Neural Computing and Applications (2022) 34:6427–6451
https://doi.org/10.1007/s00521-021-06775-0

ORIGINAL ARTICLE

An efficient binary chimp optimization algorithm for feature selection in biomedical data classification

Elnaz Pashaei[1] · Elham Pashaei[2]

Received: 9 July 2021 / Accepted: 21 November 2021 / Published online: 17 January 2022

Abstract

Accurate classification of high-dimensional biomedical data highly depends on the efficient recognition of the data's main features which can be used to assist diagnose related diseases. However, due to the existence of a large number of irrelevant or redundant features in biomedical data, classification approaches struggle to correctly identify patterns in data without a feature selection algorithm. Feature selection approaches seek to eliminate irrelevant and redundant features to maintain or enhance classification accuracy. In this paper, a new wrapper feature selection method is proposed based on the chimp optimization algorithm (ChOA) for biomedical data classification. The ChOA is a newly proposed metaheuristic algorithm whose capability for solving feature selection problems has not been investigated yet. Two binary variants of the ChoA are introduced for the feature selection problem. In the first approach, two transfer functions (S-shaped and V-shaped) are used to convert the continuous version of ChoA to binary. In addition to the transfer function, the crossover operator is utilized in the second approach to improve the ChOA's exploratory behavior. To validate the efficiency of the proposed approaches, five publicly available high-dimensional biomedical datasets, and a few datasets from different domains such as life, text, and image are employed. The proposed approaches were then compared with six well-known wrapper-based feature selection methods, including multi-objective genetic algorithm (GA), particle swarm optimization (PSO), Bat algorithm (BA), ant colony optimization (ACO), firefly algorithm (FA), and flower pollination (FP) algorithm, as well as two standard filter-based feature selection methods using three different classifiers. The experimental results demonstrate that the proposed approaches can effectively remove the least significant features and improve classification accuracy. The suggested wrapper feature selection techniques also outperform the GA, PSO, BA, ACO, FA, FP, and other existing methods in the terms of the number of selected genes, and classification accuracy in most cases.

Keywords Chimp optimization algorithm · Feature selection · Biomedical data · Classification · Optimization

1 Introduction

Biological data, such as microarrays, are one of the most essential analytical tools for medical researchers and biologists. Microarray data analysis allows for more accurate diagnosis and prognosis of patient diseases, as well as better clinical decision-making. The main challenging issues associated with microarray data are the curse of dimensionality and complex interaction between features. Microarray data consist of a small number of patient samples with a large number of features (genes), most of which are redundant and irrelevant. The presence of irrelevant and redundant features in biological datasets might obscure the important ones, causing many learning algorithms to perform poorly. To overcome this challenge, feature selection (FS) is required [1]. FS is typically considered as a preprocessing mechanism that aims to choose a subset of significant features to alleviate overfitting, improve the accuracy and interpretability of the learned model, speed up the learning process, and reduce dataset storage memory requirements [2]. From a biological

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

Elnaz Pashaei
elnazpashaei@aydin.edu.tr

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

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

perspective, FS helps molecular biologists to identify the molecular mechanism driving cancer gene expression, interpret the underlying pattern of data to discover new therapeutic targets for those selected features, and reduce clinical costs [3, 4].

The FS process involves two main stages: feature-subset search and feature-subset assessment. In the first stage, a search strategy is needed to explore the search space to find the optimal feature subset, and a learning algorithm is required in the second stage to assess the quality of the selected feature subset [5]. A wide variety of FS methods for classification issues have been proposed, which can be divided into three classes: wrapper-based, filter-based, and hybrid approaches. Filter-based approaches focus solely on the interior characteristics of training data to rank features and are independent of any learning algorithm. Wrapper-based techniques, on the one hand, rely on a specialized learning algorithm (classifier) to determine the optimal set of features. The wrapper approaches typically outperform filter approaches in terms of classification performance, but require high computational time, especially for high-dimensional data due to the frequent use of learning algorithms in their search strategy. The filter-based methods are less time-consuming than the wrapper-based method since no learning algorithm is involved in their search stage. However, the filter-based models suffer in terms of accuracy because they are not iterative models and their search strategy only comprises a single iteration, making them get stuck in local optima easily [6]. A combination of these two approaches has also been proposed as hybrid models to combine their strengths.

Recently swarm intelligence (SI)-based optimization methods have gained a lot of interest due to their high performance in tackling FS problems. Some of most popular SI-based FS methods are Whale optimization algorithm (WOA) [1], ant colony optimization (ACO) [7], bat algorithm (BA) [3, 8], artificial bee colony (ABC) [9, 10], particle swarm optimization (PSO) [11], biogeography based optimization (BBO) [12], genetic algorithm (GA) [13–15], harmony search algorithm (HSA) [16], flower pollination (FP) algorithm [17], grasshopper optimization algorithm (GOA) [18], firefly algorithm (FA) [19], and binary dragonfly (BDF) algorithm [20]. However, because of the intricate interactions between features in biomedical data, the large feature search space, and the stochastic nature of the approaches, most of these algorithms are susceptible to the stagnation problem and may suffer from degraded performance [3, 5, 21]. Therefore, the door is still open for more improvement, and a strong search method capable of exploring the search space more complete, eluding local minima, and exploiting the global optimum more reliably is demanded to better address the FS problem.

The Chimp Optimization Algorithm (ChOA) [22] is a recent SI algorithm that mimics the chimps' intelligent group hunting (IGH) behavior and their social diversity. For hunting prey, chimps are separated into four groups and undertake four different actions: dividing, chasing, blocking, and attacking. These four groups of the chimps and several operators such as diverse intelligence and sexual motivation were mathematically modeled to create ChOA with high exploration and exploitation ability. The ChOA has been successfully used to train neural network (NN) [23] parameters, and a hybrid version [24] and a modified version of ChOA [25] have also been presented. However, to the best of our knowledge, the effectiveness of ChOA has not been investigated in the feature selection problem.

In this paper, two new wrapper feature selection approaches based on ChOA are proposed to identify the optimal feature subset for biomedical data classification. In addition to using the main operators of the ChOA, some modifications need to be done on the algorithm in order to solve FS problems since the original version of the ChOA was created to address continuous problems. This paper primarily proposes two binary ChOA (BChOA) variants:

- In the first version, two transfer functions (S-shaped and V-shaped) are suggested to map the continuous data into binary ones.
- In the second version, the crossover operator is integrated with BChOA (BChOA-C) to empower the algorithm's exploration capabilities.

The fundamental idea is to assign a binary structure to each chimp in the population that indicates whether or not a feature belongs in the final list of features. As a fitness function, the accuracy of two learning algorithms, Naïve Bayes (NB) [26, 27] and K-nearest neighbor (K-NN) [26, 28] is used in the proposed algorithms. The learning algorithm is trained with the given features and tenfold cross-validation (CV) is used to evaluate each chimp (candidate solution). Extensive experiments were carried out on five popular microarray datasets (large-scale biological data), as well as various small biological, text, and image datasets. For performance evaluation, the proposed approaches are compared with six previous binary optimization algorithms: PSO, BA, ACO, GA, FA, FP, and other well-regard filtering approaches. The conducted experiments demonstrate that the proposed approaches have better performance compared to the above-mentioned algorithms and several current state-of-the-art methods in the term of accuracy and number of selected features. Furthermore, the results show that incorporating a crossover operator into the BChOA improves the classification accuracy of the model.

The rest of the paper is structured as follows. Section 2 presents a review of recent literature on the FS techniques in biomedical data classification. Section 3 provides all the details about the ChOA. The suggested binary ChOA for feature selection is discussed in Sect. 4. The details of the conducted experiments and achieved results on well-known datasets are given in Sect. 5. The conclusion and future work are presented in Sect. 6.

2 Related works

FS is an NP-hard search problem [29] that aims to find the optimal number of features (attributes) from the original dataset without sacrificing the performance of the classification. Based on the number of features in the original dataset, the complexity of the problem grows exponentially. Therefore, metaheuristic search (MHS) algorithms have been employed to improve the obtained result and the computational time in large problems. There have been several attempts to review the FS methods [30, 31]. In this section, we briefly review various FS approaches which can be categorized into three classes: filter, wrapper (MHS-Based), and hybrid models.

The filter technique ranks each feature based on its discriminating power between different classes without considering any learning algorithm. In filter methods, various criteria are utilized to find the features' importance such as information theory, cross-entropy, symmetrical uncertainty, correlation, similarity, and statistical measures. Examples include Information Gain (IG) [32], Markov blanket [33], Correlation-based FS (CFS) [11], Fast Correlation Based Filter (FCBF)[34], Fisher score [35], Relief-F [36], Chi-square [37], Random Forest Ranking (RFR) [38], simplified silhouette filter (SSF) [39], Condition Mutual Information Maximization (CMIM) [40], Double Input Symmetrical Relevance (DISR) [41], and Minimum Redundancy Maximum Relevance (mRMR) [42]. The mRMR is a popular filter method in biological data that seeks features with the greatest relevance to the class and the least redundancy between them [8, 43].

The filter methods usually carry out on microarray datasets as a preprocessing step, to reduce the dataset's high dimensionality by removing redundant and irrelevant features within a reasonable time. The obtained dataset is then fed to wrapper algorithms to find the optimal feature subset. The hybrid model, which is extensively employed in biological data, is based on the idea of sequentially applying both a filter and a wrapper technique. Since the hybrid model supplies a reduced feature set to the wrapper technique, an extended search of feature subsets is avoided [6].

The wrapper technique utilizes a learning algorithm as a fitness function to evaluate the quality of the feature subset. The wrapper model is an iterative search procedure in which the learning algorithm's accuracy is employed to direct search space at each iteration. Various learning algorithms such as Support Vector Machine (SVM) [44], Decision Tree (DT) [45], Artificial Neural Network (ANN) [46], K-NN [47], and NB [48] have been used in wrapper-based FS method for better classification of biomedical data. Generally, wrapper-based methods can be divided into two categories: greedy and random search approaches. Examples of greedy methods include sequential forward selection (SFS) [49], backward selection (SBS) [50], and hill-climbing algorithm [8] in which a single feature is added or removed iteratively in a greedy manner. The random search approaches are mainly based on MHS algorithms. MHS algorithms are nature-inspired optimization algorithm (NIOA) that use randomness in their search strategy to explore a large portion of the search space.

The majority of NIOA's have been introduced for continuous search space and they should be converted to binary form to solve discrete optimization problems like FS [51]. Several transfer functions (TFs) have been utilized for these purposes within the NIOAs such as S-shaped [52], V-shaped [53], U-shaped [54], and X-shaped [55]. The TFs play a key role in the efficiency of binary NIOAs.

Different binary NIOAs have been proposed for FS so far using these TFs, which can be classified into four groups: evolution-based, SI-based, physics-based, and human-related approaches. The most popular evolution-based FS algorithm is a binary genetic algorithm (GA) [13, 14, 26, 38] that simulates Darwinian evolution concepts. Some of the most popular physical-based FS methods which mimic the physical concept in the world are binary black hole algorithm [43, 56–58], simulated annealing (SA) [59, 60], and gravitational search algorithm (GSA) [61]. The examples of human-based approaches in FS of biomedical data include Teaching Learning-based optimization (TLBO) algorithm [59], the BrainStorm Optimization (BSO) algorithm [62], and the JAYA algorithm [63] which are inspired by human behaviors in society. SI-based algorithms (SIA) are inspired by animals' behavior in herds, flocks, colonies, or schools. SIAs have shown to be quite competitive with the other three types of NIOAs, and have several advantages over them, such as fewer parameters, fewer operators, and the ability to remember search space [22]. Some of the well-known suggested binary SIA for FS are PSO [2, 11], GOA [64], ABC algorithm [9, 10, 65], BA [3], Krill Herd algorithm (BKH) [32], Gray Wolf Optimizer (GWO) [21], Bacterial Foraging Optimization (BFO) algorithm [66], cuckoo search algorithm [67], and Moth Fame Optimization (MFO) algorithm [68]. The following is an analysis of some of the selected literature.

Wang et al. [33] proposed to integrate the Markov blanket filter technique into wrapper-based SFS for feature selection of biomedical data. This model speeds up the FS process by reducing the number of candidate features for the wrapper evaluation. K-NN, NB, and DT classifiers were used in this study as the fitness function for feature subset evaluation.

Dashtban and Balafar [26] introduced a hybrid evolutionary algorithm called an intelligent dynamic genetic algorithm (IDGA) for the FS of microarray data. First, Fisher score was used to reduce dimensionality and provide statistically significant features to the next step. Then, the IDGA method was applied to find the optimal feature subset. Moreover, three classifiers, namely SVM, NB, and K-NN, were utilized to measure the performance of IDGA. Later on, Zhou et al. [14] developed a problem-specific non-dominated sorting genetic algorithm (PS-NSGA), as a multi-objective FS algorithm for high-dimensional data classification. The suggested algorithm included a non-dominated sorting with a preference for accuracy, a rapid bit mutation operator, a mutation-retry operator, and a combination operator to solve the FS issue efficiently. Their study reported the proposed PS-NSGA approach achieves better classification performance and smaller feature subsets in comparison to existing evolutionary and traditional feature selection methods.

Shukla et al. [59] introduced a new hybrid wrapper strategy for determining the optimal feature subsets to predict cancerous-genes., based on the combination of TLBO with the SA algorithm called TLBOSA. First, CFS was utilized to filter the redundant feature from the biological datasets. Then, TLBOSA was used to identify the subset of the most informative features. Also, a new TF was proposed to convert the continuous version of TLBOSA to binary. It was found that TLBOSA outperforms other wrappers in terms of classification accuracy and a small subset of features.

An improved binary krill herd (MBKH) algorithm for FS has been developed by Zhang et al. [32]. The study utilized the IG filter method as a preprocessing step to rank and remove redundant features. Then, MBKH was applied to find out the best feature subset. In the suggested method, the hyperbolic tangent function was employed as the transfer function, and the chaos memory weight factor was introduced into the movement operators of the MBKH algorithm to enhance its local and global search abilities. A new hybrid filter-wrapper strategy was proposed in [21], in which robust mRMR (rmRMR) was used as a filter approach to choose the top-ranked features, and modified GWO (MGWO) with SVM evaluator was utilized as a wrapper approach to seeking the best subset of features. To increase the diversity of the population in the MGWO, TRIZ-inspired optimization operators were introduced in the original GWO, which result in a practical and effective FS tool to select the most informative features.

Although several binary NIOAs for FS have been presented, due to the stochastic nature of the NIOAs according to the No-Lunch theorem, there is still room for more improvements. This is one of the key inspirations for this study, in which two novel binary versions of ChOA, are developed and compared to current well-known discrete NIOAs in the literature for FS of high-dimensional biomedical data.

3 Chimp optimization algorithm

ChOA is a new SI-based optimization algorithm that was proposed by Khishe and Mosavi in 2020 [22]. The basic idea for ChOA comes from the chimps' intelligence and sexual motivation in their group haunting, which differs from that of other social hunters. Due to its simplicity, local optima avoidance, high convergence speed, and low computational overhead, this approach has been widely used to determine the best possible solutions for complex optimization problems [24, 25].

Chimps' hunting behavior is divided into two phases: exploration and exploitation. The exploration entails moving, blocking, and chasing the prey which leads to discovering a wider region of the search space globally, while exploitation entails attacking the prey that provides local search potential across the promising areas discovered during the exploration process. To implement the steps of hunting, four groups of chimps are used: driver, chaser, barrier, and attacker. Each chimp in the population represents a candidate solution in the search space, and attacker, barrier, chaser, and driver chimps represent the best (leader), second best, third best, and fourth-best solutions, respectively. At each iteration, after determining the position of the attacker (x_{attacker}), barrier (x_{barrier}), chaser (x_{chaser}), and driver (x_{driver}) chimps (i.e., the four best chimps), the rest of the chimps (x_{chimp}) are forced to update their positions according to the locations of these four best chimps using the following equations:

$$\begin{aligned}
&x_1(t+1) = x_{\text{attacker}}(t) - A1.(D_{\text{attacker}}),\ D_{\text{attacker}} \\
&\quad = \left|C1.x_{\text{attacker}} - m.x_{\text{chimp}}(t)\right| \\
&x_2(t+1) = x_{\text{barrier}}(t) - A2.(D_{\text{barrier}}),\ D_{\text{barrier}} \\
&\quad = \left|C2.x_{\text{barrier}} - m.x_{\text{chimp}}(t)\right| \\
&x_3(t+1) = x_{\text{chaser}}(t) - A3.(D_{\text{chaser}}),\ D_{\text{chaser}} \\
&= \left|C3.x_{\text{chaser}} - m.x_{\text{chimp}}(t)\right| \\
&x_4(t+1) = x_{\text{driver}}(t) - A4.(D_{\text{driver}}),\ D_{\text{driver}} \\
&\quad = \left|C4.x_{\text{driver}} - m.x_{\text{chimp}}(t)\right|
\end{aligned} \tag{1}$$

$$x_{\text{chimp}}(t+1) = \frac{x_1 + x_2 + x_3 + x_4}{4} \tag{2}$$

where t is the current iteration's number, $x_{\text{chimp}}(t)$ implies the location of each solution in iteration t. A and C indicate coefficient vectors that are formulated in Eqs. (3) and (4).

$$\begin{aligned} A_1 &= 2.f.r_{11} - f, \quad C_1 = 2.r_{12} \\ A_2 &= 2.f.r_{21} - f, \quad C_2 = 2.r_{22} \\ A_3 &= 2.f.r_{31} - f, \quad C_3 = 2.r_{32} \\ A_4 &= 2.f.r_{41} - f, \quad C_4 = 2.r_{42} \end{aligned} \tag{3}$$

$$f = 2 - t * \left(\frac{2}{T}\right) \tag{4}$$

where f decreases linearly from 2 to 0, and T indicates the maximum number of iterations. r_1 and r_2 are random scaled factors within [0,1] which are calculated as follows:

$$\begin{aligned} c_1g_1 &= 1.95 - \left(\frac{2 * t^{\frac{1}{4}}}{T^{\frac{1}{3}}}\right),\ r_{11} = c_1g_1 * rand(), \\ c_2g_1 &= \frac{2 * t^{\frac{1}{3}}}{T^{\frac{1}{3}}} + 0.5,\ r_{12} = c_2g_1 * rand() \\ c_1g_2 &= 1.95 - \left(\frac{2 * t^{\frac{1}{3}}}{T^{\frac{1}{4}}}\right),\ r_{21} = c_1g_1 * rand(), \\ c_2g_2 &= \left(2 * \frac{t^3}{T^3}\right) + 0.5,\ r_{22} = c_2g_1 * rand() \\ c_1g_3 &= \left(-3 * \frac{t^3}{T^3}\right) + 1.5,\ r_{31} = c_1g_3 * rand(), \\ c_2g_3 &= \frac{2 * t^{\frac{1}{3}}}{T^{\frac{1}{3}}} + 0.5,\ r_{32} = c_2g_3 * rand() \\ c_1g_4 &= \left(-2 * \frac{t^3}{T^3}\right) + 1.5,\ r_{41} = c_1g_4 * rand(), \\ c_2g_4 &= \left(2 * \frac{t^3}{T^3}\right) + 0.5,\ r_{42} = c_2g_4 * rand(). \end{aligned} \tag{5}$$

where $rand()$ stands for uniform distribution with a scale of 0 to 1. In Eq. (1) m indicates a chaotic value between 0 and 1 derived from one of the chaotic maps mentioned below:

$$\begin{aligned} &\text{Quadratic}: \ x_{i+1} = m = x_i^2 - c, c = 1 \\ &\text{Gauss/mouse}: \ x_{i+1} = m = \begin{cases} 1, \ x_i = 0 \\ \dfrac{1}{\text{mod}(x_i, 1)}, \ \text{otherwise} \end{cases} \\ &\text{Logistic}: \ x_{i+1} = m = \alpha x_i(1 - x_i), \alpha = 4 \\ &\text{Singer}: \ x_{i+1} = m = \mu * \left(\begin{array}{c} 7.86x_i - 23.31x_i^2 + \\ 28.75x_i^3 - 13.302875x_i^4 \end{array}\right), \ \mu \\ &\quad = 1.07 \\ &\text{Bernoulli}: \ x_{i+1} = m = 2x_i(\text{mod}1) \\ &\text{Tent}: \ x_{i+1} = m = \begin{cases} \dfrac{x_i}{0.7}, x_i < 0.7 \\ \dfrac{10}{3}(1 - \&x_i), 0.7 \le x_i \end{cases} \end{aligned} \tag{6}$$

According to Eqs. (1) and (2), the chimps update their positions according to the population's best locations where D is the distance between the chimp (x_{chimp}) and a prey. The A and C values are adjusted to monitor the areas where a solution can be found near the best solution. The m value represents the influence of the chimps' sexual motivation, which causes them to behave erratically in the final stages of the hunting process, releasing their hunting responsibilities and desperately attempting to obtain meat. Chimps use this chaotic behavior in the final stage of ChOA to overcome local optima stagnation and slow convergence speed issues when solving complex problems.

To update the chimps' position during optimization a probability of %50 is assumed to select between the chaotic model and the normal position updating process. The following equation expresses the model:

$$x_{\text{chimp}}(t+1) = f(x) = \begin{cases} \dfrac{x_1 + x_2 + x_3 + x_4}{4}, & \mathit{if}\ (p < 0.5) \\ m, & \mathit{if}\ (p \ge 0.5) \end{cases} \tag{7}$$

where p is a random number in [0,1].

The ChOA's pseudo-code is shown in Fig. 1. The algorithm starts by creating a randomly generated population and setting the positions of the attacker (x_{attacker}), barrier (x_{barrier}), chaser (x_{chaser}), and driver (x_{driver}) to zero vector. The algorithm repeats the steps below until it reaches a termination criterion. First, each solution in the population is evaluated using a fitness function. Second, the algorithm updates the positions of the attacker, barrier, chaser, driver, and their scores. Third, the algorithm updates the values of f, r_1, r_2, and m coefficients using Eqs. (4) to (6). Third, using the values of r_1, r_2, and f parameters, the values of the main coefficients of A and C are determined by Eq. (3). Finally, Eqs. (1), (2), and (7) are used to update the chimps' positions. As a result, the best possible solution, i.e., the attacker's position, is returned.

4 The proposed ChOA-based wrapper FS methods

The ChOA has been originally defined to operate in a continuous solution space and has succeeded in tackling a variety of continuous problems. However, the capability of the ChOA in solving binary high-dimensional problems, such as FS has not yet been investigated. This paper aims to generalize the ChOA to discrete settings, and two new binary variants of ChOA are proposed to address the FS problem. In the first approach, two different TFs, including S-shaped and V-shaped [52], are utilized in ChOA (BChOA) to convert continuous search space to a binary one. In the second approach, in addition to TFs, the position of the best solution in BChOA is updated using the crossover operator (BChOA-C) which improves the exploration ability of the algorithm.

Figure 2 explores the methodology of the proposed ChOA-based wrapper FS algorithms. Below steps have been followed to execute the suggested BChOA-C algorithm.

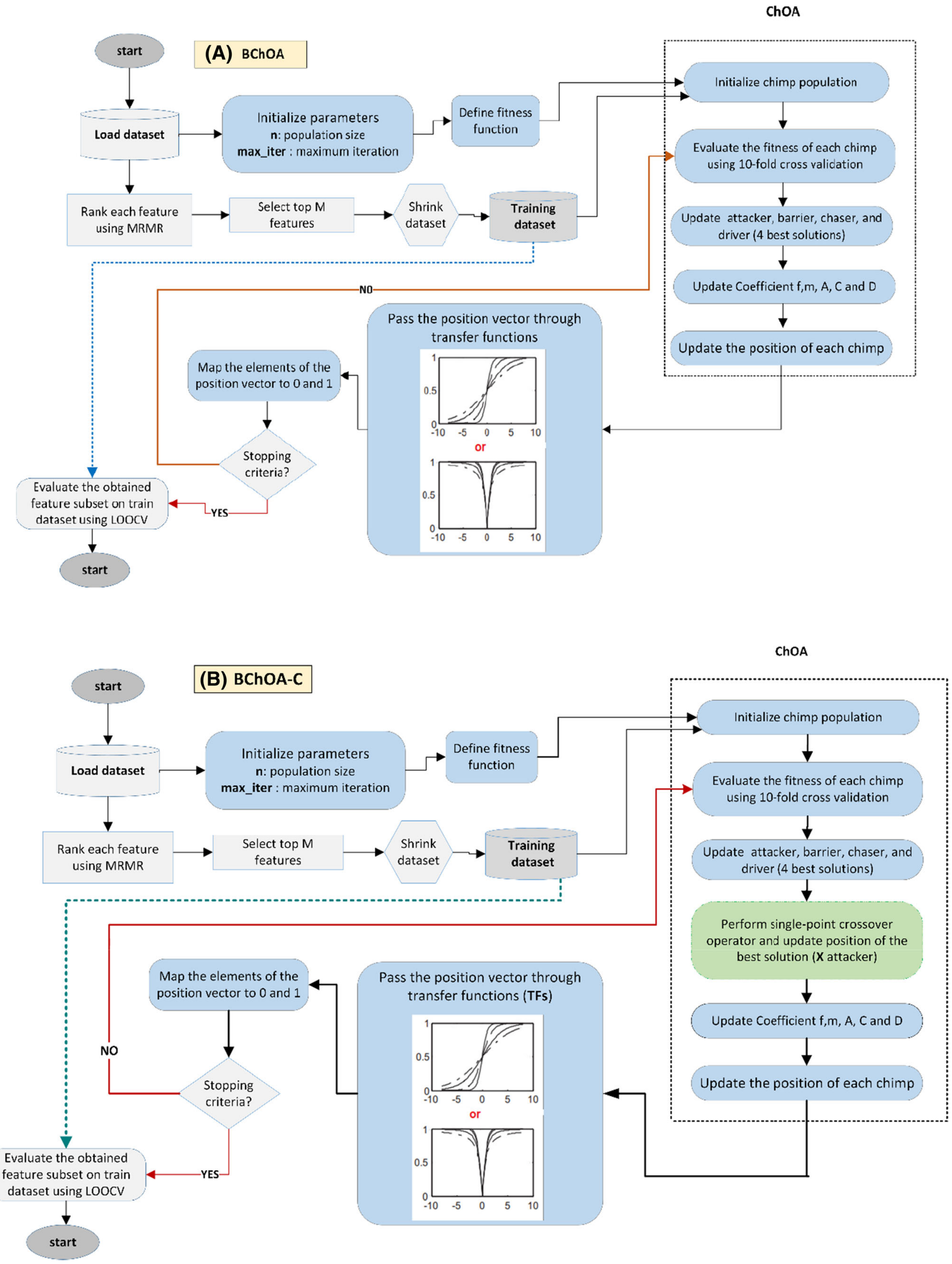
Fig. 2 Flowchart of the suggested ChOA-based feature selection ▶ wrapper approaches for biomedical data classification. (**A**) First approach: the standard binary version of ChOA (BChOA). (**B**) Second approach: an improved version of the BChOA, in which BChOA is hybridized with the crossover operator (BChOA-C)

Step 1 The original dataset is utilized to create a population of chimps. Each chimp in the swarm is regarded as a candidate feature subset. The mRMR method is used to filter out noisy and redundant genes before population initialization on microarray datasets.

Algorithm 1: Pseudo-code of ChOA

```
Initialize the population size ($n$), the maximum number of iterations ($max\_iter$)
Initialize the positions of attacker ($x_{attacker}$), barrier ($x_{\text{barrier}}$), chaser ($x_{\text{chaser}}$), driver ($x_{\text{driver}}$), and their scores ($attacker_{score}, barrier_{score}, chaser_{score}, driver_{score}$) to 0
Initialize the population of chimps randomly (initialize their positions)
while ($t < max\_iter$)
    for $i$= $1{:}\,n$ do
        fitness=calculate the fitness of each chimp with position $pos_{[i,d]}$ $i = (1, 2, \ldots, n)$ and $d = (1, .., dimension)$
        if fitness < $attacker_{score}$
            update attacker's position and score as the best search agent
        if fitness > $attacker_{score}$ and fitness < $barrier_{score}$
            update barrier's position and score as the second-best search agent
        if fitness > $attacker_{score}$ and fitness > $barrier_{score}$ and fitness < $chaser_{score}$
            update chaser's position and score as the third best search agent
        if fitness > $attacker_{score}$ and fitness > $barrier_{score}$ and fitness > $chaser_{score}$ and fitness < $driver_{score}$
            update driver's position and score as the fourth-best search agent
    End for
    $f = 2 - t * ((2)/max\_iter)$
    Calculate the dynamic coefficients $c_k g_j$, $k = 1, 2$ for each group $j = 1, 2, 3, 4$ by Eq. (5).
    Generate random values $r_{jk}$ for each group $j = 1,2,3,4$ utilizing the corresponding coefficient $c_k g_j$ by Eq. (5).
    $m$ = chaotic_value (Eq. (6))
    Calculate $A_j = 2.f.r_{j1} - f$ and $C_j = 2.r_{j2}$ for each group (Eq. (4))
    for $i = 1{:}\,n$ do
        for $d = 1$: dimension do
            $D_{attacker} = |C1\,.\,x_{attacker} - m.\,\text{POS}[i,d]|$ , $x_1 = x_{attacker} - A1.\,(D_{attacker})$
            $D_{barrier} = |C2\,.\,x_{barrier} - m.\,\text{POS}[i,d]|$ , $x_2 = x_{barrier} - A2.\,(D_{barrier})$
            $D_{chaser} = |C3.\,x_{chaser} - m.\,\text{POS}[i,d]|$ , $x_3 = x_{\text{chaser}} - A3.\,(D_{chaser})$
            $D_{driver} = |C4.\,x_{driver} - m.\,\text{POS}[i,d]|$ , $x_4 = x_{\text{driver}} - A4.\,(D_{driver})$
            generate a random value $\mu$ [0,1]
            if $\mu < 0.5$
                $\text{POS}[i,d] = \frac{x_1 + x_2 + x_3 + x_4}{4}$
            else
                $\text{POS}[i,d] = m$
        End for
    End for
    $t = t + 1$
End while
return the best solution= $x_{attacker}$
```

Fig. 1 Pseudo-code of the ChOA


ChOA
start
(A) BChOA
Load dataset
Initialize parameters
n: population size
max_iter : maximum iteration
Define fitness function
Initialize chimp population
Evaluate the fitness of each chimp using 10-fold cross validation
Update attacker, barrier, chaser, and driver (4 best solutions)
Update Coefficient f,m, A, C and D
Update the position of each chimp
Rank each feature using MRMR
Select top M features
Shrink dataset
Training dataset
NO
Pass the position vector through transfer functions
Map the elements of the position vector to 0 and 1
Stopping criteria?
YES
Evaluate the obtained feature subset on train dataset using LOOCV
start
or
1
0.5
0
-10
-5
0
5
10
ChOA
start
(B) BChOA-C
Load dataset
Initialize parameters
n: population size
max_iter : maximum iteration
Define fitness function
Initialize chimp population
Evaluate the fitness of each chimp using 10-fold cross validation
Update attacker, barrier, chaser, and driver (4 best solutions)
Perform single-point crossover operator and update position of the best solution (X attacker)
Update Coefficient f,m, A, C and D
Update the position of each chimp
Rank each feature using MRMR
Select top M features
Shrink dataset
Training dataset
NO
Pass the position vector through transfer functions (TFs)
Map the elements of the position vector to 0 and 1
Stopping criteria?
YES
Evaluate the obtained feature subset on train dataset using LOOCV
start
or

Step 2 The ChOA is performed. The population is evaluated using a fitness function by employing a classifier.

Step 3 After assigning fitness values to each solution, the four best solutions are selected from the population.

Step 4 The single-point crossover operator is performed between the best and other chimps of the population, and the better offspring is selected as the new position of attacker chimp.

Step 5 The algorithm updates the main coefficients (f, m, C, A, and D), and the rest of the chimps are forced to update their location according to the best chimp position.

Step 6 TFs are used to determine the probability of altering the elements of position vectors. Either Eqs. (9–10) or Eqs. (11–12) are used to update the elements of position vectors in BChOA. Thus, the movements are restricted to 0 and 1 values.

Step 7 The algorithm repeats the above steps until it reaches the value of the maximum iteration.

Step 8 If the termination condition is satisfied, the algorithm will stop and return the best solution, i.e., position of the attacker chimp, in the current population.

Step 9 The algorithm returns to step 2 if the termination condition is not met.

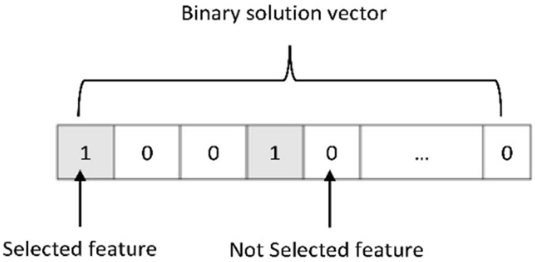


Fig. 3 Binary representation of candidate solutions

Fig. 4 Pseudo-code of the suggested BChOA for FS

Algorithm 2: Pseudo-code of proposed BChOA for FS

```
Initialize the population size ($n$), the maximum number of iterations ($max\_iter$)
Initialize the positions of chimps $pos_i \in \{0,1\}^d$, $i = (1, 2, \ldots, n)$ and $d = (1, \ldots, dimension)$
Initialize the positions of the attacker, barrier, chaser, driver, and their scores
while ($t < max\_iter$)
    for $i$= 1: $n$ do
        calculate the fitness of each chimp using KNN or NB
        update attacker's position (the best search agent)
        update barrier's position (the second-best search agent)
        update chaser's position (the third-best search agent)
        update driver's position (the fourth-best search agent)
    End for
    Update $f$, $m$, $A_j$, and $C_j$ ($j = 1, 2, 3, 4$)
    for $i$ = 1: $n$ do
        for $d$ = 1: dimension do
            $D_{attacker} = |C1 . x_{attacker} - m.\mathrm{POS}[i, d]|$ , $x_1 = x_{attacker} - A1.(D_{attacker})$
            $D_{barrier} = |C2 . x_{barrier} - m.\mathrm{POS}[i, d]|$ , $x_2 = x_{barrier} - A2.(D_{barrier})$
            $D_{chaser} = |C3. x_{chaser} - m.\mathrm{POS}[i, d]|$ , $x_3 = x_{\mathrm{chaser}} - A3.(D_{chaser})$
            $D_{driver} = |C4. x_{driver} - m.\mathrm{POS}[i, d]|$ , $x_4 = x_{\mathrm{driver}} - A4.(D_{driver})$
                $\mathrm{POS}[i, d] = \frac{x_1 + x_2 + x_3 + x_4}{4}$
                $Tf = \frac{2}{1 + e^{-2*\mathrm{POS}[i,d]}} - 1$
                if $Tf > 0.6$ then POS[$i, d$]=1 else POS[$i, d$]=0
                or
                $Tf = \left|\frac{\mathrm{POS}[i, d]}{\sqrt{1 + (\mathrm{POS}[i, d])^2}}\right|$
                if $Tf >$ r then $\mathrm{POS}[i, d] = \neg\mathrm{POS}[i, d]$ else $\mathrm{POS}[i, d] = \mathrm{POS}[i, d]$
        End for
    End for
    $t = t + 1$
End while
return the best solution= $x_{attacker}$
```

Each of the following subsections describes a component of the proposed methods in detail.

4.1 Solution representation

In general, to apply the ChOA as a wrapper-based FS technique, the search space should be modeled by candidate feature subsets or binary solutions. Each candidate solution (chimp) is represented by a one-dimensional binary vector of d elements, $pos = (f_1, f_2, \ldots, f_d)$, where d indicates the problem dimension (i.e., the number of all features in the original dataset). In the solution vector, each bit f_j has a value of "1" or "0." The value 1 indicates that the corresponding feature will be maintained, whereas 0 indicates that it will be discarded. The binary representation of a ChOA's solution can be seen in Fig. 3.

ChOA initially starts with a set of randomly generated binary solutions or candidate feature subsets. They can be seen as the position of chimps in search space. Only features coded in ones will be considered in the evaluation. The algorithm utilizes an objective (fitness) function to evaluate the effectiveness of each solution during the search.

4.2 Fitness function

The fitness function is an important factor to consider when designing any NIOA. Finding the optimal feature subset is a difficult task in the wrapper-based FS methods since it aims to find a subset with the highest accuracy and fewest number of features. The solution is better if it has fewer features and has a higher classification accuracy. An efficient fitness function should take into account these two conflicting objectives and strike a balance between them [3]. In this study, the suggested fitness function combines the number of selected features in the solution with the solution's classification accuracy in order to assign a fitness value to each subset using the following equation:

$$\text{Fitness} = \alpha \times \text{acc}(D) + \beta \times \frac{|d| - |R|}{|d|} \tag{8}$$

where acc(D) represents the prediction accuracy of a classifier on the training dataset (D) with subset features. The fitness function employs a classifier to evaluate solutions (feature subsets). In wrapper-based FS methods, the process of learning a classifier is concurrent with FS. Two widely used classifiers, NB and K-NN ($K = 5$) [26] are adopted for the fitness evaluation of solutions using tenfold CV. $|d|$ indicates the total number of features in the original dataset, and $|R|$ stands for the number of selected features in the solution. The parameters α and β determine the effects of the accuracy and number of selected features on the fitness value, respectively. α is in the interval of $[0, 1]$ and $\beta = (1 - \alpha)$. In this work, α was set to 0.8, since the classification accuracy of the solution is more important than the number of selected features.

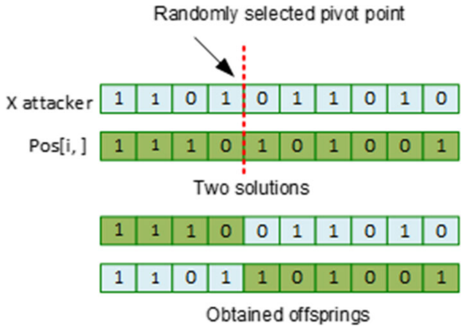


Fig. 5 The crossover process

It is worth reminding that after finding the best subset, leave-one-out- cross-validation (LOOCV) is used to report the final performance of the proposed binary ChOA on biomedical datasets.

4.3 Binary ChOA with transfer functions

The classical ChOA algorithm for continuous problems proceeds in discrete time considering n chimps $p_1, p_2, \ldots, p_n$, in which each chimp p_i has a position in step t, $pos_i^t \in \mathbb{R}^d$. In discrete problems, however, the search space contains binary position vectors $pos_i^t \in \{0, 1\}^d$. Four best solutions estimate the prey's location, and other chimps in the search space update their positions within the prey's vicinity. The main challenge in the design of binary ChOA is that how the algorithm's movement Eq. (7) 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. TFs force the chimps to move in a binary space by restricting their movements to 0 and 1 values. The ChOA has been adjusted to fit into the FS problem by employing two distinct TFs from two different families: S-shaped and V-shaped [52].

First, a hyperbolic tangent sigmoid (tansig) TF is employed in ChOA to convert the continuous algorithm to a binary version. It is utilized to modify chimps' position based on the following rules:

$$Tf\left(\text{pos}_{[i,d]}^{t}\right) = \frac{2}{1 + e^{-2*\text{pos}_{[i,d]}^{t}}} - 1 \tag{9}$$

$$\text{pos}_{[i,d]}^{t+1} = \begin{cases} 1, & Tf\left(\text{pos}_{[i,d]}^{t}\right) > 0.6 \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where $\text{pos}_{[i,d]}^{t+1}$ represents the bit value of dth dimension of ith chimp (position) in the next iteration $(t + 1)$. The tansig function generates a value in the range $[-1, 1]$ that specifies the likelihood of changing the elements of a position from 0 to 1 and vice versa. The value of the *dth* element in the *ith* position vector is set to 0 or 1 based on Eq. (10). The tansig function belongs to the S-shaped transfer function category, and it has been used in the proposed binary ChOA since it has the highest experimental performance among existing S-shaped TFs.

The TF plays a key role in the efficiency of a binary algorithm. TFs strike a balance between exploration and exploitation to reach an appropriate solution. It was shown that the choice of TFs could significantly affect the obtained results of the binary algorithm [53]. The high performance of the V-shaped family of TFs has already been proven in the literature for binary algorithms [51, 52]. So also the rules of a V-shaped function are explored in ChOA to generate the next binary positions. The function is defined as follows:

Algorithm 3: Pseudo-code of proposed BChOA-C for FS

```
Initialize the population size ($n$), the maximum number of iterations ($max\_iter$)
Initialize the positions of chimps $pos_i \in \{0,1\}^d$, $i = (1, 2, \ldots, n)$ and $d = (1, .., dimension)$
Initialize the positions of the attacker, barrier, chaser, driver, and their scores
while ($t < max\_iter$)
    for $i$= 1: $n$ do
        calculate the fitness of each chimp using KNN or NB
        update attacker's position (the best search agent)
        update barrier's position (the second-best search agent)
        update chaser's position (the third-best search agent)
        update driver's position (the fourth-best search agent)
        [$p1, p2$]=Crossover ($x_{attacker}$, POS[$i, d$] )
        Calculate the objective values of $p1, p2$
        if fitness value of $p1$ is better than fitness values of $p2$ and $x_{attacker}$
            $x_{attacker} = p1$
        else if fitness value of $p2$ is better than the fitness value of $x_{attacker}$
            $x_{attacker} = p2$
        End if
    End for
    Update $f$, $m$, $A_j$, and $C_j$ ( $j = 1, 2, 3, 4$)
    for $i = 1: n$ do
        for $d = 1$: dimension do
            $D_{attacker} = |C1 . x_{attacker} - m. \mathrm{POS}[i, d]|$    , $x_1 = x_{attacker} - A1. (D_{attacker})$
            $D_{barrier} = |C2 . x_{barrier} - m. \mathrm{POS}[i, d]|$    , $x_2 = x_{barrier} - A2. (D_{barrier})$
            $D_{chaser} = |C3. x_{chaser} - m. \mathrm{POS}[i, d]|$    , $x_3 = x_{chaser} - A3. (D_{chaser})$
            $D_{driver} = |C4. x_{driver} - m. \mathrm{POS}[i, d]|$    , $x_4 = x_{\mathrm{driver}} - A4. (D_{driver})$
            $\mathrm{POS}[i, d] = \frac{x_1 + x_2 + x_3 + x_4}{4}$
            $Tf = \left|\frac{\mathrm{POS}[i, d]}{\sqrt{1 + (\mathrm{POS}[i, d])^2}}\right|$
            if $Tf >$ r then POS[$i, d$] = ¬POS[$i, d$] else POS[$i, d$] = POS[$i, d$]
            or
            $Tf = \left|\frac{\mathrm{POS}[i, d]}{\sqrt{1 + (\mathrm{POS}[i, d])^2}}\right|$
            if $Tf >$ r then POS[$i, d$] = ¬POS[$i, d$] else POS[$i, d$] = POS[$i, d$]
        End for
    End for
    $t = t + 1$
End while
return the best solution= $x_{attacker}$
```

Fig. 6 Pseudo-code of the suggested BChOA-C for FS

Table 1 Main characteristics of the image, text, and biological datasets

Data set	#Features	#Samples	#Classes	Domain
Chess	36	3196(1669,1527)	2	Text
dbworld	242	64 (35, 29)	2	Text
Lymphography	18	148 (2,81,61,4)	4	Life
Lung cancer	56	32(9,13,10)	3	Life
Yale	1024	165	15	Face Image
SRBCT	2308	83 (29, 11, 18, 25)	4	Microarray
Prostate cancer	10,509	102 (50, 52)	2	Microarray
Leukemia	7129	72 (25, 47)	2	Microarray
Brain Tumor_1	5920	90 (60, 10, 10, 4, 6)	5	Microarray
11_Tumors	12,533	174 (26, 8, 26, 23, 12, 11, 7, 27, 6, 14, 14)	11	Microarray

Table 2 Key to comparative methods

Type	Abbreviation	Explanation	Reference
proposed ChOA-based wrapper feature selection methods	BChOA-ST	Binary Chimp Optimization Algorithm using S-shaped Transfer function	–
	BChOA-VT	Binary Chimp Optimization Algorithm using V-shaped Transfer function	
	BChOA-C	Binary Chimp Optimization Algorithm with Crossover operator	
	BChOA-VT-C	Binary Chimp Optimization Algorithm with Crossover operator using V-shaped Transfer function	
Classifiers	KNN	k-Nearest Neighbor classifier	[28]
	NB	Naïve Bayes classifier	[27]
	RBFNet	Radial Basis function-based Neural Network classifier	[70]
Validation approaches	LOOCV	Leave-One-Out Cross-Validation	[71]
	K-fold CV	K-fold Cross-Validation (k = 10 in this study)	
Filter-based feature selection methods	CFS	Correlation-based Feature Selection	[11]
	FCBF	Fast Correlation-Based Filter for feature selection	[34]
	mRMR	Maximum Relevance Minimum Redundancy algorithm	[42]
Wrapper-based feature selection methods	Flower Pollution	Flower Pollution (FP) optimization algorithm-based feature selection	[17]
	FireFly	Firefly optimization algorithm (FFA) for feature selection	[19]
	ACO	Ant Colony Optimization-based feature selection	[7]
	Bat	binary Bat algorithm for feature selection	[3]
	PSO	binary Particle Swarm Optimization algorithm	[2]
Hybrid-based feature selection methods	IG-MBKH	Information Gain and a Modified Binary Krill Herd Algorithm	[32]
	rMRMR-MGWO	Robust Maximum Relevance Minimum Redundancy filter and Modified Gray Wolf Optimizer with TRIZ-inspired operators wrapper	[21]
	SFS-MB	Wrapper-based Sequential Forward Selection with Markov Blanket	[33]
	TLBOSA-SVM	Teaching Learning-Based Optimization, Simulated Annealing, and Support Vector Machine	[59]
	F-Score-IDGA-F-SVM	Fisher score filter, Intelligent Dynamic Genetic Algorithm (IDGA) wrapper, and Support Vector Machine classifier	[26]
	VLPSO-LS-KNN	Variable-Length Particle Swarm Optimization with Local Search and k-nearest neighbor classifier	[2]
	BCO-KNN	Bacterial Colony Optimization and k-nearest neighbor classifier	[66]
	PS-NSGA	Problem-Specific Non-dominated Sorting Genetic Algorithm and k-nearest neighbor classifier	[14]

Table 3 Percentage of average performance using KNN, NB, and RBFNet classifiers on five microarray datasets

Data set	KNN				NB				RBFNet			
	ACC	Se	Sp	Fumes	ACC	Se	Sp	Fumes	ACC	Se	Sp	Fumes
SRBCT	83.13	83.1	83.8	83.1	**98.7**	**98.8**	**98.8**	**98.8**	92.77	92.8	93.7	92.8
Prostate Tumor	**84.3**	**84.3**	**84.8**	**84.3**	62.7	62.7	66.8	59.9	67.64	67.6	67.8	67.5
Leukemia	87.5	87.4	87.5	87.3	**98.6**	**98.6**	**98.6**	**98.6**	97.22	97.2	97.3	97.2
Brain Tumor_1	86.6	**86.7**	86.9	**97.0**	**87.7**	86.0	**87.8**	88.0	81.11	81.1	31.5	79.4
11_Tumors	75.8	75.9	76.6	75.2	**89.08**	**89.1**	**90.7**	**89.2**	84.48	84.5	86.5	84.5

Bold values represent the best results

$$Tf\left(\mathrm{pos}^t_{[i,d]}\right) = \left| \frac{\mathrm{pos}^t_{[i,d]}}{\sqrt{1 + \left(\mathrm{pos}^t_{[i,d]}\right)^2}} \right| \qquad (11)$$

$$\mathrm{pos}^{t+1}_{[i,d]} = \begin{cases} \neg\mathrm{pos}^t_{[i,d]}, Tf(\mathrm{pos}^t_{[i,d]}) > r \\ \mathrm{pos}^t_{[i,d]}, Tf(\mathrm{pos}^t_{[i,d]}) \leq r \end{cases} \qquad (12)$$

where $\neg\mathrm{POS}[i,d]$ is the complement operator of $\mathrm{POS}[i,d]$ and r is a random number in $\mathrm{U} \sim (0,1)$. The value of r has a significant impact on whether or not the single bit value of $pos^t_{[i,d]}$ is filliped for the next position. The flowchart of the ChOA with transfer functions is demonstrated in Fig. 2A, and Fig. 4 illustrates the general steps of the proposed Binary ChOA for FS (See lines 20 and 21).

It is worth noting that generally a filter-based FS method is applied to biomedical datasets (i.e., microarrays data) as a preprocessing step to provide strong initial data for the wrapper-based FS approach [43]. The search space of the wrapper-based FS approach involves a set of all possible feature subsets (2^d), and the size of the set grow exponentially as the number of features (d) increases. So, filter-based FS methods are utilized to effectively alleviate the complexities of the big search space. In this study, mRMR filter approach [42] was first used to reduce the dimensionality of feature space in the microarray dataset, and the features with the highest rank are selected to build a new dataset. Then, the wrapper approach is performed to seek the most informative feature subset.

4.4 Binary ChOA with crossover scheme

The crossover operator is integrated into BChOA to offer a new wrapper technique called BChOA-C, which boosts the proposed BChOA performance. The BAOA produces good results on a variety of microarray datasets, but in some datasets, it gets stuck at a sub-optimal solution. So, BChOA's behavior with crossover operator is being examined to improve BChOA exploring capability. The BChOA-C performs the crossover operator just after determining the four best solutions in the population. The crossover operation is done between the best solution $x_{attacker}$ and current solution $POS[i,]$ as shown in Eq. (13).

$$[p1, p2] = \mathrm{Crossover}(x_{\mathrm{attacker}}, \ \mathrm{POS}[i,]) \qquad (13)$$

This paper uses the single-point crossover, in which the two x_{attacker} and $POS[i,]$ mating vectors are severed at a random pivot point. Figure 5 shows an example of this technique. The binary bits are exchanged between two solutions as seen in Fig. 5, causing sudden changes in both solutions. Crossover has the potential to change the global best solution and to avoid the algorithm from getting stuck in local optima. In the meantime, the fitness values of the two offsprings produced by the crossover operator are compared with the best solution. The algorithm selects the best offspring as a new candidate solution. If the offspring has a higher fitness value than x_{attacker}, the position of x_{attacker} should be replaced and set to the offspring. Figure 2B shows the flowchart of the proposed BChOA-C with transfer functions and crossover operator, and Fig. 6 represents the pseudo-code of the algorithm.

5 Experimental result

5.1 Experimental setup

The R programming language was used to implement the suggested approaches. To evaluate the efficacy of the proposed BChOA-based techniques, five small-sized datasets in the text, life, and image domains, as well as five high-dimensional public microarray datasets with different types of diseases, were employed. The small datasets are obtained from the University of California, Irvine (UCI) Machine Learning Repository and https://jundongl.github.io/scikit-feature/datasets.html. The standard microarray gene expression datasets involving two binary and three multi-class datasets can be downloaded from https://data.mendeley.com/datasets/fhx5zgx2zj/1. Table 1 shows the characteristics of datasets used in this work. To find the optimal reduct, two wrapper FS techniques based on NB

Table 4 Classification accuracy using selected genes by filter-based gene selection methods

		50 top genes					100 top genes				
Classifier	Methods	SRBCT	Prostate Tumor	Leukemia	Brain Tumor_1	11_Tumors	SRBCT	Prostate Tumor	Leukemia	Brain Tumor_1	11_Tumors
KNN	mRMR	100	94.11	94.44	87.77	87.93	100	92.15	95.83	90	88.50
	CMIM	95.1	90.19	87.50	87.77	85.05	100	91.17	88.88	87.77	85.05
	Chi-square	98.7	94.11	93.05	77.77	81.60	100	90.19	97.22	87.77	82.18
	Relief-F	100	94.11	98.61	84.44	72.98	100	93.13	98.61	85.55	86.78
	DISR	100	90.19	94.44	82.22	83.90	100	91.17	94.44	85.55	88.50
NB	mRMR	100	90.19	97.22	86.66	86.78	100	89.21	95.83	88.88	89.65
	CMIM	95.1	89.21	98.61	84.44	85.01	100	87.25	97.22	83.33	86.78
	Chi-square	87.5	92.15	97.22	85.55	79.88	100	90.19	97.22	86.66	86.20
	Relief-F	100	94.11	95.83	85.55	81.03	100	94.11	97.22	82.22	85.05
	DISR	100	88.23	94.44	85.55	87.35	100	87.25	95.83	85.55	86.78
RBFNet	mRMR	100	88.23	97.22	90.00	87.35	100	88.23	94.44	86.66	87.35
	CMIM	97.59	85.29	97.22	88.55	82.18	100	90.19	100	84.44	88.50
	Chi-square	100	94.11	95.83	86.66	81.60	100	89.21	95.83	86.66	86.20
	Relief-F	100	93.17	95.83	83.33	81.03	100	92.15	95.83	86.66	89.65
	DISR	100	92.15	94.44	86.66	82.18	100	92.15	95.83	85.55	82.18

with Gaussian kernel and KNN classifiers ($K = 5$) are utilized.

Tenfold CV is used for the assessment of candidate solutions in the suggested methods. The experiments were carried out in two stages. Five high-dimensional microarray datasets (SRBCT, Prostate cancer, Leukemia, Brain Tumor_1, 11_Tumors) were utilized in the first phase to evaluate the suggested methodologies, and several small datasets in the text, life, and image domains were employed in the second phase. The trials were run on an Intel system with a Core i5 CPU running at 2.2 GHz and 8 GB of RAM. Both the population size and the maximum iteration parameter were set to 50. BChOA-based approaches are compared to state-of-the-art FS algorithms and other NIOAs, based on average classification accuracy and feature count from 20 independent runs.

The Weka program [69] (https://www.cs.waikato.ac.nz/ml/weka/), which is an open-source machine learning platform, was utilized for comparison. Table 2 summarizes all the proposed models and the comparative studies examined.

5.2 Results for microarray datasets

This study employs three commonly used classification algorithms; KNN with $k = 5$, NB, and radial basis function (RBF) neural networks (RBFNet). The performance of these classifiers on five microarray datasets is shown in

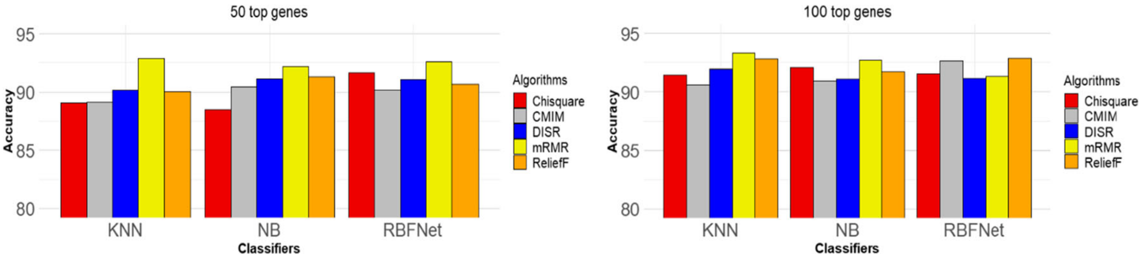


Fig. 7 Average classification accuracy of KNN, NB, and RBFNet classifiers for 50 and 100 top genes selected by different filter-based FS algorithms

Table 5 Comparison of two proposed BChOA with V-shaped and S-shaped TFs using **KNN** classifier based on the average number of selected genes, fitness value, classification accuracy (%), and computational time (second)

Data set	Metrics	# Genes		Fitness value		Accuracy		Time	
		BChOA-ST	BChOA-VT	BChOA-ST	BChOA-VT	BChOA-ST	BChOA-VT	BChOA-ST	BChOA-VT
SRBCT	*AVG*	7.2	**6.2**	80.1672	**80.1712**	100	100	168.756	**138.196**
	best	6	**4**	80.172	**80.18**	100	100	88.63	**77.99**
	worst	8	**7**	80.164	**80.168**	100	100	**220.98**	231.36
	STDEV	**0.83666**	1.30384	**0.0033**	0.00521	0.00	0.00	**64.16187**	78.30062
Prostate Tumor	*AVG*	6.8	**6**	78.27789	**78.34103**	**97.49493**	95.79832	78.978	**73.18167**
	best	4	**3**	78.72145	78.72145	**98.03**	97.05882	72.46	**60.48**
	worst	11	**8**	**77.97018**	77.92945	**95.09804**	94.11765	89.88	**80.12**
	STDEV	2.774887	**1.632993**	**0.403214**	0.4113963	**1.044394**	1.090877	**6.755769**	7.101891
Leukemia	*AVG*	**3.2**	3.4	**80.1916**	80.1912	100	100	**130.306**	137.602
	best	3	3	80.192	80.194	100	100	**117.01**	122.41
	worst	**4**	5	**80.19**	80.186	100	100	**142.95**	168.77
	STDEV	**0.4472136**	1.140175	0.0008944	**0.00228035**	0.00	0.00	**9.3311**	20.61237
Brain Tumor_1	*AVG*	10.8	**10.33333**	77.75862	**78.4918**	95.77778	95.7777	**222.746**	293.2867
	best	8	**6**	78.582	78.582	96.66667	96.66667	**145.2**	154.91
	worst	15	17	76.168	**76.98**	94.44444	**95.55556**	**351.74**	417.3
	STDEV	**2.588436**	3.777124	0.9855074	**0.6367874**	0.9296249	**0.4536088**	**77.80552**	110.3203
11_Tumors	*AVG*	20.28571	**17.8**	**75.74996**	75.68692	**93.77504**	93.75862	485.89	**197.876**
	best	16	16	**77.17038**	76.73978	**95.4023**	94.82759	220.61	**175.45**
	worst	24	**20**	**74.53495**	74.46158	90.22989	90.22989	718.19	**226.58**
	STDEV	2.9277	**1.48324**	1.019654	**0.9933987**	**1.872808**	2.056155	212.7285	**18.67895**

Bold values indicate best performance

Table 3. Based on Table 3, we conclude that all three classifiers perform well. Various filter-based algorithms, such as mRMR, CMIM, Chi-square, Relief-F, and DISR, were compared to validate the effectiveness of the recommended approach. The Min–Max technique was used to standardize the microarray expression data before applying filter-based methods. The average classification accuracy of KNN, NB, and RBFNet classifiers for 50 and 100 top genes selected by the above-mentioned filter-based FS algorithms are shown in Table 4 and Fig. 7. Based on Table 4 and Fig. 7, we can infer that the classification performance of the mRMR method with three classifiers is the best. Therefore, mRMR is used to select relevant top genes before employing the NIOA-based feature selection approaches.

Also from Table 4, the desired number of genes to be selected was considered 50 for the SRBCT and Prostate Tumor datasets, and 100 for the remaining datasets (Leukemia, Brain Tumor_1, and 11_Tumors).

5.2.1 Evaluation of the proposed BChOA without crossover

In this section, the performance of the suggested BChOAs based on the S-shaped Transfer function (BChOA-ST) and V-shaped Transfer function (BChOA-VT) are examined on microarray datasets using two classifiers (NB and KNN). A comparative study was carried out to compare the accuracy of the developed BChOA models. Tables 5 and 6 illustrate the results of these comparisons. Tables 5 and 6 show the performance of the proposed approaches (BChOA-ST and BChOA-VT) in terms of the three objectives (average number of genes, fitness value, and classification accuracy), as well as the computing time for the NB and KNN classifiers. In terms of the average number of selected features and CPU time, BChOA-VT outperformed BChOA-ST in most datasets. In addition, the KNN classifier yielded to a better outcome than NB. The classification accuracies and fitness values presented in the same tables produce almost identical results.

Figure 8 demonstrates the average computational results of classification accuracy, fitness function, CPU time, and the number of selected features with error bars for two suggested BChOA with V-shaped and S-shaped transformation functions utilizing NB and KNN classifiers.

Table 6 Comparison of two proposed BChOA with V-shaped and S-shaped TFs using **NB** classifier based on the average number of selected genes, fitness value, classification accuracy (%), and computational time (second)

Data set	Metrics	# Genes		Fitness value		Accuracy		Time	
		BChOA-ST	BChOA-VT	BChOA-ST	BChOA-VT	BChOA-ST	BChOA-VT	BChOA-ST	BChOA-VT
SRBCT	*AVG*	7.4	**7**	80.1672	**80.168**	100	100	296.164	**287.212**
	best	7	**6**	80.172	80.172	100	100	253.59	**244.97**
	worst	9	9	80.16	80.16	100	100	347.88	**333.6**
	STDEV	**0.8944272**	1.224745	**0.00438178**	0.0048989	0.00	0.00	**36.46541**	37.37997
Prostate Tumor	*AVG*	5.4	**5.2**	**78.22531**	77.99338	96.47059	96.47059	249.504	**199.39**
	best	**3**	4	**78.71745**	77.99818	**98.03922**	97.05882	195.6	**169.85**
	worst	7	7	77.19818	**77.98618**	95.09804	**96.07843**	346.1	**229.14**
	STDEV	1.81659	**1.30384**	0.678952	**0.0052153**	1.11782	**0.53698**	61.82561	**23.80237**
Leukemia	*AVG*	4.2	**3.8**	80.1896	**80.1904**	100	100	**150.28**	165.884
	best	3	3	80.192	80.194	100	100	**90.1**	112.41
	worst	5	5	80.188	80.188	100	100	**223.8**	298.03
	STDEV	1.095445	1.095445	0.00219089	0.00219089	0.00	0.00	**55.4955**	76.04223
Brain Tumor_1	*AVG*	**8.833333**	9	**77.48404**	77.32074	94.25925	**94.6296**	480.7033	**469.4217**
	best	**6**	7	**79.378**	78.578	96.66667	96.66667	247.63	**242.68**
	worst	10	10	**76.09111**	76.08911	92.2222	93.33333	618.38	**604.92**
	STDEV	1.602082	**1.264911**	1.230991	**0.8760792**	1.780098	**1.298942**	136.5332	**136.0959**
11_Tumors	*AVG*	25	**23.66667**	**75.66339**	75.25169	**92.52874**	92.24138	1000.83	**530.695**
	best	21	**20**	**76.14106**	75.83515	**93.67816**	93.10345	522.4	**451.56**
	worst	28	28	**75.12101**	74.70263	91.37931	91.37931	1383.68	**611.09**
	STDEV	**3.082207**	3.669696	0.3937991	0.4433911	0.908701	**0.6027657**	387.3495	**59.93123**

Bold values indicate best performance

5.2.2 Evaluation of the proposed BChOA with crossover

In this section, we assess the performance of BChOA-VT combined with crossover (BChOA-VT-C) and compare its performance to the basic BChOA-VT that has no crossover operator.

Table 7 shows the experimental outcomes for BChOA and BChOA-C in terms of average classification accuracy, fitness value, and the average number of selected genes. Both techniques were tested in ten separate runs. The Wilcoxon signed-rank statistical test between BChOA-C and BChOA is also shown in Table 7. Wilcoxon signed-rank statistical test was performed to reveal a substantial statistical difference between the two approaches. The best performances are highlighted in bold font.

In Table 7, the T − sig row, with a probability range of $\alpha \leq 0.05$, ‘ $*$ ‘ connotes that the BChOA-C technique produces substantially better results than the BChOA, whereas ‘‘ − ‘ connotes that the BChOA-C method produces results that are not significantly better than the

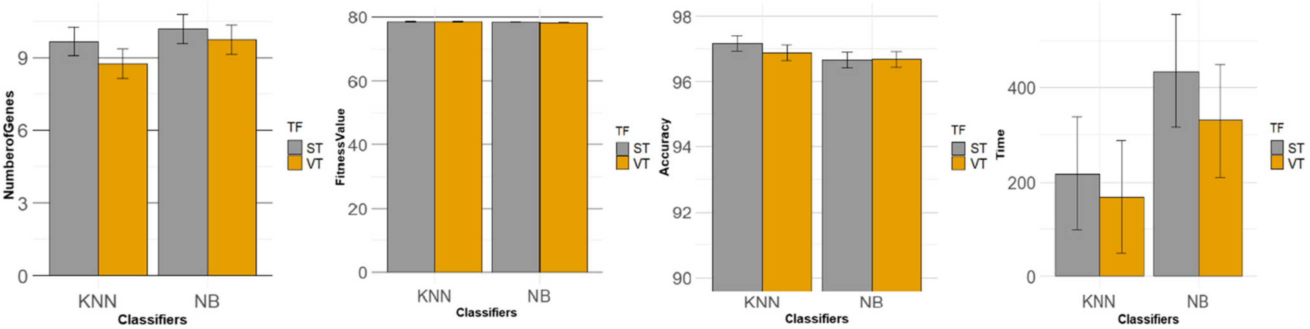


Fig. 8 Average Classification performance of two proposed BChOA with V-shape and S-shape TFs using NB and KNN classifiers on all datasets

Table 7 Comparison between BChOA and BChOA-C with V-shape (VT) transformation function

Algorithms	Metrics	Datasets				
		SRBCT	Prostate Tumor	Leukemia	Brain Tumor_1	11_Tumors
BChOA	l# Genesl	6.2	6	3.4	**10.33**	**17.8**
	Accuracy	100	95.79832	100	95.7777	93.75
	Fitness value	80.171	78.34103	80.191	78.4918	75.68
BChOA-C	l# Genesl	**4.4**	**5.75**	**3.1**	13.22	22.6
	Accuracy	**100**	**97.52**	**100**	**95.85**	**95.14**
	Fitness value	**80.179**	**78.686**	**80.1928**	**78.62**	**76.90**
	T—sing	–	*	–	–	*

Bold values represent the best results

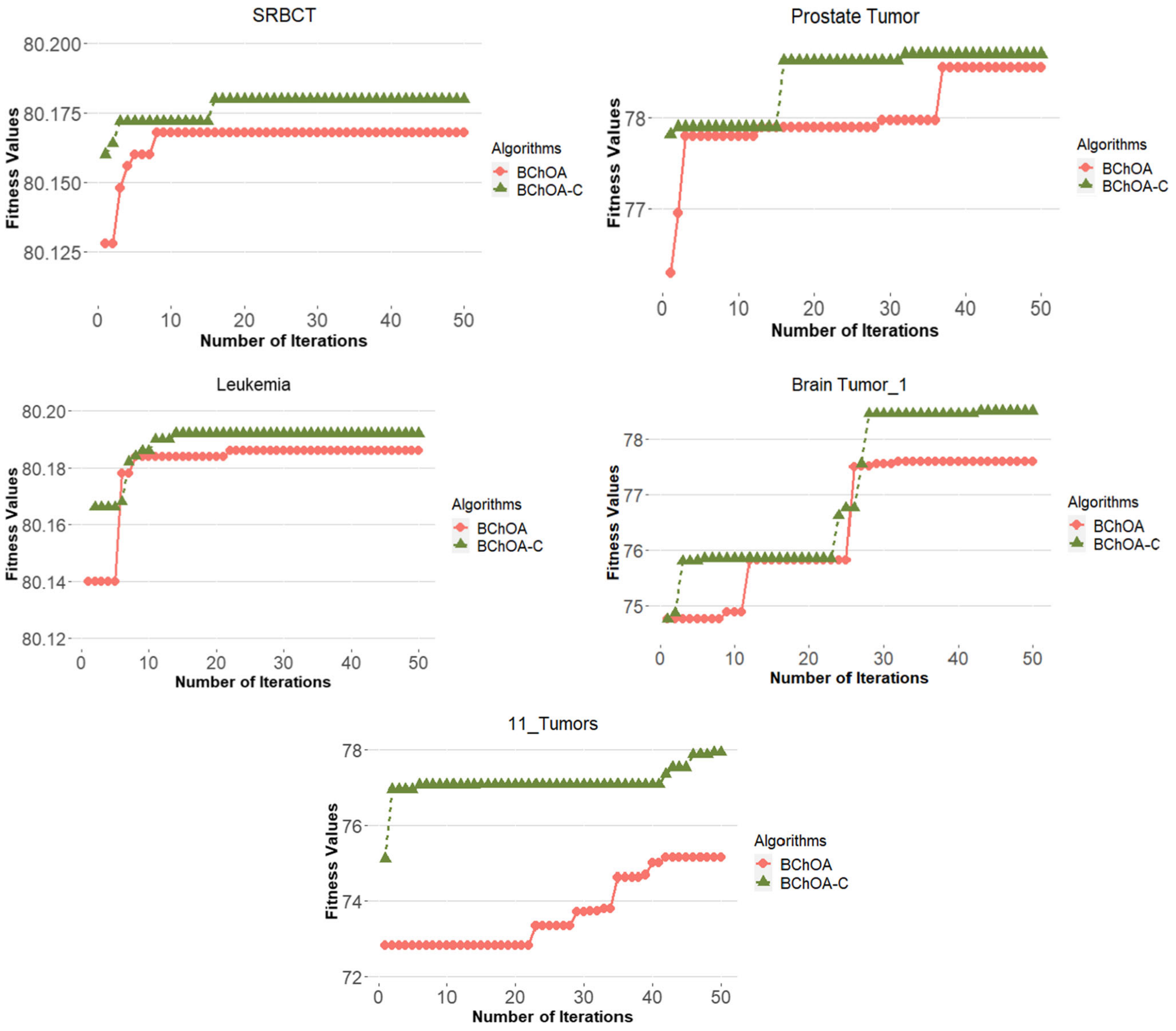


Fig. 9 The convergence behavior of BChOA and BChOA-C for five microarray datasets

BChOA. To conduct statistical calculations, Wilcoxon signed-rank statistical test uses just the accuracy metric. As shown in Table 7, BChOA-C outperformed BChOA on three out of five datasets in terms of classification accuracy. Both methods achieved the highest classification accuracy in the remaining two datasets (SRBCT and Leukemia) (100%). On all datasets, BChOA-C yields higher fitness values. In Brain Tumor_1 and 11_Tumors datasets, BChOA showed slightly better results than BChOA-C in terms of the average number of selected genes. On two datasets (i.e., Prostate Tumor and 11_Tumors), substantial differences in favor of BChOA-C may be deduced. The convergence behavior of both methods on all datasets is shown in Fig. 9. In terms of fitness value, the convergence behavior trend of BChOA-C is significantly better than BChOA on all datasets.

5.2.3 Comparison with other NIOAs

Table 8 shows that both proposed methods (BChOA and BChOA-C with VT transform function) outperform other NIOAs in terms of accuracy and the number of selected genes. Figures 10 and 11 illustrate the average number of selected genes and the accuracy of the ten approaches, respectively.

In this study, we utilized the paired t-test for statistical evaluation of BChOA-C performance. We compared our algorithm's accuracy and the number of selected genes with the other nine methods (Table 9). From Table 9 it can be seen that the p-values produced by the paired t-test among BChOA-C and other algorithms are mostly below the usual significance level of 0.05 for the number of selected genes metric. In other words, the suggested technique outperforms current methods in terms of the number of selected genes and the results are statistically significant. The proposed BChOA-C marginally outperforms the current NIOAs in terms of accuracy, but there is no statistically significant difference between the result of the proposed technique and other metaheuristic algorithms (MHAs). As a consequence, we may infer that the proposed approach has a significant difference in performance and indeed performs better than most of the compared approaches in the term of the number of selected genes.

5.2.4 Comparison with other state-of-the-art approaches

In this part, the suggested method's results are compared against state-of-the-art gene selection approaches in the literature to further examine its performance. The average classification accuracy and the average of selected genes, which appear between parentheses, are utilized as performance metrics in the assessment (Table 10).

Table 8 Average LOOCV classification accuracy ± STD (in %) and the number of genes over 20 runs of different feature selection methods using KNN ($K = 5$) classifier

Data set	Metrics	CFS	FCBF	Flower Pollution	FireFly	ACO	Bat	PSO	NSGAII	BChOA (VT)	BChOA-C (VT)
SRBCT	*ACC*	100	98.79	100 ± 0.0	100 ± 0.0	100 ± 0.0	100 ± 0.0	100 ± 0.0	100 ± 0.0	100 ± 0.0	100 ± 0.0
	#Genes	36	24	7.4 ± 0.89	20 ± 3.391	8.2 ± 2.16	15.6 ± 1.81	6.5 ± 1.32	6.33 ± 1.50	6.2 ± 1.30	4.4 ± 0.54
Prostate Tumor	*ACC*	93.13	93.13	97.703 ± 0.50	97.04 ± 0.88	97.33 ± 0.47	97.24 ± 0.43	97.34 ± 0.42	97.33 ± 0.47	97.49 ± 1.04	97.52 ± 0.55
	#Genes	25	23	7.33 ± 1.861	16 ± 3.34	8.833 ± 1.581	13.4 ± 3.847	7.4 ± 1.94	7.5 ± 1.76	6.8 ± 1.63	5.75 ± 1.91
Leukemia	*ACC*	97.22	95.83	99.72 ± 0.62	99.16 ± 0.76	99.72 ± 0.62	99.44 ± 0.76	97.2 ± 0.621	99.44 ± 1.24	100 ± 0.0	100 ± 0.0
	#Genes	34	27	7.166 ± 1.72	24.2 ± 6.41	15.2 ± 3.96	24 ± 9.082	14 ± 2	4.2 ± 1.09	3.4 ± 1.14	3.1 ± 0.54
Brain Tumor_1	*ACC*	87.77	86.66	93.74 ± 2.96	93.51 ± 1.63	91.267 ± 1.75	92.378 ± 1.18	93.607 ± 1.54	92.08 ± 1.24	95.77 ± 0.92	95.85 ± 0.499
	#Genes	42	25	15.66 ± 3.44	23.16 ± 5.03	20 ± 4.08	26 ± 4.76	18.75 ± 3.615	6 ± 1.30	10.33 ± 3.77	13.22 ± 3.734
11_Tumors	*ACC*	87.35	86.78	92.40 ± 0.48	93.19 ± 1.33	93 ± 0.92	93.673 ± 1.54	93.67 ± 1.32	92.887 ± 0.87	93.75 ± 1.87	95.14 ± 1.45
	#Genes	57	50	41 ± 6.70	50.33 ± 4.88	45.8 ± 5.06	48 ± 2.28	43.33 ± 4.92	41.16 ± 8.75	17.8 ± 1.463	22.6 ± 3.3

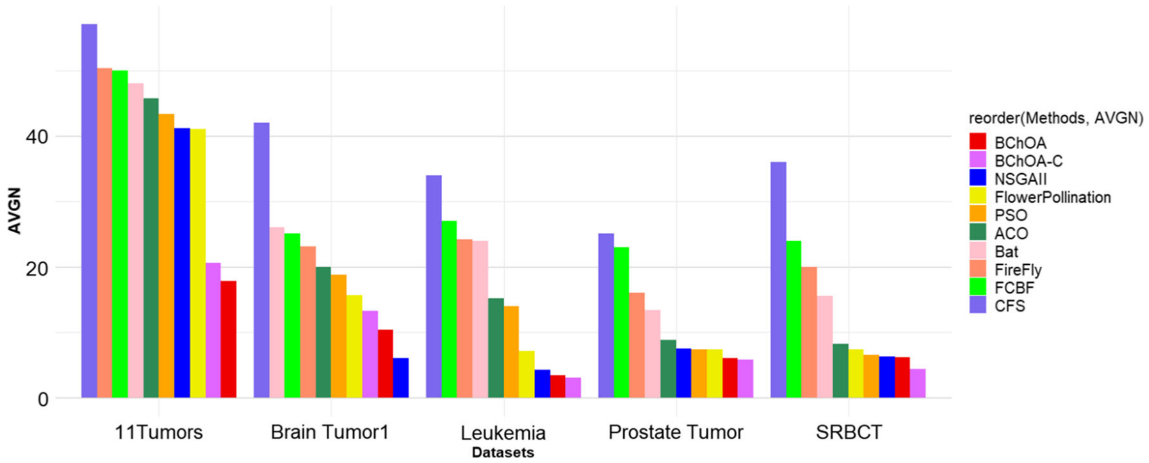


Fig. 10 The average number of genes of all datasets

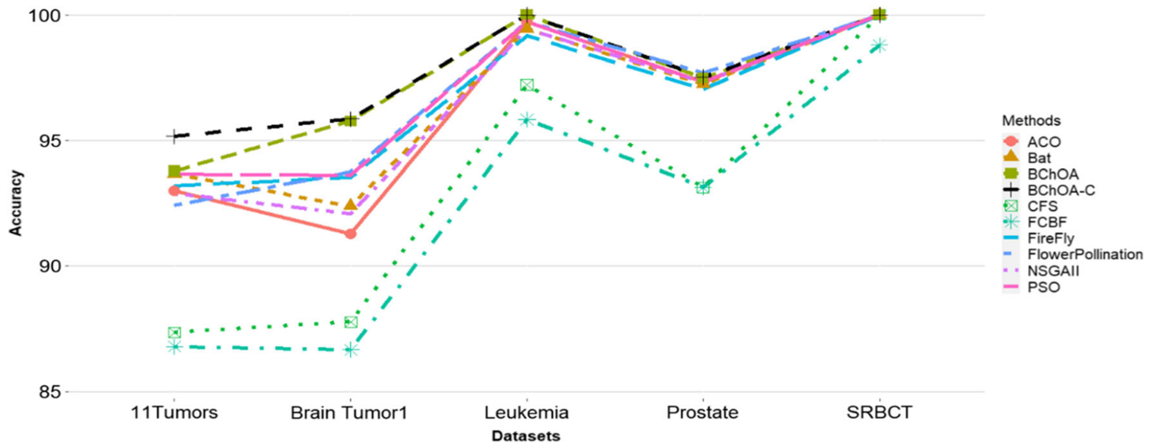


Fig. 11 The average accuracy comparison

Table 9 The *p*-Values of paired t-test of BChOA-C with other algorithms in the terms of accuracy and the number of selected genes

Paired t-Test		BChOA-C								
		vs								
		CFS	FCBF	Flower pollination	FireFly	ACO	Bat	PSO	NSGAII	BChOA
P-Value	Accuracy	0.03949	0.02053	0.06474	0.06474	0.1757	0.1402	0.07235	0.1342	0.3331
	#Genes	0.00048	0.002402	0.009547	0.009539	0.06663	0.01117	0.09287	0.4719	0.5129

On most datasets, the BChOA-C provided equal or higher classification performance, as shown in Table 10. Meanwhile, the suggested BChOA-C was able to reduce the number of genes in each dataset while obtaining high classification accuracy. On SRBCT and 11_Tumors datasets, BChOA-C had the best results in terms of classification accuracy and the number of selected genes. On two datasets (Prostate cancer and Brain Tumor 1), BChOA-C chose the fewest number of genes. Furthermore, for Brain Tumor 1, TLBOSA-SVM achieved the best accuracy, whereas BCO-KNN picked the fewest genes for the Leukemia dataset.

Table 10 Comparing the performance of the proposed approach with the literature methods

Algorithm	SRBCT	Prostate Tumor	Leukemia	Brain Tumor_1	11_Tumors
Proposed (BChOA-C-KNN)	**100 (4.4)**	**97.52 (5.75)**	**100(3.1)**	95.85(13.22)	**95.14** (22.6)
Proposed (BChOA-KNN)	100 (6.20)	97.49 (6)	100(3.4)	95.77(**10.33**)	93.75(17.8)
IG-MBKH	100(6.30)	–	100.00(4.20)	–	–
rMRMR-MGWO	100(37.5)	–	100 (5.06)	–	–
SFS-MB	97.50(35)	97.42(29)	96.19(23)	90.37(25)	72.31(39)
TLBOSA-SVM	99.91(11)	99.13(10.8)	95.35(12)	**96.98**(12)	92.23(**13**)
IDGA-F-SVM	100(18)	96.3(14)	100(15)	–	–
VLPSO-LS-KNN	99.7 (71.4)	92.58 (56.4)	–	75.54(102.1)	82.81(367.4)
BCO-KNN	100(7.4)	**100**(7)	100(**3**)	96.30 (15.5)	89.62(24.1)
PS-NSGA-KNN	96.35(18.6)	89.44(65)	–	73.81(57.8)	83.94(338.3)

Bold values represent the best results

Table 11 The best subset of selected genes from the gene selection method BChOA-C for binary datasets

	Index of Genes	ProbeID	Gene.Description	Specification
SRBCT	2144	308,231	Homo sapiens incomplete cDNA for a mutated allele of a myosin class I, myh-1c (myosin IB)	Myo1b has a potential role in the progression of several cancers, including prostate cancer, head, and neck squamous cell carcinoma (HNSCC), and cervical cancer (CC)
	509	207,274	Human DNA for insulin-like growth factor II (IGF-2); exon 7 and additional ORF	IGF2 is a protein hormone that affects cell proliferation, growth, migration, differentiation, and survival. IGF2 is linked to an increased risk of developing malignancies such as colorectal, breast, prostate, and lung
	545	1,435,862	antigen identified by monoclonal antibodies 12E7, F21, and O13 (CD99)	CD99 is a 32-kD T-cell surface glycoprotein involved in spontaneous rosette formation with erythrocytes.CD99 can influence tumor cell migration, invasion, and metastasis
	742	812,105	transmembrane protein (TMEM)	A transmembrane protein (TMEM) is a type of protein that spans biological membranes. TMEMs have been linked to tumor growth and invasion, as well as chemoresistance
Leukemia	4211	X51521_at	VIL2 Villin 2 (ezrin)	Cytovillin is a microvillar cytoplasmic peripheral membrane protein that is expressed strongly in certain human tumors
	4377	X62654_rna1_at	ME491 gene extracted from H. sapiens gene for Me491/CD63 antigen	The melanoma-associated antigen ME491 is expressed strongly during the early stages of progression of the tumor
	6855	M31523_at	transcription factor 3(E2A immunoglobulin enhancer-binding factors E12/E47), TCF3	Dysregulation of E2A leads to leukemia and tumorigenesis of some solid tumors

In summary for gene selection problems, BChOA-C looks to be competitive and in some cases superior against state-of-art methods.

5.2.5 The biological meaning of the selected genes by the proposed BChOA-C approach

The biological meanings of the best subset of selected genes derived from our suggested approach are presented in this section. The names, prob-IDs, and descriptions of obtained genes as well as their specifications for SRBCT and Leukemia datasets are listed in Table 11. The interpretation of acquired genes from Prostate cancer, Brain Tumor_1, 11_ Tumors datasets is not feasible since no names have been assigned to genes in the datasets.

The biological meanings of the selected genes were obtained using the OMIM (https://omim.org/) and NCBI (https://pubmed.ncbi.nlm.nih.gov/) websites. Table 11 shows that our suggested method can successfully identify cancer-related genes for each dataset.

Fig. 13 Box plot diagrams of gene expression for the best subset of selected genes ▶

For biological interpretation of gene expression data, the heatmap combined with the clustering method was used. The heatmap was created in R using the “gplots” package. In heatmaps (Fig. 12), 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. Figure 10 depicts the profile of samples for different datasets. It clusters together genes with common expression profiles and confirms that the expression of most of the genes is coordinately down-regulated.

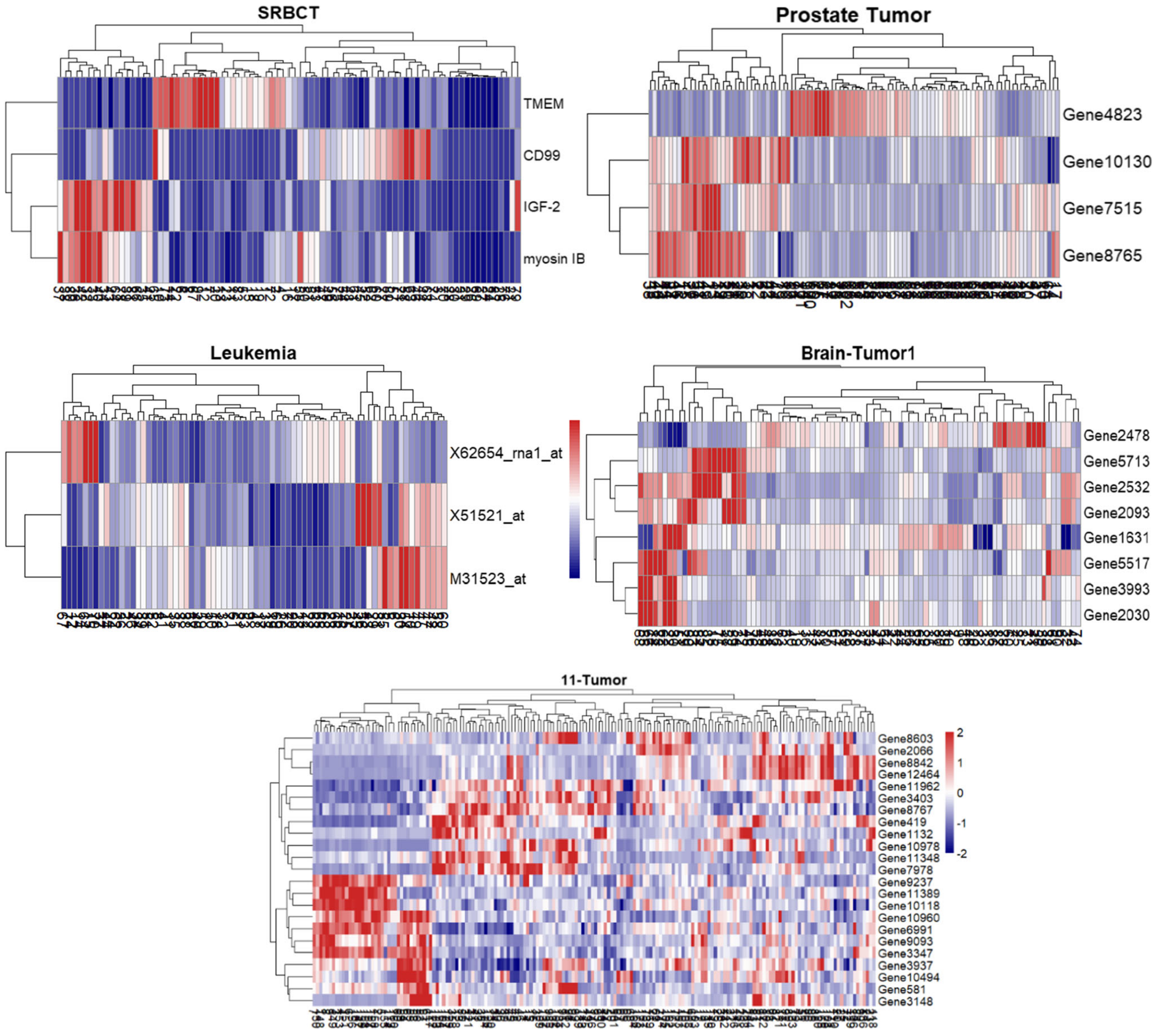
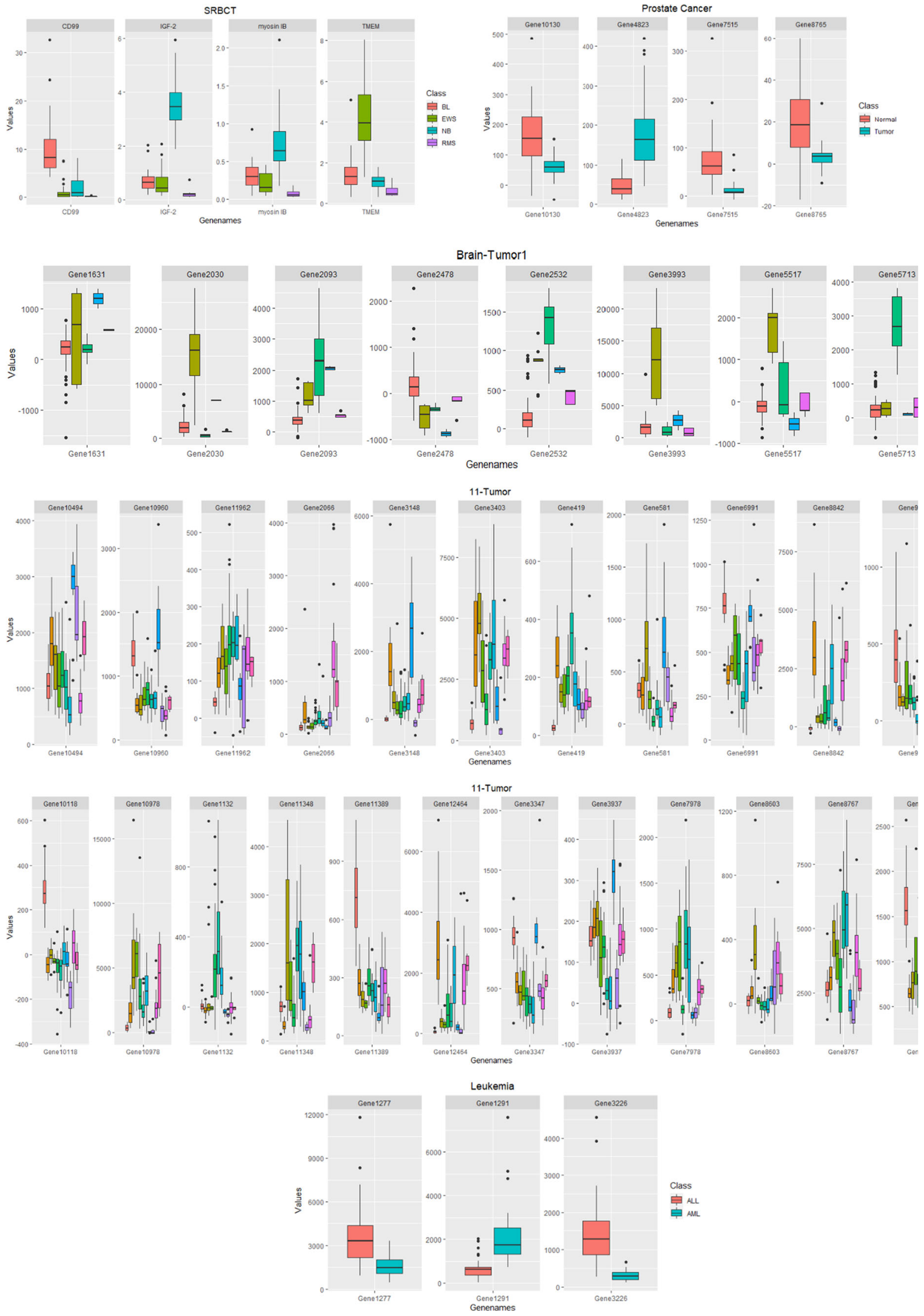


Fig. 12 The gene expression level of the best subset of selected genes shown as a heatmap


SRBCT
CD99
IGF-2
myosin IB
TMEM
Values
Genenames
Class
BL
EWS
NB
RMS
Prostate Cancer
Gene10130
Gene4823
Gene7515
Gene8765
Values
Genenames
Class
Normal
Tumor
Brain-Tumor1
Gene1631
Gene2030
Gene2093
Gene2478
Gene2532
Gene3993
Gene5517
Gene5713
Values
Genenames
11-Tumor
Gene10494
Gene10960
Gene11962
Gene2066
Gene3148
Gene3403
Gene419
Gene581
Gene6991
Gene8842
Gene9
Values
Genenames
11-Tumor
Gene10118
Gene10978
Gene1132
Gene11348
Gene11389
Gene12464
Gene3347
Gene3937
Gene7978
Gene8603
Gene8767
Gen
Values
Genenames
Leukemia
Gene1277
Gene1291
Gene3226
Values
Genenames
Class
ALL
AML

Boxplot was used to assess the validity of selected genes. In Fig. 13, the expression of the best subset of genes was shown as a box plot. Figure 13 shows that the selected genes are able to separate cancer groups by differences in their gene expressions.

5.3 Results for small-sized datasets

We show and evaluate the results produced by our suggested methods on tiny datasets in this section. Table 12 compares the performance of both proposed methods with three classical (CFS, FCBF, and SSF) and three NIOA-based (Cuckoo, PSO, and NSGAII) feature selection approaches on small datasets. The best results among all feature selection methods have been highlighted and marked in bold type. It is worth mention that for the Yale dataset initially, we used the IG filter-based approach to select the top 100 relevant genes. The kappa2 measurement with the NB classifier from the "Irr" package in R was employed as a fitness function in these datasets.

As shown in Table 12, our method obtained fewer genes than CFS, FCBF, SSF, Cuckoo, PSO, and NSGAII for the majority of datasets. On the other hand, for most datasets, our approach obtains somewhat higher or second higher classification accuracy than other approaches. From this comparison, we conclude that BChOA-ST-NB is a suitable algorithm for FS problems on the different types of datasets.

Table 12 Average LOOCV classification accuracy ± STD (in %) and the number of features over 20 runs of different feature selection methods using NB classifier

Data set	Metrics	CFS	FCBF	SSF	Cuckoo	PSO	NSGAII	BChOA(ST)	BChOA-C (ST)
Chess	*ACC*	91.99	91.99	94.27	94.55 ± 0.35	94.63 ± 0.33	94.27 ± 0.13	94.66 ± 1.01	**94.75** ± 0.3
	# of features	7	7	6	11 ± 4.54	10.66 ± 1.52	5.5 ± 1.73	**5.285** ± 1.38	10 ± 2.38
dbworld	*ACC*	92.18	90.62	90.62	93.83 ± 0.90	93.79 ± 0.80	92.49 ± 1.71	93.77 ± 1.16	**94.55** ± 0.8
	# of features	12	10	**5**	61.5 ± 6.02	54 ± 10.56	7 ± 1.58	11.8 ± 3.96	50 ± 7.24
lymphography	*ACC*	81.75	82.43	86.38	85.53 ± 1.02	86.21 ± 0.36	**87.38** ± 1.03	86.541 ± 1.23	86.88 ± 1.25
	# of features	11	**6**	10	11.4 ± 2.40	10.2 ± 1.923	9.66 ± 0.57	**6** ± 1.22	8.25 ± 1.25
Lung cancer	*ACC*	68.75	62.5	**87.5**	82.29 ± 1.80	86.71 ± 1.56	84.89 ± 2.61	84.99 ± 2.47	85.12 ± 1.7
	# of features	9	**6**	**6**	10.33 ± 3.78	14.8 ± 3.42	8.4 ± 2.30	**6** ± 1.58	10 ± 3.1
Yale	*ACC*	60.60	57.57	64.24	69.08 ± 1.78	69.87 ± 1.048	69.05 ± 1.26	69.88 ± 1.15	**70.10 ± 1.3**
	# of features	33	**9**	11	32.5 ± 6.75	34.25 ± 4.78	11.66 ± 1.52	11 ± 9.5	18 ± 3.6

Bold values represent the best results

6 Conclusion

Feature subset selection plays an important role in classification tasks, as it enhances the general abilities of classifiers, simplifies the learning model, and reduces the computational cost. In this paper, the problem of feature selection in high-dimensional biomedicine data classification has been considered and solved through a novel binary ChOA which extends ChOA from the continuous version to the discrete domain. To the best of our knowledge, this is the first binary variant of ChOA which has been developed for the task of feature selection. An enhanced binary ChOA-based optimizer with a crossover scheme was also presented. Two different transfer functions, S-shaped and V-shaped, were utilized to convert the continuous form of ChOA to binary form. Moreover, the widely used KNN and NB classifiers were served as evaluators of feature subsets in the proposed wrapper BChOA approach.

To verify the effectiveness of the proposed BChOA-based approaches five well-known biomedical datasets and several datasets with different domains were used. The results were compared to two standard filter feature selection methods and six popular wrapper techniques:

PSO, BA, ACO, GA, FA, and FP. The experimental results show that the wrapper BChOA-based FS approaches are able to select a small number of the most prominent features whilst achieve a higher classification accuracy. The proposed BChOAs also outperform other current state-of-the-art techniques in the literature in terms of classification accuracy using fewer features. Moreover, the performance of BChOA-C is better than BChOA in the terms of classification accuracy due to the enhancement of the algorithm's exploration capability. In conclusion, both suggested BChOA and BChOA-C can be used as ideal feature selection tools for high-dimensional biomedical data, allowing for better biological data mining in fields of disease diagnosis.

As future work, BChOA may be proposed as a filter feature selection approach and be examined on the classification of biomedical data using a variety of classifiers. Development of binary multi-objective ChOA for feature selection and its performance comparison with the continuous multi-objective ChOA is recommended as well.

Author contributions Elnaz Pashaei and Elham Pashaei 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.

Ethical approval This article does not contain any data, or other information from studies or experimentation, with the involvement of human or animal subjects.

References

1. 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
2. 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
3. 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
4. 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 annual international conference of the IEEE engineering in medicine and biology society, EMBS. Institute of Electrical and Electronics Engineers Inc., pp 3076–3079
5. Too J, Mirjalili S (2021) A hyper learning binary dragonfly algorithm for feature selection: a COVID-19 case study. Knowl Based Syst 212:106553. https://doi.org/10.1016/j.knosys.2020.106553
6. 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
7. 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. https://doi.org/10.1155/2019/7828590
8. 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
9. Alshamlan HM (2018) Co-ABC: correlation artificial bee colony algorithm for biomarker gene discovery using gene expression profile. Saudi J Biol Sci 25:895–903. https://doi.org/10.1016/j.sjbs.2017.12.012
10. Alshamlan H, Badr G, Alohali Y (2015) mRMR-ABC: a hybrid gene selection algorithm for cancer classification using microarray gene expression profiling. Biomed Res Int 2015:604910. https://doi.org/10.1155/2015/604910
11. 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
12. 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
13. Shukla AK (2019) Multi-population adaptive genetic algorithm for selection of microarray biomarkers. Neural Comput Appl. https://doi.org/10.1007/s00521-019-04671-2
14. 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
15. 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
16. 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
17. Venkatasalam K, Rajendran P, Thangavel M (2019) Improving the accuracy of feature selection in big data mining using accelerated flower pollination (AFP) algorithm. J Med Syst 43:1–11. https://doi.org/10.1007/s10916-019-1200-1
18. Tumuluru P, Ravi B (2018) Chronological grasshopper optimization algorithm-based gene selection and cancer classification. J Adv Res Dyn Control Syst 10:80–94
19. Srivastava A, Chakrabarti S, Das S, et al (2013) Hybrid firefly based simultaneous gene selection and cancer classification using support vector machines and random forests. In: Advances in intelligent systems and computing. Springer Verlag, pp 485–494
20. 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
21. 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
22. Khishe M, Mosavi MR (2020) Chimp optimization algorithm. Exp Syst Appl 149:113338. https://doi.org/10.1016/j.eswa.2020.113338

23. Khishe M, Mosavi MR (2020) Classification of underwater acoustical dataset using neural network trained by Chimp Optimization Algorithm. Appl Acoust 157:107005. https://doi.org/10.1016/j.apacoust.2019.107005
24. Kaur M, Kaur R, Singh N, Dhiman G (2021) SChoA: an newly fusion of sine and cosine with chimp optimization algorithm for HLS of datapaths in digital filters and engineering applications. Eng Comput. https://doi.org/10.1007/s00366-020-01233-2
25. Jia H, Sun K, Zhang W, Leng X (2021) An enhanced chimp optimization algorithm for continuous optimization domains. Complex Intell Syst 1:3. https://doi.org/10.1007/s40747-021-00346-5
26. 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
27. Ahmed MS, Shahjaman M, Rana MM, Mollah MNH (2017) Robustification of Naïve Bayes classifier and its application for microarray gene expression data analysis. Biomed Res Int. https://doi.org/10.1155/2017/3020627
28. Maleki N, Zeinali Y, Niaki STA (2021) A k-NN method for lung cancer prognosis with the use of a genetic algorithm for feature selection. Exp Syst Appl 164:113981. https://doi.org/10.1016/J.ESWA.2020.113981
29. Tumar I, Hassouneh Y, Turabieh H, Thaher T (2020) Enhanced binary moth flame optimization as a feature selection algorithm to predict software fault prediction. IEEE Access 8:8041–8055. https://doi.org/10.1109/ACCESS.2020.2964321
30. Agrawal P, Abutarboush HF, Ganesh T, Mohamed AW (2021) Metaheuristic algorithms on feature selection: a survey of one decade of research (2009–2019). IEEE Access 9:26766–26791. https://doi.org/10.1109/ACCESS.2021.3056407
31. Shukla AK, Tripathi D, Reddy BR, Chandramohan D (2020) A study on metaheuristics approaches for gene selection in microarray data: algorithms, applications and open challenges. Evol Intell 13:309–329
32. 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
33. 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
34. Djellali H, Guessoum S, Ghoualmi-Zine N, Layachi S (2017) Fast correlation based filter combined with genetic algorithm and particle swarm on feature selection. In: 2017 5th International conference on electrical engineering - boumerdes, ICEE-B 2017. Institute of Electrical and Electronics Engineers Inc., pp 1–6
35. Sun L, Zhang XY, Qian YH et al (2019) Joint neighborhood entropy-based gene selection method with fisher score for tumor classification. Appl Intell 49:1245–1259. https://doi.org/10.1007/s10489-018-1320-1
36. Sun L, Kong X, Xu J et al (2019) A hybrid gene selection method based on relieff and ant colony optimization algorithm for tumor classification. Sci Rep 9:1–14. https://doi.org/10.1038/s41598-019-45223-x
37. Zhang H, Li L, Luo C et al (2014) Informative gene selection and direct classification of tumor based on chi-square test of pairwise gene interactions. Biomed Res Int. https://doi.org/10.1155/2014/589290
38. Pashaei E, Pashaei E (2019) Gene selection using intelligent dynamic genetic algorithm and random forest. In: 2019 11th international conference on electrical and electronics engineering (ELECO). pp 470–474
39. Covões TF, Hruschka ER (2011) Towards improving cluster-based feature selection with a simplified silhouette filter. Inf Sci (Ny) 181:3766–3782. https://doi.org/10.1016/J.INS.2011.04.050
40. 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
41. Meyer PE, Schretter C, Bontempi G (2008) Information-theoretic feature selection in microarray data using variable complementarity. IEEE J Sel Top Signal Process 2:261–274. https://doi.org/10.1109/JSTSP.2008.923858
42. Alomari OA, Khader AT, Al-Betar MA, Abualigah LM (2017) MRMR BA: a hybrid gene selection algorithm for cancer classification. J Theor Appl Inf Technol 95:2610–2618
43. 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
44. Devi Arockia Vanitha C, Devaraj D, Venkatesulu M (2014) Gene expression data classification using Support Vector Machine and mutual information-based gene selection. In: Procedia computer science. Elsevier B.V., pp 13–21
45. Chen KH, Wang KJM, Tsai ML et al (2014) Gene selection for cancer identification: a decision tree model empowered by particle swarm optimization algorithm. BMC Bioinform 15:1–10. https://doi.org/10.1186/1471-2105-15-49
46. Zeebaree DQ, Haron H, Abdulazeez AM (2018) Gene selection and classification of microarray data using convolutional neural network. In: ICOASE 2018 - International conference on advanced science and engineering. institute of electrical and electronics engineers Inc., pp 145–150
47. 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
48. 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
49. Lin HY (2016) Gene discretization based on EM clustering and adaptive sequential forward gene selection for molecular classification. Appl Soft Comput J 48:683–690. https://doi.org/10.1016/j.asoc.2016.07.015
50. Haq AU, Li J, Memon MH, et al (2019) Heart disease prediction system using model of machine learning and sequential backward selection algorithm for features selection. In: 2019 IEEE 5th international conference for convergence in technology, I2CT 2019. Institute of Electrical and Electronics Engineers Inc.
51. Beheshti Z (2021) UTF: upgrade transfer function for binary meta-heuristic algorithms. Appl Soft Comput 106:107346. https://doi.org/10.1016/j.asoc.2021.107346
52. 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
53. Hammouri AI, Mafarja M, Al-Betar MA et al (2020) An improved Dragonfly Algorithm for feature selection. Knowl Based Syst 203:106131. https://doi.org/10.1016/j.knosys.2020.106131
54. Mirjalili S, Zhang H, Mirjalili S et al (2020) A novel U-shaped transfer function for binary particle swarm optimisation. In: Advances in intelligent systems and computing. Springer, pp 241–259
55. Beheshti Z (2021) A novel x-shaped binary particle swarm optimization. Soft Comput 25:3013–3042. https://doi.org/10.1007/s00500-020-05360-2
56. 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

57. 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
58. 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)
59. 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
60. Baliarsingh SK, Muhammad K, Bakshi S (2021) SARA: a memetic algorithm for high-dimensional biomedical data. Appl Soft Comput 101:107009. https://doi.org/10.1016/j.asoc.2020.107009
61. Nagpal S, Arora S, Dey S, Shreya S (2017) Feature selection using gravitational search algorithm for biomedical data. In: Procedia Computer Science. Elsevier B.V., pp 258–265
62. Tuba E, Strumberger I, Bezdan T et al (2019) Classification and feature selection method for medical datasets by brain storm optimization algorithm and support vector machine. In: Procedia Computer Science. Elsevier B.V., pp 307–315
63. 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
64. Sharifai GA, Zainol Z (2020) Feature selection for high-dimensional and imbalanced biomedical data based on robust correlation based redundancy and binary grasshopper optimization algorithm. Genes (Basel) 11:1–26. https://doi.org/10.3390/genes11070717
65. Coleto-Alcudia V, Vega-Rodríguez MA (2020) Artificial Bee Colony algorithm based on Dominance (ABCD) for a hybrid gene selection method. Knowl Based Syst 205:106323. https://doi.org/10.1016/j.knosys.2020.106323
66. 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
67. Alzaqebah M, Briki K, Alrefai N et al (2021) Memory based cuckoo search algorithm for feature selection of gene expression dataset. Informat Med Unlocked 24:100572. https://doi.org/10.1016/J.IMU.2021.100572
68. 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
69. Hall M, Frank E, Holmes G et al (2009) The WEKA data mining software. ACM SIGKDD Explor Newsl 11:10–18. https://doi.org/10.1145/1656274.1656278
70. Chouhan SS, Kaul A, Singh UP, Jain S (2018) Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: an automatic approach towards plant pathology. IEEE Access 6:8852–8863. https://doi.org/10.1109/ACCESS.2018.2800685
71. Qi C, Diao J, Qiu L (2019) On estimating model in feature selection with cross-validation. IEEE Access 7:33454–33463. https://doi.org/10.1109/ACCESS.2019.2892062