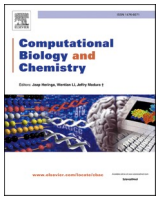
Computational Biology and Chemistry 101 (2022) 107767

Contents lists available at ScienceDirect

Computational Biology and Chemistry

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



Mutation-based Binary Aquila optimizer for gene selection in cancer classification

Elham Pashaei [1]

Department of Computer Engineering, Istanbul Gelisim University, Istanbul, Turkey

ARTICLE INFO

Keywords:
Cancer classification
Feature selection
Aquila optimizer
Optimization
Mutation

ABSTRACT

Microarray data classification is one of the hottest issues in the field of bioinformatics due to its efficiency in diagnosing patients' ailments. But the difficulty is that microarrays possess a huge number of genes where the majority of which are redundant or irrelevant resulting in the deterioration of classification accuracy. For this issue, mutated binary Aquila Optimizer (MBAO) with a time-varying mirrored S-shaped (TVMS) transfer function is proposed as a new wrapper gene (or feature) selection method to find the optimal subset of informative genes. The suggested hybrid method utilizes Minimum Redundancy Maximum Relevance (mRMR) as a filtering approach to choose top-ranked genes in the first stage and then uses MBAO-TVMS as an efficient wrapper approach to identify the most discriminative genes in the second stage. TVMS is adopted to transform the continuous version of Aquila Optimizer (AO) to binary one and a mutation mechanism is incorporated into binary AO to aid the algorithm to escape local optima and improve its global search capabilities. The suggested method was tested on eleven well-known benchmark microarray datasets and compared to other current state-of-the-art methods. Based on the obtained results, mRMR-MBAO confirms its superiority over the mRMR-BAO algorithm and the other comparative GS approaches on the majority of the medical datasets strategies in terms of classification accuracy and the number of selected genes. R codes of MBAO are available at https://github.com/el-pashaei/MBAO.

1. Introduction

Microarray technology is widely utilized for a variety of purposes, including the diagnosis of diseases such as cancer.

Using DNA microarray data, researchers can discover all the differences in gene expression between two different cell types, such as healthy and cancerous cells in a single experiment. In medical research, the use of microarray data for disease classification based on distinct patterns of gene expression is crucial. Microarray data classification plays an important role in real-world clinical practice, particularly in the diagnosis of heart disease, infectious disease, and cancer research (Nguyen et al., 2015). Microarray classification is a supervised learning task that uses an expression array phenotype to determine the diagnostic category of a tissue sample (Sánchez-Maroño et al., 2019). However, the classification of microarray data faces significant challenges due to the high dimensionality and high complexity of data. Microarray datasets contain tens of thousands of genes (features) and a low number of samples, often from less than hundreds of patients. Although the huge number of genes in microarray datasets may appear to be beneficial, many of these genes are noisy, redundant, or irrelevant, causing a deterioration of the classification quality. Furthermore, because most of the genes in microarray data are directly or indirectly connected with each other, microarray data can be perceived as very complex. To address these issues, gene selection (GS) approaches are used to identify the most informative genes before the classification process. GS techniques aid in reducing data dimensionality, simplifying the learning model, speeding up the learning process, improving classification accuracy, and increasing the interpretability of data (Alanni et al., 2019). The best GS approach can be defined as one that reduces the number of selected genes while increasing the classifier's accuracy.

Filter approaches, wrapper approaches, and hybrid approaches are the three commonly used types of GS techniques. Filter approaches remove irrelevant and redundant genes without employing a classifier. They utilize some principal characteristics of the training data to calculate a score for each gene and then choose the highest-ranking genes. Some of the widely used filter approaches are Information Gain (IG) (Zhang et al., 2020), Mutual Information Maximization (MIM) (Dabba et al., 2021), Fisher-Score (Dashtban and Balafar, 2017), ReliefF

E-mail addresses: epashaei@gelisim.edu.tr, elham.pashaei@gmail.com.
[1] ORCID: https://orcid.org/0000-0001-7401-4964.

https://doi.org/10.1016/j.compbiolchem.2022.107767
Received 24 March 2022; Received in revised form 10 July 2022; Accepted 29 August 2022
Available online 5 September 2022

(Shukla and Tripathi, 2019), Chi-Square (Kanti Ghosh et al., 2021), Symmetrical Uncertainty (SU) (Shreem et al., 2022), and Minimum Redundancy Maximum Relevance (mRMR) (Alomari et al., 2018). Wrapper approaches use a classifier as a fitness evaluator to measure the usefulness of genes or gene subsets. They build a lot of models with different subsets of input genes and then choose the ones that produce the best model based on the fitness value. Although the wrapper approaches may obtain better performances, they are not suitable for high-dimensional datasets since they are computationally expensive. In comparison, filter approaches are less computationally expensive but yield less classification accuracy. For this reason, hybrid methods have been suggested to combine the best properties of both filter and wrapper approaches.

Finding the best gene subsets has already been demonstrated as an NP-hard problem (Pashaei and Pashaei, 2021a; Alomari et al., 2021). Therefore, nature-inspired optimization algorithms (NIOAs) have been adopted in wrapper-based approaches to address feature selection (FS) problems. The majority of NIOAs begin with a random population initialization and continue with solution evaluation based on the fitness function at each iteration, solution updating, and finally identifying the optimal solution based on the termination criterion (Elgamal et al., 2020). Binary Krill Herd (BKH) algorithm (Zhang et al., 2020), Binary Black Hole Algorithm (BBHA) (Pashaei and Pashaei, 2021a, 2020; Pashaei et al., 2016a), Gray Wolf Optimizer (GWO) (Alomari et al., 2021), Moth Flame Optimization Algorithm (MFOA) (Dabba et al., 2021), improved Binary Clonal Flower Pollination Algorithm (IBCFPA) (Yan et al., 2019a), improved Shuffled Frog Leaping Algorithm (ISFLA) (Hu et al., 2018), Harmony Search Algorithm (HSA) (Dash, 2021), Harris hawks optimization (HHO) algorithm (Abdel-Basset et al., 2021), and Binary Coral Reefs Optimization with Simulated Annealing and tournament selection strategy (BCROSAT) (Yan et al., 2019b) are some examples of wrapper techniques that have been used to address GS problem.

Several hybrid filter/wrapper approaches have also been introduced to tackle the GS problem. Examples of these implementations are IG/ modified BKH algorithm (MKHA) (Zhang et al., 2020), Fisher-Score/ Intelligent Dynamic Genetic Algorithm (IDGA) (Dashtban and Balafar, 2017), Random Forest Ranking (RFR)/ IDGA (Pashaei and Pashaei, 2019), robust mRMR/ bat-inspired algorithm (BA)-β hill-climbing (Alomari et al., 2018), mRMR/ hybrid of Simulated Annealing and Rao Algorithm (SARA) (Baliarsingh et al., 2021), Correlation Feature Selection (CFS)/ Improved Particle Swarm Optimization (IPSO) (Jain et al., 2018), SU/ reference set HAS (RSHSA) (Shreem et al., 2022), RFR/ BBHA (Pashaei et al., 2016b), RFR/ hybrid of BBHA and PSO (Pashaei et al., 2019), MIM/ MFOA (Dabba et al., 2021), mRMR/ hybrid of BBHA and binary dragonfly optimization algorithm (DBH) (Pashaei and Pashaei, 2021a), "Technique for Order Preference by Similarity to Ideal Solution" (TOPSIS) filtering/ binary Jaya algorithm (Chaudhuri and Sahu, 2021), and Joint Mutual Information (JMI)/ bacterial algorithm (Wang et al., 2017).

Most of the above-mentioned approaches, however, suffer from stagnation in local optima that result from the intricate interplay between genes and the huge space search (Alomari et al., 2018, 2021). As a result, a robust search technique is still required to select the optimum gene subset in an acceptable amount of time in order to enhance classification accuracy. Aquila Optimizer (AO) is one of the efficient NIOAs that has recently been suggested by Abualigah et al (Abualigah et al., 2021). This algorithm is inspired by the behavior of Aquilas in nature while catching their prey. Compared with other recognized NIOAs, AO has unique characteristics, such as easy implementation, flexibility, and robustness to control parameters. As a result, the AO and its hybrid version (Wang et al., 2021) have been utilized to tackle several optimization problems since its creation in 2021, such as industrial engineering design problems (Wang et al., 2021), Adaptive Neuro-Fuzzy Inference System (ANFIS) parameter tuning for oil production forecasting (Alrassas et al., 2021), image enhancement (Rajinikanth et al., 2022), FS for COVID-19 image classification (Abd Elaziz et al., 2021), and FS for intrusion detection system (Fatani et al., 2021). However, to our knowledge, AO is not yet properly investigated for GS and cancer classification problems.

In this paper, a hybrid filter/wrapper GS method is proposed for cancer microarray data classification using the mRMR and Mutated Binary Aquila Optimizer (MBAO) algorithm. A time-varying mirrored S-shaped (TVMS) transfer function (Beheshti, 2020) is applied to convert continuous search space to a binary one. The TVMS transfer function can balance exploration and exploitation in BAO. To improve the efficiency of Binary AO (BAO) in dealing with complex high-dimensional microarrays data, a mutation mechanism is incorporated into the BAO. Mutation operation randomly modifies one or more elements of the local best solution and therefore can empower BAO's global search capabilities, and avoid the algorithm from getting stuck in the local optimum. The suggested method is called mRMR-MBAO and it uses Support Vector Machine (SVM) classifier (Pashaei and Aydin, 2018; Pashaei et al., 2016c) to evaluate candidate gene subsets. Before performing MBOA for microarray data classification, a subset of the most discriminative genes must be selected from thousands of genes. The mRMR filter strategy is used first in the proposed hybrid method to select M top-ranked genes. Thereafter, MBAO uses these genes as a powerful initial input to determine the final most informative subset of genes.

The main contributions of this paper are as follows:

- A novel wrapper approach based on an improved Aquila Optimizer is applied to GS for microarray data.
- The TVMS transfer function is introduced in continuous AO to design a new binary AO.
- A mutation genetic operator is combined with BAO to improve the search performance of the original BAO.
- Eleven well-known microarray datasets are used to evaluate whether the suggested method (mRMR-MBAO) can produce a gene subset with fewer genes and higher classification accuracy than current GS approaches.

For performance analysis and comparison evaluations, three metrics have been used: classification accuracy, number of selected genes, and fitness value. The performance of mRMR is compared against other well-regarded filtering approaches. The performance of BAO with and without mutation mechanism is also studied. For comparative evaluation, the proposed mRMR-MBAO method is compared with current state-of-art approaches. The conducted experiments demonstrate that mRMR-MBAO is able to obtain comparatively better results in terms of accuracy and number of selected genes than other previously proposed methods.

The remainder of this paper is organized as follows: The AO method and mRMR filtering method are briefly discussed in Section 2. In Section 3, the stages of the developed method are presented. The experimental setup, the obtained results, and their discussions are reported in Section 4. Finally, the conclusion and future works are given in Section 5.

2. Background

2.1. Aquila optimizer algorithm

AO (Abualigah et al., 2021) is a new stochastic population-based NIOA that mimics the hunting techniques of Aquila, the most popular raptor in the Northern Hemisphere. Aquila uses four different hunting strategies (S) in nature, depending on the type of prey. High soar with vertical stoop (S^1), contour flight with a short glide attack (S^2), low fly with gradual descent attack (S^3), and strolling and grabbing prey (S^4) are the strategists in question, which are modeled in the AO algorithm. Depending on the situation, Aquila skillfully and quickly switches between those hunting strategies. The AO algorithm contains two search

Algorithm 1 Aquila Optimizer

Initialize the AO's parameters
Initialize the Aquila population $X_i(i = 1,2,\ldots,N)$
while $(t < T)$
  evaluate each solution (Aquila) using fitness values
  Determine the best solution $X_b(t)$ (i.e. prey's location)
  for (each solution (X_i)) **do**
    calculate the $X_M(t)$, $QF(t)$, $Levy(D)$ and r
    if $(t \le \left(\frac{2}{3}\right) \times T)$ **then**
      if $(rand \le 0.5)$
        move Aquila to a new location, X_i^{S1}, using the S^1 tactic (Eq. (2))
        if the fitness value of the X_i^{S1} solution is better than the current solution X_i, then update X_i
        if the fitness value of the X_i^{S1} solution is better than the best solution X_b, then update X_b
      else
        move Aquila to a new location, X_i^{S2}, using S^2 tactic (Eq. (4))
        if the fitness value of the X_i^{S2} solution is better than the current solution X_i, then update X_i
        if the fitness value of the X_i^{S2} solution is better than the best solution X_b, then update X_b
    else
      if $(rand \le 0.5)$
        move Aquila to a new location, X_i^{S3}, using S^3 tactic (Eq. (7))
        if the fitness value of the X_i^{S3} solution is better than the current solution X_i, then update X_i
        if the fitness value of the X_i^{S3} solution is better than the best solution X_b, then update X_b
      else
        move Aquila to a new location, X_i^{S4}, using S^4 tactic (Eq. (8))
        if the fitness value of the X_i^{S4} solution is better than the current solution X_i, then update X_i
        if the fitness value of the X_i^{S4} solution is better than the best solution X_b, then update X_b
    end if
  end for
  $t = t + 1$
end while
Output: the best solution X_b

Fig. 1. Pseudocode of AO.

phases: the exploration phase, which conducts a global search using S^1 and S^2 tactics, and the exploitation phase, which conducts a local search using S^3 and S^4 tactics. This condition $t \le (2/3) \times T$ determines the AO algorithm's transition between exploration and exploitation phases. t and T present the current iteration and the maximum number of iterations, respectively. When the condition is true, the AO conducts the exploration phase; otherwise, it executes the exploitation phase. The following subsections detail each phase of the AO algorithm:

2.1.1. Initialization phase

The search space and the fitness function are defined in this phase. The AO solutions are initialized randomly in the search space, considering the area's boundaries:

$$X_{ij} = rand \times \left(UB_j - LB_j\right) + LB_j \tag{1}$$

where X_{ij} is the position of the ith candidate solution (Aquila) in the jth dimension. $i = 1, 2, \ldots, N$ (N is the total number of solutions) and $j = 1, 2, \ldots, D$ (D is the dimension of the solutions). UB and LB are the maximum and minimum bounds of the given problem. $rand$ is a random number in $U \sim [0, 1]$.

2.1.2. Exploration phase

The S^1 and S^2 tactics are used to provide AO's exploratory behavior. In S^1, Aquila scouts the search zone from a high altitude to find the prey. S^1 mathematically formulated as follows:

$$X_i^{S1}(t+1) = X_b(t) \times \left(1 - \frac{t}{T}\right) + (X_M(t) - X_b(t)) \times rand \tag{2}$$

$$X_M(t) = \frac{1}{N}\sum_{i=1}^{N} X_i(t) \quad \forall \quad j = 1, \quad 2, \quad \ldots, \quad D \tag{3}$$

where $X_i^{S1}(t+1)$ presents the next position of the i^{th} solution in the continuous search space using S^1 tactic. $X_b(t)$ is the prey position (i.e. the best solution so far), and the $X_M(t)$ is the average position of Aquilae (solutions).

The Aquila then circles above the prey in the S^2 tactic, and narrowly explores the prey's vicinity in a spiral shape in preparation for the attack. S^2 mathematically formulated in Eq. (4).

$$X_i^{S2}(t+1) = X_b(t) \times Levy(D) + X_R(t) + (r \times \cos(\theta) - r \times \sin(\theta)) \times rand \tag{4}$$

$$Levy(D) = 0.01 \times ((\mu \times \sigma) \Big/ |v|^{1/\beta}), \quad \sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\pi\beta/2)}{\Gamma((1+\beta)/2) \times \beta \times 2^{((1-\beta)/2)}}\right)^{\frac{1}{\beta}} \tag{5}$$

$$r = r_1 + 0.0265 \times [1 \quad 2 \quad 3\ldots D], \quad \theta = -0.005 \times \quad [1 \quad 2 \quad 3\ldots D] + (3 \times \pi/2) \tag{6}$$

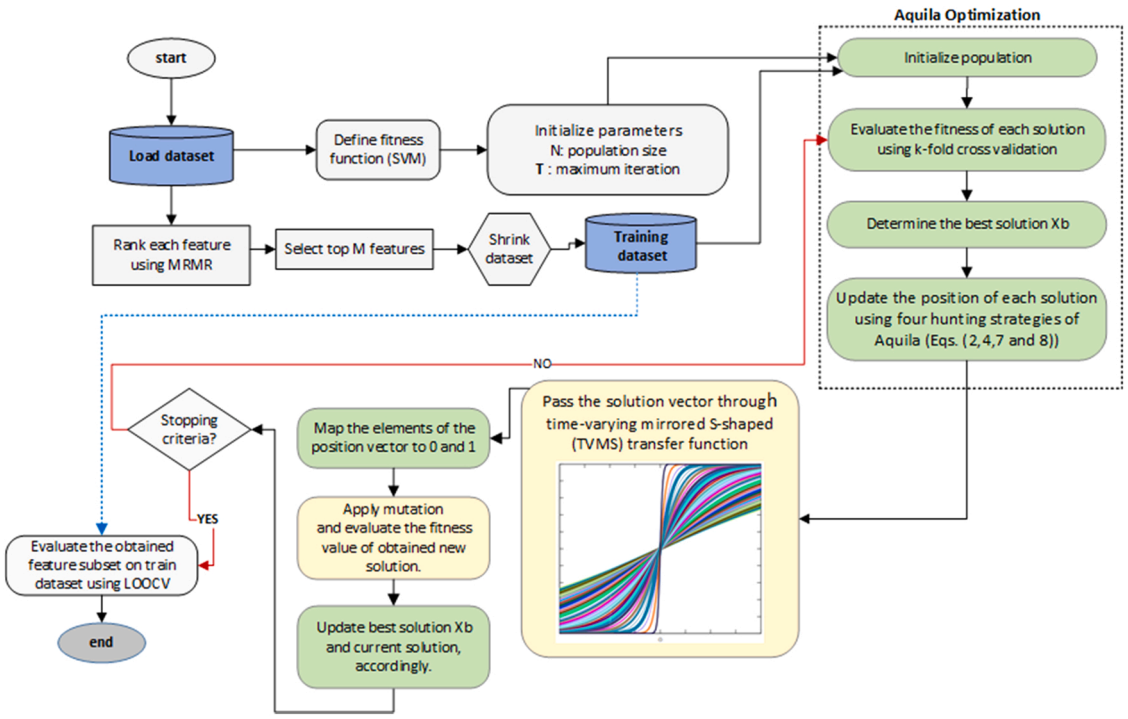


Fig. 2. Flowchart of proposed mRMR-MBAO approach for GS.

where $X_R(t)$ is a randomly selected solution from N candidate solutions in the current iteration t. r_1 is a random value in the range [0,20], and [1,2,3…D] is a vector from 1 to dimension size D. In Levy flight distribution ($Levy(D)$), β= 1.5, μ and ν are two random numbers within $[0,1]$, and the Gamma function Γ for an integer z is expressed as $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$.

The S^1 and S^2 techniques in the exploration phase also are conditioned by a random number. If the random value is less than 0.5, then the S^1 tactic is performed; otherwise, the S^2 tactic is conducted.

2.1.3. Exploitation phase

The S^3 and S^4 tactics are employed to offer AO's exploitation search behavior. Aquila is ready to land and attack in S^3. It makes a vertical descent with a preliminary attack to see how the prey reacts. This technique is mathematically presented as follows:

$$X_i^{S3}(t+1) = (X_b(t) - X_M(t)) \times 0.1 - rand + (rand \times (UB - LB) + LB) \times 0.1 \quad (7)$$

In S^4 the Aquila finally attacks the prey to grab it, which is mathematically defined as follows:

$$X_i^{S4}(t+1) = QF \times X_b(t) - ((2 \times rand - 1) \times X_i(t) \times rand) - (2 \times (1 - (t/T)) \times Levy(D) + rand \times (2 \times rand - 1)) \quad (8)$$

$$QF(t) = t^{(2\times rand-1)/(1-T^2)} \quad (9)$$

where $QF(t)$ represents the value of the quality function in the tth iteration that is employed to balance the search strategy.

Again, a random number influences the S^3 and S^4 strategies. If the random value is greater than 0.5, then the S^3 tactic will be called to action; otherwise, the S^4 tactic will be used. The pseudocode of the AO algorithm is presented in Fig. 1. The AO is a relatively new approach that originally has been suggested to handle numerical optimization problems as well as real-world engineering design optimization challenges. This study is one of the first to put the AO to the test in terms of discrete GS problems.

2.2. The mRMR filtering approach

The mRMR, as a widely used GS filtering approach (Pashaei and Pashaei, 2021a, 2022; Radovic et al., 2017), favors features that have a high correlation with the class (output) but a low correlation with each other. The mutual Information Difference (MID) objective function is used in mRMR to evaluate genes in the search space. MID denotes the relevance and redundancy difference of genes, while mutual information is utilized to calculate the correlation between genes (redundancy) and correlation with the class (relevance). Let $M = \{x_{kj}\}_{K\times D}$ represents a microarray gene expression matrix, where x_{kj} represents the expression of gene j in sample k. K and D represent the total number of samples, and genes in M, respectively. Let $x_j = (x_{1j}, x_{2j}, \ldots, x_{Kj})$ denotes expression of j^{th} gene across samples. The indexed set of genes are $G = \{1, 2, \ldots, D\}$. The method begins with a gene that has maximum mutual information with the class $\mathscr{C}$ and inserts it into a subset $Z \subset G$. Then, it adds the next gene with a maximum value of MID into the set of already selected genes (Z subset).

$$MID(j) = I(\mathscr{C};j) - \frac{1}{|Z|}\sum_{j' \in Z, j' \neq j} I(j;j') \quad (10)$$

$$I(\mathscr{C};j) = \sum_{x_j} p(\mathscr{C};x_j) \log\left(p(\mathscr{C};x_j)/p(\mathscr{C})p(x_j)\right) \quad (11)$$

where $I(\mathscr{C};j)$ is the mutual information between class labels $\mathscr{C}$ and gene j, which quantify the relevancy of gene j for the classification. $I(j;j')$ represents the redundancy of gene j with the other genes in the subset Z.

3. The proposed mRMR-MBAO algorithm for GS

This section introduces a novel GS technique based on mRMR and two different BAO variants (BAO-TVMS with and without mutation). The least important genes in the dataset are first pruned using the mRMR filtering method. Then, as shown in Fig. 2, the proposed MBAO with TVMS transfer function is utilized to determine the best gene subset from the current set of genes. The proposed method is described in-depth in

Algorithm 2 Binary Aquila Optimizer with TVMS transfer function

Initialize the AO's parameters
Initialize the Aquila population $X_i(i = 1,2,\ldots,N)$
while $(t < T)$
  Evaluate each solution (Aquila) using fitness values
  Determine the best solution $X_b(t)$ (i.e. prey's location)
  for (each solution (X_i) i) **do**
    Calculate the $X_M(t)$, $QF(t)$, $Levy(D)$ and r
    if $(t \le \left(\frac{2}{3}\right) \times T)$ **then**
      if $(rand \le 0.5)$
        Move Aquila to a new location using S^1 tactic, $x_i^{new} = X_i^{S1}$ (Eq. (2)
      else
        Move Aquila to a new location using S^2 tactic, $x_i^{new} = X_i^{S2}$ (Eq. (4))
    else
      if $(rand \le 0.5)$
        Move Aquila to a new location using S^3 tactic, $x_i^{new} = X_i^{S3}$ (Eq. (7))
      else
        Move Aquila to a new location using S^4 tactic, $x_i^{new} = X_i^{S4}$ (Eq. (8))
    end if
    Calculate ω using Eq. (14)
    for(each dimension j**) do**
      Compute $S\left(x_{ij}^{new}, \omega\right)$ using Eq. (12)
      if $(S\left(x_{ij}^{new}, \omega\right) > rand)$ **then**
        $P_{ij} = 1$
      else
        $P_{ij} = 0$
      end if
      Compute $S'\left(x_{ij}^{new}, \omega\right)$ using Eq. (13)
      if $(S'\left(x_{ij}^{new}, \omega\right) < rand)$ **then**
        $P'_{ij} = 1$
      else
        $P'_{ij} = 0$
      end if
    End for *j*
    if (fitness(P_i) > fitness(P_i') **then** $x_i^{new} = P_i$ **else** $x_i^{new} = P_i'$ (Eq.(17))
    if (fitness(x_i^{new}) > fitness (X_i), **then** $X_i = x_i^{new}$
    if (fitness(x_i^{new}) > fitness (X_b), **then** $X_b = x_i^{new}$
  End for *i*
  $t = t + 1$
end while
Output: the best solution X_b

Fig. 3. Pseudocode of BAO with TVMS transfer function.

the following subsections.

3.1. Preprocessing of genes using mRMR

The mRMR filter method performs data preprocessing to prepare the input data for suggested BAO by eliminating noisy genes. mRMR selects the *M* top-ranked genes based on the score achieved from their correlation with each other and class label. The selected genes, which served as input for the construction of the initial population in the BAO wrapper GS approach, are more useful for cancer classification. The filtering method, in addition to improvement of classification accuracy, reduces the computing burden associated with an exhaustive search across all possible gene subsets of wrapper approaches, since the number of gene subsets grows exponentially as the number of genes increases.

3.2. Binary Aquila optimizer with TVMS transfer function

The selected features from the previous preprocessing step are fed to the suggested BAO wrapper approach to find the best gene subset. As a result, the *M* top-ranking genes from mRMR are narrowed down even more at this stage to provide the smallest subset of informative genes with the highest fitness value for cancer classification.

GS is a binary optimization problem. This problem's solutions are limited to binary values of 0 and 1. A candidate solution *X* is represented as a binary string of size *D*, $X = \{g_1, g_2, \ldots, g_D\}$, where *D* denotes the problem dimension, and g_j is *j*th element (gene) in the solution ($j =$

Algorithm 3 Binary Aquila Optimizer with TVMS transfer function

Initialize the AO's parameters
Initialize the Aquila population $X_i(i = 1,2,\ldots,N)$
while $(t < T)$
  Evaluate each solution (Aquila) using fitness values
  Determine the best solution $X_b(t)$ (i.e. prey's location)
  for (each solution i) **do**
    Move Aquila to a new location, x_i^{new}, using four hunting strategies (Eqs.2,4,7 and 8)
    Calculate ω using Eq. (14)
    for(each dimension j) **do**
      Compute $S\left(x_{ij}^{new},\sigma\right)$ and $S'\left(x_{ij}^{new},\sigma\right)$ using Eqs. (12) and (13)
      if $(S\left(x_{ij}^{new},\sigma\right) > rand)$ **then** $P_{ij} = 1$ **else** $P_{ij} = 0$
      if $(S'\left(x_{ij}^{new},\sigma\right) < rand)$ **then** $P'_{ij} = 1$ **else** $P'_{ij} = 0$
    End for j
    if (fitness(P_i) > fitness(P'_i) **then** $x_i^{new} = P_i$ **else** $x_i^{new} = P'_i$ (Eq.(17))
    if (fitness(x_i^{new}) > fitness (X_i), **then** $X_i = x_i^{new}$
    if (fitness(x_i^{new}) > fitness (X_b), **then** $X_b = x_i^{new}$
    Perform mutation on current solution, x_i^m
    if (fitness(x_i^m) > fitness (X_i), **then** $X_i = x_i^m$
    if (fitness(x_i^m) > fitness (X_b), **then** $X_b = x_i^m$
  End for i
  $t = t + 1$
end while
Output: the best solution X_b

Fig. 4. Pseudocode of BAO with mutation.

Table 1
Benchmark gene expression datasets.

Dataset Name	#Samples	#Genes	#Classes	Diagnostic task
Colon Tumor	62	2000	2 (Binary class)	'Tumor':40, 'Normal':22
CNS	60	7129	2 (Binary class)	'MS':39, 'TF:'21
Ovarian	253	15154	2 (Binary class)	'Cancer':162, 'Normal':91
ALL-AML	72	7129	2 (Binary class)	'ALL':47, 'AML':25
Breast Cancer	97	24481	2 (Binary class)	'relapse':46, 'non-relapse':51
Prostate Tumor	102	10509	2 (Binary class)	'Tumor':50, 'Normal':52
MLL	72	12582	3 (Multi-class)	'AML':28, 'ALL':24, 'MLL':20
Leukemia-3c	72	7129	3 (Multi-class)	'B-cell':38, 'T-cell':9, 'AML':25
Leukemia-4c	72	7129	4 (Multi-class)	'AML-B':38, 'AML-P':9, 'ALL-B':21, 'ALL-T':4
SRBCT	83	2308	4 (Multi-class)	'EWS':29, 'RMS':25, 'NB':18, 'BL':11
Lung Cancer	203	12600	5 (Multi-class)	'A':139, 'N':17, 'SM':6, 'SQ':21, 'P':20

$\{1,2,..,D\}$). In the binary solution X, the gene g_j will be preserved if the value is 1, and it will be eliminated if the value is 0. Thus, only genes that are coded in one are considered in the evaluation.

AO works in a continuous solution space whereas the GS problem has a discrete (binary) solution space. The continuous space must be turned into discrete space before applying the AO to the GS problem. Therefore, AO should be modified to handle binary optimization problems. This means that each solution's elements should be set to 0's or 1's. This modification is done using a transfer function (TF). Although there are various TFs in the literature including V-shaped, S-shaped, and U-shaped, TF selection is not a trivial task (Chaudhuri and Sahu, 2021; Beheshti, 2021). The choice of TFs was demonstrated to have a substantial impact on the binary algorithm's output (Hammouri et al., 2020). This study uses the recently developed TVMS transfer function (Beheshti, 2020) to introduce a binary version of AO (BAO). The AO is a new efficient NIOA whose capability to solve the GS problem has not been investigated yet.

Table 2
Parameter setting.

Parameters	BAO	MBAO	Explanation
T	50	50	Max number of iteration
N	35	35	Population size
r_1	10	10	Parameters inside BAO/MBAO movement
β	1.5	1.5	equations, which were set according to the
ω_{min}	1	1	original AO's parameter values
ω_{max}	10	10	
p_m	–	[0.005,0.9]	Mutation rate

The TVMS transfer function mathematically is formulated in Eqs. (12), (13), and (15).

$$S\left(x_{ij}(t+1),\quad \omega\right) = sigmoid\quad \left(x_{ij}(t+1),\quad \omega\right) = 1/\left(1+e^{\ \omega(-x_{ij}(t+1))}\right) \tag{12}$$

$$S'\left(x_{ij}(t+1),\quad \omega\right) = sigmoid\quad \left(x_{ij}(t+1),\quad \omega\right) = 1/\left(1+e^{\ \omega(x_{ij}(t+1))}\right) \tag{13}$$

Two sigmoid functions are used to convert the real results to binary ones by generating the probability of changing the element x_{ij} to 0 or 1. $x_{ij}(t+1)$ demonstrates the value of the jth dimension of the ith solution in the new iteration $(t+1)$, and ω is a time-varying variable, which is defined as follows:

Table 3
Choosing the optimal value of mutation parameter p_m for some datasets. The best values are highlighted in bold and the selected parameter values are highlighted by underlining.

p_m values	Colon Tumor		CNS		Breast Cancer		Prostate Tumor	
	ACC	#G	ACC	#G	ACC	#G	ACC	#G
0.005	96.67	11	80.48	27	85.22	28	95.27	9
0.01	95.00	11	88.24	23	85.22	28	95.27	9
0.02	92.04	11	88.33	41	82.67	32	97.09	11
0.03	87.80	10	85.14	32	86.54	12	96.09	8
0.04	91.66	12	82.38	19	86.78	10	97.18	10
0.05	**96.90**	9	85.48	21	85.67	26	97.09	15
0.06	94.26	13	**89.90**	40	**91.78**	25	96.09	8
0.07	91.76	11	88.24	39	90.85	30	96.28	7
0.08	91.76	11	85.24	41	90.82	34	96.27	7
0.09	91.28	11	85.48	25	85.56	**8**	95.00	24
0.1	95.11	28	84.67	38	88.21	21	95.09	19
0.2	90.21	24	80.14	34	84.11	15	96.09	**3**
0.3	94.83	21	83.24	18	86.78	18	96.09	7
0.4	88.81	19	88.24	52	87.76	42	**98.00**	7
0.5	94.00	12	82.81	41	86.56	24	95.18	9
0.6	89.57	36	81.23	46	87.56	17	96.17	18
0.7	94.35	26	88.03	47	84.56	17	96.18	24
0.8	92.09	14	85.38	**4**	90.87	26	96.09	25
0.9	91.21	**4**	82.99	11	83.44	10	95.27	5

Table 4
Comparison of classification accuracy (ACC), True Positive (TP), and False Positive (FP) rate of different filtering approaches on eleven biological datasets using SVM classifier.

Class	Datasets	Metrics	mRMR	IG	Chi-square	ReliefF
Binary Class	Colon Tumor	ACC	**85.48**	77.41	79.03	80.64
		TPR	**85.5**	77.4	79	80.6
		FPR	**1.82**	26.7	27.9	27
	CNS	ACC	63.33	**75.00**	73.33	66.66
		TPR	63.3	**75**	73.33	66.7
		FPR	4.83	**26.6**	34.1	39.9
	Ovarian	ACC	**100**	100	100	100
		TPR	**100**	100	100	100
		FPR	**100**	100	100	100
	ALL-AML	ACC	**98.61**	98.61	97.22	97.22
		TPR	**98.6**	98.6	97.2	97.2
		FPR	**2.8**	2.6	3.3	3.3
	Breast	ACC	75.25	74.22	74.22	**77.31**
		TPR	75.3	74.2	74.2	**77.3**
		FPR	2.49	2.6	2.5	**2.32**
	Prostate_Tumor	ACC	**94.11**	88.23	86.27	93.13
		TPR	**94.1**	88.2	86.3	93.1
		FPR	**5.9**	11.8	13.8	6.8
Multi Class	MLL	ACC	**97.22**	95.83	97.22	97.22
		TPR	**97.2**	95.8	97.2	97.2
		FPR	1.4	2.0	1.4	1.1
	Leukemia-3c	ACC	95.83	**97.22**	94.44	94.44
		TPR	95.8	**97.2**	94.44	94.44
		FPR	3	**1.8**	3.8	3.8
	Leukemia-4c	ACC	**94.44**	94.44	94.44	87.5
		TPR	**94.4**	94.4	94.4	87.5
		FPR	**3.3**	3.3	3.3	6.1
	SRBCT	ACC	**100**	100	100	100
		TPR	**100**	100	100	100
		FPR	**100**	100	100	100
	Lung Cancer	ACC	**96.55**	93.59	92.11	91.62
		TPR	**96.6**	93.6	92.11	91.6
		FPR	**4.4**	8.8	9	13.1

The best results are highlighted in bold font.

$$\omega = (\ \omega_{max} - \ \omega_{min})(t/T) + \ \omega_{min} \tag{14}$$

where t indicates the current iteration and T indicates the maximum iteration. $\omega_{min} = 1$ and $\omega_{max} = 10$ are the lower and upper bound of the variable ω. To transition seamlessly from exploration to exploitation, ω is initially set to ω_{max} and gradually dropped to ω_{min}.

Eqs. (15) and (16) determine the binary solutions of each transfer function. Then, to update the new solution, the best solution among $P_{ij}(t+1)$ and $P'_{ij}(t+1)$ is chosen according to their fitness value. The elements of the candidate solution are set to 0 and 1 based on Eq. (17).

$$P_{ij}(t+1) = \left\{ \begin{matrix} 1 & ,\&if \quad rand < S\left(x_{ij}(t+1), \quad \omega\right) \\ 0 & , \quad if \quad rand \geq S\left(x_{ij}(t+1), \quad \omega\right) \end{matrix} \right\} \tag{15}$$

$$P'_{ij}(t+1) = \left\{ \begin{matrix} 1 & ,\&if \quad rand > S'\left(x_{ij}(t+1), \quad \omega\right) \\ 0 & , \quad if \quad rand \leq S'\left(x_{ij}(t+1), \quad \omega\right) \end{matrix} \right\} \tag{16}$$

$$x_{ij}(t+1) = \left\{ \begin{matrix} P_{ij}(t+1) & ,\&if \quad fitness\left(P_{ij}(t+1)\right)\rangle \quad fitness\left(P'_{ij}(t+1)\right) \\ P'_{ij}(t+1), & if \quad fitness\left(P_{ij}(t+1)\right) \leq \quad fitness\left(P'_{ij}(t+1)\right) \end{matrix} \right\} \tag{17}$$

A set of randomly generated binary solutions is used to start the wrapper feature selection approaches. During the search process, the approaches employ a fitness (objective) function to evaluate each solution in the population. The fitness function is an important factor to consider when designing any optimization algorithm because the fitness function guides the algorithm to find the optimal solution within the large search space. Machine learning classifiers such as support vector machine (SVM) (Pashaei and Pashaei, 2021a; Pashaei and Aydin, 2018; Pashaei et al., 2016c), k-nearest neighbor (KNN) (Dashtban and Balafar, 2017), artificial neural network (ANN) (Pashaei and Pashaei, 2021b), and Naïve Bayes (NB) (Ahmed et al., 2017) are used as the fitness function to evaluate the predictive accuracy of candidate gene subsets. SVM with linear kernel function is utilized in this study to calculate the fitness value of each solution using 10-fold cross-validation (CV). A better solution has a higher fitness value, which permits more accurate cancer classification. It's worth noting that wrapper FS techniques try to reduce the number of genes while improving classification accuracy. If two subset genes have the same classification accuracy, the subset with fewer genes is chosen in the evaluation procedure. Fig. 3 depicts the pseudocode of the proposed BAO for GS.

Each gene subset can be seen as a candidate solution (Aquila position) in BAO. Each solution may contain D genes, where D is the number of genes found during the previous filtering stage. The algorithm begins with a population of randomly generated binary solutions. The fitness function is then employed to assess each solution in the population. The population's best solution is found after fitness values have been assigned. The core loop of BAO is repeated multiple times. Several random values within [0,1] are used to choose between the exploration and exploitation phases. Four hunting strategies of Aquila (Eqs. (2), (4), (7), and (8)) are utilized to update the solutions. Then, the TVMS transfer function is carried out to convert real values to binary values. First, P_i and P'_i are calculated by Eqs. (15) and (16), and the best solution between P_i and P'_i is selected as the next solution. Then, the obtained solution is compared with the current solution and global best solution in terms of classification accuracy and number of selected genes, and both are updated accordingly. The algorithm repeats the steps until it reaches the value of the maximum iteration.

3.3. Proposed BAO with mutation (MBOA)

The proposed BAO performs well on a wide range of microarray datasets. On some benchmark datasets, however, it fails to provide sufficient performance. To address this issue, the mutation mechanism is

Table 5
Comparison between BAO and MBAO.

Algorithm		Datasets										
		Colon	CNS	Ovarian	SRBCT	Lung Cancer	Leukemia-4c	MLL	ALL-AML	Leukemia-3c	Breast	Prostate_Tumor
BAO	#G	**11**	31.8	5.857	9.2	42.71	**21.33**	11.29	**7.66**	**9.8**	**21.87**	16.87
	ACC	93.43	84.58	100	100	98.08	98.18	99.06	99.583	99.9	87.23	95.99
MBAO	#G	16.11	**21.37**	**5.166**	**7.5**	**23.833**	21.33	**11.28**	8.71	12.28	23.58	**15.85**
	ACC	**95.74**	**88.57**	100	100	**98.54**	**98.70**	**99.64**	**100**	**100**	**89.12**	**97.038**
	T–Sig.	*	*	–	–	–	–	–	–	–	*	*

The best results are highlighted in bold font.

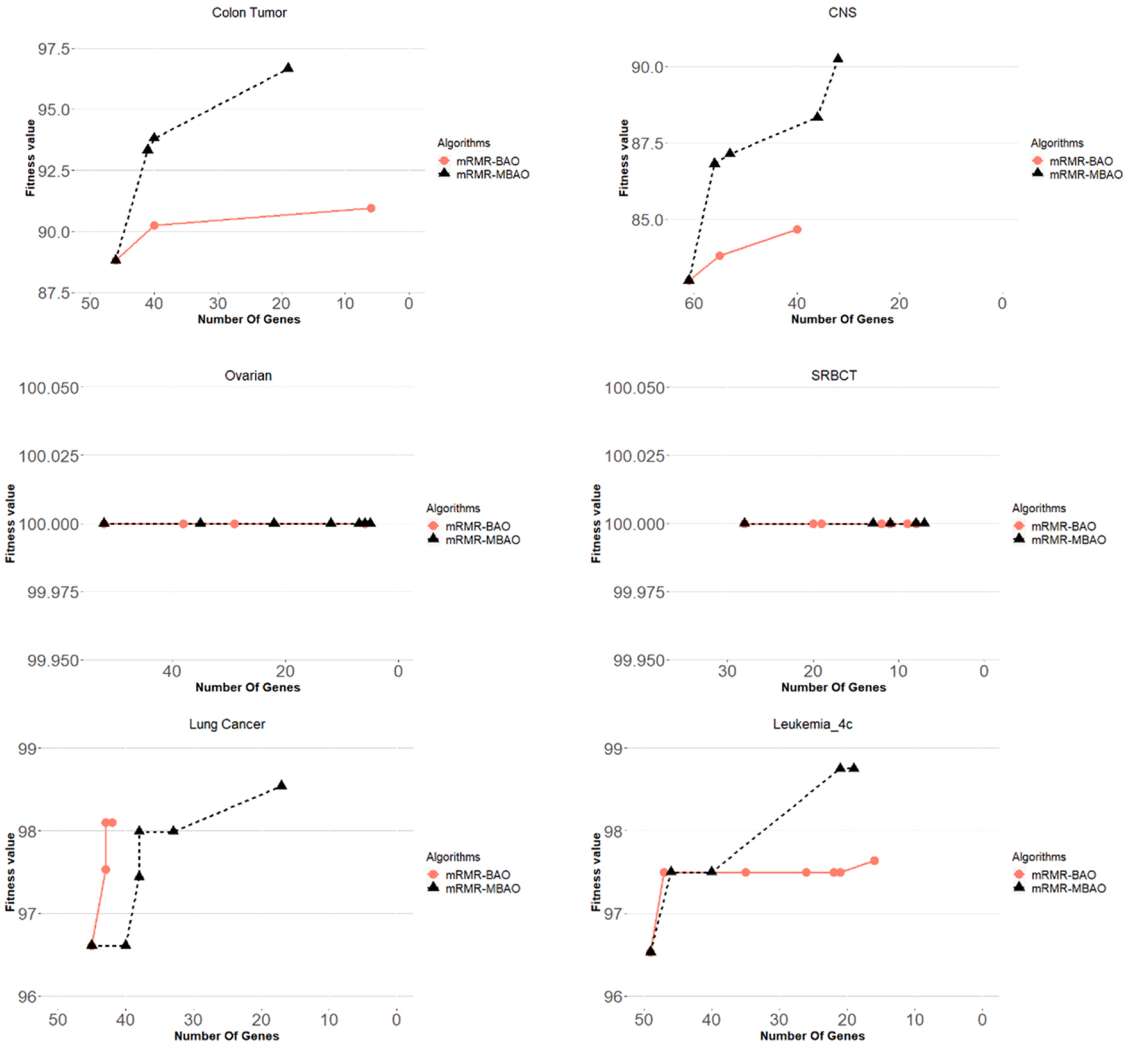


Fig. 5. Classification accuracy versus the number of selected genes by mRMR-BAO and mRMR-MBAO for 11 datasets using the SVM classifier.

introduced to the BAO algorithm, and an MBAO algorithm is suggested. The local searching performance of the BAO is improved, the diversity of solutions is increased, and early convergence is avoided by incorporating the mutation mechanism. The mutation operator is added to the BAO structure to produce a mutation vector x_i^m for the current solution x_i. The mutation operator modifies specific elements in the solution x_i with a probability p_m (mutation rate), resulting in a more diverse solution to assist the search process in escaping local optimal. The values of p_m have a significant impact on the performance of mutation operators in BAO. Each solution has a mutation rate p_m, which determines the number of mutated elements. Mutation rate p_m is set to a random value within the range $[0.01, 0.9]$ for each solution at each iteration.

Assume the solution after the tth iteration is $x_i = (1, 0, 0, 1, 1, 1)$ in MBAO. One or more elements in the x_i are chosen randomly, the elements are modified by reversing their values, and the other elements are left unchanged (e.g. two elements (1 and 3) are chosen). The mutated solution $x_i^m = (0, 0, 1, 1, 1, 1)$ can be obtained via this mutation technique. The BAO evaluates the mutated vector x_i^m using the fitness function. If $x_i^m$is better than x_i in terms of fitness value and number of selected genes, x_i will be replaced by x_i^m. Otherwise, the x_i remained unchanged. Furthermore, if x_i^m has a higher fitness value and a fewer number of selected genes than the best solution, x_b will be updated by x_i^m. (i.e. Lines 17–19 in Algorithm 3). The proposed MBAO algorithm pseudocode is shown in Fig. 4.

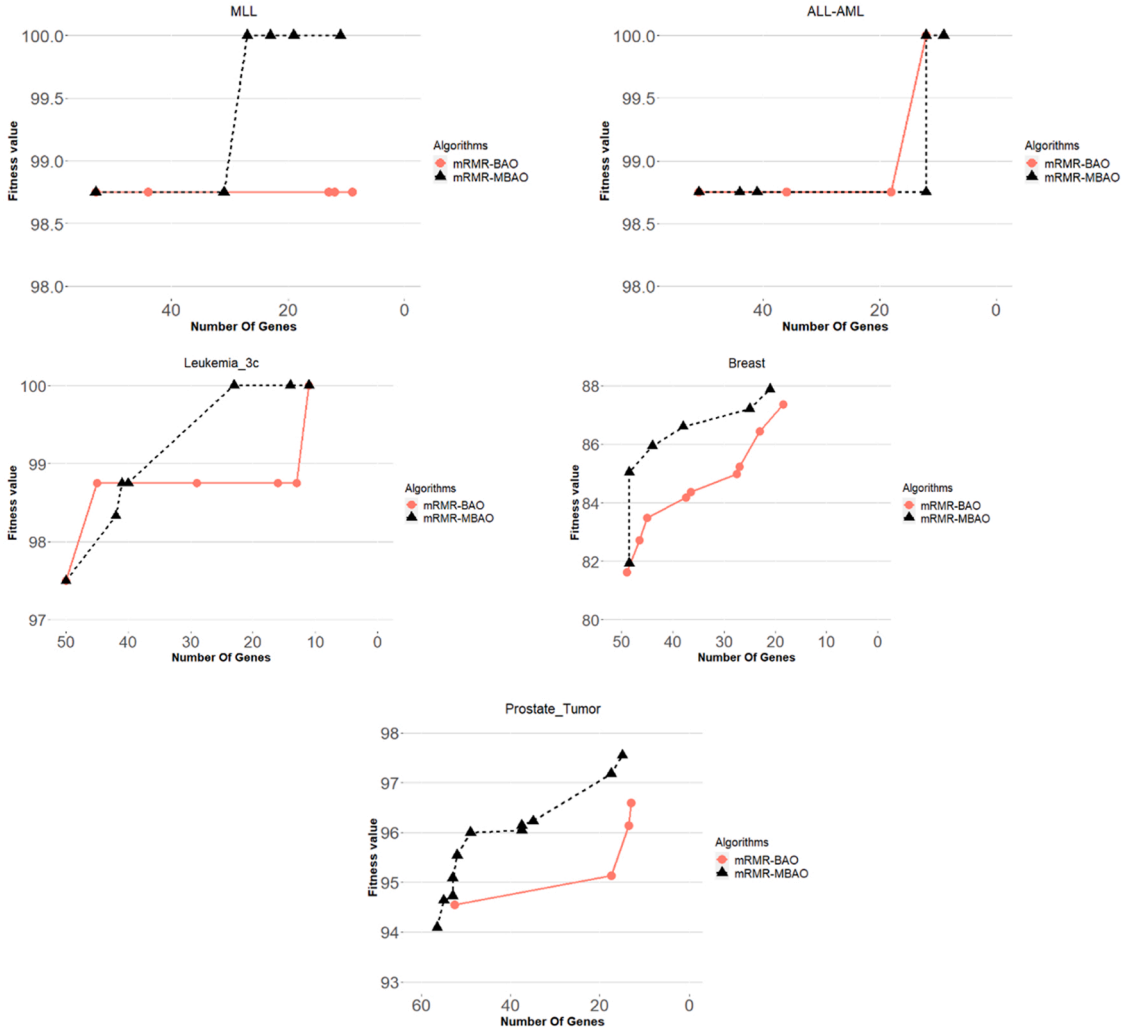


Fig. 5. (*continued*).

4. Experimental results

In this section, the performance of the suggested method was assessed using benchmark microarray datasets. Section 4.1 describes the details of the datasets. The min-max normalization and mRMR filtering method were applied to datasets at the first stage to normalize our data to [0,1] and eliminate noisy and redundant features. The impact of the mRMR on classification accuracy is investigated in Section 4.2. In the next stage, suggested BAO and MBAO algorithms were employed to find the best gene subset. Therefore, Section 4.3 evaluates the performance of BAO and MBAO on microarray datasets where the BAO with and without mutation are compared. Section 4.4 discusses the comparison of the proposed mRMR-MBAO-SVM algorithm with existing state-of-the-art approaches such as IG-MBKH (Zhang et al., 2020), SU-RSHSA (Shreem et al., 2022), MIM-MFOA (Dabba et al., 2021), IBCFPA (Yan et al., 2019a), ISFLA (Hu et al., 2018), BCROSAT (Yan et al., 2019b), mRMR-DBH (Pashaei and Pashaei, 2021a), SARA (Baliarsingh et al., 2021), and TOPSIS-Jaya (Chaudhuri and Sahu, 2021) in the terms of classification accuracy and the number of selected genes. Finally, Section 4.5 presents the biological interpretation of the marker genes selected by the mRMR-MBAO method.

The proposed algorithm was implemented in R programming language and simulations are conducted on an Intel Core 2.2 GHz Core i5 CPU with 8 GB of RAM and a Windows 10 operating system. The 'praznik' R package was used to implement mRMR and package 'e1071' was utilized to implement the SVM classifier. Weka, an open-source machine learning platform, was used to implement the other filter techniques (ReliefF, Chi-Square, and IG). The heat map was generated using the "pheatmap" package of R.

4.1. Datasets and experimental setup

The suggested algorithms were evaluated on eleven publicly available microarray gene expression datasets with different types of diseases (https://data.mendeley.com/datasets/fhx5zgx2zj/1). The details of these datasets are summarized in Table 1 and include the dataset name, the number of samples, the number of genes, the number of classes, and the diagnostic task. The datasets included binary classes and multi-classes datasets with thousands of genes. The datasets' dimensional scopes span from 2000 to 15154 (high-dimensional data) while sample sizes (number of patients) are (very) small.

SVM classifier with a 10-fold CV was utilized to validate the fitness performance of the suggested algorithm on each of the datasets, while Leave-One-Out CV (LOOCV) approach provided the final evaluation of the suggested method with the selected genes. Moreover, a linear kernel was chosen in SVM to carry out the classification task. For the fairness of the experiment, the experiments of all methods were also repeated 10 times for each dataset, and the mean values were reported as the final result. Table 2 provides the proposed methods' parameter setting values. These parameter values were chosen based on preliminary experiments and previous studies on AO (Abualigah et al., 2021; Wang et al., 2021). Parameter values of other comparison algorithms are the same as Ref

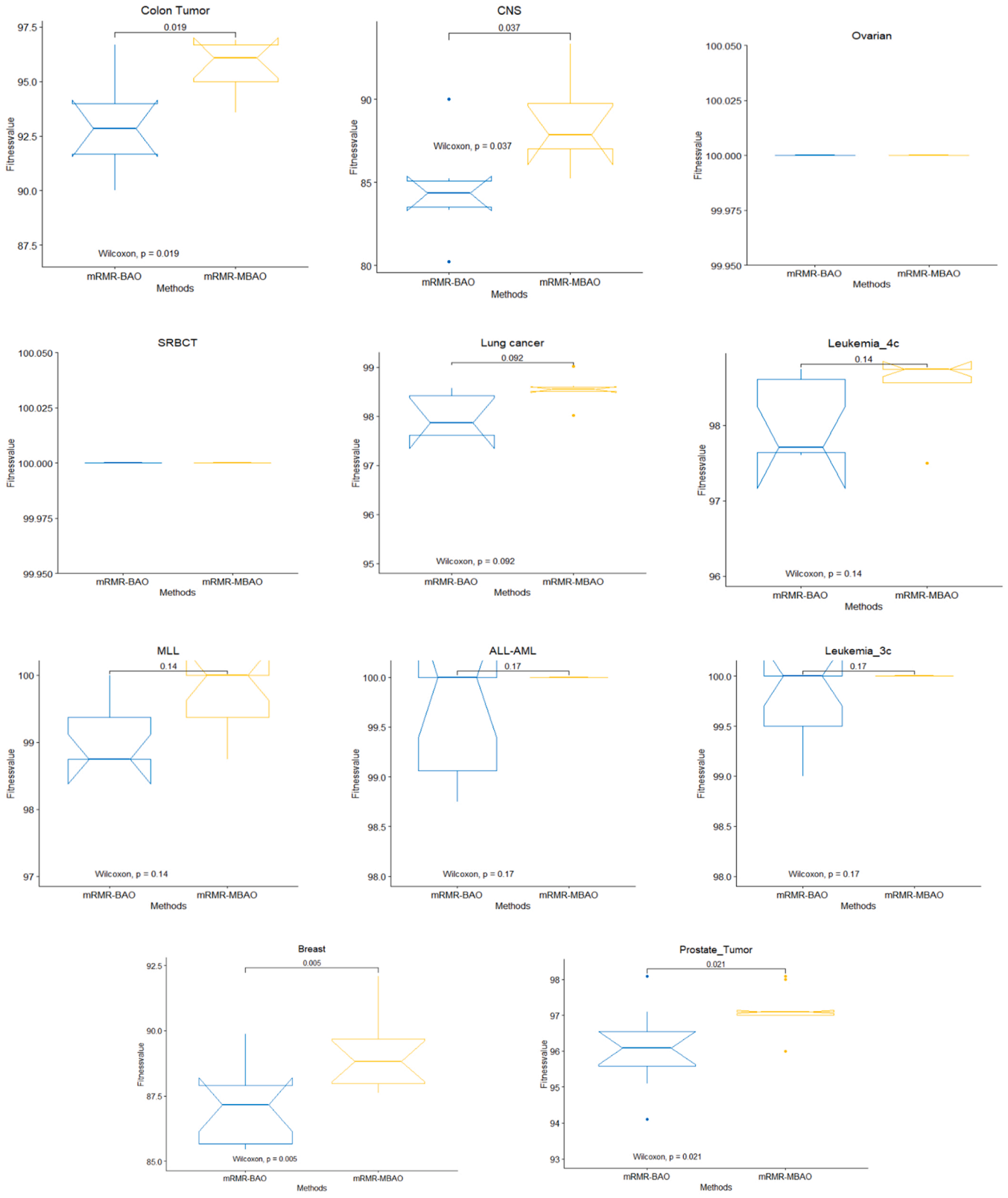


Fig. 6. Boxplots for all datasets to show the diverse behavior of the proposed mRMR-MBAO.

(Zhang et al., 2020). The population size N, the maximum number of iterations T, and mutation rate p_m are the three main hyperparameters of the proposed algorithm. For both BAO and MBAO algorithms, population size and iteration number are set to 35 and 50, respectively, in our trials. While performing several experiments, the mutation hyperparameter comes out to be important since the slight variation of this parameter causes significant changes in the results. The variation of classification accuracies (ACC) and the number of selected features (#G) concerning the mutation rate on several microarray datasets are given in Table 3. It is worth noting that the reported results in Table 3 were

Table 6
Experimental results by mRMR-MBAO on all datasets.

Dataset	Accuracy				#Genes			
	Best	Worst	Avg.	SD	Best	Worst	Avg.	SD
Colon Tumor	96.90	95	95.74	1.183	12	20	16.11	3.75
CNS	93.33	85.24	88.57	2.85	11	32	21.37	8.423
Ovarian	100	100	100	0	4	6	5.16	0.752
SRBCT	100	100	100	0	7	8	7.5	0.547
Lung Cancer	99.02	98.02	98.54	0.318	13	27	23.83	7.88
Leukemia_4c	98.75	97.5	98.70	0.431	9	29	21.33	7.88
MLL	100	98.75	99.64	0.609	7	14	11.28	2.21
ALL-AML	100	100	100	0	6	12	8.714	2.13
Leukemia_3c	100	100	100	0	9	16	12.28	2.49
Breast	92.09	87.62	89.12	1.33	12	39	23.58	7.82
Prostate_Tumor	98.09	96	97.038	0.604	11	24	15.85	4.29

Table 7
Results of comparison between proposed mRMR-MBAO and the state-of-art methods.

Datasets	Metric	Proposed approach	SU-RSHSA	mRMR-DBH	IBCFPA	MIM-MFOA	BCROSAT	ISFLA	SARA	TOPSIS-Jaya	IG-MBKH
Colon Tumor	| # G|	16.11	**7. 59**	12	25.9	24.25	20.5	37.1	9	18.90	17.10
	ACC	95.74	93.17	97.02	92.16	**99.19**	92.31	89.56	97.02	97.76	96.47
CNS	| # G|	21.37	13.15	39.75	25.2	17	21.40	41.1	–	**8.7**	14.70
	ACC	88.57	89.36	**97.19**	84.82	85.00	82.00	77.46	–	96.22	90.34
Ovarian	| # G|	5.166	20.47	**2.66**	48.8	33.20	–	33.3	6	18.50	3.40
	ACC	**100**	99.61	**100**	99.06	97.63	–	97.29	99.15	99.52	100
SRBCT	| # G|	7.5	8.37	–	49.8	13.50	33.0	43.10	**5**	15.80	6.3
	ACC	**100**	97.98	–	98.02	95.18	95.76	93.72	99.81	100	100
Lung Cancer	| # G|	23.833	–	–	82.2	19.50	23.3	40.3	**5**	9.9	23.80
	ACC	**98.54**	–	–	94.44	87.13	93.57	89.56	90.22	94.24	96.12
Leukemia_4c	| # G|	21.33	**12.02**	–	45.6	–	30.9	32.2	–	19.50	15.80
	ACC	98.70	97.11	–	94.35	–	90.90	90.91	–	99.72	**99.44**
MLL	| # G|	11.28	7.83	**5.25**	42.21	24.50	35.6	40.7	–	12.90	11.10
	ACC	99.64	99.94	**100**	96.51	69.30	98.04	92.59	–	99.62	99.72
ALL-AML	| # G|	8.71	21.64	**4**	29.9	8.23	–	35.8	7	16.10	4.2
	ACC	**100**	100	**100**	99.37	99.31	–	96.34	97.65	100	100
Leukemia_3c	| # G|	12.28	10.72	–	49.6	10.50	32	40.0	7	**6.6**	8.80
	ACC	**100**	100	–	97.97	93.75	94.50	94.00	98.02	100	100
Breast Cancer	| # G|	23.58	18.31	**14**	–	22.50	–	–	–	–	–
	ACC	89.12	80.40	**90.21**	–	84.11	–	–	–	–	–
Prostate Tumor	| # G|	15.85	–	28	–	**14.00**	–	–	–	–	–
	ACC	97.038	–	**98.19**	–	86.63	–	–	–	–	–

The best results are highlighted in bold.

produced by a single run of the proposed MBAO algorithm using 10 fold CV. From Table 3, it can observe that the optimal choice of p_m hyperparameter can significantly affect the resulting model's performance. Therefore, the mutation parameter should be adjusted for each dataset to maximize the proposed model performances. Fortunately, fine-tuning the mutation parameter is not a difficult task. It can be determined by a grid search as seen in Table 3.

4.2. Effect of mRMR in the proposed method

The classification behavior of mRMR is compared with several well-known filtering algorithms such as IG, ReliefF, and Chi-square in order to explore the effect of the mRMR on the performance of the proposed algorithm. The mRMR filtering approach was used to find the most important genes and reduce the dataset's dimensionality. For comparative purposes, the SVM classifier with LOOCV was utilized to measure the performance of all filtering methods. Table 4 shows the performance of the mRMR and other comparative methods in terms of classification accuracy (ACC), True Positive rate (TPR), and False Positive rate (FPR) for all binary classes and multi-class microarray datasets. The best results are highlighted in bold font. Note that top-ranked 100 genes are selected to make up the initial set of candidate genes after the pre-filter operation except CNS (125), following previous studies (Pashaei and Pashaei, 2021a, 2022). According to the results presented in Table 4, the mRMR-SVM gives admirable classification performance in almost all datasets. It can be seen that mRMR is able to find very competitive results where some of which are the best-recorded.

4.3. Performance evaluation of BAO and MBAO

In this section, the effect of mutation on the performance of the proposed BAO algorithm is studied. The BOA with mutation (i.e. MBAO) and BAO without mutation are compared together. These comparisons are shown in Table 5, Figs. 5 and 6. Both algorithms were run in 10 independent runs because they are stochastic algorithms. It should be noted that as a first step the mRMR filtering method was used to exclude noisy features and 100 top-ranked genes were picked for BAO and MBAO. The average classification accuracy (ACC), the average number of selected genes (#G), and the Wilcoxon signed-rank statistical test are utilized in Table 5 for exposing the obtained experimental results. The Wilcoxon signed-rank statistical test was performed on classification accuracy to determine if there is a statistically significant difference between MBAO and BAO algorithms. The results obtained were summarized in the 'T-Sig' row of Table 5 with the probability range of $p - \text{value} \leq 0.05$. Symbol '*' indicates that the MBAO's results are significantly better than BAO, whereas symbol '-' indicates the MBAO results are not significantly better than BAO. Moreover, for all tested datasets, the performance of MBAO vs BAO was depicted in Fig. 5. As can be seen

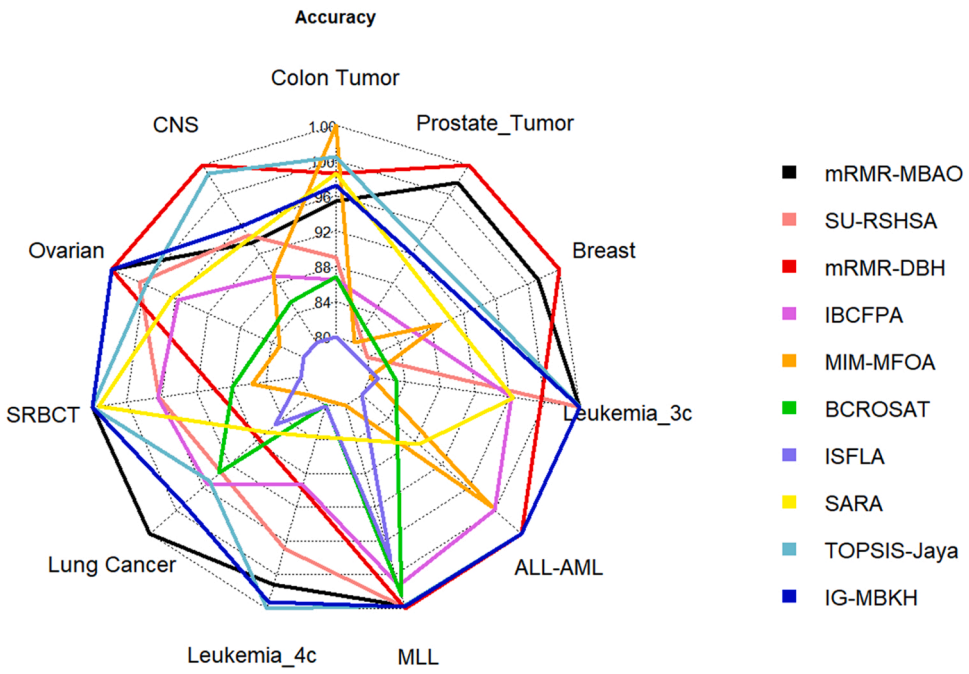


Fig. 7. Comparison between the proposed algorithm and the state-of-art methods regarding the accuracy.

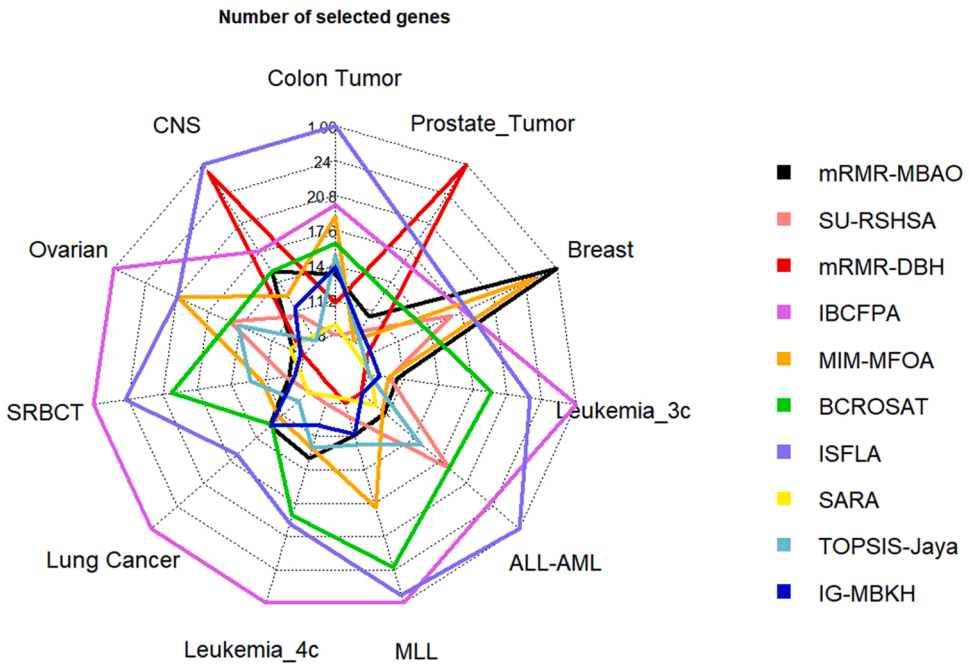


Fig. 8. Comparison between the proposed algorithm and the state-of-art methods regarding the number of selected genes.

in Table 5 and Fig. 5, MBAO is able to obtain higher classification accuracy than BAO on nine out of eleven datasets (i.e. Colon Tumor, CNS, Lung Cancer, Leukemia_4c, MLL, ALL-AMLL, Leukemia_3c, Breast, and Prostate_Tumor) and similar best-recorded classification accuracy (100%) in rest of datasets (i.e. Ovarian, and SRBCT).

In terms of the number of selected genes, MBAO performs better than BAO in two out of eleven datasets (i.e., CNS and Lung Cancer), and performs approximately the same in five datasets (i.e. Ovarian, SRBCT, Leukemia_4c, MLL, and Prostate_Tumor). For Colon Tumor, ALL-AMLL, Breast, and Leukemia_3c datasets, the BAO approach selected slightly fewer genes than MBAO; however, its classification accuracy is smaller. Furthermore, MBAO gets significantly better results than BAO in four out of eleven datasets, while no significant differences were found in the remaining seven datasets. It shows on some datasets, the mutation mechanism enhanced the performance of the BAO, but on others, the results were nearly identical. This proves the capability of naïve BAO with TVMS transfer function to GS selection. Based on the result reported in Table 5 and Fig. 5, both proposed BAO and MBAO are capable of determining a minimum number of the most informative genes to obtain competitive or higher classification accuracy. However, the results of MBAO are better than BAO in most of the datasets in terms of classification accuracy and the number of selected genes. These fruitful results are due to the integration of a mutation operator with BAO, which acts as a local search and so increases the BAO algorithm's

Table 8
The best subsets of genes obtained from the proposed approach for each dataset.

Dataset	Index of Genes	Accuracy	TPR	FPR
Colon Tumor	1976, 822, 1221, 1494, 377, 1920, 143, 1730, 317, 1102, 1562, 1520	96.90	96.9	3.8
CNS	5637, 6345, 2474, 6248, 3239, 2087, 5355, 1787, 6565, 4536, 4469, 4062, 5563	93.33	93.3	10.2
Ovarian	2238, 6781, 183, 2196	100	100	0.0
SRBCT	1, 1003, 545, 338, 1601, 1613, 1207	100	100	0.0
Lung Cancer	3191, 10188, 8457, 1422, 9164, 10573, 12121, 12375, 9134, 6949, 12524, 5934, 7298	99.02	99	2
Leukemia_4c	4050, 6855, 3469, 6225, 5543, 2121, 5300, 3237, 1926, 5688	98.75	98.7	2
MLL	7666, 6067, 7232, 9845, 1132, 7961, 6089, 7155	100	100	0.0
ALL-AML	2354, 4847, 6376, 1779, 5593, 4951	100	100	0.0
Leukemia_3c	2642, 4055, 4377, 6236, 5466, 5300, 3847	100	100	0.0
Breast	10889, 8776, 1872, 6541, 2769, 13625, 3232, 2177, 8782, 18811, 6836, 1364, 18761, 15833, 21752, 9480, 19044, 7206, 20859, 13058, 24107, 22834, 1409, 8899	92.09	92	7.4
Prostate_Tumor	4823, 8765, 7451, 2439, 8009, 5227, 4346, 275, 10130, 6105, 10504	98.09	98	2.6

exploitation capabilities.

Detailed results of MBAO when solving all datasets are presented in Table 6. The accuracy and number of selected genes (#Genes) are two evaluation metrics that are used to quantify the performance of the prediction model. Besides average (Avg.), best, worst, and Standard Deviation (SD) are adopted to assess the robustness of the algorithm. Smaller SD indicates that the algorithm performs more stable. For almost all datasets, the SD achieved by mRMR-MBAO is relatively small in terms of classification accuracy. However, the SD is a little bit high in CNS, Lung cancer, Leukemia_4c, and Breast cancer datasets.

Moreover, the boxplot enriched with $p-\text{value} \leq 0.05$ (Wilcoxon signed-rank statistical test) was utilized as a graphical representation to provide a better understanding of the varied behaviors of the proposed strategies. Fig. 6 statistically compares the means of BAO and MBAO algorithms in the terms of fitness value and demonstrates the diversity of results during the search. The $p-\text{value} \leq 0.05$ shows that the MBAO generates significantly better results than BAO; however, the $p-\text{value} > 0.05$ indicates that the MBAO does not produce significantly better results than BAO. Although significant differences in favor of the MBAO can be seen for only four datasets, the classification accuracy of MBAO for the remaining datasets is much better than BAO. In addition, the diversity of produced results by MBAO is smaller when compared with BAO.

4.4. Comparative evaluation

The experiment in this study includes the comparison of mRMR-MBAO with other state-of-the-art GS approaches. Nine well-known GS approaches including MIM-MFOA (Dabba et al., 2021), IBCFPA (Yan et al., 2019a), ISFLA (Hu et al., 2018), BCROSAT (Yan et al., 2019b), mRMR-DBH (Pashaei and Pashaei, 2021a), SARA (Baliarsingh et al., 2021), SU-RSHSA (Shreem et al., 2022), TOPSIS-Jaya (Chaudhuri and Sahu, 2021), and IG-MBKH (Zhang et al., 2020) are utilized to further analyze the proposed algorithm performance. Table 7 presents the experiment result of multiple independent runs for each approach in terms of classification accuracy (ACC) and the number of selected genes (#G). Moreover, to demonstrate mRMR-MBAO performance intuitively, Figs. 7 and 8 show the average accuracy comparison and average selected gene numbers for all datasets, respectively.

As seen in Table 7 and Fig. 7, for one of the eleven experiment datasets, mRMR-MBAO outperformed other comparing approaches in terms of classification accuracy (i.e., Lung Cancer). In addition, in four out of eleven datasets (i.e. Ovarian, SRBCT, ALL-AML, and Leukemia_3c), the mRMR-MBAO is able to produce the best-recorded classification accuracy (100%) as done by other comparative methods. The mRMR-MBAO was ranked second in Prostate-tumor and Breast cancer datasets, and third in the Leukemia_4c dataset among the approaches. In the case of the MLL dataset, mRMR-MBAO has competitive accuracy, while for Colon and CNS datasets it has slightly lower performance compared to other state-of-art methods. In general, the classification accuracy generated by the mRMR-MBAO is promising for all datasets in comparison to existing approaches.

In terms of the number of the selected genes, mRMR-MBAO shows an acceptable performance based on the results in Table 6 and Fig. 8. Taking into account the number of genes as well as the classification accuracy, the mRMR-MBAO achieves the best performance on two datasets (Lung Cancer and the Prostate-Tumor), where it manages to select the fewest number of genes with the highest classification accuracy. In the Lung Cancer dataset TOPSIS-Jaya, SARA, and MIM-MFOA select fewer genes (9, 5, 19) and achieve 94.24%, 90.22%, and 87.13% classification accuracy. In comparison, the mRMR-MBAO method select quite higher genes (23) but achieved the highest classification accuracy (98.54%). In the Prostate-Tumor dataset, MIM-MFOA selects 14 genes with 86.63% accuracy, while mRMR-MBAO selects approximately the same number of genes (15) with 97.83% classification accuracy. The mRMR-MBAO was ranked second in three datasets (i.e. Colon, Breast, and SRBCT) and third in five datasets (i.e. CNS, Ovarian, Leukemia_3c, Leukemia_4c, and ALL-AML) between the algorithms compared in terms of the number of genes chosen and their accuracy. In the MLL dataset, mRMR-MBAO was ranked fourth among algorithms although it achieves 99.69% accuracy with 11 gene numbers.

In sum, it can be observed that the mRMR-MBAO has a classification rate of more than 88%. In terms of the number of selected genes, the mRMR-MBAO was able to obtain fewer than 24 for eleven datasets on average. Furthermore, mRMR-MBAO appears to be competitive and in some cases superior to state-of-arts in the GS problem.

4.5. Biological interpretation

From a biological standpoint, only a small number of genes in microarray data are important for cancer diagnosis (Pashaei and Pashaei, 2021a; Shukla et al., 2018). The suggested strategy seeks to find the smallest gene subset with the highest classification accuracy. It is critical to analyze these genes by determining the obtained genes, as well as their impact and biological meaning. Table 8 shows the best subset of gens obtained by using the proposed method in each dataset. As can be seen in Table 8, the SVM classifier achieves high accuracy and TPR on the selected optimal gene subsets (minimum 92%) on eleven high-dimensional microarray data. Also, some of the obtained genes are common with others reported genes in the literature, for example, the 24107 gene index in the Breast cancer dataset has been reported before as a marker gene in (Pashaei and Pashaei, 2021a) study. We can observe that the proposed mRMR-MBAO can identify and select informative genes. Fig. 9 displays a heatmap created for the identified best subset of gens to show their expression levels. The heatmap correctly clusters samples and reorders the genes into blocks with similar expression patterns.

5. Conclusion

The high-dimensional, complex, and noisy data pose a great challenge to recognize disease-related gene expression patterns embedded in the microarray datasets. Finding a small set of biologically meaningful

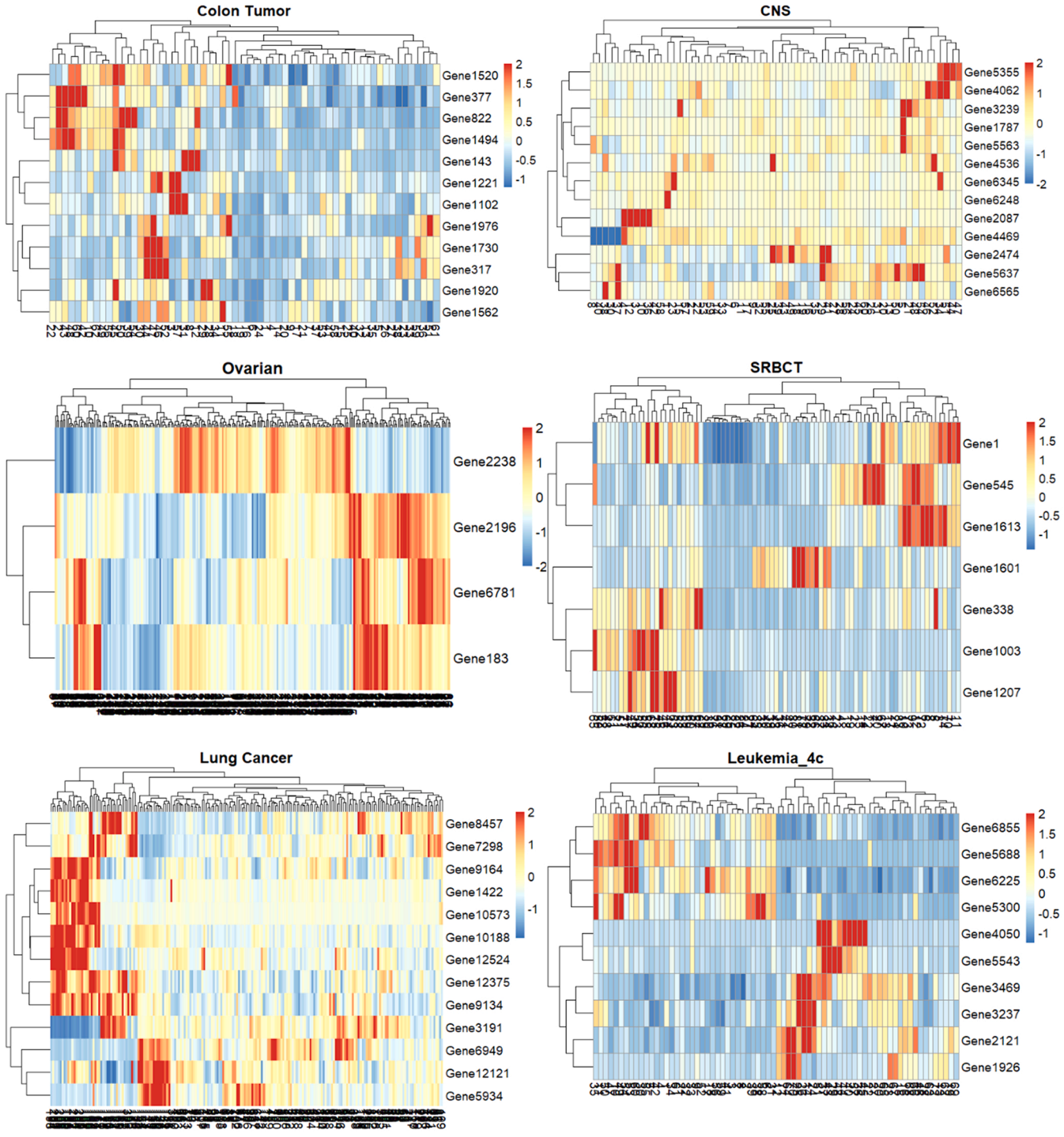


Fig. 9. The heat map of the actual expression profiles for the best subset of genes produced by the proposed method.

genes is, therefore, necessary but a difficult issue in microarray expression data analysis. In this paper, a new Binary version of the Aquila Optimizer (BAO) algorithm is proposed for solving the GS problem. The mutation mechanism and TVMS transfer function are combined with BAO to develop a novel algorithm, called MBAO, to efficiently reduce the dimensionality of the microarray dataset. To our knowledge, this is the first time the AO has been employed to solve the FS problem in high-dimensional microarray datasets.

The suggested hybrid approach works as follows. First, the mRMR filter approach is used to generate a robust gene list from the input dataset in order to achieve high classification accuracy. Second, the proposed MBAO with SVM classifier is used to find the best gene subset from the most informative genes obtained by the mRMR approach. To make BAO more suited for the complex gene selection search space, the mutation mechanism was added to the BAO algorithm to enlarge the searching range and avoid local minima. The TVMS transfer function was also used to convert AO from continuous to binary, which aids in finding the ideal balance between exploration and exploitation. To evaluate the suggested approach, eleven well-known benchmark microarray datasets were used. The performance of MBAO was deeply

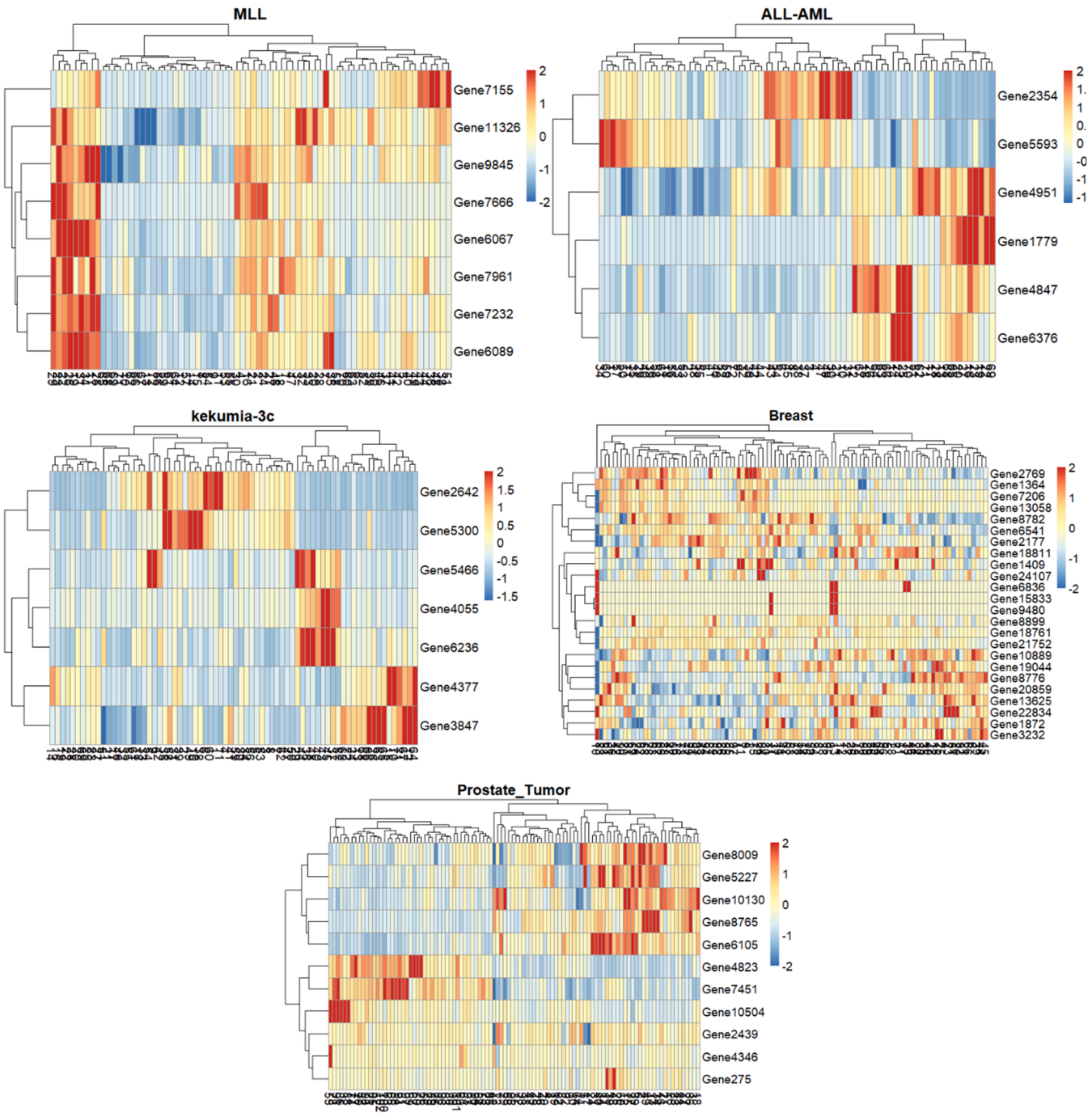


Fig. 9. (*continued*).

examined in comparison with other current state-of-art methods using several evaluation measures, such as the fitness values, classification accuracy, and the number of selected genes. Simulating results show that MBAO has better performance than BAO and almost all state-of-art methods due to its good ability to strike a balance between exploration and exploitation while avoiding the local optimum.

For future works, mutation-based AO can be applied to solve other optimization and real-world problems such as clustering, sentiment analysis, and intrusion detection. Furthermore, other NIOAs can be combined with BOA to make the algorithm more effective when tackling various optimization problems.

CRediT authorship contribution statement

Elham Pashaei designed the model and the computational framework. She carried out the implementation, performed the experiment, and wrote the manuscript.

Declaration of Competing Interest

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

References

Abd Elaziz, M., Dahou, A., Alsaleh, N.A., et al., 2021. Boosting COVID-19 image classification using MobileNetV3 and aquila optimizer algorithm. Entropy 23, 1383. https://doi.org/10.3390/E23111383.

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.

Abualigah, L., Yousri, D., Abd Elaziz, M., et al., 2021. Aquila optimizer: a novel meta-heuristic optimization algorithm. Comput. Ind. Eng. 157, 107250 https://doi.org/10.1016/J.CIE.2021.107250.

Ahmed, M.S., Shahjaman, M., Rana, M.M., Mollah, M.N.H., 2017. Robustification of Naïve Bayes classifier and its application for microarray gene expression data analysis. BioMed Res. Int. 2017, 3020627. https://doi.org/10.1155/2017/3020627.

Alanni, R., Hou, J., Azzawi, H., Xiang, Y., 2019. Deep gene selection method to select genes from microarray datasets for cancer classification. BMC Bioinform. 20, 1–15. https://doi.org/10.1186/S12859-019-3161-2/FIGURES/5.

Alomari, O.A., Khader, A.T., Al-Betar, M.A., Awadallah, M.A., 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.

Alomari, O.A., Makhadmeh, S.N., Al-Betar, M.A., 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.

Alrassas, A.M., Al-Qaness, M.A.A., Ewees, A.A., et al., 2021. Optimized ANFIS model using aquila optimizer for oil production forecasting. Processes 9, 1194. https://doi.org/10.3390/PR9071194.

Baliarsingh, S.K., 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.

Beheshti, Z., 2020. A time-varying mirrored S-shaped transfer function for binary particle swarm optimization. Inf. Sci. 512, 1503–1542. https://doi.org/10.1016/J.INS.2019.10.029.

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.

Chaudhuri, A., Sahu, T.P., 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.

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.

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.

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.

Elgamal, Z.M., Yasin, N.B.M., Tubishat, M., et al., 2020. An improved Harris Hawks optimization algorithm with simulated annealing for feature selection in the medical field. IEEE Access 8, 186638–186652. https://doi.org/10.1109/ACCESS.2020.3029728.

Fatani, A., Dahou, A., Al-qaness, M.A.A., et al., 2021. Advanced feature extraction and selection approach using deep learning and aquila optimizer for IoT intrusion detection system. Sensors 22, 140. https://doi.org/10.3390/S22010140.

Hammouri, A.I., Mafarja, M., Al-Betar, M.A., et al., 2020. An improved Dragonfly algorithm for feature selection. Knowl. Based Syst. 203, 106131 https://doi.org/10.1016/j.knosys.2020.106131.

Hu, B., Dai, Y., Su, Y., et al., 2018. Feature selection for optimized high-dimensional biomedical data using an improved shuffled frog leaping algorithm. IEEE/ACM Trans. Comput. Biol. Bioinform. 15, 1765–1773. https://doi.org/10.1109/TCBB.2016.2602263.

Jain, I., Jain, V.K., 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.

Kanti Ghosh, K., Begum, S., Sardar, A., et al., 2021. Theoretical and empirical analysis of filter ranking methods: experimental study on benchmark DNA microarray data. Expert Syst. Appl. 169, 114485 https://doi.org/10.1016/J.ESWA.2020.114485.

Nguyen, T., Khosravi, A., Creighton, D., Nahavandi, S., 2015. Hierarchical gene selection and genetic fuzzy system for cancer microarray data classification. PLoS One 10, e0120364. https://doi.org/10.1371/JOURNAL.PONE.0120364.

Pashaei, E., Aydin, N., 2018. Markovian encoding models in human splice site recognition using SVM. Comput. Biol. Chem. 73, 159–170. https://doi.org/10.1016/j.compbiolchem.2018.02.005.

Pashaei, E., Pashaei, E., 2021a. 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.

Pashaei, E., Pashaei, E., 2021b. Training feedforward neural network using enhanced black hole algorithm: a case study on COVID-19 related ACE2 gene expression classification. Arab J. Sci. Eng. 46, 3807–3828. https://doi.org/10.1007/s13369-020-05217-8.

Pashaei, E., Pashaei, E., 2022. An efficient binary chimp optimization algorithm for feature selection in biomedical data classification. Neural Comput. Appl. 34, 6427–6451. https://doi.org/10.1007/S00521-021-06775-0.

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.

Pashaei, E., Ozen, M., Aydin, N., 2016a. Biomarker discovery based on BBHA and AdaboostM1 on microarray data for cancer classification. In: Proceedings of the 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS). IEEE, pp. 3080–3083.

Pashaei, E., Ozen, M., Aydin, N., 2016b. Gene selection and classification approach for microarray data based on Random Forest Ranking and BBHA. In: Proceedings of the 3rd IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2016. Institute of Electrical and Electronics Engineers Inc., pp. 308–311.

Pashaei, E., Yilmaz, A., Aydin, N., 2016c. A combined SVM and Markov model approach for splice site identification. In: Proceedings of the 6th International Conference on Computer and Knowledge Engineering (ICCKE 2016). IEEE, pp. 200–204.

Pashaei, E., Pashaei, E., 2019. Gene selection using intelligent dynamic genetic algorithm and random forest. In: Proceedings of the 11th International Conference on Electrical and Electronics Engineering (ELECO). IEEE, pp. 470–474.

Pashaei, E., Pashaei, E., 2020. Gene selection for cancer classification using a new hybrid of binary black hole algorithm. In: Proceedings of the 28th IEEE Conference on Signal Processing and Communications Applications (SIU2020). IEEE, pp. 1–4.

Radovic, M., Ghalwash, M., Filipovic, N., Obradovic, Z., 2017. Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. BMC Bioinform. 18, 9. https://doi.org/10.1186/s12859-016-1423-9.

Rajinikanth, V., Aslam, S.M., Kadry, S., Thinnukool, O., 2022. Semi/fully-automated segmentation of gastric-polyp using aquila-optimization-algorithm enhanced images. Comput. Mater. Contin. 70, 4087–4105. https://doi.org/10.32604/CMC.2022.019786.

Sánchez-Maroño, N., Fontenla-Romero, O., Pérez-Sánchez, B., et al., 2019. Classification of microarray data. In: Bolón-Canedo, V., Alonso-Betanzos, A. (Eds.), Microarray Bioinformatics. Springer New York, New York, NY, pp. 185–205.

Shreem, S.S., Ahmad Nazri, M.Z., Abdullah, S., Sani, N.S., 2022. Hybrid symmetrical uncertainty and reference set harmony search algorithm for gene selection problem. Mathematics 10, 374. https://doi.org/10.3390/MATH10030374.

Shukla, A.K., Tripathi, D., 2019. Identification of potential biomarkers on microarray data using distributed gene selection approach. Math. Biosci. 315, 108230 https://doi.org/10.1016/J.MBS.2019.108230.

Shukla, A.K., 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.

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.

Wang, S., Jia, H., Abualigah, L., et al., 2021. An improved hybrid aquila optimizer and Harris Hawks algorithm for solving industrial engineering optimization problems. Processes 9, 1551. https://doi.org/10.3390/PR9091551.

Yan, C., Ma, J., Luo, H., et al., 2019a. A novel feature selection method for high-dimensional biomedical data based on an improved binary clonal flower pollination algorithm. Hum. Hered. 84, 34–46. https://doi.org/10.1159/000501652.

Yan, C., Ma, J., Luo, H., Patel, A., 2019b. 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.

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.