The Journal of Supercomputing (2022) 78:15598–15637
https://doi.org/10.1007/s11227-022-04507-2

Hybrid binary arithmetic optimization algorithm with simulated annealing for feature selection in high-dimensional biomedical data

Elham Pashaei[1] · Elnaz Pashaei[2]

Accepted: 6 April 2022 / Published online: 20 April 2022

Abstract

Gene expression data play a significant role in the development of effective cancer diagnosis and prognosis techniques. However, many redundant, noisy, and irrelevant genes (features) are present in the data, which negatively affect the predictive accuracy of diagnosis and increase the computational burden. To overcome these challenges, a new hybrid filter/wrapper gene selection method, called mRMR-BAOAC-SA, is put forward in this article. The suggested method uses Minimum Redundancy Maximum Relevance (mRMR) as a first-stage filter to pick top-ranked genes. Then, Simulated Annealing (SA) and a crossover operator are introduced into Binary Arithmetic Optimization Algorithm (BAOA) to propose a novel hybrid wrapper feature selection method that aims to discover the smallest set of informative genes for classification purposes. BAOAC-SA is an enhanced version of the BAOA in which SA and crossover are used to help the algorithm in escaping local optima and enhancing its global search capabilities. The proposed method was evaluated on 10 well-known microarray datasets, and its results were compared to other current state-of-the-art gene selection methods. The experimental results show that the proposed approach has a better performance compared to the existing methods in terms of classification accuracy and the minimum number of selected genes.

Keywords Cancer classification · Feature selection · Arithmetic optimization algorithm · Gene selection · Optimization

✉ Elham Pashaei
elham.pashaei@gmail.com; epashaei@gelisim.edu.tr

Elnaz Pashaei
elnazpashaei@aydin.edu.tr

[1] Department of Computer Engineering, Istanbul Gelisim University, Istanbul, Turkey

[2] Department of Software Engineering, Istanbul Aydin University, Istanbul, Turkey

1 Introduction

As biomedical research and healthcare continue to advance in the genomic era, one of the most critical challenges facing bioinformatics is the high dimensionality of genomic data. Limited sample size and imbalanced class in data make it even more difficult. DNA microarray gene expression datasets are good examples of such high dimensional data [1]. Microarray datasets have offered expression measurements on tens of thousands of genes (dimensions), opening up new avenues for prognosis and diagnosis of life-threatening diseases like cancer, Alzheimer's, diabetes, etc. The extraction of disease-related biomarkers and the precise classification of cancer types are the two most important applications of microarray data. However, a large proportion of genes in microarray data are redundant and irrelevant for clinical use. Extracting disease-related information from a massive amount of redundant data and noise is a challenging task. It negatively influences the accuracy and computational cost of classification [2, 3] and makes biomarker discovery and data interpretation more difficult [4]. Therefore, effective extraction of the informative genes from a high dimensional microarray dataset referred to as gene (feature) selection, is crucial. Gene selection (GS) plays an important role in achieving better classification accuracy, increasing data interpretability, and reducing clinical costs by identifying the most promising gene subset. GS is the most challenging preprocessing step in biomedical data analysis due to the existence of a huge number of genes, a small number of patient samples, and complicated gene interactions [2]. It's been already proven that finding the best gene subsets is an NP-hard problem [5, 6]. All this motivates researchers to search for possible solutions and to suggest different algorithms. Although several Nature-inspired optimization algorithms (NIOAs) have already been studied for gene selection, due to the complicated interplay between the genes and large-scale search space, most of the NIOA based GS techniques suffer from the stagnation problem and often drop in local optima [1, 5–8]. Moreover, because the NIOAs based strategies are stochastic, there is no certainty that the best set of genes will be found in GS problems. Therefore, there is still a need for powerful search strategies to improve the performance of gene selection. One of the efficient NIOAs that has been recently suggested is Arithmetic Optimization Algorithm (AOA).

AOA, introduced by Abualigah et al. [9], has been inspired by the distribution behavior of the four basic arithmetic operators: multiplication, division, subtraction, and addition. AOA has some advantages over other NIOAs such as simplicity, flexibility, easy implementation, less number of parameters, and local optima avoidance. As a result, AOA and its binary [10], enhanced [11–13], hybrid [14, 15] versions have been successfully applied to a variety of optimization issues, including global optimization problems [11–13], image processing [16], neural network training [17], and feature selection [10, 14, 15]. However, to our knowledge, binary AOA (BAOA) is not yet properly investigated for GS and cancer classification problems.

The Simulated Annealing (SA) algorithm, on the other hand, is a hill-climbing method that is often employed in conjunction with other NIOAs to overcome the problem of local optima stagnation and so handle feature selection problems more effectively. Some of the examples include the hybrid of SA with Whale Optimization Algorithm (WOA) [18], SA with binary Corel Reefs Optimization (BCRO) [8], SA with Harris Hawks Optimization (HHO) algorithm [19], SA with Salp Swarm Algorithm (SSA) [20], SA with binary Dragonfly Algorithm (BDA or BDF) [21], SA with Teaching–Learning-based Optimization (TLBO) [22], and SA with Grey Wolf Optimizer (GWO) [23]. This inspired us to combine the SA with the BAOA to find robust and stable discriminative genes for cancer classification due to the difficulties of the learning process induced by the intricate interaction between the genes and the huge gene search space.

This paper presents a new BAOA-based hybrid GS approach, called mRMR-BAOAC-SA. The Minimum Redundancy Maximum Relevance (mRMR) [5] filter approach is initially used in the suggested hybrid method to find top-ranked genes in order to fine-tune the search space to the wrapper method. Then, hybrid BAOA with crossover operator (C) and SA algorithm is utilized as a new wrapper GS approach to efficiently tackle the gene selection problem and find the best gene subset. SA is combined with BAOA to enhance the exploitation property of BAOA and avoid falling into the optima problem, while the crossover operator improves the global search ability of BAOA and accelerates the convergence speed. Note that Random Forest (RF) [24] classifier is employed to evaluate each candidate gene subset in BAOAC-SA. Especially, the main contribution of this work can be summarized as follows:

- This is the first time the BAOA has been applied to gene selection where two versions of BAOA are proposed (i.e. BAOAC and BAOAC-SA).
- The crossover operator and SA are added to BAOA in order to enhance the local exploitation of the algorithm.
- Hybridization of mRMR and BAOAC-SA is suggested for gene selection utilizing an RF classifier.
- The performance of the suggested method (mRMR-BAOAC-SA) is investigated to see whether it can obtain the smallest subset of genes with higher classification accuracy when compared with current state-of-the-art GS methods.

To assess the efficacy of the proposed method, extensive experiments were carried out on ten widely used microarray datasets. The conducted experiments demonstrate that the suggested hybrid mRMR-BAOASA has a better performance compared to current state-of-the-art techniques in terms of accuracy and number of selected genes.

The paper's organization is as follows: Sect. 2 provides a literature review of the GS techniques. Section 3 briefly introduces the AOA, SA algorithm, and crossover operator. The primary concepts and details of the suggested mRMR-BAOAC-SA schema are elaborated in Sect. 4. The performed experiments on well-known datasets and achieved results are presented and analyzed in Sect. 5. Finally, Sect. 6 presents the conclusions and future direction.

2 Related works

There are three types of GS techniques in literature: filter, wrapper, and hybrid methods. The filter models are independent of any classifier and instead rely on interior properties of training data such as distance, information theory, and probability distribution. As a result, they are extremely fast when dealing with high-dimensional microarray data [25]. Filter methods can be divided into two categories: univariate and multivariate approaches. Univariate filter models assign a rank to each gene independently of the other genes, while multivariate filter approaches consider the possible dependency between genes to rank the significance of genes [26]. The highest-ranking genes can then be chosen for future analysis. Some of the widely used filter approaches are Fisher-score [4], mRMR [5], Information Gain (IG) [27], Mutual Information (MI)[7], etc.

The wrapper methods use a classifier to evaluate the quality of gene subsets in the search space iteratively. The search strategy and evaluation are the two basic components of wrapper approaches. The predictive accuracy of candidate gene subsets generated by the search strategy is evaluated using the machine learning classifiers. An increase in the number of genes causes the gene search space to grow exponentially, which gives rise to a laborious search over all possible combinations of features [28]. NIOAs have been devoted as search techniques in wrapper methods to solve various optimization problems. Some of them are binary Black Hole Algorithm (BBHA) [2, 29], GWO [6], Moth Flame Optimization Algorithm (MFOA) [7], Harmony Search Algorithm (HSA) [30], TLBO [31], Ant Colony Optimization (ACO) [32], WOA [33], etc. [34–36]. Wrapper approaches are more computationally expensive than filter methods, but in terms of classification accuracy, they outperform filter methods. Hybrid methods combine filter and wrapper approaches to take advantage of the computational efficiency of filter methods as well as the wrapper methods' proper performance [1]. However, the hybrid methods are still in their infancy, and more research is needed to develop more efficient hybrid approaches.

Various hybrid methods have been introduced in the literature to solve GS problems. A hybrid GS method based on the Binary Jaya algorithm and a multi-attribute decision-making (MADM) method was proposed by Chaudhuri and Sahu [1]. Technique for order preference by similarity to ideal solution (TOPSIS) approach, as a MADM method, is used to filter noisy genes at the first stage. A binary Jaya algorithm with a time-varying transfer function is utilized in the second stage to choose informative genes. In [2], the authors introduced a GS approach based on the BDF and BBHA using a Support Vector Machine (SVM). The mRMR filtering approach is first applied to eliminate noisy and redundant genes, then the BDF-BBHA wrapper approach is employed to find the best genes. Zhang et al. [3] proposed a hybrid strategy based on IG and the modified binary krill herd (MBKH) algorithm. A hyperbolic tangent transfer function, an adaptive transfer factor, and a chaos memory weight factor were introduced into MBKH to facilitate a better searching of the possible gene subsets. Dashtban and Balafar [4] utilized two well-known filter methods, Laplacian (L) and Fisher-score

(F), to reduce the dimensionality of feature space. Then, a hybrid evolutionary algorithm called an intelligent dynamic genetic algorithm (IDGA) based on the genetic algorithm (GA) and some artificial intelligence concepts, was used to determine the biomarkers. Alomari et al. [5] suggested Robust mRMR (RMRMR) and Hybrid Bat-inspired Algorithm (RMRMR-HBA) for gene selection. The BA algorithm was hybridized with β hill-climbing local search to empower BA's search capabilities. In another study, Alomari et al. [6] used RMRMR and Modified GWO (MGWO) to seek further small sets of genes. New optimization operators inspired by the TRIZ-inventive solution were combined with the original GWO in MGWO to boost the population's diversity. In [24], Random Forest Ranking (RFR) method was utilized as a filter approach to choose the top-ranked genes, and IDGA was used as a wrapper approach to seeking the most informative genes. In another GS method, called MIM-MFOA, the Mutual Information Maximization (MIM) was combined with the MFOA [7]. A two-phase hybrid model based on Correlation Feature Selection (CFS) and improved Binary Particle Swarm Optimization (iBPSO) was introduced in [37], for cancer diagnosis and classification. Pashaei et al. [38] proposed a hybrid RFR-BBHA approach for gene selection and classification of microarray data. It employed the bagging classifier as the fitness evaluator. In [39], BBHA was combined with the Chi-squared filtering approach and RF classifier. A hybrid BPSO- BBHA approach was developed for gene selection in [40], in which RF Recursive Feature Elimination (RF-RFE) technique was utilized to provide initial input for the model. Sparse Partial Least Squares Discriminant Analysis (SPLSDA) classifier played the role of classifier in the model. Wang et al. [41] introduced Bacterial Colony Optimization (BCO) for gene selection using the K-nearest neighbor (KNN) classifier. A two-stage method based on the combination of ensemble gene selection (EGS) and Adaptive GA (AGA) algorithm was proposed in [42], to find biomarkers that can help diagnose cancer. A combination of TLBO and SA algorithm, called TLBOSA, was suggested in [22]. As a preprocessing step, CFS was used to filter the redundant genes from the biological datasets. A wrapper-based gene selection with sequential forward selection (SFS) was employed in [43] utilizing the KNN classifier. Wang et al. [44] proposed integrating the Markov blanket (MB) with the wrapper-based SFS method to select informative genes using 1NN, NB, and C4.5 classifiers. Lu et al. [45] developed a hybrid of MIM and AGA for gene selection and cancer classification using an Extreme learning machine (ELM). A variable-length PSO with Local Search (VLPSO-LS) was proposed in [46]. A distributed ranking filter (DRF) approach removing the features with IG zero from the ranking (DRF0) was introduced in [47]. The DRF0 model employed the CFS and INTERACT algorithm which is based on symmetrical uncertainty (SU) to discriminate informative genes. Recently, Zhou et al. [48] introduced a problem-specific non-dominated sorting genetic algorithm (PS-NSGA) to find the optimal gene subset. Mollaee and Moattar [49] developed a three-stage ensemble schema for cancer diagnosis and classification. At first, an Ensemble filter-based gene selection method (EF) is used, which includes modified Bayesian logistic regression (BLogReg), T-test, and Fisher score ratio. In the second stage, the PSO-dICA method, which is a modification of dICA (discriminant independent component analysis), is used to choose the most informative genes. Finally, the SVM classifier [50] is used to classify selected genes. Medjahed et al. [51] utilized SVM Recursive Feature Elimination (SVM-RFE) to pre-filter the genes and used BDF to find important biomarkers. A hybrid of mRMR and Bat algorithm (BA) was proposed in [52], and a hybrid approach that embeds the MB filter approach into the HAS was introduced in [53] for gene selection problems. In another method, MB was embedded into the genetic algorithm (MBEGA) to identify the smallest possible set of genes that can achieve good predictive accuracy [54]. A wrapper GS method based on a Binary Differential Evolution (BDE) algorithm with a rank-based filtering method was developed in [55].

Table 1 shows a comparison of the studied GS approaches in terms of technique used, classifier, datasets, and validation method. In general, many NIOAs-based methods have been developed to solve the GS problem, and they have shown good performance compared to traditional methods. Most of the above-mentioned approaches, however, have limited accuracy in one or more of the datasets studied and suffer from stagnation because of the intricate interplay between genes and large-scale search space, and frequently dip in local optima [1, 5–8]. Therefore, there is still room for improvement, and better search algorithms are required to identify biomarkers with the highest classification accuracy.

3 Methods

3.1 Arithmetic optimization algorithm

AOA [9] is a new stochastic population-based NIOA that is inspired by arithmetic, a fundamental aspect of number theory. The behavior of classic arithmetic operators, such as division (D), multiplication (M), addition (A), and subtraction (S), in solving arithmetic problems, are used in AOA. Originally, the approach was suggested to handle numerical optimization problems as well as real-world engineering design optimization challenges.

The algorithm consists of two search phases: the exploration phase, which uses D and M operators to conduct a global search, and the exploitation phase in which A and S operators are utilized to carry out a local search. After random initialization, a function called Math Optimizer Accelerated (MOA) is used to choose between those two search phrases. The MOA's value at the tth iteration is obtained using Eq. (1):

$$MOA(t) = Min + t * \left(\frac{Max - Min}{T}\right) \tag{1}$$

where t and T denote the current and the maximum number of iterations, and Min and Max denote the minimum and maximum values of the accelerated function, respectively.

The following subsections detail each search phase:

Table 1 Literature details of hybrid gene selection approaches

Literature	Technique used	Microarray datasets		Classifier	Validation method
(Chaudhuri and Sahu, 2021) [1]	TOPSIS-Jaya	Colon	Lung cancer	Naïve Bayes (NB)	*k*-fold cross-validation (CV) ($k = 10$)
		CNS	Lymphoma		
		ALL-AML	MLL		
		ALL-AML-3C	Ovarian		
		ALL-AML-4C	SRBCT		
Pashaei and Pashaei (2021) [2]	mRMR-BDF-BBHA	Colon	Ovarian	SVM	*k*-fold CV ($k = 10$)
		MLL	Prostate Cancer		
		CNS	DLBCL		
		ALL-AML	Breast Cancer		
Zhang et al. (2020) [3]	IG-MBKH	Colon	Lung Cancer	SVM	*k*-fold CV ($k = 10$)
		CNS	MLL	NB	
		ALL-AML	Ovarian	KNN	
		ALL-AML-3C	SRBCT		
		ALL-AML-4C			
Dashtban and Balafar (2017) [4]	F-IDGA	SRBCT	Leukemia	SVM	leave-one-out-cross-validation (LOOCV)
	L-IDGA	Breast Cancer	Prostate Cancer	NB	
		DLBCL		KNN	
Alomari et al. (2018) [5]	RMRMR-HBA	Colon	Breast Cancer	SVM	*k*-fold CV ($k = 10$)
		CNS	Lymphoma	With radial basis function (RBF) KERNEL	
		ALL-AML	MLL		
		ALL-AML-3C	Ovarian		
		ALL-AML-4C	SRBCT		

Table 1 (continued)

Literature	Technique used	Microarray datasets		Classifier	Validation method
Alomari et al. (2021) [6]	RMRMR-MGWO	Colon	Lung Cancer	SVM	*k*-fold CV (*k* = 10)
		CNS	MLL	with RBF kernel	
		ALL-AML	Ovarian		
		ALL-AML-3C	SRBCT		
		ALL-AML-4C			
Pashaei and Pashaei (2019) [24]	RFR-IDGA	Colon	Leukemia	RF Classifier	LOOCV
Dabba et al. (2021) [7]	MIM-MFOA	Breast Cancer	Brain_Tumor1	SVM	LOOCV
		CNS	Brain_Tumor2		
		Colon	SRBCT		
		9_Tumors	MLL		
		11_Tumors	Leukemia1		
		Lymphoma DLBCL	Leukemia2		
		Ovarian	Prostate_Tumor		
			Lung_Cancer		
Jian et al. (2018) [37]	CFS-iBPSO	Colon	MLL	NB	*k*-fold CV (*k* = 10)
		CNS	Ovarian		
		ALL-AML	SRBCT		
		ALL-AML-3C	Breast		
		ALL-AML-4C	Lymphoma		
		Lung ancer			
Pashaei et al. (2016) [38]	RFR-BBHA	Colon	MLL	Bagging	*k*-fold CV (k = 10)
		CNS	ALL-AML-4C		
Pashaei and Aydin (2017) [39]	Chi-squared-BBHA	Ovarian	ALL-AML	RF Classifier	*k*-fold CV (*k* = 10)

Table 1 (continued)

Literature	Technique used	Microarray datasets		Classifier	Validation method
		Breast Cancer	Colon	C4.5	
			CNS	C5.0	
				Bagging	
Pashaei et al. (2019) [40]	RFE-PSO-BBHA	Breast Cancer	Real GSE datasets	SPLSDA	*k*-fold CV ($k=10$)
		CNS		KNN	
				NB	
Wang et al. (2017) [41]	BCO	Breast Cancer	Brain_Tumor1	KNN ($K=5$)	Hold-out CV
		CNS	Brain_Tumor2		(70% Train)
		Colon	SRBCT		
		9_Tumors	Leukemia		
		11_Tumors	Leukemia1		
		14_Tumors	Leukemia2		
		DLBCL	Prostate_Tumor		
			Lung_Cancer		
Shukla et al. (2018) [42]	EGS-AGA	Breast cancer	Leukemia	SVM	*k*-fold CV ($k=10$)
		Colon cancer	SBRCT	NB	
		DLBCL	Lung cancer		
Shukla et al. (2019) [22]	CFS-TLBO-SA	Colon	Brain_Tumor1	SVM	*k*-fold CV ($k=10$)
		9_Tumors	SRBCT		
		11_Tumors	Leukemia1		
		DLBCL	Leukemia2		
		Lung_Cancer	Prostate_Tumor		

Table 1 (continued)

Literature	Technique used	Microarray datasets		Classifier	Validation method
Wang et al. (2017) [43]	SFS	Colon	Leukemia2	KNN ($K=3$)	k-fold CV ($k=5$)
		CNS	DLBCL		
		Prostate	Ovarian		
		Leukemia1	SRBCT		
Wang et al. (2017) [44]	SFS-MB	Colon	Prostate	C4.5	k-fold CV ($k=5$)
		SRBCT	Bladder	KNN ($K=1$)	
		Leukemia1	Gastric	NB	
		Leukemia2	Tox		
		DLBCL	Blastomi		
Lu et al. (2017) [45]	MIM-AGA	Leukemia	Breast Cancer	ELM	–
		Colon	SBRCT	SVM	
		Lung_Cancer	Prostate_Tumor	Backpropagation neural network	
				Regularized ELM	
Tran et al. (2019) [46]	VLPSO-LS	Brain_Tumor2	Brain_Tumor1	KNN	k-fold CV ($k=10$)
		9_Tumors	SRBCT		
		11_Tumors	Leukemia1		
		DLBCL	Leukemia2		
		Lung_Cancer	Prostate_Tumor		
Bolón-Canedo et al. (2015) [47]		Breast Cancer	Leukemia	SVM	Hold-out CV
		CNS	Prostate	NB	(2/3 Train and 1/3 Test)
		Colon	Lung	KNN	
		Ovarian	DLBCL	C4.5	

Table 1 (continued)

Literature	Technique used	Microarray datasets		Classifier	Validation method
Zhou et al. (2021) [48]	PS-NSGA	SRBCT	11Tumor	KNN ($k=1$)	Stratified k-fold CV ($k=10$)
		Leukemia 1	Lung Cancer		
		DLBCL	Adenocarcinoma		
		9Tumor	Breast3		
		Brain Tumor 1	Lymphoma		
		Brain Tumor 2	Nci		
		Prostate	Prostate		
		Leukemia			
Mollaee and Moattar (2016) [49]	EF-PSO dICA	Colon	Leukemia 1	SVM	k-fold CV ($k=10$)
		SRBCT	DLBCL		
		Prostate Cancer	Lung		
Medjahed et al. (2017) [51]	RFE-BDF	Colon	Leukemia	C-SVM,	Hold-out CV
		Breast Cancer	DLBCL	υ–SVM	(50% Train, 30% validation and 20% Test)
		Ovarian	Lung Cancer	LS-SVM	
Alomari et al. (2017) [52]	MRMR-BA	Colon	Breast Cancer	SVM	k-fold CV ($k=10$)
		CNS	Lymphoma	with RBF kernel	
		ALL-AML	MLL		
		ALL-AML-3C	Ovarian		
		ALL-AML-4C	SRBCT		

Table 1 (continued)

Literature	Technique used	Microarray datasets		Classifier	Validation method
Shreem et al. (2014) [53]	HAS-MB	Colon	Breast Cancer	NB	k-fold CV ($k=10$)
		CNS	Lymphoma		
		ALL-AML	MLL		
		ALL-AML-3C	Ovarian		
		ALL-AML-4C	SRBCT		
Zhu et al. (2007) [54]	MBEGA	Colon	Lung Cancer	SVM	.632 + bootstrap
		CNS	MLL		$k=30$
		ALL-AML	Ovarian		
		ALL-AML-3C	SRBCT		
		ALL-AML-4C	Breast Cancer		
			Lymphoma		
Apolloni et al. (2016) [55]	BDE-χ_{Rank}	Colon tumor	Leukemia	SVM	Hold-out CV
		Ovarian	Lymphoma-DLBCL	NB	(2/3 Train and 1/3 Test)
		Prostate Cancer	Lung Cancer	KNN	
				C4.5	

3.1.1 Exploration phase

The D and M mathematical operators are used to provide AOA's exploratory behavior since D and M operators have higher dispersed values than S and A. Their high dispersion ensures that the algorithm reaches various promising regions in the search space. The exploration phase is conditioned by MOA's value. If MOA $< r1$ ($r1$ is a random number), then the exploration phase is carried out using D and M operators. The D and M operators also are conditioned by a random number. If $r2$ is greater than 0.5, the D operator will be engaged to perform; otherwise, the M operator will be executed. The algorithm's solution update rules in the exploration phase are expressed in Eq. (2)

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X^*}(t) \div (MOP + \varepsilon) \times ((ub - lb) \times \mu + lb)\, r2 > 0.5 \\ \overrightarrow{X^*}(t) \times MOP \times ((ub - lb) \times \mu + lb)\, r2 \leq 0.5 \end{cases} \tag{2}$$

where $X(t)$ denotes the current solution, X^* is the best solution that has been found so far. ε represents a small integer number, and μ is a constant parameter (0.5) to adjust the exploration search. lb and ub represent the lower and upper bounds of the current solution, and the coefficient Math Optimizer Probability (MOP) at iteration t is defined as:

$$MOP(t) = 1 - \left(\frac{t^{\frac{1}{\alpha}}}{T^{\frac{1}{\alpha}}} \right) \tag{3}$$

where α is symbolized as a control value of the exploitation accuracy over the iteration and set to 5.

3.1.2 Exploitation phase

The S and A mathematical operators are employed to offer AOA's exploitation search behavior as these operators provide high-dense results. They can easily approach the target and detect the near-optimal solution in a given region due to their low dispersion, unlike the D and M operators. The exploitation phase, like the exploration phase, is conditioned by the value of MOA. If MOA $\geq r1$, then the exploitation phase is carried out using S and A operators. A random number also influences the S and A operators. If $r3$ is greater than 0.5, the S operator will be called to action; otherwise, the A operator will be used. In the exploitation phase, the algorithm's solution updating rules are written as:

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X^*}(t) - MOP \times ((ub - lb) \times \mu + lb) \;\; r3 > 0.5 \\ \overrightarrow{X^*}(t) + MOP \times ((ub - lb) \times \mu + lb) \;\; r3 \leq 0.5 \end{cases} \tag{4}$$

The pseudocode of the algorithm is presented in Fig. 1. The AOA is a relatively new approach, and this study is one of the first to put it to the test in terms of GS.

Algorithm 1 Arithmetic Optimization Algorithm

```
Initialize the AOA's parameters $\alpha$ and $\mu$
Initialize the first population $X_i (i = 1,2, \ldots, N)$
Compute each solution's objective value
$X^*$=the best solution
while $(t < T)$
    calculate MOP and MOA values
    for each solution
        Generate random numbers between 0 and 1 for $r_1$, $r_2$, and $r_3$
        if $(r_1 > \text{MOA})\{$
            if $(r_2 > 0.5)$
                update position using the first rule in Eq. (2) (D operator)
            else
                update position using the second rule in Eq. (2) (M operator)
        else
            if $(r_3 < 0.5)$
                update position using the first rule in Eq. (4) (S operator)
            else
                update position using the second rule in Eq. (4) (A operator)
        end if
    end for
    Examine whether any solution extends outside the search space and correct it
    Compute each solution's objective value
    Update $X^*$ if there is a better solution
    $t = t + 1$
end while
Output: the best solution = $X^*$
```

Fig. 1 Pseudocode of AOA

3.2 Simulated annealing algorithm

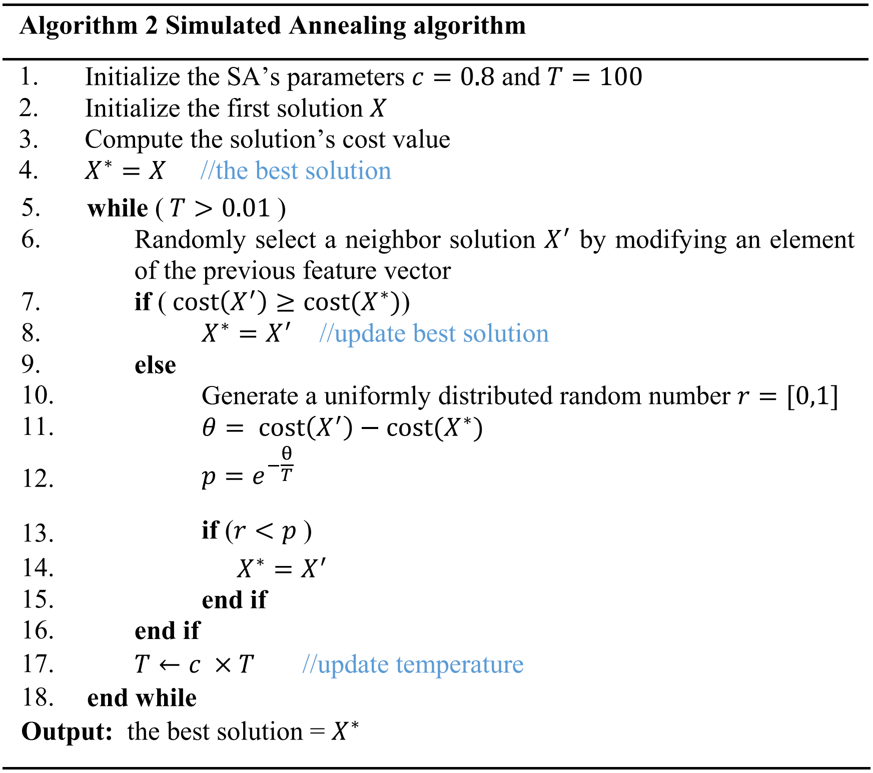
One of the most used heuristic methods for tackling optimization problems is the SA algorithm [56]. It is a local search algorithm based on a single solution. The algorithm has been inspired by the physical phenomenon that occurs during the solidification of fluids. In general, the SA algorithm simulates the annealing process of metals, in which an atomic configuration for a solid (optimal solution of the problem) that minimizes internal energy (objective function) must be discovered. During the annealing process, heat and cooling rates are controlled. To achieve the best atomic configuration throughout the annealing process, the temperature T or heat must be gradually decreased according to some cooling schedule.

In the SA-based feature selection approach, the initial solution is chosen randomly and assumed as the best solution. The energy of the initial solution (cost) is then calculated using the previously established objective function. When temperature T fails to meet the termination condition, a neighboring solution of the current optimal solution is chosen by modifying an element of the current solution and its cost is calculated. The current optimal solution is replaced with a newly selected adjacent solution if the cost of the newly selected neighboring solution is better than or equal to the current solution. If the cost of the nearby solution is worse than the current optimal solution, a random number r in the range of (0, 1) is generated. The current optimal solution can only be replaced if the generated random number r is less than an acceptance probability p, which is mathematically formulated by Eq. (5). The temperature T is then decreased using Eq. (6).

$$\theta = \cos t(neighbor\,solution) - \cos t(optimal\,solution),\ p = e^{-\frac{\theta}{T}} \tag{5}$$

$$T \leftarrow c \times T \tag{6}$$

where c is the temperature's reduction rate (cooling rate), and the objective function includes the computation of the coefficient of determination (R^2) to determine the cost of solutions. The coefficient of determination is similar to the correlation coefficient (R), which is used to determine how strong a linear relationship exists between two variables. The abovementioned steps are repeated until T meets the termination criterion. Figure 2 shows the pseudocode of the SA for feature selection.



Algorithm 2 Simulated Annealing algorithm

```
Initialize the SA's parameters $c = 0.8$ and $T = 100$
Initialize the first solution $X$
Compute the solution's cost value
$X^* = X$    //the best solution
while ( $T > 0.01$ )
    Randomly select a neighbor solution $X'$ by modifying an element
    of the previous feature vector
    if ( cost($X'$) $\geq$ cost($X^*$))
        $X^* = X'$    //update best solution
    else
        Generate a uniformly distributed random number $r = [0,1]$
        $\theta =$ cost($X'$) $-$ cost($X^*$)
        $p = e^{-\frac{\theta}{T}}$
        if ($r < p$ )
            $X^* = X'$
        end if
    end if
    $T \leftarrow c \times T$    //update temperature
end while
Output: the best solution = $X^*$
```

Fig. 2 Pseudocode of SA for feature selection

3.3 Crossover scheme

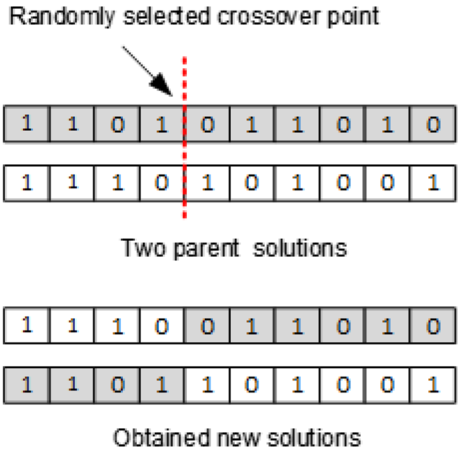
Crossover, also known as recombination, is one of the main genetic operators in GA that allows two parent chromosomes to combine their genetic information in order to produce new offspring. Crossover is based on the premise that if the new chromosome inherits the finest traits from both parents, it may be superior to both. In binary NIOAs, the crossover operator is accomplished by cutting a portion of each parent solution and inserting it into the other. This paper uses the single-point crossover, in which both parental solutions are split at a single randomly determined crossover point. The first portions of the parent solutions are then swapped between two parents. Figure 3 shows the single-point crossover process. As seen in Fig. 3, binary bits are exchanged between two solutions, causing both solutions to change abruptly. This property of crossover can change the global optimal solution and keep the algorithm from getting stuck in local optima.

It is also worth noting that a hybrid of AOA with GA operators, i.e. crossover and mutation, called AOAGA, has been considered in the literature as a technique to increase the AOA's global search capabilities for solving feature selection problems [14]. The position of the solutions in AOAGA is updated using GA or AOA operators that are reliant on a transition mechanism, but in our suggested method, a crossover operator is utilized after performing AOA operators without considering any conditions or transition mechanism.

4 The proposed algorithm for GS

GS is a challenging task due to some properties of gene expression microarray datasets. Microarray data contains a huge number of genes but only a few samples. Moreover, a vast majority of those genes are noisy and irrelevant. Subsequently, machine learning classifiers perform poorly on data with respect to predictive

Fig. 3 Single point crossover



accuracy and run time [6, 57]. The goal of GS is to maximize the accuracy of the classifier with the fewest number of biologically important genes.

This section introduces a novel two-stage gene selection strategy, which is based on the mRMR filtering and the BAOAC-SA wrapper method. First, the mRMR filtering approach is used to prune the least important genes from the dataset. Then, the suggested BAOAC-SA approach is utilized to determine the best gene subset from the current set of genes. The framework of the developed GS technique is depicted in Fig. 4.

After the original dataset is loaded and the training and testing samples are separated, the top M genes, which were selected using the mRMR filtering approach, are used to build new training data with a smaller gene set, as shown in Fig. 4. The training sub-data is only utilized to build a classifier and evaluate candidate solutions during the wrapper process, whereas the test sub-data is used to evaluate the final solution. Subsequently, the proposed wrapper BAOAC-SA approach starts by creating a population of candidate solutions (i.e. subsets of genes) using the new reduced training dataset. The population is evaluated using a fitness function that employs a classifier. After assigning fitness, the best candidate solution X^* is determined. The main loop in AOA is iterated several times and in each iteration, the solutions are updated using the operators of AOA. The solutions then are passed through a V-shaped hyperbolic tangent transfer function and accordingly the binary solution vectors are created. Each solution in the population is combined with the best global solution X^* using a single-point crossover operator. Between obtained solutions, the fittest solution is maintained. Then, if necessary, the global best solution X^* is updated. After implementing the crossover operator, SA is used to improve the quality of the current and best solutions at the end of each BAOA iteration. This

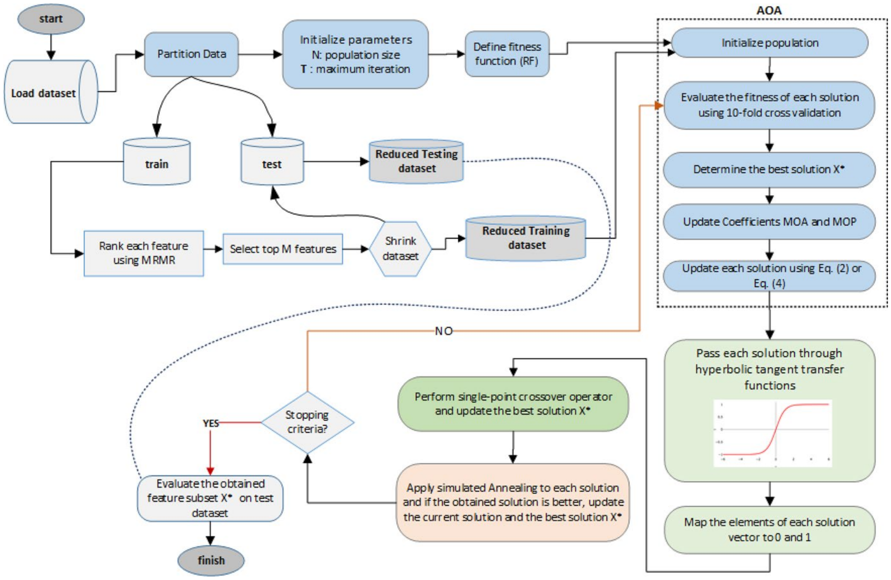

start
Load dataset
Partition Data
Initialize parameters
N: population size
T : maximum iteration
Define fitness function (RF)
AOA
Initialize population
Evaluate the fitness of each solution using 10-fold cross validation
Determine the best solution X*
Update Coefficients MOA and MOP
Update each solution using Eq. (2) or Eq. (4)
train
test
Reduced Testing dataset
Rank each feature using MRMR
Select top M features
Shrink dataset
Reduced Training dataset
NO
Pass each solution through hyperbolic tangent transfer functions
Perform single-point crossover operator and update the best solution X*
YES
Stopping criteria?
Evaluate the obtained feature subset X* on test dataset
Apply simulated Annealing to each solution and if the obtained solution is better, update the current solution and the best solution X*
Map the elements of each solution vector to 0 and 1
finish

Fig. 4 Flowchart of suggested mRMR-BAOAC-SA approach for gene selection and cancer classification

process is repeated until satisfying the stopping criteria, which is usually the maximum number of iterations. Once the process was completed, the best subset of candidate genes is obtained and evaluated using the test sub-data and a classifier. One consideration is that the quality of the best solution should be measured over the testing samples or using leave-one-out-cross-validation (LOOCV) [4] as exhibited in Fig. 4.

The following subsections provide a detailed description of both stages of the suggested method.

4.1 Stage I: gene filtering using mRMR

As the number of genes expands, the number of all possible subsets of genes in the search space grows exponentially. For N genes, the total number of possible gene subsets is 2^N [58]. Instead of doing an exhaustive search through feature subsets, a filter approach is utilized in the GS problem at the first stage of hybrid methods to decrease the original dataset's high dimensionality [40, 59]. The filter approach calculates a score for each gene, ranks them, and chooses the top M genes. To put it another way, the filter method prunes the search space and prepares it for the wrapper method. Then, the wrapper approach is utilized to choose the smallest number of highly discriminative genes among *n* top genes selected by the filter approach. As a result, the computational cost of the GS process is reduced and classification accuracy is improved [60].

In this paper, the mRMR filter approach was utilized as a strong initial stage to filter away noisy genes from high-dimensional microarray data. As a useful algorithm, mRMR has been widely used in GS and cancer classification problems [2, 60]. The mRMR uses mutual information (MI), a measure of relevance/redundancy, to provide gene scores. The relevance is determined by the MI of genes with targeted classes, and the correlation, or redundancy, is decided by the level of MI between genes. The objective of the mRMR approach is to select genes with the highest relevance and the least redundancy. After assigning a score to each gene, the gene with the highest score is included in the optimal subset iteratively in a greedy forward manner [61]. The MI between two variables X and Y is calculated using Eq. (7):

$$I(Y;X) = H(Y) - H(Y|X) \tag{7}$$

$$H(Y) = -\sum_{y} p(y) \log_2 (p(y)) \tag{8}$$

$$H(Y|X) = \sum_{x} p(y)(-\sum_{y} p(y|x) \log_2(p(y|x)) \tag{9}$$

where $H(.)$ and p represent entropy and probability mass function, respectively.

The score of each gene $X_k (k = 1, \dots , m)$ in a dataset with n sample, m genes and a target class Y is determined as:

$$J_{MRMR}(X_K) = I(Y;X_k) - \frac{1}{|S|}\sum_{X_j \in S} I(X_k;X_j) \tag{10}$$

Let S denote a subset of genes we're looking for. $I(Y;X_k)$ quantifies the relevance of X_k for the classification task. The value of S is initially set to $S = \{X_k\}$ with $I(Y;X_k) = \underset{j \in \{1,\dots,m\}}{max} I(Y;X_j)$. The term $\frac{1}{|S|}\sum_{X_j \in S} I(X_k;X_j)$ assesses the mean information that the X_k gene shares with selected genes in S to determine its redundancy. The gene that achieves the highest possible score is added to S at each iteration.

4.2 Stage II: BAOAC-SA wrapper approach

In this stage, the wrapper gene selection method based on the hybrid BAOA with SA and crossover operator uses the M top-ranked gene subset given by mRMR to find the optimal subset of genes. BAOA is hybridized with the SA algorithm and crossover operator to promote the searching process and effectively explore the interaction among the genes. Figure 5 depicts the pseudocode for the proposed BAOAC-SA algorithm.

4.2.1 Solution representation

GS is a binary optimization problem. However, AOA works in a continuous solution space. The continuous version must be converted to binary form before AOA can be applied to the GS problem. The search space in a wrapper-based strategy should be expressed in binary format. Each candidate solution X (or gene subset) is represented as a one-dimensional binary vector of size Z, $X = (G_1, G_2, \dots, G_Z)$, where Z indicates the problem dimension (i.e. the number of genes in the dataset). In the solution vector X, each bit G_j has a value of "1" or "0". The value 1 implies that the associated gene will be kept, whilst the value 0 indicates that it will be removed. So, the evaluation process only takes into account genes that are coded in one.

4.2.2 Fitness function

The wrapper feature selection techniques begin with a set of randomly generated binary solutions. During the search process, the techniques use a fitness (objective) function to give each solution a fitness score. Selecting a proper fitness function is critical for the efficiency of wrapper approaches since the fitness function guides model to find the best solutions within the large search space. RF classifier [24, 62] is utilized in this study to determine the fitness value of each solution. In other words, the fitness function is defined as RF's classification accuracy on the given solution using tenfold cross-validation (CV) [63]. A solution with higher classification accuracy is a better solution. It is worth reminding that the major purpose of the model is to enhance classification accuracy with fewer genes. So, in the evaluation process, if two subset genes have the same classification accuracy, the subset with the fewer genes is picked.

Algorithm 3 BAOA with SA and crossover operator (BAOAC-SA)

```
Initialize the AOA's parameters $\alpha$ and $\mu$
$X = \{G_1, G_2, \dots, G_z\}$
Initialize the first population $X_i (i = 1,2, \dots, N)$
Compute each solution's objective value using RF
$X^*$=the best solution
while ($t < T$)
    calculate MOP and MOA values
    for $i = 1$ to $N$ do
      for $j = 1$ to $Z$ do
          Generate random numbers between 0 and 1 for $r_1$, $r_2$, and $r_3$
          if ($r_1 >$ MOA)
              if ($r_2 > 0.5$)
                  update $X_{ij}$ using the first rule in Eq. (2) (D operator)
              else
                  update $X_{ij}$ using the second rule in Eq. (2) (M operator)
          else
              if ($r_3 < 0.5$)
                  update $X_{ij}$ using the first rule in Eq. (4) (S operator)
              else
                  update $X_{ij}$ using the second rule in Eq. (4) (A operator)
          end if
          if ($abs(\tanh(X_{ij})) > 0.6$ )
              $X_{ij} = 1$
          else
              $X_{ij} = 0$
          end if
      End for  //inner loop j
      Compute the $X_i$ objective value
      if ($X_i$ is better in the terms of accuracy and number of selected genes)
          update $X^*$
      end if
      //Perform crossover operator
      [$p1, p2$]=Crossover($X^*$, $X_i$)
      Calculate the objective values of $p1, p2$
      if (fitness ($p1$) > fitness ($X^*$) or (fitness ($p1$) == fitness ($X^*$) and $p1$ has fewer genes) )
          $X^* = p1$
      else if (fitness ($p2$) >= fitness ($X^*$))
          $X^* = p2$
      //Update the current solution by SA
      $X_{new}$ = SimulatedAnnealing ($X_i$, temperature = 100, c= 0.8 )
      if (fitness ($X_{new}$) > fitness ($X_i$))
          $X_i = X_{new}$ // Accept the new solution
      if (fitness ($X_{new}$) > fitness ($X^*$))
          $X^* = X_{new}$
    End for  //outer loop i
    $t = t + 1$
end while
Output: the best solution = $X^*$
```

Fig. 5 Pseudocode of BAOA with SA and crossover schema

4.2.3 Binay AOA

In AOA each gene subset can be seen as a candidate solution. Each subset may contain Z genes, where Z is the number of genes discovered in the preceding filtering stage. The algorithm begins with a population of randomly generated binary solutions. The fitness function is then used to evaluate each solution in the population. After assigning fitness values, the population's best solution is determined.

AOA's core loop is repeated several times. The algorithm updates the main coefficients (MOP and MOA) at each iteration. MOA which contains values $> r_1$ or $< r_1$ (a random value in [0,1]) is used to choose between the exploration and exploitation phases. Accordingly, the solutions (positions) are updated with the rules in Eq. (2) and Eq. (4). The algorithm repeats the steps until it reaches the value of the maximum iteration.

The main challenge in the design of binary AOA is how the algorithm's updating equations (i.e. Equation (2) and Eq. (4)) in real space can be interpreted in discrete domains. The most straightforward technique to convert a continuous search space to a binary one is to use a transfer function (TF). The first binary version of AOA was proposed by Bansal et al. in [10] and they have utilized S-shaped (sigmoid) and V-shaped TFs to perform feature selection. The high performance of the V-shaped family of TFs has already been proven in the literature for binary algorithms [64, 65]. So a V-shaped hyperbolic tangent TS is utilized in this study to binarize AOA, which is defined as follows:

$$TF\left(X_{iz}(t+1)\right) = abs\left(\tanh(X_{iz}(t+1))\right) \tag{11}$$

$$X_{iz}(t+1) = \begin{cases} 1, & TF\left(X_{iz}(t+1)\right) > 0.6 \\ 0, & otherwise \end{cases} \tag{12}$$

where $X_{iz}(t+1)$ represents the bit value of Z th dimension of i th solution in the next iteration $(t+1)$. The tangent function determines the probability of updating a solution element to 0 (not selected) or 1 (selected), and $TF\left(X_{iz}(t+1)\right)$ represents the probability value. The value of the Z th element in the i th solution vector is set to 0 or 1 by Eq. (12), depending on the probability value obtained from Eq. (11).

4.2.4 BAOA with SA and crossover operator

The BAOA produces good results on a variety of microarray datasets. However, it fails to come up with satisfactory performance on some benchmark datasets. To overcome this and boost BAOA's performance, two improvements are introduced to the original BAOA framework. The first improvement includes the use of the crossover operator. In the proposed method, a crossover step is added to the BAOA (BAOAC) to allow for the effective exploration of promising regions of the search space. The suggested approach performs the crossover operator just after converting the solution vector to binary form. The crossover operation is taken between the best solution X^* and the current solution X_i as shown in Eq. (13).

$$\left[p_1, p_2\right] = Crossover\left(X^*, X_i(t)\right) \tag{13}$$

After the crossover, the fitness of the best solution is compared with that of the two offspring, and the best one is taken as the new best solution (i.e. Lines 32 to 37 in Algorithm 3 (Fig. 5)).

The second improvement is to embed the SA in the BAOA to enhance its exploitation capability and avoid being stuck in local optima. The standard AOA has flaws in terms of exploitation capability (local search) and the trade-off between exploitation and exploration search strategies. The poor exploitation in NIOAs may lead to skipping the most optimal solution even present in the vicinities of the current solution which results in a poor local convergence rate that ultimately degrades the solution quality. To address this issue, SA is incorporated into the BAOA solution updating stage in order to develop a powerful gene selection algorithm with a better balance between exploration and exploitation capabilities and a better convergence rate.

After implementing the crossover operator and achieving the best solution, SA is used at the end of each BAOA iteration to enhance the current solution (i.e. X_i). The new solution X_{new} obtained by SA is accepted if it has a higher fitness value than the current solution. As a result, X_{new} is included in the population. Furthermore, the best solution X^* is updated when the fitness value of X_{new} is better than the global best solution (i.e. Lines 38 to 42 in Algorithm 3).

4.3 Time complexity analysis

The time complexity of BAOAC-SA depends on the complexity of the BAOAC and SA algorithm. The time complexity of wrapper BAOAC depends on initialization, fitness evaluation, and updating of candidate solution processes. The time complexity of the initialization process is $O(N)$ where N represents the number of all candidate solutions (population size). The computational complexity of updating process is $O(T * N) + O(T * N * Z)$, which consists of finding the best solution and updating all solutions. T represents the maximum number of iterations and Z indicates the dimension of the search space. The fitness evaluation process helps the process of updating to find the best solution. On the other hand, the time complexity of the SA algorithm is $O(T * I * S)$, where I and S are symbolized by the number of iterations and search strategy in the SA, respectively. Consequently, the time complexity of BAOAC-SA is $O(N * (T + TZ + 1) + T * I * S)$.

5 Experimental setup and results

The suggested method is a two-phase iterative procedure. In the first phase, the Min–Max normalization and mRMR techniques are used as a preprocessing step to standardize and remove redundant and irrelevant genes from the microarray expression data. Then, in the second phase, the suggested BAOAC-SA technique is employed to obtain an optimum gene subset. The suggested mRMR-BAOAC-SA gene selection technique was applied for the classification of ten binary and multiclass high-dimensional microarray cancer datasets. The RF classifier with 500 trees is used as the fitness evaluator. The tenfold CV method was performed over each dataset to validate and assess each candidate gene subset (fitness value). The final model evaluation with obtained features is then given by the LOOCV procedure. Since NIOAs are stochastic models, proposed algorithms were performed independently several times for each dataset and the average LOOCV results were reported.

Note that in many earlier works, researchers typically split the original microarray datasets into training and testing sets randomly. Gene selection is then conducted on the training set and the unseen test set is used to evaluate the quality of selected genes. However, due to the small number of samples, such an approach is now recognized by the community as unreliable [54]. Instead, Ambroise and McLachlan [66] suggested splitting the data using external CV (tenfold) or 0.632 + bootstrap. Considering recent studies [4, 7, 60], LOOCV is utilized to evaluate obtained results after each run of algorithms to confirm that the whole dataset is used in the training and testing phases.

The algorithms used in our studies were written in the R programming language. The 'praznik' package was used to implement mRMR, the 'randomForest' package was employed to construct the RF classifier, and the 'FSinR'package was utilized to implement SA, WOA, and GA. All of the experiments were done on a 2.2 GHz Core i5 CPU with 8 GB of RAM.

5.1 Dataset used

This study has utilized two types of microarray datasets: binary class and multi-class datasets. In total, ten (10) gene expression datasets from various diseases were used. These datasets are frequently used in many studies and include small, medium, and large dimensional datasets. CNS, Colon, ALL-AML, Breast, and Ovarian are binary-class microarray datasets, while lymphoma, MLL, SRBCT, Brain_Tumors1, and 11_Tumors and multi-class microarray datasets. Table 2 summarizes the characteristic of the selected datasets in detail.

5.2 Parameter settings

For all algorithms, the number of populations and the maximum number of iterations were set to 50. The number of genes selected after the pre-filter operation was

Table 2 Summary of gene expression datasets

Dataset name	No. of samples	No. of features	No. of classes	Distribution of class label
SRBCT	83	2308	4 (Multi-class)	29, 25, 18, 11
MLL	72	12,582	3 (Multi-class)	28, 24, 20
Lymphoma	62	4026	3 (Multi-class)	46, 11, 9
Brain Tumor_1	90	5920	5 (Multi-class)	60, 10, 10, 4, 6
11_Tumors	174	12,533	11 (Multi-class)	26, 8, 26, 23, 12, 11, 7, 27, 6, 14, 14
Ovarian	253	15,154	2 (Binary-class)	162, 91
CNS	60	7129	2 (Binary-class)	39, 21
ALL-AML	72	7129	2 (Binary-class)	47, 25
Colon Cancer	62	2000	2 (Binary-class)	40, 22
Breast Cancer	97	24,481	2 (Binary-class)	51, 46

Table 3 The LOOCV classification accuracy performance of the mRMR method with an RF classifier for all microarray datasets

Class	Dataset	Number of selected genes						
		10	25	50	75	100	125	150
Binary class	Ovarian	96.83	98.02	99.20	99.60	99.60	99.20	99.20
	CNS	81.66	81.66	83.33	78.33	75	73.33	71.66
	ALL-AML	97.22	97.22	97.22	98.61	98.61	98.61	98.61
	Colon	85.48	88.70	87.09	87.09	87.09	85.48	85.48
	Breast	82.47	82.47	83.50	83.50	82.47	86.59	83.50
Multi-class	SRBCT	98.79	97.59	97.59	98.79	100	100	100
	MLL	93.05	94.44	97.22	97.22	97.22	95.83	97.22
	Lymphoma	92.42	98.48	98.48	98.48	98.48	98.48	98.48
	Brain Tumor_1	90	90	88.88	90	88.88	88.88	90
	11_Tumors	76.43	86.20	91.37	90.22	91.95	91.37	89.08

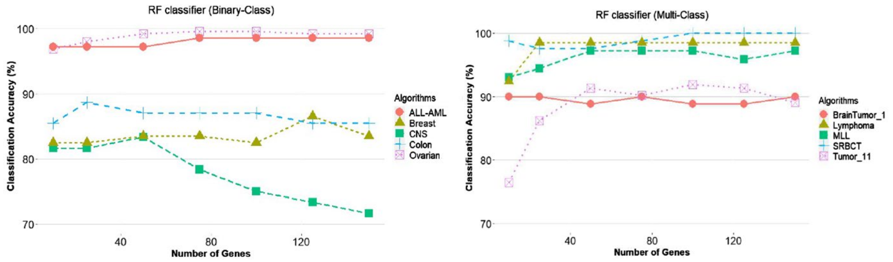


Fig. 6 The classification accuracy performance of the mRMR method with RF classifier for all datasets

set to 100 except for CNS (125). Furthermore, after using mRMR in the Breast dataset, a multivariate step (correlation matrix) was used to remove the redundancy among highly-correlated genes (correlation coefficient >= 0.85). In this study, we ran each dataset ten times in order to conduct our experiments. MOP Max, MOP Min, Alpha, and Mu are the parameters utilized in the BAOA wrapper method. These parameters sequentially have 1, 0.2, 5, and 0.499 values, which were chosen based on the study in [9]. Determination Coefficient was used as a filter evaluator for SA. SA parameters including temperature, reduction, and innerIter are set to 100, 0.8, and 35, respectively. The classification accuracy and number of selected features were chosen as measurement metrics for evaluating the performance of the optimization algorithms. For both WOA and GA, the default parameter settings of the 'FSinR' package were used.

5.3 Comparison of BAOA, BAOAC, and BAOAC-SA

Initially, we used the mRMR filtering approach to find the most significant genes to eliminate noisy genes from high dimensional microarray data using an RF

Table 4 Experimental results by mRMR-BAOA on all datasets

Class	Dataset	Accuracy				#Genes			
		Best	Worst	Avg	S.D	Best	Worst	Avg	S.D
Binary class	Ovarian	100	100	100	0.0	3	4	3.04	0.8
	CNS	93.33	86.9	90.37	2.155	2	11	3.833	1.169
	ALL-AML	100	100	100	0.0	2	5	4.34	1.2
	Colon	95.13	92.18	93.36	1.20	3	19	8.4	6.22
	Breast	94.84	87.63	90.88	2.67	7	19	12.71	4.34
Multi-class	SRBCT	100	100	100	0.0	4	7	6	1.2
	MLL	100	98.67	99.42	0.8	3	5	4.1	0.5
	Lymphoma	100	100	100	0.0	2	3	2.3	0.7
	Brain Tumor_1	95.67	92.46	94.41	1.178	6	14	9.4	3.78
	11_Tumors	94.88	93.78	94.49	0.467	24	41	28.8	6.94

Table 5 Experimental results by mRMR-BAOAC on all datasets

Class	Dataset	Accuracy				#Genes			
		Best	Worst	Avg	S.D	Best	Worst	Avg	S.D
Binary class	Ovarian	100	100	100	0.0	2	4	2.9	0.57
	CNS	96.92	91.77	94.44	2.32	3	15	6.37	3.77
	ALL-AML	100	100	100	0.0	2	5	4.22	1.1
	Colon	95.38	92.18	94.79	1.18	4	13	8.57	3.82
	Breast	97	93.89	95.89	1.51	8	17	12.62	3.5
Multi-class	SRBCT	100	100	100	0.0	4	7	6	1.2
	MLL	100	100	100	0.0	3	6	3.9	0.6
	Lymphoma	100	100	100	0.0	2	3	2.3	0.7
	Brain Tumor_1	95.79	94.62	95.88	0.488	6	13	8.87	2.47
	11_Tumors	96.58	94.98	95.99	0.656	23	40	27	6.67

classifier. Table 3 and Fig. 6 exhibit the classification accuracy performance of the mRMR approach utilizing an RF classifier for binary class and multiclass microarray datasets. It can be seen from Table 3 and Fig. 6 that the top 100 genes give good results for all microarray datasets.

Tables 4, 5, and 6 demonstrate the experimental results of three versions of BAOA based GS methods (i.e. BAOA, BAOAC, and BAOAC-SA) in terms of the best, worst, average, and standard deviation (S.D.) of the number of selected genes and classification accuracy. All three techniques were tested in ten separate runs. From Tables 4, 5, and 6, it is evident that the BAOAC-SA outperforms BAOA and BAOAC in all datasets.

Table 6 Experimental results by mRMR-BAOAC-SA on all datasets

Class	Dataset	Accuracy				#Genes			
		Best	Worst	Avg	S.D	Best	Worst	Avg	S.D
Binary class	Ovarian	100	100	100	0.0	2	4	2.6	0.57
	CNS	97.14	92.14	94.60	2.06	3	8	5.2	1.97
	ALL-AML	100	100	100	0.0	3	4	3.3	1
	Colon	98.57	93.81	95.68	1.61	5	11	7.66	2.65
	Breast	97.5	95.78	96.188	0.48	8	14	10.8	3.11
Multi-class	SRBCT	100	100	100	0.0	4	7	5.8	1.1
	MLL	100	100	100	0.0	3	5	3.6	0.5
	Lymphoma	100	100	100	0.0	2	2	2	0.0
	Brain Tumor_1	97.1	94.89	96.021	0.687	6	10	7.5	1.414
	11_Tumors	97.84	96.32	96.946	0.601	19	39	24	8.48

To benchmark the performance of the proposed BAOAC-SA approach, a comparison between BAOAC-SA against BAOA, BAOAC, and two other optimizers (WOA and GA) is conducted. These results are presented in Table 7 where.

the experimental outcomes are reported in terms of average classification accuracy (ACC), the average number of chosen genes ($|\#G|$), and the average execution time (minutes). The best results of ACC and $|\#G|$ are highlighted in bold font. In terms of classification accuracy, BAOAC-SA outperformed WOA, GA, BAOA, and BAOAC across five datasets, as shown in Table, 7. Table 7 shows that in the remaining five datasets, all techniques obtained flawless classification accuracy (100%) (i.e. MLL, Ovarian, SRBCT, Lymphoma, and ALL-AML). Overall, On nine datasets (Ovarian, ALL-AML, Colon, Breast, SRBCT, MLL, Lymphoma, Brain Tumor_1, and 11_Tumors), BAOAC-SA yields higher classification accuracy and a lower number of selected genes. However, in the CNS dataset, BAOA shows better performance than BAOAC and BAOAC-SA in terms of the average number of selected genes.

Although execution time is not a prominent factor in the gene selection domain because it is not real-time, the average execution times of 10 independent runs for GA, WOA, BAOA, BAOAC, and BAOAC-SA are outlined in Table 7. The mRMR filtering approach is extremely fast and takes only a few seconds to complete and remove irrelevant features.

Considering Table 7, it is apparent that BAOA has the lowest time compared to all approaches, with an average time of 22.2 min across all datasets. BAOAC has a second lower time, with an average of 62 min across all datasets, while BAOAC-SA has the highest time value with an average of 208.2 min across all datasets since it is a hybrid version and hence requires more execution time. It's worth noting that the time values obtained by BAOA, BAOAC, and BAOAC-SA are all reasonably acceptable. It can be seen that BAOA, BAOAC, WOA, and GA are 9.37, 3.35, 1.4, and 2 times faster than BAOAC-SA, respectively. Although BAOAC-SA takes longer to run than other methods, it can produce a smaller

Table 7 Performance comparison between WOA, GA, BAOA, BAOAC, and BAOAC-SA

Algorithms		Dataset									
		Binary Class					Multi-Class				
		Ovarian	CNS	ALL-AML	Colon	Breast	SRBCT	MLL	Lymphoma	Brain Tumor_1	11_Tumors
WOA	$\|\#G\|$	41.2	60.1	45.2	65.6	40.5	42.4	52.5	39.1	46.2	50.2
	ACC	100	90.23	100	93.33	92.55	100	100	100	93.88	96.02
	$Time$	283:04	177:5	99:99	170:5	127:6	81:62	84:27	49:27	124:73	290:13
GA	$\|\#G\|$	51.9	54.3	47.2	49.5	43.1	57.4	44.8	48.2	52.2	39.7
	ACC	100	89.04	100	93.57	93.09	100	100	100	94.88	96.74
	T ime	125:06	86:77	111:98	68:12	125:9	104:7	75:55	55:75	77:73	202:37
BAOA	$\|\#G\|$	3.04	3.833	4.34	8.4	12.71	6	4.1	2.3	9.4	28.8
	ACC	100	90.37	100	93.36	90.88	100	99.42	100	94.41	94.49
	T ime	15:30	17:1	21:66	22:62	32:4	16:08	10:28	5:36	12:40	72:99
BAOAC	$\|\#G\|$	2.9	6.37	4.22	8.57	12.62	6	3.9	2.3	8.87	27
	ACC	100	94.44	100	94.79	95.89	100	100	100	95.88	95.99
	$Time$	58:66	38:01	30:53	75:47	99:53	78:45	29:08	17:52	40:48	156:3
BAOAC-SA	$\|\#G\|$	**2.6**	**5.2**	**3.3**	**7.66**	**10.8**	**5.8**	**3.6**	**2**	**7.5**	**24**
	ACC	**100**	**94.60**	**100**	**95.68**	**96.188**	**100**	**100**	**100**	**96.021**	**96.946**
	T ime	275:34	150:8	235:8	210:04	230:2	222:4	120:1	110:2	220:8	310:9

set of reliable genes with higher/equivalent classification accuracy on almost all datasets.

The boxplot, in conjunction with statistical analysis and p-values, was utilized as a graphical representation to offer a better understanding of the diverse behaviors of the suggested procedures and to statistically assess the effectiveness of the proposed method (see Fig. 7).

The boxplot with statistical analysis was created in R using the " ggpubr " package. The boxplot in Fig. 7 shows that the proposed BAOAC-SA may retain variety during the search while delivering highly accurate results. The means of the five

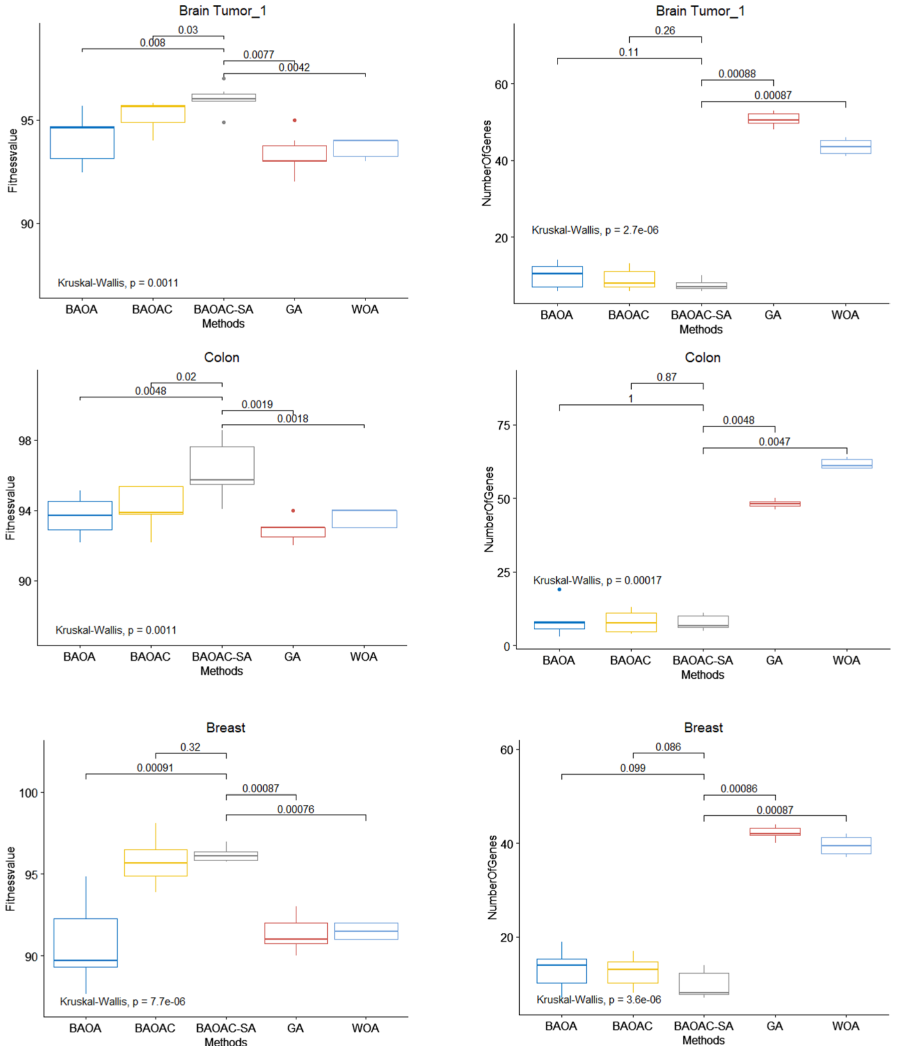


Fig. 7 Boxplots with P-values and significance levels to demonstrate the diversity behavior and statistical significance of the proposed method in comparison to other techniques

techniques were also compared, and the boxplots were enriched with p-values and significance levels. To compare the means of techniques, the Wilcoxon test was utilized. P-value ≤ 0.05 indicates that the BAOAC-SA methodology delivers considerably better outcomes than other techniques, whilst P-value > 0.05 indicates that the BAOAC-SA technique produces results that are not significantly better than the other approaches. Both fitness values and the number of selected genes were statistically computed using the Wilcoxon signed-rank statistical test for five datasets (i.e. Breast, Brain Tumor 1, 11 Tumors, CNS, and Colon). Significant differences in favor of BAOAC-SA can be determined for these datasets. For the remaining datasets with perfect classification accuracy, the average number of chosen genes by BAOAC-SA is slightly better than other techniques.

Furthermore, the BAOAC-SA is compared to the BAOA and BAOAC to explore the impact of SA and crossover operator on the convergence behavior of BAOA. The convergence behaviors of the three methods on all datasets are shown in Figs. 8 and 9. In terms of classification accuracy, the convergence behavior trend of BAOAC-SA is significantly better than BAOA and BAOAC on six datasets (i.e. CNS, Colon, Breast, MLL, Brain Tumor_1, and 11_Tumors). Figure 9 shows that for the remaining datasets with the perfect classification accuracy

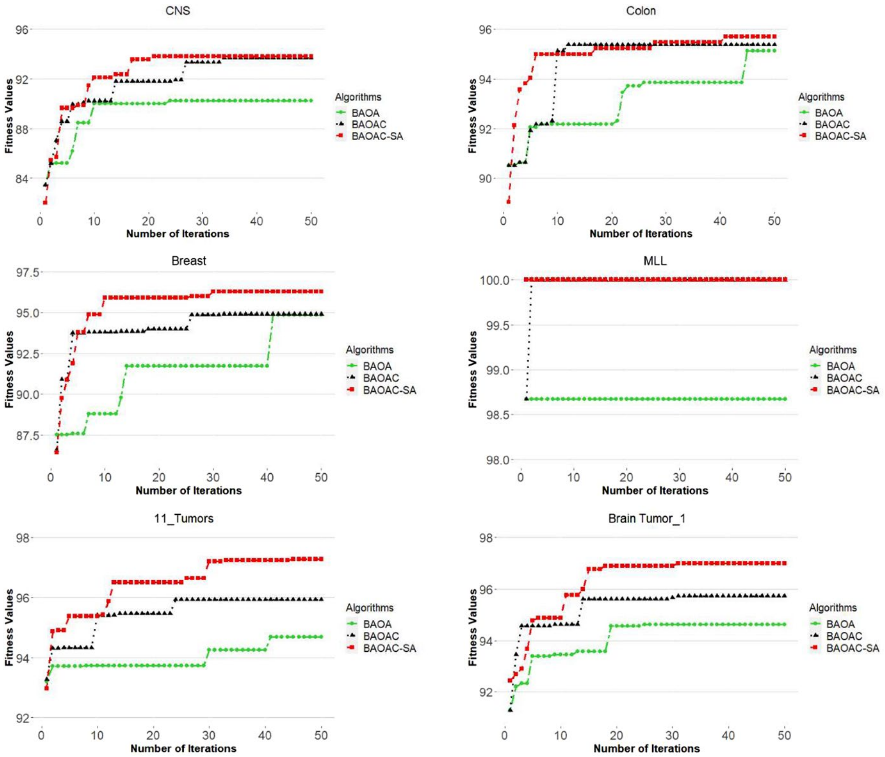


Fig. 8 The convergence behavior of BAOA, BAOAC, and BAOAC-SA for 6 datasets

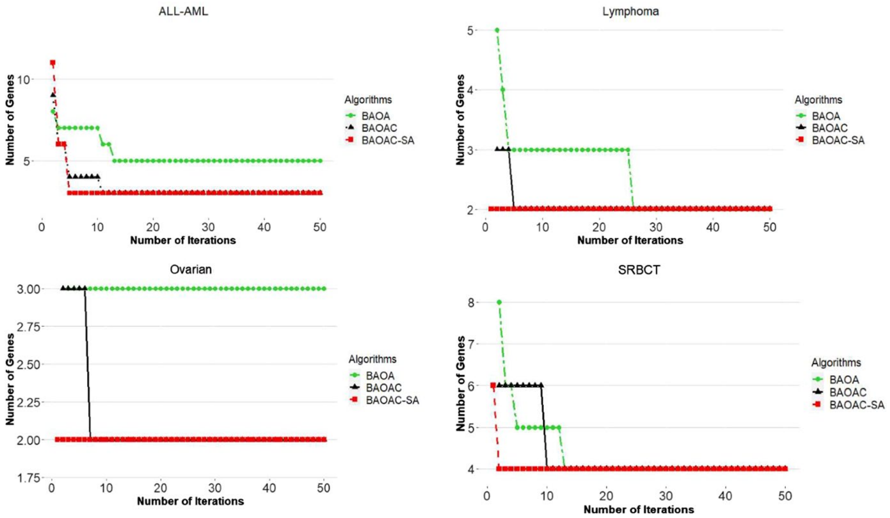


Fig. 9 The convergence behavior of BAOA, BAOAC, and BAOAC-SA for 4 datasets with perfect accuracy (100%)

(100%), BAOAC-SA can converge better than BAOA and BAOAC in terms of the number of selected genes.

In brief, the experimental results revealed that BAOAC-SA achieved the best balance between the number of selected genes and the classification accuracy on all datasets. The experimental results confirm that using the crossover operator and SA can help BAOA in further exploring the interactions between the genes, leading to the most intriguing part of the search space, which is comprised of a limited set of informative genes.

5.4 Comparison with other GS techniques

The efficacy of the suggested approach is further evaluated in this section by comparing mRMR-BAOAC-SA to current state-of-the-art methods. Table 8 displays the outcomes based on the average classification accuracy (ACC) using LOOCV and the number of selected genes. The BAOAC-SA was performed independently several times for each dataset, because it is not a filter approach, and a different subset of genes with various performances can be obtained over independent runs. After each run, the generated results were assessed using LOOCV, which is the most promising evaluation criterion and is not prone to change [4]. Table 8 shows the average LOOCV results, which are compared to numerous well-known state-of-the-art approaches in the terms of classification accuracy and the number of selected genes. The comparing results are either LOOCV or tenfold CV. For more detailed information, see Table 1. İt is worth mentioning that the purpose of gene selection is to make the classifier as accurate as possible while using the fewest number of biologically significant genes. This allows for straightforward model interpretation.

Table 8 Comparing the performance of the proposed mRMR-BAOAC-SA with the literature methods over 10 public cancer datasets using a random forest classifier

Dataset	Method	Accuracy	Reference	Dataset	Method	Accuracy	Reference
Brain Tumor1	MIM-MFOA-SVM	83.15 (12)	[7]	*Colon Cancer*	TOPSIS-Jaya-NB	97.76 (18.90)	[1]
	CFS-TLBO-SA-SVM	96.98 (12)	[22]		BDE-SVM_{Rankf}	75.0 (4)	[55]
	VLPSO-LS-KNN	75.54 (102.1)	[46]		DRF0-CFS-KNN	90 (10)	[47]
	BCO-KNN	96.30 (20.5)	[41]		SFS-MB-KNN	82.40 (3.7)	[44]
	PS-NSGA-KNN	73.81 (57.8)	[48]		RFR-IDGA-RF	93.39 (4.7)	[24]
	BAOAC-SA	96.21 (7.5)			EF-PSO dICA-SVM	94.73 (15)	[49]
					CFS-TLBO-SA-SVM	99.01 (11)	[22]
					RFR-BBHA-Bagging	93.33 (4.5)	[38]
					MRMR-BDF-BBHA-SVM	97.02 (14.4)	[2]
					BAOAC-SA	95.68 (7.66)	
11_Tumors	MIM-MFOA-SVM	79.48(33)	[7]	*Breast cancer*	RFE-BDF-SVM	86.22 (7237)	[51]
	CFS-TLBO-SA-SVM	92.23(13)	[22]		DRF0-INT-SVM	84.21 (97)	[47]
	VLPSO-LS-KNN	82.81(367.4)	[46]		MIM-AGA-ELM	82.47(6)	[45]
	BCO-KNN	89.62(24.1)	[41]		Chi-squared -BBHA-RF	87.77 (6.2)	[39]
	PS-NSGA-KNN	83.94(338.3)	[48]		EGS-AGA-SVM	88.64 (17)	[42]
	BAOAC-SA	96.946 (24)			MRMR-BA-SVM	88.8 (18.3)	[52]
					RFE- PSO-BBHA/SPLSDA	97.72 (12.9)	[40]
					MRMR-BDF-BBHA-SVM	90.21(12)	[2]
					BAOAC-SA	96.188 (10.8)	

Table 8 (continued)

Dataset	Method	Accuracy	Reference	Dataset	Method	Accuracy	Reference
MLL	RMRMR-HBA-SVM	100 (8)	[5]	*Ovarian Cancer*	RMRMR-HBA-SVM	100 (3.07)	[5]
	HAS-MB-NB	99.55 (6.6)	[53]		HAS-MB-NB	99.81 (5.73)	[53]
	MRMR-BA-SVM	79.86 (19.03)	[52]		MRMR-BA-SVM	100 (3.83)	[52]
	MBEGA	94.33 (32.1)	[54]		MBEGA	99.71 (9)	[54]
	CFS-iBPSO	100 (30.8)	[37]		CFS-iBPSO	100 (3.3)	[37]
	TOPSIS-Jaya-NB	99.62 (12.90)	[1]		TOPSIS-Jaya-NB	99.52 (18.50)	[1]
	MRMR-BDF-BBHA-SVM	100 (5.25)	[2]		MRMR-BDF-BBHA-SVM	100 (2.66)	[2]
	BAOAC-SA	100 (3.6)			BAOAC-SA	100 (2.6)	
ALL-AML	RMRMR-HBA-SVM	100 (4.07)	[5]	*CNS*	RMRMR-HBA-SVM	100 (11.2)	[5]
	HAS-MB-NB	99.34 (5)	[53]		HAS-MB-NB	84.17 (7.43)	[53]
	MRMR-BA-SVM	100 (5.23)	[52]		MRMR-BA-SVM	94.22 (19.2)	[52]
	MBEGA	95.89 (12.8)	[54]		MBEGA	72.21 (2.5)	[54]
	CFS-iBPSO-NB	100 (4.3)	[37]		CFS-iBPSO-NB	95.84 (10.5)	[37]
	TOPSIS-Jaya-NB	100 (16.10)	[1]		TOPSIS-Jaya-NB	96.22 (8.7)	[1]
	RFR- IDGA-RF	98.06 (7.4)	[24]		RFR-BBHA-Bagging	90 (3.33)	[38]
	MRMR-BDF-BBHA-SVM	100 (4)	[2]		RFE-PSO-BBHA/SPLSDA	99.16 (10.5)	[40]
	BAOAC-SA	100 (3.3)			MRMR-BDF-BBHA-SVM	97.19 (39.7)	[2]
					BAOAC-SA	94.60 (5.2)	

Table 8 (continued)

Dataset	Method	Accuracy	Reference	Dataset	Method	Accuracy	Reference
SRBCT	IG-MBKH	100 (6.30)	[3]	*Lymphoma*	RMRMR-HBA-SVM	100 (8.13)	[5]
	RMRMR-MGWO	100 (37.5)	[6]		HAS-MB-NB	99.99 (3.75)	[53]
	SFS-MB-KNN	89.40 (5.7)	[44]		MRMR-BA-SVM	86.36 (37.73)	[52]
	CFS-TLBO-SA-SVM	99.91 (11)	[22]		MBEGA	97.68 (34.3)	[54]
	F-IDGA-SVM	100 (18)	[4]		CFS-iBPSO-NB	100 (24)	[37]
	TOPSIS-Jaya-NB	100 (15.80)	[1]		TOPSIS-Jaya-NB	98.33 (15.20)	[1]
	VLPSO-LS-KNN	99.7 (71.4)	[46]		BAOAC-SA	100 (2)	
	BCO-KNN	100 (7.4)	[41]				
	PS-NSGA-KNN	96.35 (18.6)	[48]				
	BAOAC-SA	100 (5.8)					

The number in the parentheses denotes the number of genes

According to Table 8, BAOAC-SA achieved higher or equivalent classification accuracy with a lower number of selected genes than other comparative approaches, on seven out of ten datasets (i.e. Brain Tumor_1, 11_Tumors, MLL, ALL_AML, SRBCT, Ovarian, Lymphoma). The proposed method is ranked second in the Breast cancer dataset in the terms of classification accuracy and ranked third in terms of the number of selected genes. Considering the Colon datasets, the proposed approach rated fourth in terms of classification accuracy and fifth in terms of the number of selected genes when compared with nine state-of-the-art techniques. Moreover, MRMR-BAOAC-SA is rated sixth in the CNS dataset in terms of classification accuracy, but ranked third in terms of the number of selected genes. To sum up, the proposed method performs better than existing methods on seven out of ten datasets and has a comparable performance with other methods on the rest of the datasets. Therefore, the BAOAC-SA can be considered an efficient and effective optimization algorithm for solving the GS problem.

5.5 Biological interpretation

From a biological standpoint, only a few genes (biomarkers) are significant for cancer diagnosis in microarray datasets [42]. As a result, identifying such biomarkers using an NIOA based GS method will be beneficial to medical experts for proper disease diagnosis. The suggested strategy tries to find a significant subset of tiny genes with the highest classification accuracy. The stability of the wrapper-based gene selection methods is typically determined by comparing the overlap of selected genes across successive runs. The Index and genes names of the most frequently repeated genes (> 7 out of 10 runs) derived from the suggested BAOAC-SA approach are presented in Table 9. We can observe that the proposed BAOAC-SA

Table 9 The most frequently repeated genes selected by the proposed BAOAC-SA approach for each dataset

Dataset	Index of Genes	Gene names
CNS	2426, 2474, 5637	D43682_s_at, S71824_at, M98539_at
Colon	765, 1671, 1473,1562, 1954, 1644	M76378, M26383, R54097, R49459, M35531, R80427
Breast	1409, 3232, 18,761, 20,342, 24,107	NM_001756, NM_020123, Contig55662_RC, Contig55574_RC, Contig49670_RC
Ovarian	182, 1683	MZ2.8234234, MZ246.12233
ALL-AML	1882, 2354	M27891_at, M92287_at
MLL	8212, 11,297	35307_at, 1389_at
Lymphoma	1622, 3739	GENE2668X, GENE1602X
SRBCT	1003, 509, 255, 1955, 335, 188, 2046, 1319	–
11_Tumors	5860, 3917, 3334, 10,162, 7185, 10,509, 11,348, 3929, 4336, 7978, 3406	–
Brain Tumor_1	4801, 2116, 1563, 2330, 3113, 5175	–

behaves consistently and ensures adequate stability in selecting discriminative genes.

Heatmaps are commonly used in biology to show the expression levels of numerous genes in different samples. Moreover, the unique patterns between different

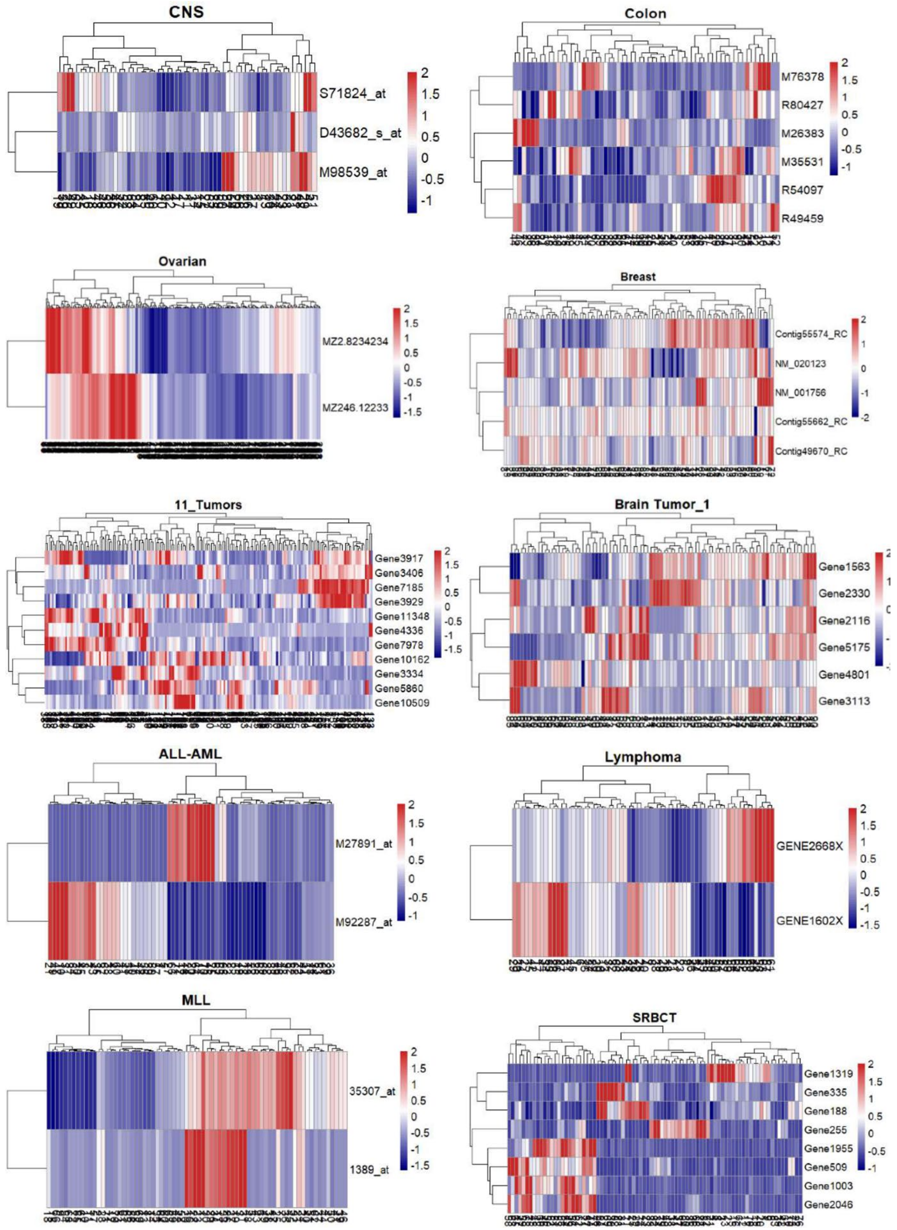


Fig. 10 Clustering and heatmap for representing expression level of all selected genes

groups such as disease groups and control groups can be viewed through clustering. Heatmap can help identify genes that are frequently regulated or find biological signatures associated with a specific disease. Figure 10 depicts the heatmaps of discriminating genes of each dataset obtained from the proposed model (Table 9), to investigate differentially-expressed genes. In heatmaps, each column demonstrates a sample and each row demonstrates a gene. Changes in gene expression are represented by the color and intensity of the boxes. Red denotes up-regulated genes, blue denotes down-regulated genes, and white represents unchanged expression in the heatmaps.

To do gene expression profiling, more people are turning to RNA-Seq rather than Microarray. Our proposed algorithm can also be utilized to analyze RNAseq data. Identifying differentially expressed genes between different sample groups is fundamental research in many RNAseq studies. The capacity of ML-based algorithms to identify differentially expressed genes (DEGs) from RNAseq data has been demonstrated in previous research [2, 67, 68] confirming the applicability of our proposed technique in discovering DEGs.

6 Conclusion

For cancer diagnosis and treatment, a precise classification of cancer into their types is a critical issue. Due to the large number of genes involved and the small number of associated tissues, selecting meaningful genes from high-dimensional biomedical data is one of the most difficult problems in cancer classification. This study introduced an efficient wrapper gene selection technique based on an improved binary AOA to find the biologically significant genes that collaborate for cancer detection. The suggested method employs mRMR as a filter, hybrid BAOAC-SA as a wrapper, and the RF classifier as an evaluator to predict cancerous genes. The mRMR filter technique is used to fine-tune the search space initially. Next, the SA algorithm and crossover operator are combined with BAOA to design the hybrid algorithm BAOAC-SA in order to find the best gene subset. AOA is a recently suggested NIOA whose capability for gene selection problems has not yet been properly investigated. The wrapper BAOA technique is hybridized with SA to improve BAOA's exploitation capabilities, which leads to an increase in the stability of chosen genes. The suggested algorithm also includes a crossover schema to boost the exploratory behavior of BAOA. Ten benchmark microarray datasets were used to test the performance of the suggested approach. Experimental results indicated that in terms of the number of selected genes and classification accuracy, the proposed approach can produce better results than the other current state-of-the-art methods on most of the datasets. The finding proves that mRMR-BAOAC-SA is a promising approach for solving the GS problem in biomedical data analysis.

The proposed approach can be used to solve other complex high-dimensional problems such as text mining and image data in future studies. The combination of BAOA with other NIOAs, filters, and classifiers can also be investigated in order to increase model accuracy in solving various optimization issues. Furthermore, the performance of the algorithm for gene selection in clustering of single-cell RNA-seq data can be investigated.

Author contributions EP and EP designed the model and the computational framework. Both carried out the implementation and performed the experiment and wrote the manuscript.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Chaudhuri A, Sahu TP (2021) A hybrid feature selection method based on binary Jaya algorithm for micro-array data classification. Comput Electr Eng 90:106963. https://doi.org/10.1016/j.compeleceng.2020.106963
2. Pashaei E, Pashaei E (2021) Gene selection using hybrid dragonfly black hole algorithm: a case study on RNA-seq COVID-19 data. Anal Biochem 627:114242. https://doi.org/10.1016/j.ab.2021.114242
3. Zhang G, Hou J, Wang J et al (2020) Feature selection for microarray data classification using hybrid information gain and a modified binary krill herd algorithm. Interdiscip Sci Comput Life Sci 12:288–301. https://doi.org/10.1007/s12539-020-00372-w
4. 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
5. Alomari OA, Khader AT, Al-Betar MA, Awadallah MA (2018) A novel gene selection method using modified MRMR and hybrid bat-inspired algorithm with β-hill climbing. Appl Intell 48:4429–4447. https://doi.org/10.1007/s10489-018-1207-1
6. Alomari OA, Makhadmeh SN, Al-Betar MA et al (2021) Gene selection for microarray data classification based on gray wolf optimizer enhanced with TRIZ-inspired operators. Knowl Based Syst 223:107034. https://doi.org/10.1016/J.KNOSYS.2021.107034
7. Dabba A, Tari A, Meftali S, Mokhtari R (2021) Gene selection and classification of microarray data method based on mutual information and moth flame algorithm. Expert Syst Appl 166:114012. https://doi.org/10.1016/J.ESWA.2020.114012
8. Yan C, Ma J, Luo H, Patel A (2019) Hybrid binary coral reefs optimization algorithm with simulated annealing for feature selection in high-dimensional biomedical datasets. Chemom Intell Lab Syst 184:102–111. https://doi.org/10.1016/j.chemolab.2018.11.010
9. Abualigah L, Diabat A, Mirjalili S et al (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609. https://doi.org/10.1016/J.CMA.2020.113609
10. Bansal P, Gehlot K, Singhal A, Gupta A (2022) Automatic detection of osteosarcoma based on integrated features and feature selection using binary arithmetic optimization algorithm. Multimed Tools Appl 81:8807–8834. https://doi.org/10.1007/S11042-022-11949-6/TABLES/6
11. Agushaka JO, Ezugwu AE (2021) Advanced arithmetic optimization algorithm for solving mechanical engineering design problems. PLoS ONE 16:e0255703. https://doi.org/10.1371/JOURNAL.PONE.0255703
12. Premkumar M, Jangir P, Kumar BS et al (2021) A new arithmetic optimization algorithm for solving real-world multiobjective CEC-2021 constrained optimization problems: diversity analysis and validations. IEEE Access 9:84263–84295. https://doi.org/10.1109/ACCESS.2021.3085529
13. Chauhan S, Vashishtha G (2021) Mutation-based arithmetic optimization algorithm for global optimization. In: 2021 Int Conf Intell Technol (CONIT). https://doi.org/10.1109/CONIT51480.2021.9498358
14. Ewees AA, Al-qaness MAA, Abualigah L et al (2021) Boosting arithmetic optimization algorithm with genetic algorithm operators for feature selection: case study on cox proportional hazards model. Mathematics 9:2321. https://doi.org/10.3390/MATH9182321

15. Ibrahim RA, Abualigah L, Ewees AA et al (2021) An electric fish-based arithmetic optimization algorithm for feature selection. Entropy 2021 23:1189. https://doi.org/10.3390/E23091189
16. Abualigah L, Diabat A, Sumari P, Gandomi AH (2021) A novel evolutionary arithmetic optimization algorithm for multilevel thresholding segmentation of COVID-19 CT images. Processes 9:1155. https://doi.org/10.3390/PR9071155
17. Khatir S, Tiachacht S, Le Thanh C et al (2021) An improved artificial neural network using arithmetic optimization algorithm for damage assessment in FGM composite plates. Compos Struct 273:114287. https://doi.org/10.1016/J.COMPSTRUCT.2021.114287
18. Mafarja M, Mirjalili S (2017) Hybrid whale optimization algorithm with simulated annealing for feature selection. Neurocomputing 260:302–312. https://doi.org/10.1016/j.neucom.2017.04.053
19. Abdel-Basset M, Ding W, El-Shahat D (2021) A hybrid Harris Hawks optimization algorithm with simulated annealing for feature selection. Artif Intell Rev 54:593–637. https://doi.org/10.1007/s10462-020-09860-3
20. Khamees M, Albakry A, Shaker K (2018) Multi-objective feature selection: hybrid of Salp Swarm and simulated annealing approach. In: Al-mamory SO, Alwan JK, Hussein AD (eds) Al-mamory S, Alwan J, Hussein A (eds) New Trends in Information and Communications Technology Applications. NTICT 2018. Communications in Computer and Information Science. Springer, Cham, pp 129–142
21. Chantar H, Tubishat M, Essgaer M, Mirjalili S (2021) Hybrid binary dragonfly algorithm with simulated annealing for feature selection. SN Comput Sci 2:1–11. https://doi.org/10.1007/s42979-021-00687-5
22. Shukla AK, Singh P, Vardhan M (2019) A new hybrid wrapper TLBO and SA with SVM approach for gene expression data. Inf Sci (Ny) 503:238–254. https://doi.org/10.1016/j.ins.2019.06.063
23. Pandey AC, Rajpoot DS (2019) Feature selection method based on grey wolf optimization and simulated annealing. Recent Adv Comput Sci Commun 14:635–646. https://doi.org/10.2174/2213275912666190408111828
24. Pashaei E, Pashaei E (2019) Gene selection using intelligent dynamic genetic algorithm and random forest. In: 11th International Conference on Electrical and Electronics Engineering (ELECO), pp 470–474
25. Paniri M, Dowlatshahi MB, Nezamabadi-pour H (2020) MLACO: A multi-label feature selection algorithm based on ant colony optimization. Knowl Based Syst 192:105285. https://doi.org/10.1016/J.KNOSYS.2019.105285
26. Tabakhi S, Moradi P (2015) Relevance-redundancy feature selection based on ant colony optimization. Pattern Recognit 48:2798–2811. https://doi.org/10.1016/j.patcog.2015.03.020
27. Gao L, Ye M, Lu X, Huang D (2017) Hybrid method based on information gain and support vector machine for gene selection in cancer classification. Genomics Proteomics Bioinform 15:389–395. https://doi.org/10.1016/j.gpb.2017.08.002
28. Al-Betar MA, Alomari OA, Abu-Romman SM (2020) A TRIZ-inspired bat algorithm for gene selection in cancer classification. Genomics 112:114–126. https://doi.org/10.1016/j.ygeno.2019.09.015
29. Pashaei E, Ozen M, Aydin N (2016) Biomarker discovery based on BBHA and AdaboostM1 on microarray data for cancer classification. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS. Institute of Electrical and Electronics Engineers Inc., pp 3080–3083
30. Dash R (2021) An adaptive harmony search approach for gene selection and classification of high dimensional medical data. J King Saud Univ Comput Inf Sci 33:195–207. https://doi.org/10.1016/j.jksuci.2018.02.013
31. Shukla AK, Singh P, Vardhan M (2020) An adaptive inertia weight teaching-learning-based optimization algorithm and its applications. Appl Math Model 77:309–326. https://doi.org/10.1016/j.apm.2019.07.046
32. Bir-Jmel A, Douiri SM, Elbernoussi 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
33. Kundu R, Chattopadhyay S, Cuevas E, Sarkar R (2022) AltWOA: altruistic whale optimization algorithm for feature selection on microarray datasets. Comput Biol Med 144:105349. https://doi.org/10.1016/J.COMPBIOMED.2022.105349

34. Ghobaei-Arani M (2021) A workload clustering-based resource provisioning mechanism using biogeography based optimization technique in the cloud based systems. Soft Comput 25:3813–3830. https://doi.org/10.1007/S00500-020-05409-2/FIGURES/11
35. Ghobaei-Arani M, Shahidinejad A (2021) An efficient resource provisioning approach for analyzing cloud workloads: a metaheuristic-based clustering approach. J Supercomput 77:711–750. https://doi.org/10.1007/S11227-020-03296-W/FIGURES/14
36. Aslanpour MS, Dashti SE, Ghobaei-Arani M, Rahmanian AA (2018) Resource provisioning for cloud applications: a 3-D, provident and flexible approach. J Supercomput 74:6470–6501. https://doi.org/10.1007/S11227-017-2156-X/FIGURES/20
37. 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
38. Pashaei E, Ozen M, Aydin N (2016) Gene selection and classification approach for microarray data based on random forest ranking and BBHA. In: 3rd IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2016. Institute of Electrical and Electronics Engineers Inc., pp 308–311
39. Pashaei E, Aydin N (2017) Binary black hole algorithm for feature selection and classification on biological data. Appl Soft Comput J 56:94–106. https://doi.org/10.1016/j.asoc.2017.03.002
40. Pashaei E, Pashaei E, Aydin N (2019) Gene selection using hybrid binary black hole algorithm and modified binary particle swarm optimization. Genomics 111:669–686. https://doi.org/10.1016/j.ygeno.2018.04.004
41. Wang H, Jing X, Niu B (2017) A discrete bacterial algorithm for feature selection in classification of microarray gene expression cancer data. Knowl Based Syst 126:8–19. https://doi.org/10.1016/j.knosys.2017.04.004
42. 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
43. Wang A, An N, Chen G et al (2015) Accelerating wrapper-based feature selection with K-nearest-neighbor. Knowl Based Syst 83:81–91. https://doi.org/10.1016/j.knosys.2015.03.009
44. Wang A, An N, Yang J et al (2017) Wrapper-based gene selection with Markov blanket. Comput Biol Med 81:11–23. https://doi.org/10.1016/j.compbiomed.2016.12.002
45. 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
46. Tran B, Xue B, Zhang M (2019) Variable-length particle swarm optimization for feature selection on high-dimensional classification. IEEE Trans Evol Comput 23:473–487. https://doi.org/10.1109/TEVC.2018.2869405
47. Bolón-Canedo V, Sánchez-Maroño N, Alonso-Betanzos A (2015) Distributed feature selection: An application to microarray data classification. Appl Soft Comput J 30:136–150. https://doi.org/10.1016/j.asoc.2015.01.035
48. Zhou Y, Zhang W, Kang J et al (2021) A problem-specific non-dominated sorting genetic algorithm for supervised feature selection. Inf Sci (Ny) 547:841–859. https://doi.org/10.1016/j.ins.2020.08.083
49. Mollaee M, Moattar MH (2016) A novel feature extraction approach based on ensemble feature selection and modified discriminant independent component analysis for microarray data classification. Biocybern Biomed Eng 36:521–529. https://doi.org/10.1016/j.bbe.2016.05.001
50. Pashaei E, Yilmaz A, Aydin N (2016) A combined SVM and Markov model approach for splice site identification. In: 6th International Conference on Computer and Knowledge Engineering (ICCKE 2016), pp 200–204
51. Medjahed SA, Saadi TA, Benyettou A, Ouali M (2017) Kernel-based learning and feature selection analysis for cancer diagnosis. Appl Soft Comput J 51:39–48. https://doi.org/10.1016/j.asoc.2016.12.010
52. Ahmad Alomari O, Tajudin Khader A, Azmi Al-Betar M, Mohammad Abualigah L (2017) Gene selection for cancer classification by combining minimum redundancy maximum relevancy and bat-inspired algorithm. Int J Data Min Bioinform 19:32–51. https://doi.org/10.1504/IJDMB.2017.088538
53. 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
54. 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

55. Apolloni J, Leguizamón G, Alba E (2016) Two-hybrid wrapper-filter feature selection algorithms applied to high-dimensional microarray experiments. Appl Soft Comput J 38:922–932. https://doi.org/10.1016/j.asoc.2015.10.037
56. Delahaye D, Chaimatanan S, Mongeau M (2019) Simulated Annealing: From basics to applications. In: Handbook of Metaheuristics. International Series in Operations Research and Management Science. Springer, Cham, pp 1–35
57. Hameed SS, Hassan WH, Latiff LA, Muhammadsharif FF (2021) A comparative study of nature-inspired metaheuristic algorithms using a three-phase hybrid approach for gene selection and classification in high-dimensional cancer datasets. Soft Comput 2513(25):8683–8701. https://doi.org/10.1007/S00500-021-05726-0
58. Mafarja M, Mirjalili S (2018) Whale optimization approaches for wrapper feature selection. Appl Soft Comput J 62:441–453. https://doi.org/10.1016/j.asoc.2017.11.006
59. Pashaei E, Pashaei E (2020) Gene selection for cancer classification using a new hybrid of binary black hole algorithm. In: The 28th IEEE Conference on Signal Processing and Communications Applications (SIU2020). Institute of Electrical and Electronics Engineers Inc.
60. Dabba A, Tari A, Meftali S (2021) Hybridization of moth flame optimization algorithm and quantum computing for gene selection in microarray data. J Ambient Intell Humaniz Comput 12:2731–2750. https://doi.org/10.1007/s12652-020-02434-9
61. Bommert A, Sun X, Bischl B et al (2020) Benchmark for filter methods for feature selection in high-dimensional classification data. Comput Stat Data Anal 143:1–19. https://doi.org/10.1016/j.csda.2019.106839
62. Pashaei E, Ozen M, Aydin N (2016) Random forest in splice site prediction of human genome. In: Kyriacou E, Christofides S, Pattichis C (eds) XIV Mediterranean Conference on Medical and Biological Engineering and Computing. IFMBE Proceedings, vol 57. Springer, Cham, pp 518–523
63. Pashaei E, Yilmaz A, Ozen M, Aydin N (2016) A novel method for splice sites prediction using sequence component and hidden Markov model. In: Proceedings of the 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS. Institute of Electrical and Electronics Engineers Inc., pp 3076–3079
64. Mirjalili S, Lewis A (2013) S-shaped versus V-shaped transfer functions for binary particle swarm optimization. Swarm Evol Comput 9:1–14. https://doi.org/10.1016/j.swevo.2012.09.002
65. Beheshti Z (2021) UTF: Upgrade transfer function for binary meta-heuristic algorithms. Appl Soft Comput 106:1–28. https://doi.org/10.1016/j.asoc.2021.107346
66. Ambroise C, McLachlan GJ (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. Proc Natl Acad Sci 99:6562–6566. https://doi.org/10.1073/pnas.102102699
67. Wenric S, Shemirani R (2018) Using supervised learning methods for gene selection in RNA-Seq case-control studies. Front Genet 9:297. https://doi.org/10.3389/FGENE.2018.00297/BIBTEX
68. Feng J, Niu X, Zhang J, Wang JH (2022) Gene selection and classification of scRNA-seq data combining information gain ratio and genetic algorithm with dynamic crossover. Wirel Commun Mob Comput 2022:1–16. https://doi.org/10.1155/2022/9639304