016) Two hybrid wrapper-filter feature selection algorithms applied to high-dimensional microarray experiments. *Appl Soft Comput* 38:922–932. <https://doi.org/10.1016/j.asoc.2015.10.037>
30. Bir-Jmel A, Douiri SM, Elberoussi S (2019) Gene selection via a new hybrid ant colony optimization algorithm for cancer classification in high-dimensional data. *Comput Math Methods Med* 2019:1–20. <https://doi.org/10.1155/2019/7828590>
31. Vafae Sharbaf F, Mosafer S, Moattar MH (2016) A hybrid gene selection approach for microarray data classification using cellular learning automata and ant colony optimization. *Genomics* 107:231–238. <https://doi.org/10.1016/j.ygeno.2016.05.001>
32. Dash R, Dash R, Rautray R (2019) An evolutionary framework based microarray gene selection and classification approach using binary shuffled frog leaping algorithm. *J. King Saud Univ Comput Inf Sci.* <https://doi.org/10.1016/j.jksuci.2019.04.002>
33. Shukla AK, Singh P, Vardhan M (2020) Gene selection for cancer types classification using novel hybrid metaheuristics approach. *Swarm Evol Comput.* <https://doi.org/10.1016/j.swevo.2020.100661>
34. Elyasiogomari V, Lee DA, Screen HRC, Shaheed MH (2017) Development of a two-stage gene selection method that incorporates a novel hybrid approach using the cuckoo optimization

- algorithm and harmony search for cancer classification. *J Biomed Inform* 67:11–20. <https://doi.org/10.1016/j.jbi.2017.01.016>
35. Fei L, Juan L (2008) Optimal genes selection with a new multi-objective evolutionary algorithm hybridizing NSGA-II with EDA. *Biomed Eng Informatics New Dev Futur - Proc 1st Int Conf Biomed Eng Informatics, BMEI 2008* 1:327–331. <https://doi.org/10.1109/BMEI.2008.313>
 36. Hasnat A, Molla AU (2017) Feature selection in cancer microarray data using multi-objective genetic algorithm combined with correlation coefficient. *Proc IEEE Int Conf Emerg Technol Trends Comput Commun Electr Eng ICETT*. <https://doi.org/10.1109/ICETT.2016.7873741>
 37. Banerjee M, Mitra S, Banka H (2007) Evolutionary rough feature selection in gene expression data. *IEEE Trans Syst Man Cybern Part C Appl Rev* 37:622–632. <https://doi.org/10.1109/TSMCC.2007.897498>
 38. Mohamad MS, Omatu S, Deris S, Yoshioka M (2008) Multi-objective optimization using genetic algorithm for gene selection from microarray data. *Proc Int Conf Comput Commun Eng 2008, ICCCE08 Glob Links Hum Dev* 1331–1334. <https://doi.org/10.1109/ICCCE.2008.4580821>
 39. Annavarapu CSR, Dara S, Banka H (2016) Cancer microarray data feature selection using multi-objective binary particle swarm optimization algorithm. *Excli J* 15:460–473. <https://doi.org/10.17179/excli2016-481>
 40. Chakraborty G, Chakraborty B (2013) Multi-objective optimization using pareto GA for gene-selection from microarray data for disease classification. *Proc - 2013 IEEE Int Conf Syst Man Cybern SMC 2013*:2629–2634. <https://doi.org/10.1109/SMC.2013.449>
 41. Divya S, Kiran ELN, Rao MS, Vemulapati P (2020) Prediction of Gene Selection Features Using Improved Multi-objective Spotted Hyena Optimization Algorithm. In: *Advances in Intelligent Systems and Computing*. pp 59–67
 42. Sharma A, Rani R (2019) C-HMOSHSSA: gene selection for cancer classification using multi-objective meta-heuristic and machine learning methods. *Comput Method Progr Biomed* 178:219–235. <https://doi.org/10.1016/j.cmpb.2019.06.029>
 43. Ratnoo S, Ahuja J (2017) Dimension reduction for microarray data using multi-objective ant colony optimisation. *Int J Comput Syst Eng* 3:58. <https://doi.org/10.1504/ijcsyse.2017.10004024>
 44. Yang P, Hwa Yang Y, B Zhou B, Y Zomaya A (2010) A review of ensemble methods in bioinformatics. *Curr Bioinform* 5:296–308. <https://doi.org/10.2174/157489310794072508>
 45. Lan K, Tong WD, Fong S et al (2018) A survey of data mining and deep learning in bioinformatics. *J Med Syst*. <https://doi.org/10.1007/s10916-018-1003-9>
 46. Khoshgoftaar TM, Dittman DJ, Wald R, Awada W (2013) A review of ensemble classification for DNA microarrays data. *Proc Int Conf Tools with Artif Intell ICTAI*. <https://doi.org/10.1109/ICTAI.2013.64>
 47. Amaratunga D, Cabrera J, Lee Y-S (2008) Enriched random forests. *Bioinformatics* 24:2010–2014
 48. Polikar R (2012) Ensemble learning. In: Zhang C, Ma Y (eds) *Ensemble machine learning*. Springer, pp 1–34
 49. Xue X, Yao M, Wu Z (2018) A novel ensemble-based wrapper method for feature selection using extreme learning machine and genetic algorithm. *Knowl Inf Syst* 57:389–412. <https://doi.org/10.1007/s10115-017-1131-4>
 50. Huda S, Yearwood J, Jelinek HF et al (2016) A hybrid feature selection with ensemble classification for imbalanced healthcare data: a case study for brain tumor diagnosis. *IEEE Access* 4:9145–9154. <https://doi.org/10.1109/ACCESS.2016.2647238>
 51. Fallahpour S, Lakvan EN, Zadeh MH (2017) Using an ensemble classifier based on sequential floating forward selection for financial distress prediction problem *J. Retail Consum Serv* 34:159–167. <https://doi.org/10.1016/j.jretconser.2016.10.002>
 52. Franken H, Lehmann R, Häring HU, et al (2011) Wrapper- and ensemble-based feature subset selection methods for biomarker discovery in targeted metabolomics. *Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics)* 7036 LNBI:121–132. https://doi.org/10.1007/978-3-642-24855-9_11
 53. Panthong R, Srivihok A (2015) Wrapper feature subset selection for dimension reduction based on ensemble learning algorithm. *Procedia Comput Sci* 72:162–169. <https://doi.org/10.1016/j.procs.2015.12.117>
 54. Yang P, Liu W, Zhou BB, et al (2013) Ensemble-based wrapper methods for feature selection and class imbalance learning. *Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics)* 7818 LNAI:544–555. https://doi.org/10.1007/978-3-642-37453-1_45
 55. Yu E, Cho S (2006) Ensemble based on GA wrapper feature selection. *Comput Ind Eng* 51:111–116. <https://doi.org/10.1016/j.cie.2006.07.004>
 56. Idris A, Khan A (2017) Churn prediction system for telecom using filter-wrapper and ensemble classification. *Comput J* 60:410–430. <https://doi.org/10.1093/comjnl/bxv123>
 57. Zhou Y, Cheng G, Jiang S, Dai M (2020) Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Comput Networks* 174:107247. <https://doi.org/10.1016/j.comnet.2020.107247>
 58. Mukhopadhyay A, Member S, Maulik U, Member S (2014) A survey of multiobjective evolutionary algorithms for data mining: Part I. *IEEE Trans Evol Comput* 18:4–19
 59. Baliarsingh SK, Vipsita S, Muhammad K, Bakshi S (2019) Analysis of high-dimensional biomedical data using an evolutionary multi-objective emperor penguin optimizer. *Swarm Evol Comput* 48:262–273. <https://doi.org/10.1016/j.swevo.2019.04.010>
 60. Li X, Yin M (2013) Multiobjective binary biogeography based optimization for feature selection using gene expression data. *IEEE Trans Nanobioscience* 12:343–353. <https://doi.org/10.1109/TNB.2013.2294716>
 61. Ghaemi M, Feizi-Derakhshi M-R (2014) Forest optimization algorithm. *Expert Syst Appl* 41:6676–6687. <https://doi.org/10.1016/j.eswa.2014.05.009>
 62. Ghaemi M, Feizi-Derakhshi M-R (2016) Feature selection using forest optimization algorithm. *Pattern Recognit* 60:121–129. <https://doi.org/10.1016/j.patcog.2016.05.012>
 63. Huang G-B, Zhu Q-Y, Siew C-K (2006) Extreme learning machine: theory and applications. *Neurocomputing* 70:489–501
 64. Bin HG, Wang DH, Lan Y (2011) Extreme learning machines: a survey. *Int J Mach Learn Cybern* 2:107–122. <https://doi.org/10.1007/s13042-011-0019-y>
 65. Marler RT, Arora JS (2004) Survey of multi-objective optimization methods for engineering. *Struct Multidiscip Optim* 26:369–395. <https://doi.org/10.1007/s00158-003-0368-6>
 66. Coello CAC, Lamont GB, Van Veldhuizen DA (2007) *Evolutionary algorithms for solving multi-objective problems*. Springer, US, Boston, MA
 67. Hall MA (1999) *Correlation-based Feature Selection for Machine Learning*
 68. Kira K, Rendell LA (1992) A Practical Approach to Feature Selection. In: *Machine Learning Proceedings 1992*. Elsevier, pp 249–256
 69. Shreem SS, Abdullah S, Nazri MZA (2014) Hybridising harmony search with a Markov blanket for gene selection problems. *Inf Sci (Ny)* 258:108–121. <https://doi.org/10.1016/j.ins.2013.10.012>

70. Gangavarapu T, Patil N (2019) A novel filter–wrapper hybrid greedy ensemble approach optimized using the genetic algorithm to reduce the dimensionality of high-dimensional biomedical datasets. *Appl Soft Comput J* 81:105538. <https://doi.org/10.1016/j.asoc.2019.105538>
71. Ozger ZB, Bolat B, Diri B (2019) A probabilistic multi-objective artificial bee colony algorithm for gene selection. *J Univers Comput Sci* 25:418–443
72. Abo-Hammour Z, Alsmadi O, Momani S, Abu Arqub O (2013) A genetic algorithm approach for prediction of linear dynamical systems. *Math Probl Eng* 2013:1–12. <https://doi.org/10.1155/2013/831657>
73. Abo-Hammour Z, Abu Arqub O, Alsmadi O et al (2014) An optimization algorithm for solving systems of singular boundary value problems. *Appl Math Inf Sci* 8:2809–2821. <https://doi.org/10.12785/amis/080617>
74. Momani S, Abo-Hammou ZS, Alsmad OM (2016) Solution of inverse kinematics problem using genetic algorithms. *Appl Math Inf Sci* 10:225–233
75. Abu Arqub O, Abo-Hammour Z, Momani S, Shawagfeh N (2012) Solving singular two-point boundary value problems using continuous genetic algorithm. *Abstr Appl Anal*. <https://doi.org/10.1155/2012/205391>
76. Hammad M, Iliyasa AM, Subasi A et al (2021) A multitier deep learning model for arrhythmia detection. *IEEE Trans Instrum Meas*. <https://doi.org/10.1109/TIM.2020.3033072>
77. Sedik A, Hammad M, Abd El-Samie FE et al (2021) Efficient deep learning approach for augmented detection of coronavirus disease. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-020-05410-8>
78. Alghamdi AS, Polat K, Alghoson A et al (2020) A novel blood pressure estimation method based on the classification of oscillometric waveforms using machine-learning methods. *Appl Acoust* 164:107279. <https://doi.org/10.1016/j.apacoust.2020.107279>
79. Hammad M, Alkinani MH, Gupta BB, Abd El-Latif AA (2021) Myocardial infarction detection based on deep neural network on imbalanced data. *Multimed Syst*. <https://doi.org/10.1007/s00530-020-00728-8>
80. Hancer E, Xue B, Zhang M et al (2018) Pareto front feature selection based on artificial bee colony optimization. *Inf Sci (Ny)* 422:462–479. <https://doi.org/10.1016/j.ins.2017.09.028>
81. Agarwalla P, Mukhopadhyay S (2018) Feature selection using multi-objective optimization technique for supervised cancer classification. In: Mandal JK, Mukhopadhyay S, Dutta P (eds) *Multi-objective optimization*. Springer, pp 195–213
82. Nouri-Moghaddam B, Ghazanfari M, Fathian M (2020) A novel filter-wrapper hybrid gene selection approach for microarray data based on multi-objective forest optimization algorithm. *Decis Sci Lett* 9:271–290. <https://doi.org/10.5267/j.dsl.2020.5.006>
83. Shahbeig S, Rahideh A, Helfroush MS, Kazemi K (2018) Gene selection from large-scale gene expression data based on fuzzy interactive multi-objective binary optimization for medical diagnosis. *Biocybern Biomed Eng* 38:313–328. <https://doi.org/10.1016/j.bbe.2018.02.002>
84. Mishra S, Shaw K, Mishra D (2012) A new meta-heuristic bat inspired classification approach for microarray data. *Procedia Technol* 4:802–806. <https://doi.org/10.1016/j.protcy.2012.05.131>
85. Dashtban M, Balafar M, Suravajhala P (2018) Gene selection for tumor classification using a novel bio-inspired multi-objective approach. *Genomics* 110:10–17. <https://doi.org/10.1016/j.ygeno.2017.07.010>
86. Nouri-Moghaddam B, Ghazanfari M, Fathian M (2021) A novel multi-objective forest optimization algorithm for wrapper feature selection. *Expert Syst Appl* 175:114737. <https://doi.org/10.1016/j.eswa.2021.114737>
87. Javidi M, Hosseinpoufard R (2015) Chaos genetic algorithm instead genetic algorithm. *Int Arab J Inf Technol* 12:163–168
88. Gharooni-fard G, Moein-darbari F, Deldari H, Morvaridi A (2010) Scheduling of scientific workflows using a chaos-genetic algorithm. *Procedia Comput Sci* 1:1445–1454. <https://doi.org/10.1016/j.procs.2010.04.160>
89. Snaselova P, Zboril F (2015) Genetic algorithm using theory of chaos. *Procedia Comput Sci* 51:316–325. <https://doi.org/10.1016/j.procs.2015.05.248>
90. Baliarsingh SK, Vipsita S, Muhammad K et al (2019) Analysis of high-dimensional genomic data employing a novel bio-inspired algorithm. *Appl Soft Comput* 77:520–532. <https://doi.org/10.1016/j.asoc.2019.01.007>
91. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6:182–197
92. Zhu Z, Ong YS, Dash M (2007) Markov blanket-embedded genetic algorithm for gene selection. *Pattern Recognit* 40:3236–3248. <https://doi.org/10.1016/j.patcog.2007.02.007>
93. OpenML. <https://www.openml.org/>
94. Singh N, Singh P (2020) Stacking-based multi-objective evolutionary ensemble framework for prediction of diabetes mellitus. *Biocybern Biomed Eng* 40:1–22. <https://doi.org/10.1016/j.bbe.2019.10.001>
95. Asadi S, Roshan SE (2021) A bi-objective optimization method to produce a near-optimal number of classifiers and increase diversity in Bagging. *Knowl Based Syst* 213:106656. <https://doi.org/10.1016/j.knsys.2020.106656>
96. Ribeiro VHA, Reynoso-Meza G (2020) Ensemble learning by means of a multi-objective optimization design approach for dealing with imbalanced data sets. *Expert Syst Appl* 147:113232
97. Taguchi G, Chowdhury S, Wu Y (2005) *Taguchi's quality engineering handbook*. Wiley
98. Xue B, Zhang M, Browne WN (2013) Particle swarm optimization for feature selection in classification: a multi-objective approach. *IEEE Trans Cybern* 43:1656–1671. <https://doi.org/10.1109/TSMCB.2012.2227469>
99. Xue B, Fu W, Zhang M (2014) Multi-objective feature selection in classification: a differential evolution approach. In: *Asia-Pacific Conference on Simulated Evolution and Learning*. pp 516–528
100. Hancer E, Xue B, Zhang M, et al (2015) A multi-objective artificial bee colony approach to feature selection using fuzzy mutual information. *2015 IEEE Congr Evol Comput CEC 2015 Proc* 2420–2427. <https://doi.org/10.1109/CEC.2015.7257185>
101. Xue B, Zhang M, Browne WN (2012) Multi-objective particle swarm optimisation (PSO) for feature selection. *Proc fourteenth Int Conf Genet Evol Comput Conf GECCO '12* 81. <https://doi.org/10.1145/2330163.2330175>
102. Amoozegar M, Minaei-Bidgoli B (2018) Optimizing multi-objective PSO based feature selection method using a feature elitism mechanism. *Expert Syst Appl* 113:499–514. <https://doi.org/10.1016/j.eswa.2018.07.013>
103. Auger A, Bader J, Brockhoff D, Zitzler E (2009) Theory of the hypervolume indicator: Optimal μ -distributions and the choice of the reference point. In: *Proceedings of the 10th ACM SIGEVO Workshop on Foundations of Genetic Algorithms, FOGA'09*. ACM Press, New York, New York, USA, pp 87–102
104. Brockhoff D, Friedrich T, Neumann F (2008) Analyzing hypervolume indicator based algorithms. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. pp 651–660

105. Sierra MR, Coello Coello CA (2005) Improving PSO-Based Multi-objective Optimization Using Crowding, Mutation and ϵ -Dominance. In: International conference on evolutionary multi-criterion optimization. Springer, pp 505–519
106. Jensen MT (2003) Reducing the run-time complexity of multi-objective EAs: the NSGA-II and other algorithms. *IEEE Trans Evol Comput* 7:503–515. <https://doi.org/10.1109/TEVC.2003.817234>
107. Lv J, Peng Q, Chen X, Sun Z (2016) A multi-objective heuristic algorithm for gene expression microarray data classification. *Expert Syst Appl* 59:13–19. <https://doi.org/10.1016/j.eswa.2016.04.020>
108. Wang Y, Sun X (2018) A many-objective optimization algorithm based on weight vector adjustment. *Comput Intell Neurosci*. <https://doi.org/10.1155/2018/4527968>
109. Chand S, Wagner M (2015) Evolutionary many-objective optimization: a quick-start guide. *Surv Oper Res Manag Sci* 20:35–42. <https://doi.org/10.1016/j.sorms.2015.08.001>
110. Khan B, Wang Z, Han F et al (2017) Fabric weave pattern and yarn color recognition and classification using a deep ELM network. *Algorithms*. <https://doi.org/10.3390/a10040117>
111. Tissera MD, McDonnell MD (2016) Deep extreme learning machines: supervised autoencoding architecture for classification. *Neurocomputing* 174:42–49. <https://doi.org/10.1016/j.neucom.2015.03.110>
112. Jiang XW, Yan TH, Zhu JJ et al (2020) Densely connected deep extreme learning machine algorithm. *Cognit Comput* 12:979–990. <https://doi.org/10.1007/s12559-020-09752-2>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.