

Article

Optimization of Gene Selection for Cancer Classification in High-Dimensional Data Using an Improved African Vultures Algorithm

Mona G. Gafar ^{1,2,*} , Amr A. Abohany ^{3,†} , Ahmed E. Elkhoully ^{4,†} and Amr A. Abd El-Mageed ^{5,†} 

- ¹ Department of Computer Engineering and Information, College of Engineering in Wadi Alddawasir, Prince Sattam bin Abdulaziz University, Kharj 16278, Saudi Arabia
- ² Machine Learning and Information Retrieval Department, Artificial Intelligence, Kafrelsheikh University, Kafrelsheikh 33511, Egypt
- ³ Faculty of Computers and Information, Kafrelsheikh University, Kafrelsheikh 33511, Egypt; amrabohany8@gmail.com
- ⁴ Department of Biomedical Engineering, Faculty of Electrical Engineering, Menofia University, Menofia 32951, Egypt; eng.elkholy.ahmed@gmail.com
- ⁵ Department of Information Systems, Sohag University, Sohag 82511, Egypt; amr.atf@commerce.sohag.edu.eg
- * Correspondence: m.gafar@psau.edu.sa or dr.mona.gamal85@gmail.com
- † These authors contributed equally to this work.

Abstract: This study presents a novel method, termed RBAVO-DE (Relief Binary African Vultures Optimization based on Differential Evolution), aimed at addressing the Gene Selection (GS) challenge in high-dimensional RNA-Seq data, specifically the rnaseqv2 IlluminaHiSeq rnaseqv2 un edu Level 3 RSEM genes normalized dataset, which contains over 20,000 genes. RNA Sequencing (RNA-Seq) is a transformative approach that enables the comprehensive quantification and characterization of gene expressions, surpassing the capabilities of micro-array technologies by offering a more detailed view of RNA-Seq gene expression data. Quantitative gene expression analysis can be pivotal in identifying genes that differentiate normal from malignant tissues. However, managing these high-dimensional dense matrix data presents significant challenges. The RBAVO-DE algorithm is designed to meticulously select the most informative genes from a dataset comprising more than 20,000 genes and assess their relevance across twenty-two cancer datasets. To determine the effectiveness of the selected genes, this study employs the Support Vector Machine (SVM) and k-Nearest Neighbor (*k*-NN) classifiers. Compared to binary versions of widely recognized meta-heuristic algorithms, RBAVO-DE demonstrates superior performance. According to Wilcoxon's rank-sum test, with a 5% significance level, RBAVO-DE achieves up to 100% classification accuracy and reduces the feature size by up to 98% in most of the twenty-two cancer datasets examined. This advancement underscores the potential of RBAVO-DE to enhance the precision of gene selection for cancer research, thereby facilitating more accurate and efficient identification of key genetic markers.

Keywords: RNA sequencing; gene selection; high dimensionality; meta-heuristic algorithms; African vultures algorithm; differential evolution; relief technique



Citation: Gafar, M.G.; Abohany, A.A.; Elkhoully, A.E.; Abd El-Mageed, A.A. Optimization of Gene Selection for Cancer Classification in High-Dimensional Data Using an Improved African Vultures Algorithm. *Algorithms* **2024**, *17*, 342. <https://doi.org/10.3390/a17080342>

Academic Editors: Muhammad Adnan Khan and Frank Werner

Received: 27 April 2024

Revised: 16 July 2024

Accepted: 1 August 2024

Published: 6 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Deoxyribonucleic Acid (DNA) makes up our genetic code, containing the recipe for our existence. Although every cell has the same DNA, each tissue structure is unique and serves a distinct purpose. The RNA transcription mechanism determines which genes in a cell are active, enabling RNA to be transformed into proteins responsible for the cell's structure and functionality [1]. We analyze RNA-Seq gene expression data (GED) to determine genetic changes and evaluate disease biomarkers. Differential expression analysis identifies quantitative distinctions in gene expression, allowing us to categorize genes whose expression changes under various conditions. This method helps us understand

illnesses and find ways to manage them. Gene expression profiling technologies have significantly developed over the years [2].

Two popular techniques are micro-array, which uses a hybridization-based approach, and RNA-Seq, which is based on next-generation sequencing. These technologies have demonstrated their value in advancing our understanding of gene expression [3]. Notable studies, such as those by Chen et al. [4] and Nunez et al. [5], have highlighted the potential of these techniques, while Wang et al. [6] have provided valuable insights into the RNA-Seq approach.

The two methods under consideration serve the purpose of quantifying gene expression for classification and statistical investigation. This paper has chosen the quantification data obtained through the next-generation sequencing (NGS)-based RNA-Seq approach because it offers more accuracy in detecting RNA quantification levels compared to micro-array data [7]. RNA-Seq has overcome several limitations of micro-array analysis, such as dependence on prior sequencing knowledge, which restricted the detection range [8,9]. Unlike micro-arrays, RNA-Seq does not require any previous knowledge and significantly expands the dynamic detection range. This enhancement improves the accuracy of results and enables the identification of a comprehensive set of genes, providing a precise understanding of disease biomarkers. As a result, RNA-Seq has emerged as a potent tool for researchers, offering valuable insights into the underlying molecular mechanisms of various diseases [10].

The “dimensionality curse” [11] has become a common challenge in contemporary times owing to the abundance of available data. This has caused a surge in the development of feature selection (FS) techniques and algorithms. FS algorithms can be classified into four distinct methods: filter, wrapper, embedded, and hybrid [12,13]. These approaches aim to identify the most informative features that can distinguish between different classes. In our case, we are concerned with identifying the genes linked to tumors.

The filter approach is a widely used technique in FS that involves evaluating the relevance of individual genes based on statistical scores. This method is known for its high accuracy in selecting the most satisfactory group of genes. Nevertheless, the filter approach has some limitations, as functioning on each gene individually ignores the interrelationships between genes, which can result in a local optima issue [14]. It is worth noting that the filter approach can be categorized into two sub-types—univariate and multivariate—with the latter considering the correlations between genes. Some examples of the filter approach include Relief [15], Fisher score [16], *t*-test [17], and information gain [18].

The wrapper method investigates all feasible gene subsets to test and create a subset of genes to determine their implications. A specific classifier is employed to determine the outcome of each subset, and the categorization technique is used multiple times for each assessment. Compared to the filter approach, the wrapper approach delivers superior performance by using a categorization technique that directs the learning procedure. However, this method requires substantial time and computational resources, especially when dealing with large-scale data [19].

Meta-heuristic methods (MHMs) are high-level heuristics used in mathematical optimization and computer science to find satisfactory solutions to optimization problems when information is imperfect or resources are limited [20]. MHMs sample a subset of solutions, making them useful for various optimization issues due to their few assumptions about the problem [21]. However, they do not guarantee globally optimal solutions. Many MHMs employ stochastic optimization, meaning the solution is based on random variables generated during the process [22].

Meta-heuristic methods (MHMs) are more practical than traditional iterative methods and optimization algorithms because they can explore a broader range of possible solutions. Consequently, MHMs have become a preferred strategy for solving optimization problems [23]. Numerous research papers have demonstrated that among the different wrapper methods, MHMs are well suited to address the feature selection (FS) issue. Stochastic methods, including MHMs, can produce optimal or near-optimal results quickly. They

offer benefits such as flexibility, self-management without detailed mathematical properties, and the ability to assess numerous outcomes simultaneously. Various MHMs have recently been developed to solve the FS problem, providing reliable near-optimal solutions at significantly reduced computational costs [24].

Embedded FS methods utilize a learning technique to select appropriate genes that interact with the classification process. This method combines the FS technique as part of the learning model. The learning algorithm is trained with an initial attribute subset to estimate a measure for evaluating the rank values of attributes [25]. The ultimate goal is to decrease the computation time for reclassifying different subsets by incorporating the FS stage into the training process. Some techniques in this approach perform feature weighting based on regularization models with objective functions that minimize fitting errors while enforcing feature coefficients to be small or precisely zero. Examples of embedded methods are the First-Order Inductive Learner (FOIL) rule-based feature subset selection algorithm and SVM based on Recursive Feature Elimination (SVM-RFE) [26].

The hybrid method is a well-crafted technique that amalgamates the filter and wrapper methods to leverage the strengths of each. This approach begins with a filter that reduces the feature space dimensionality, generating multiple subsets with intermediate complexity. Subsequently, a wrapper is employed as a learning technique to choose the most suitable candidate subset [27]. The hybrid approach integrates the accuracy of wrappers and the efficiency of filters, resulting in an optimal methodology [28,29].

1.1. Motivation and Contributions

An effective global meta-heuristic optimization algorithm is presented in this paper, which is called the African Vultures Optimization (AVO) algorithm [30], which imitates the living and eating habits of African vultures. The proposed algorithm consists of four basic phases: population division into three groups—the best solution, the second-best solution, and the remaining solutions; famine-level measurement to formulate a mathematical model for the vultures' exploitation, exploration, and transfer; exploration of vultures employing two strategies to allow vultures to cover large distances over extended periods to find food at random sites; and exploitation of vultures involving two sub-phases. Two distinct strategies are used in the first sub-phase: siege fight for strong vultures and rotational flight, which forms a spiral activity between the outstanding vulture and the rest. Two strategies are utilized in the second sub-phase: assembling vultures around the food source and a hostile siege fight, where vultures become more aggressive and attempt to steal food left behind by healthy vultures.

This study proposes an enhanced binary variant of the AVO technique, namely the RBAVO-DE technique, which is a productive approach that demonstrates accurate performance in handling the GS problem. At first, there is a high possibility that the recommended technique will steer clear of local optima and accomplish sufficient examination precision, quick convergence, and improved stability. Compared to recent MHMs, the proposed RBAVO-DE technique achieves enhanced efficacy by producing optimal or substantially optimal solutions for many of the analyzed situations. The Relief algorithm is utilized to identify only the related features to produce the final classification dataset. RBAVO-DE combines the Relief algorithm with the DE approach to increase exploration capability and obtain the best results within the solution space via repetitions. It also utilizes a transfer function to transform real position values into binary values. The RBAVO-DE approach makes sense in GS because it is simple to comprehend and implement, can deal with various optimization problems, produces valuable results in an acceptable amount of time, requires less computing power, and uses a small number of control parameters. This paper's primary contributions are as follows:

- Level 3 data based on next-generation sequencing (RNA-Seq) have been pre-processed.
- The ability of the proposed AVO meta-heuristic algorithm to address the GS problem has not been investigated, as RNA-Seq GED has never been used with the AVO algorithm.

- To construct a binary version known as the RBAVO-DE algorithm, the AVO algorithm is altered and recreated.
- The proposed RBAVO-DE algorithm combines the binary variant of the AVO with a Relief approach and DE to improve the exploration ability of the search space and enhance the achieved optimal results.
- The proposed RBAVO-DE algorithm is applied to RNA-Seq GED for the first time.
- Several performance indicators, including average fitness, classification accuracy, number of selected genes, precision, recall, and F1-score, are used to assess the outcomes.
- A comparison is made between the impact of the presented RBAVO-DE algorithm, employing the two recommended ML classifiers (SVM and k -NN), and other algorithms in the literature.
- Twenty-two distinct cancer datasets are used to assess the proposed RBAVO-DE method, and the results are shown.
- The chosen genes are investigated using biomarkers associated with cancer.

1.2. Structure

This paper is structured into five main sections. Section 2 reviews previous research on FS using RNA-Seq GED. This is followed by Section 3, which offers an in-depth discussion of the RBAVO-DE algorithm, an enhanced version of AVO, including its parameters for addressing GS. Section 4 showcases the experimental findings compared to several recent MHMs. Lastly, Section 6 concludes this study and proposes avenues for future investigation.

2. Related Works

In this section, we discuss the literature that focuses on the methods researchers use to classify RNA-Seq GED, which typically has high dimensionality. To achieve optimal performance of classification methods, it is crucial to disregard irrelevant and unrelated genes; hence, selecting appropriate genes is a vital stage before utilizing ML and deep learning (DL) techniques [31] or any other classification techniques. In this regard, we have explored some relevant papers in this domain to accomplish the objective of RNA-Seq categorization for cancer identification.

Yaqoob et al. [32] introduced a cutting-edge technique for GS known as the Sine-Cosine-Cuckoo Search Algorithm (SCCSA), a hybrid method tailored to function alongside established ML models such as SVM. This innovative GS algorithm was evaluated using a breast cancer benchmark dataset, where its performance was meticulously analyzed and compared with other GS methodologies. To refine the selection of features, the minimum Redundancy Maximum Relevance method was initially applied as a preliminary filtering step. Subsequently, the hybrid SCCSA approach was deployed to further improve and fine-tune the GS process. The final stage involved using the SVM classifier to classify the dataset based on the selected genes. Considering the critical importance of GS in decoding complex biological datasets, SCCSA emerges as a crucial asset for the classification of cancer-related datasets.

Joshi et al. [33] introduced an innovative optimization strategy named PSO-CS integrated with DL for brain tumor classification. This method enhances the efficiency of the Particle Swarm Optimization (PSO) technique by incorporating the Cuckoo Search (CS) algorithm, optimizing the classification process. Following this optimization, PSO-CS utilizes DL to classify GED related to brain tumors, identifying various classes associated with specific tumors alongside the PSO-CS optimization method. By integrating the PSO-CS technique with DL, it significantly outperformed other DL and ML models regarding classification accuracy, as evidenced by various performance measures.

Mahto et al. [34] unveiled a groundbreaking approach for cancer classification through an integrated method based on CS-SMO (CS and Spider Monkey Optimization) for GS. Initially, the fitness function of the Spider Monkey Optimization (SMO) algorithm is modified using the CS algorithm. This modification leverages the strengths of both MHMs

to identify a subset of genes capable of predicting cancer at an early stage. To further refine the accuracy of the CS-SMO algorithm, a pre-processing step, MRMR, is employed to reduce the complexity of cancer gene expression datasets. Subsequently, these gene subsets are processed using DL to classify different cancer types. The efficacy of the CS-SMO approach coupled with DL was evaluated using eight benchmark micro-array gene expression datasets for cancer, examining its performance across various measures. The CS-SMO method integrated with DL demonstrated superior classification accuracy across all examined large-scale gene expression datasets for cancer, outperforming existing DL and ML models.

Neggaz et al. [35] introduced an improved version of the manta ray foraging optimization, called MRFO-SC, which used trigonometric operators inspired by the sine-cosine (SC) algorithm to handle the GS issue. The k -NN model was used for gene-set selection. In addition, the statistical significance of the MRFO-SC was evaluated using Wilcoxon's rank-sum test at a 5% significance level. The results were evaluated and compared with some recent MHMs. The comparison and experimental results confirmed the effective performance of the proposed MRFO-SC on high- and low-dimensional benchmark datasets by obtaining the greatest classification accuracy on 85% of the GS benchmark datasets.

Lyu et al. [36] explored cancer biomarkers by focusing on genes' significance in their impact on classification. They followed a two-phase approach—data pre-processing and utilizing a convolutional neural network—to classify the type of tumor. In the second phase, they created heat maps for each category to identify genes related to pixels with the highest intensities in the heat maps. They then evaluated the selected genes' pathways. During pre-processing, they removed the levels of gene expression that had not been modified during the GS phase, using a variance threshold of 1.19, which decreased the number of genes from 19,531 to 10,381. The final classification accuracy obtained was 95.59, which is good but could still be improved using a more effective FS methodology to further decrease data dimensionality.

Khalifa et al. [37] built on the aforementioned paper [36]. The focus of their study was on five types of cancer data: Uterine Corpus Endometrial Carcinoma (UCEC), Lung Squamous Cell Carcinoma (LUSC), Lung Adenocarcinoma (LUAD), Kidney Renal Clear Cell Carcinoma (KIRC), and Breast Invasive Carcinoma (BRCA). The benchmark dataset used for the study comprised 2086 records and 972 attributes. Each record provided detailed sample information, while each attribute included the RNA-Seq values for a specific gene, represented as RPKM (Reads Per Kilobase per Million) [38]. The researchers employed a mixed approach using binary PSO with the decision trees (BPSO-DT) method to pre-process the data. Out of 971 attributes, 615 were selected as the best attributes of RNA-Seq. The proposed method achieved an overall testing classification accuracy of 96.90%, as demonstrated by the suggested outcomes and evaluation measures used.

Xiao et al. [39] assessed their methodology using three RNA-Seq datasets: Stomach Adenocarcinoma (STAD), BRCA, and LUAD. Their approach relied on DL techniques, wherein they employed five classifiers and subsequently utilized the DL technique to ensemble each output of the five classifiers. This led to an improvement in the classification accuracy of all the predictions, with the BRCA dataset achieving 98.4% accuracy, STAD achieving 98.78% accuracy, and LUAD achieving 99.20% accuracy.

Liu et al. [40] used micro-array data and a hybrid approach to address the problem. Unlike the previously mentioned papers, they studied each cancer class separately. To assess performance, they employed four gene benchmarks related to small round blue cell tumors, colon cancer, lung cancer, and leukemia. Their proposed method used Relief as the pre-processing technique to eliminate genes with a lower correlation with the specific cancer class, followed by PSO as the search technique. Finally, they employed the SVM model to evaluate the classification accuracy of the selected subset of genes and obtain the conclusive optimum gene subset for each cancer type.

Based on the existing research, it seems that most studies using RNA-Seq GED are still in their early stages, with researchers attempting to implement and test various ideas

in this promising area. While the literature contains a plethora of experiments employing multiple techniques, such as FS and DL recent methods, no single technique is perfect due to the high dimensionality of RNA-Seq GED. FS of RNA-Seq GED plays a crucial role in determining the relationship between a gene and its category. It is a critical pre-processing task for validating gene biomarkers of cancer and overcoming the dimensionality curse. Consequently, this study aims to introduce a new wrapper approach, the RBAVO-DE algorithm, and apply it to RNA-Seq data for the first time. It also compares the proposed algorithm's effectiveness with that of other FS techniques.

3. Proposed RBAVO-DE for GS

An improved variant of AVO known as RBAVO-DE is proposed in this paper to discover the smallest relevant gene subsets from the classification process and to disregard irrelevant genes. RBAVO-DE utilizes a Relief-based binary AVO algorithm combined with the DE technique. RBAVO-DE's primary feature is its ability to maximize accuracy while utilizing the fewest features possible. The proposed RBAVO-DE consists of two primary steps. First, there is a pre-processing step in which the Relief algorithm is used to determine which features are significant by assigning each feature a weight and then removing the features that are irrelevant and have the lowest weights. In the second step, the binary AVO algorithm and the DE technique are applied to identify the more pertinent and unique features. The AVO algorithm is prone to the local optimum trap when handling large-scale problems. The AVO algorithm incorporates the DE technique to prevent this.

The proposed RBAVO-DE algorithm for tackling the GS strategy requires applying the Relief algorithm, initializing, position boosting using the AVO algorithm, binary conversion, fitness appraisal, and integration with DE. The following subsections explain these steps.

3.1. Applying the Relief Algorithm for Feature Filtration

This step aims to pre-process the population using the Relief algorithm [41], which is considered a fast, easy, and efficient filtering technique for finding features related to one another. This algorithm's primary goal is to find characteristics that differentiate sample values and group similar samples close together. As a result, the method depends on the weighted ranking of features, where a feature with a higher weight indicates better classification performance.

After choosing a sample randomly, the Relief algorithm examines two different kinds of closest samples: near-hit samples, which are associated with samples from the same class, and near-miss samples, which are associated with samples from other classes. The near-hit and near-miss values can be used to determine the features' weights. To assess the importance in the classification process, the features' weights are arranged from most significant to least significant. Finally, the features with the most significant weights are selected. The following formula can be used to determine the weight W for feature A :

$$W_A = \sum_{j=1}^N \left(x_A^j - NM(x^j)_A \right)^2 - \left(x_A^j - NH(x^j)_A \right)^2. \quad (1)$$

where W_A denotes the weight of feature A , x_A^j denotes the value of feature A for data x^j , and N denotes the number of samples. The nearest data points to x^j in the same or different classes are $NH(x^j)$ and $NM(x^j)$.

The Relief algorithm narrows its focus to only the necessary features and minimizes the search area to help the AVO algorithm find better features more quickly.

3.2. Initializing the Population

The proposed BAVO algorithm starts by randomly generating a population of N positions. A vector of dimensions D equivalent to the number of features in the original dataset describes each position as a possible solution at its constrained lower and upper

bounds. This randomly initialized step, bounded within the $[-1, 1]$ range for each position vector's variable, uses the randomly produced position.

3.3. Boosting Positions via the AVO Algorithm

This paper presents the AVO algorithm [30], a meta-heuristic optimization algorithm inspired by vultures in Africa and their living and feeding behaviors. Based on fundamental ideas about vultures, the AVO algorithm is configured as follows: The African vulture population size is initially presumed by the AVO algorithm to consist of N vultures. The population of African vultures is then divided into three groups to reflect the primary natural function of vultures, based on the computed fitness function. The first group consists of the strongest vulture, which is the best solution; the second group contains a vulture weaker than the first, which is the second-best solution; and the final group encompasses the remaining weaker vultures, which are the worst solutions.

Based on the above, the proposed AVO algorithm is composed of four fundamental phases to simulate the behavior of different types of vultures. The following subsections clarify these phases.

3.3.1. Phase of Dividing the Population

The initial population is split into groups by assessing the solution's fitness function. The best solution is chosen as the best vulture in the first group, while the second group contains the second-best solution. The third group contains the remaining solutions. As the solutions constantly strive to approach the best and second-best solutions, the population needs to be re-evaluated for each iteration, as follows:

$$R^g = \begin{cases} BestVulture_1^g, & \text{if } pr^g = L_1, \\ SecondBestVulture_2^g, & \text{if } pr^g = L_2, \end{cases} \quad (2)$$

$$pr^g = \frac{v^g}{\sum_{g=1}^N v^g}. \quad (3)$$

The best vulture in the first group and the second-best vulture in the second group at the g^{th} iteration are denoted by the expressions $BestVulture_1^g$ and $SecondBestVulture_2^g$, respectively. The probability of selecting the best solution for each group at the g^{th} iteration is defined using the roulette wheel approach, as shown in Equation (3), pr^g . Two random parameters within the range $[0, 1]$ are L_1 and L_2 .

3.3.2. Phase of Measuring the Famine Level

This phase is utilized to formulate a mathematical model for the exploitation, exploration, and transfer among the vultures. When they are not famished, the vultures have a greater ability to search for food and can fly further. Furthermore, when famished, vultures cannot fly long distances in search of food and may become violent. The i^{th} vulture's famine level (F_i^g) during the g^{th} iteration can be written as follows, which is used to develop a mathematical model for the vultures' exploitation, exploration, and transfer:

$$F_i^g = (2 \times rand + 1) \times z \times \left(1 - \frac{g_i}{G_{max}}\right) + t, \quad (4)$$

$$t = h \times \left(\sin^w \left(\frac{\pi}{2} \times \frac{g_i}{G_{max}}\right) + \cos \left(\frac{\pi}{2} \times \frac{g_i}{G_{max}}\right) - 1\right). \quad (5)$$

The vultures are presumably full since the variable F_i^g shows the vultures' shift from exploration to exploitation. A random number between 0 and 1 is called a *rand*. z denotes a random value within the interval $[-1, 1]$. The current iteration number is denoted by g_i , while the maximum iteration number is denoted by G_{max} . Equation (5) calculates the t value to help solve complicated optimization problems more effectively and prevent reaching a local optimum. An arbitrary value within the interval $[-2, 2]$ is represented by h . The probability of carrying out the exploration process is regulated by the predefined

constant parameter w ; the exploration likelihood grows as its value increases. Exploration is less probable as its value declines.

Equation (4) states that as the number of iterations increases, F_i^g gradually lessens. As a result, the next step in the proposed AVO algorithm can be defined as follows:

$$\begin{cases} \text{Exploration phase (Search food in diverse areas),} & \text{if } |F_i^g| \geq 1, \\ \text{Exploitation phase (Search food in surrounding area),} & \text{if } |F_i^g| < 1. \end{cases} \quad (6)$$

3.3.3. Phase of Exploration

The vultures are characterized by their great ocular ability to locate appropriate food during this phase. There are two different strategies used in this phase to enable vultures to travel great distances for lengthy periods to search for food in random locations. These locations are selected using a random number $rand_{P_1}$ and a preset parameter P_1 , both of which have values within the range $[0, 1]$. In the exploration phase, the famine level $|F_i^g|$ is greater than or equal to 1. The exploration techniques are described as follows:

$$X_i^{g+1} = \begin{cases} R^g - D_i^g \times F_i^g, & \text{if } rand_{P_1} \leq P_1, \\ R^g - F_i^g + rand \times ((UB - LB) \times rand + LB), & \text{if } rand_{P_1} > P_1, \end{cases} \text{ if } |F_i^g| \geq 1, \quad (7)$$

$$D_i^g = |(2 \times rand) \times R^g - X_i^g|. \quad (8)$$

where the upcoming updated position at the next $(g + 1)^{th}$ iteration is denoted by X_i^{g+1} . The best vulture chosen for the current iteration g is denoted by R^g . This is determined using Equation (2). The vultures move randomly to protect the food from other vultures and to provide a high degree of randomness in their search behavior. $rand$ is a random value between zero and one. The variables' upper bound is denoted by UB ; their lower bound by LB ; and the current position at the g^{th} iteration is X_i^g .

3.3.4. Phase of Exploitation

In this phase, $|F_i^g|$ is less than 1. The proposed AVO algorithm's efficacy is evaluated in two sub-phases that comprise the exploitation phase. Each of these sub-phases uses two different strategies. For each internal sub-phase, the appropriate strategy is chosen using two predefined parameters: P_2 for the first sub-phase and P_3 for the second, with values ranging from 0 to 1. These two internal sub-phases are explained as follows:

- **First sub-phase of exploitation:** This sub-phase applies two different strategies when $|F_i^g|$ is less than 1 and greater than or equal to 0.5. The selection of one of these two strategies is made using a random value $rand_{P_2}$ within the range $[0, 1]$ and a specified parameter P_2 .

The initial strategy of this sub-phase is called **siege fight**, which involves sufficiently powerful and somewhat satiated vultures. More robust and healthier vultures try not to share food with others because they convene around a single food source. By swarming near the healthy vultures and engaging in small fights, the weaker vultures try to take food from them. Conversely, the second strategy is called **rotational flight**; it creates a spiral movement between a superior vulture and the others. The following illustrates the strategies for the first exploitation sub-phase:

$$X_i^{g+1} = \begin{cases} D_i^g \times (F_i^g + rand) - d_t^g, & \text{if } rand_{P_2} \leq P_2, \\ R^g - (S_1^g + S_2^g), & \text{if } rand_{P_2} > P_2, \end{cases} \text{ if } 1 > |F_i^g| \geq 0.5, \quad (9)$$

$$d_t^g = R^g - X_i^g, \quad (10)$$

$$S_1^g = R^g \times \left(\frac{rand \times X_i^g}{2\pi} \right) \times \cos(X_i^g), \quad (11)$$

$$S_2^g = R^g \times \left(\frac{rand \times X_i^g}{2\pi} \right) \times \sin(X_i^g). \quad (12)$$

where the vulture's next updated position at the next $(g+1)^{th}$ iteration is denoted by X_i^{g+1} , D_i^g is defined using Equation (8), and $rand$ is an arbitrary value between 0 and 1. The distance d_i^g between the vulture and one of the best two vultures is estimated using Equation (10); R^g denotes the appropriate best vulture in the current g^{th} iteration, which is computed using Equation (2); S_1^g and S_2^g are computed employing Equations (11) and (12), respectively; and X_i^g denotes the current position at the g^{th} iteration.

- **Second sub-phase of exploitation:** This sub-phase is carried out when $|F_i^g|$ is less than 0.5. Various vulture species gather around the food supply and engage in many sieges and brawls during this sub-phase. This sub-phase employs two different strategies. A predefined parameter P_3 and an arbitrary value $rand_{P_3}$, with values ranging from 0 to 1, are used to decide which of these two strategies to use.

This sub-phase's initial strategy is called **assemble vultures around the food source**. In this strategy, different kinds of vultures search for food and may compete near a single source. The second strategy is known as **hostile siege fight**. In this strategy, the vultures become more aggressive and try to plunder the leftover food from the healthy vultures by flocking around them in different ways. The healthy vultures, on the other hand, deteriorate and are unable to fend off the other vultures. The following illustrates the strategies for the second exploitation sub-phase:

$$X_i^{g+1} = \begin{cases} \frac{A_1^g + A_2^g}{2}, & \text{if } rand_{P_3} \leq P_3, \\ R^g - |d_i^g| \times F_i^g \times Levy_d, & \text{if } rand_{P_3} > P_3, \end{cases} \text{ if } |F_i^g| < 0.5, \quad (13)$$

$$A_1^g = BestVulture_1^g - \frac{BestVulture_1^g \times X_i^g}{BestVulture_1^g - (X_i^g)^2} \times F_i^g, \quad (14)$$

$$A_2^g = SecondBestVulture_2^g - \frac{SecondBestVulture_2^g \times X_i^g}{SecondBestVulture_2^g - (X_i^g)^2} \times F_i^g, \quad (15)$$

$$Levy_d = 0.01 \times \frac{\mu \times \sigma}{|v|^{\frac{1}{\beta}}}, \quad (16)$$

$$sigma = \left(\frac{\Gamma(1+\beta) \times \sin(\frac{\Pi\beta}{2})}{\Gamma(1+\beta/2) \times \beta \times 2^{\frac{\beta-1}{2}}} \right)^{\frac{1}{\beta}}. \quad (17)$$

where the next updated position of the vulture at the next $(g+1)^{th}$ iteration, reflecting the assembly of vultures, is denoted by X_i^{g+1} . A_1^g and A_2^g are evaluated using, respectively, Equations (14) and (15). To increase the effectiveness of the AVO algorithm, $Levy_d$ is the levy flight distribution function obtained using Equation (16). d is the dimensional space; σ is defined by Equation (17), where $\beta = 1.5$ is a constant value; and μ and v are random numbers distributed equally within the range $[0, 1]$.

3.4. Converting to Binary Nature

In the presented AVO algorithm, the positions are shown as real values. As such, they are not immediately applicable to the GS binary issue. Therefore, there is a need to convert these real position values into binary values to conform to GS's binary nature and maintain the original algorithm's structure. In this conversion, 1s represent the real values of the pertinent selected genes in the binarization vector, whereas 0s represent the real values of the unselected genes, which are not pertinent. At each iteration g , the following mathematical expression can be used to convert the real position X_i^g to a binary position $(X_i^g)_{bin}$:

$$(X_i^g)_{bin} = \begin{cases} 1 & \text{if } X_i^g > \delta, \\ 0 & \text{otherwise.} \end{cases} \quad (18)$$

where an arbitrary threshold point within $[0, 1]$ is represented by δ . According to this fundamental binary conversion approach, the binary “1” (selected feature) replaces its real value if $(X_i^g)_{bin}$ is greater than δ . On the other hand, if $(X_i^g)_{bin}$ is smaller than δ , its real value is set to the binary “0” (a feature that was not chosen).

3.5. Appraising the Fitness Function Value

Finding the fewest number of selected features and optimizing the classification accuracy of the available classifiers (k -NN and SVM models) are two conflicting objectives that should be balanced to achieve the best solution and determine its quality. Since the k -NN and SVM classifiers’ accuracies might be hampered if the number of selected features is lower than the optimal, the fitness function balances the selected features’ size and accuracy. The fitness function concentrates on lowering the classification error rate rather than accuracy, as follows:

$$fit = w_1 \times Err_{rate} + w_2 \times \frac{|feat_{picked}|}{|D|}, \quad (19)$$

$$w_1 \in [0, 1], w_2 = 1 - w_1.$$

where $feat_{picked}$ denotes the number of selected features, D denotes the total number of features in the dataset, and Err_{rate} represents the classification error rate from the k -NN and SVM classifiers. The classification accuracy importance and the number of selected features are denoted by the weight parameters w_1 and w_2 , respectively. The values of w_1 (0.99) and w_2 (0.01) are determined through extensive trials conducted in previous studies [21,42,43].

3.6. Incorporating the DE Technique

DE [44] is characterized by its high efficiency and simplicity in finding a suitable solution for complex optimization problems. It can quickly produce value-added results. Three main processes—mutation, crossover, and selection—are necessary for DE. The differential mutation process seeks to produce a modified vector v_i for each iteration’s solution vector. This can be computed mathematically as follows:

$$\vec{v}_i = \vec{X}_{r_1} + W_M(\vec{X}_{r_2} - \vec{X}_{r_3}) \quad (20)$$

Within the range $[1, \text{population size}]$, three randomly selected asymmetric vectors are denoted by the symbols X_{r_1} , X_{r_2} , and X_{r_3} . W_M denotes the mutation weighting factor within the interval $[0, 1]$.

After the mutation operation, DE performs a crossover operation to increase the population’s variety. An offspring vector u_i is produced by combining values from the target vector X_i and the altered vector v_i . The most commonly used and basic factor of crossover search is binary crossover, which has the following mathematical expression:

$$u_{i,d} = \begin{cases} v_{i,d}, & \text{if } rand \leq C_R \text{ or } d = j_{rand}, \\ X_{i,d}, & \text{otherwise.} \end{cases} \quad (21)$$

where a uniformly distributed random number $j_{rand} \in [1, 2, \dots, D_X]$ is used to ensure that the mutated vector has at least one dimension. The possibility of crossing each element is calculated using the crossover rate C_R , frequently determined to be a big amount ($C_R = 0.9$).

The selection operation is then carried out, as shown in Equation (22). Here, a comparison between the fitness function $f(X_i)$ of the target vector and the corresponding offspring vector $f(u_i)$ is performed, and the lowest-valued fitness function is kept for the upcoming iteration.

$$X_i = \begin{cases} u_i, & \text{if } f(u_i) < f(X_i) \\ X_i, & \text{otherwise.} \end{cases} \quad (22)$$

3.7. The Complete RBAVO-DE Algorithm

To handle the GS strategy, the steps of the recommended RBAVO-DE algorithm are described in the following subsections. The pseudo-code for the proposed RBAVO-DE algorithm is provided in Algorithm 1. A flowchart of the proposed RBAVO-DE algorithm is shown in Figure 1, illustrating its main steps.

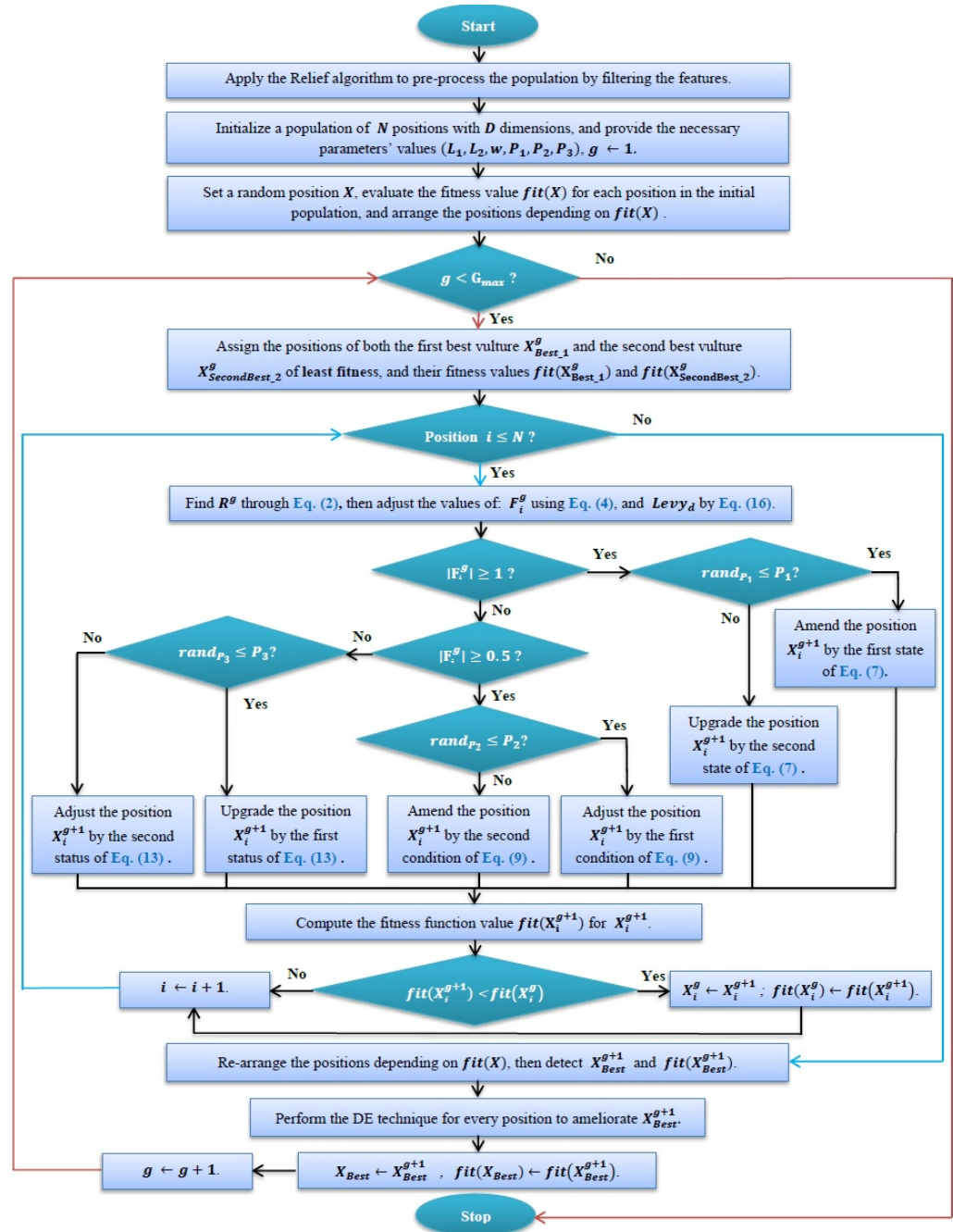


Figure 1. Flowchart of the RBAVO-DE algorithm.

Algorithm 1 The proposed RBAVO-DE algorithm.**Input:**

N —total number of positions (size of population)
 G_{max} —maximum number of permitted iterations
 D —problem's dimensional space
 LB —lower bounds of variables
 UB —upper bounds of variables
 C_R —crossover rate
 W_M —mutation weighting factor

Output:

X_{Best} —the global best vulture's position found while searching
 $fit(X_{Best})$ —the global best fitness function value found, which should be lessened

```

1: Start
2:   Apply the Relief approach for selecting the related features and filtering them, as demonstrated in Section 3.1;
3:   Initialize a population of  $N$  positions, and provide the values of the necessary parameters ( $L_1$ ,  $L_2$ ,  $w$ ,  $P_1$ ,  $P_2$ , and  $P_3$ );
4:   Set a random position  $X$  in the initial population;
5:   Evaluate the fitness values  $fit(X)$  for each position in the initial population;
6:   Arrange the positions in ascending order depending on their fitness function  $fit(X)$ ;
7:    $g \leftarrow 1$ ; ▷ Current number of iterations
8:   while  $g < G_{max}$  do
9:     Assign the positions of both the first-best vulture  $X_{Best_1}^g$  and the second-best vulture  $X_{SecondBest_2}^g$ , as well as their fitness
       values  $fit(X_{Best_1}^g)$  and  $fit(X_{SecondBest_2}^g)$ , among all positions in the population;
10:    for vulture's position  $i = 1 : N$  do
11:      Find the best vulture  $R^g$  using Equation (2);
12:      Adjust the vulture's famine level  $F_i^g$  using Equation (4);
13:      Modify the levy flight distribution function  $Levy_d$  using Equation (16);
14:      if  $|F_i^g| \geq 1$  then
15:        if  $rand_{p_1} \leq P_1$  then
16:          Amend the vulture's position  $X_i^{g+1}$  based on the first stage of the exploration phase using Equation (7);
17:        else if  $rand_{p_1} > P_1$  then
18:          Upgrade the vulture's position  $X_i^{g+1}$  based on the second stage of the exploration phase using Equation (7);
19:        end if
20:      else if  $|F_i^g| < 1$  then
21:        if  $|F_i^g| \geq 0.5$  then
22:          if  $rand_{p_2} \leq P_2$  then
23:            Adjust the vulture's position  $X_i^{g+1}$  based on the first condition of the first exploitation sub-phase using
24:            Equation (9);
25:          else if  $rand_{p_2} > P_2$  then
26:            Amend the vulture's position  $X_i^{g+1}$  based on the second condition of the first exploitation sub-phase
27:            using Equation (9);
28:          end if
29:        else if  $|F_i^g| < 0.5$  then
30:          if  $rand_{p_3} \leq P_3$  then
31:            Upgrade the vulture's position  $X_i^{g+1}$  based on the first status of the second exploitation sub-phase using
32:            Equation (13);
33:          else if  $rand_{p_3} > P_3$  then
34:            Adjust the vulture's position  $X_i^{g+1}$  based on the second status of the second exploitation sub-phase using
35:            Equation (13);
36:          end if
37:        end if
38:      end if
39:       $fit(X_i^{g+1}) \leftarrow$  Compute the fitness function value for  $X_i^{g+1}$ ;
40:      if  $fit(X_i^{g+1}) < fit(X_i^g)$  then
41:         $X_i^g \leftarrow X_i^{g+1}$ ;
42:         $fit(X_i^g) \leftarrow fit(X_i^{g+1})$ ;
43:      end if
44:    end for
45:    Re-arrange the positions in ascending order depending on their fitness function  $fit(X)$ ;
46:    Detect the global best position  $X_{Best}^{g+1}$  and its global best fitness value  $fit(X_{Best}^{g+1})$  when the current iteration  $g + 1$  is over;
47:    Perform the DE technique for every position to ameliorate  $X_{Best}^{g+1}$ , as shown in Section 3.6;
48:     $X_{Best} \leftarrow X_{Best}^{g+1}$ ;
49:     $fit(X_{Best}) \leftarrow fit(X_{Best}^{g+1})$ ;
50:     $g \leftarrow g + 1$ ;
51:  end while
52: End

```

4. Experimental Results and Discussion

This section presents the empirical findings of the proposed RBAVO-DE and its counterparts: Binary Artificial Bee Colony (BABC) [45], Binary Salp Swarm Algorithm (BSSA) [46], Binary PSO (BPSO) [47], Binary Bat Algorithm (BBA) [48], Binary Grey-Wolf Optimization (BGWO) [49], Binary Grasshopper Optimization Algorithm (BGOA) [50], Binary Whale Optimization Algorithm (BWOA) [51], Binary ASO (BASO) [52], Binary Bird Swarm Algorithm (BBSA) [53], Binary HGSO (BHGSO) [54], and Binary Harris Hawks Op-

timization (BHHO) [55]. The optimizers undergo evaluation through training and testing benchmarks, with conclusive results derived from the average values of the evaluation metrics. The benchmarks employed for assessing the performance of the proposed model are detailed in Section 4.1. The parameters used in the operational environments are outlined in Section 4.2. The metrics used for evaluation are described in Section 4.3. The analysis of the experimental outcomes is discussed in Section 4.4.

4.1. Dataset Description

In-depth experimental approaches and various wrapper algorithms were applied to twenty-two datasets of gene descriptions; the data comprise normalized Level 3 RNA-Seq gene expression data for twenty-two kinds of tumors from the Broad Institute. These data are publicly accessible and can be found in [56]. We adhered to the methodology described in [36] and observed discrepancies between the data referenced from GitHub in the paper and the figures reported within the document, which were sourced from the website. The website's data included a mix of tumor and standard samples, whereas the paper treated the data uniformly as tumor samples. Consequently, we conducted a detailed examination of the data. Initially, the website offered various formats of the identical dataset we intended to analyze. Upon delving into the data, we encountered the following challenges:

- Certain genes are identified by ID but lack an associated symbol.
- Some genes are absent from the annotation file.
- There is a mix-up of samples, including both normal and tumor types.

Consequently, pre-processing was necessary to segregate and identify samples, distinguish between normal and tumor samples for use in binary classification, and streamline the GS process. We addressed the challenges above in the following manner:

- We looked up the corresponding gene symbol for each ID found in the annotation file.
- After cross-referencing with the annotating file, over one hundred genes were eliminated.
- Based on the sample report, every Excel sheet's row was organized according to the kind of sample for binary classification.

Moreover, the Relief approach, as detailed in Section 3.1, was utilized in the pre-processing stage to calculate the weight of each gene within the benchmark. These weights were subsequently ordered from highest to lowest. Genes with lower weights were then removed. The Relief approach was useful for discarding genes that do not contribute to classification.

Following the pre-processing phase, the data were refined and ready for the GS process. Contrary to the approach used in [36], which tackled the multi-classification of all types of cancer together, we opted to analyze every cancer type individually for greater specificity. Table 1 [57] presents a comprehensive overview of all twenty-two tumor types and their respective sample counts. The number of features utilized in the benchmark datasets is 32,000 features (genes).

Table 1. Datasets employed in this paper [57].

#	Cancer	Normal Records	Tumor Records
1	Bladder Urothelial Carcinoma (BLCA)	19	408
2	Thyroid Cancer (THCA)	59	501
3	Cervical and Endocervical Cancers (CESCs)	3	304
4	Cholangiocarcinoma (CHOL)	9	36
5	Colon Adenocarcinoma (COAD)	41	458
6	Esophageal Cancer (ESCA)	13	184
7	Glioblastoma Multiforme (GBM)	5	153

Table 1. *Cont.*

#	Cancer	Normal Records	Tumor Records
8	Thymoma (THYM)	2	120
9	Head and Neck Squamous Cell Carcinoma (HNSC)	44	520
10	Kidney Chromophobe (KICH)	25	66
11	KIRC	72	533
12	Kidney Renal Papillary (KIRP)	32	290
13	Liver Hepatocellular Carcinoma (LIHC)	50	371
14	LUAD	59	515
15	LUSC	51	501
16	Pancreatic Adenocarcinoma (PAAD)	4	178
17	UCEC	35	176
18	Pheochromocytoma and Paraganglioma (PCPG)	3	179
19	Rectum adenocarcinoma (READ)	10	94
20	Sarcoma (SARC)	2	259
21	Skin Cutaneous Melanoma (SKCM)	1	103
22	STAD	37	415

4.2. Parameter Setting

The proposed RBAVO-DE algorithm was compared against binary variants of different meta-heuristic optimizers, such as BABC, BSSA, BPSO, BBA, BGWO, BWOA, BBSA, BGOA, BHHO, BASO, and BHGSO. The critical parameters for the ML models employed in this study are detailed in Table 2.

Table 2. The key parameters of the employed ML models.

Model	Parameters
SVM	Polynomial kernel = 2
k -NN	Euclidean distance metric $k = 5$ [21,58,59]

Within the proposed framework, the validity of the results was verified using a 10-fold cross-validation approach to ensure the reliability of the outcomes. This involved randomly dividing each dataset into two distinct subsets: 80% for training and the remaining 20% for testing. The training portion was employed to train the presented classifiers using optimization techniques, while the testing portion was used to evaluate the selected genes' effectiveness. The parameters most commonly applied across all compared algorithms are summarized in Table 3. To execute all experiments in this paper, Python was utilized in a computational environment with a Dual Intel® Xeon® Gold 5115 2.4 GHz CPU and 128 GB of RAM on the Microsoft Windows Server 2019 operating system.

Table 3. Parameter configurations of all utilized optimizers.

Optimizer	Parameters
All optimizers	Execution count = 30 Maximum count of iterations $G_{max} = 100$ Size of population $N = 10$

Table 3. Cont.

Optimizer	Parameters
Proposed RBAVO-DE	Dimensionality D = The count of genes in the employed datasets Upper boundaries UB Lower boundaries LB Parameter $L_1 = 0.7$ Parameter $L_2 = 0.2$ Parameter $w = 2$ Parameter $P_1 = 0.6$ Parameter $P_2 = 0.6$ Parameter $P_3 = 0.5$
BSSA	Count of scroungers $SD = 0.1 \times N$ Count of producers $PD = 0.2 \times N$ Safety threshold $ST = 0.8$
BABC	Count of employed bees = 16 Count of scout bees = 3 Count of onlooker bees = 4
BPSO	Inertia weight ($\omega_{max} = 0.9\omega_{min} = 0.4$) Acceleration coefficients ($c_2 = c_1 = 1.2$)
BBA	Loudness $A = 0.8$ Pulse emission rate $r = 0.95$
BWOA	Lower and upper pulse frequencies = 0, 10 a is linearly decreased from 2 to 0 $p = 0.5$ $b = 1.0$
BHHO	Rabbit energy $E \in [-1, 1]$
BGWO	a is linearly decreased from 2 to 0
BGOA	$C_{min} = 0.00004$ and $C_{max} = 1$
BBSA	Frequency of flight $ff = 10$ Effect on birds' vigilance Followed coefficient $fl = 0.5$ Probability of foraging for food $p = 0.8$ Acceleration coefficients ($c_1 = c_2 = 1.5$) behaviors ($a_1 = a_2 = 1.0$)
BASO	Depth weight $\alpha = 50$ Multiplier weight $\beta = 0.2$
BHGSO	Count of clusters = 2 $\alpha = \beta = 0.1$ $K = 1.0$, and $l_3 = 1E - 02$ $l_1 = 5E - 03$, $l_2 = 1E + 02$

4.3. Evaluation Criteria

To evaluate the proposed RBAVO-DE's effectiveness relative to other methods, each strategy was independently tested thirty times on each benchmark to ensure statistical validation of the results. For this purpose, the following established performance metrics for the GS issues were employed:

- **Mean classification accuracy** ($Mean_{AC}$): This measure represents the accuracy of correctly classifying data, calculated by running the algorithm independently thirty times. It is determined in the following way:

$$Mean_{AC} = \sum_{k=1}^{30} \sum_{r=1}^m match(PL_r, AL_r) \quad (23)$$

In this formula, m denotes the number of samples in the test dataset, while PL_r and AL_r denote the predicted labels from the classifier and the actual class labels for sample r , respectively. The function $match(PL_r, AL_r)$ serves as a comparison mechanism, where if PL_r is equal to AL_r , then $match(PL_r, AL_r)$ is assigned a value of 1; if they do not match, it is assigned a value of 0.

- **Mean fitness** ($Mean_{Fi}$): This measure calculates the average $meanFi$, achieved by running the proposed approach thirty times independently. It gauges the balance between lowering the classification error rate and minimizing the number of genes selected. A lower value indicates a superior solution, assessed based on fitness:

$$MeanFi = \frac{1}{30} \sum_{k=1}^{30} f_*^k \quad (24)$$

where f_*^k shows the optimum fitness value achieved in the k – th execution.

- **Mean number of chosen genes** ($MeanGe$): This metric calculates the average number of genes chosen (or GS ratio) by running the proposed approach thirty times independently, and it is defined as follows:

$$MeanGe = \frac{1}{30} \sum_{k=1}^{30} \frac{|d_*^k|}{|D|} \quad (25)$$

Here, $|d_*^k|$ denotes the total count of genes chosen in the optimal solution for the k^{th} execution, and $|D|$ represents the total number of genes present in the initial benchmark.

- **Standard Deviation** ($STDE$): Based on the outcomes above, the overall average results derived from thirty separate executions of each optimizer on every benchmark were assessed for stability in the following manner:

$$STDE = \sqrt{\frac{1}{29} \sum_{k=1}^{30} (Y_*^k - Mean_Y)^2} \quad (26)$$

where Y shows the measure to be utilized, Y_*^k denotes the measure value Y in the k^{th} run, and $Mean_Y$ is the mean of the measure from thirty independent executions.

The data in the subsequent tables represent the mean values obtained from thirty independent executions, focusing on classification accuracy, the number of chosen genes, average fitness, precision, recall, and F1-score. The ensuing subsections thoroughly examine and discuss these experimental findings, with bold figures highlighting the optimal results.

4.4. Results of Comparing the Proposed RBAVO-DE with Various Meta-Heuristic Algorithms

The proposed RBAVO-DE approach with the SVM and k -NN models was compared with other recent meta-heuristic methods executed under the same conditions to show its superiority over its counterparts. The proposed RBAVO-DE algorithm was compared with binary variants of several optimizers, including BABC, BSSA, BPSO, BBA, BGWO, BWOA, BBSA, BGOA, BHHO, BASO, and BHGSO.

4.4.1. Results Employing the k -NN Model

Table 4 shows the results of the proposed RBAVO-DE algorithm and other optimization techniques employing the k -NN model, based on accuracy metrics compared under similar conditions. The experimental outcomes reveal that the proposed RBAVO-DE achieved the most promising outcomes in four benchmarks. It is noteworthy that all competitive methods, including the proposed RBAVO-DE employing the k -NN model, produced comparable outcomes across eighteen benchmarks.

The rest of the metrics for the k -NN model are shown in Appendix A.1. In terms of the average fitness values, the proposed RBAVO-DE algorithm demonstrated higher efficiency compared to its counterparts using the k -NN model under similar conditions. The RBAVO-DE algorithm yielded the lowest fitness results and the most competitive STDE across all benchmarks. Also, all utilized benchmarks are high-dimensional, demonstrating that the proposed RBAVO-DE can run effectively on all benchmarks regardless of size. The RBAVO-DE algorithm shows promise by effectively balancing exploitation and exploration of the search space, avoiding becoming trapped in local optima during iterations. Unlike

many other algorithms that tend to become trapped, it demonstrated the ability to escape such traps.

Regarding the mean results for the genes selected by the proposed RBAVO-DE algorithm and its counterparts employing k -NN, the proposed RBAVO-DE algorithm surpassed the other methods across all benchmarks concerning the number of chosen genes. Furthermore, the RBAVO-DE's capacity to identify significant genes is due to its effective exploration of possible areas while enhancing accuracy.

In terms of the mean precision of the proposed RBAVO-DE algorithm and its counterparts with k -NN, the proposed RBAVO-DE outperformed alternative approaches for three of the twenty-two datasets. For nineteen datasets, BWOA achieved similar results, while BABC, BPSO, BGWO, BGOA, and BHHO obtained results similar to those of the proposed RBAVO-DE. BASO and BHGSO, which produced the same outcomes as the proposed RBAVO-DE on seventeen datasets, ranked fourth. Ultimately, BBA ranked lowest among all approaches, producing similar outcomes to the proposed RBAVO-DE for fifteen datasets.

Regarding the mean recall of the proposed RBAVO-DE and its counterparts employing k -NN, RBAVO-DE outperformed the alternative approaches for three of the twenty-two datasets. On the other hand, BSSA, BABC, BPSO, BGWO, BWOA, BGOA, and BBBSA produced the same outcomes as RBAVO-DE for 19 datasets, while BHHSO achieved similar results for 18 datasets. BHGSO, which obtained the same outcomes as the proposed RBAVO-DE for seventeen datasets, ranked fourth. Ultimately, BASO and BBA ranked lowest among all approaches, producing similar results to RBAVO-DE for sixteen datasets.

With regard to the mean F1-score of the proposed RBAVO-DE and its counterparts employing k -NN, the proposed RBAVO-DE outperformed the alternative approaches for three of the twenty-two datasets. In contrast, for eighteen datasets, BSSA, BABC, BPSO, BGWO, BGOA, and BBSA produced the same outcomes as RBAVO-DE, while BWOA and BHHO produced comparable outcomes for seventeen datasets. BASO and BHGSO, which obtained the same outcomes for 16 datasets as the proposed RBAVO-DE, ranked fourth. Ultimately, BBA ranked lowest among all approaches, producing similar outcomes to the proposed RBAVO-DE for fourteen datasets.

Table 4. The results of the proposed RBAVO-DE algorithm and its counterparts using k -NN regarding average classification accuracy results.

[illegible]

Table 4. Cont.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
PCPG	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
READ	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
SARC	Average	0.981	0.981	0.981	0.981	0.981	0.981	0.981	0.981	0.981	0.981	0.981	0.981
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
SKCM	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
STAD	Average	0.999	0.996	0.999	0.996	0.989	0.996	0.995	0.997	0.992	0.995	0.996	0.989
	STDE	0.003	0.005	0.004	0.005	0.000	0.005	0.006	0.005	0.005	0.006	0.005	0.000
THCA	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
THYM	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
UCEC	Average	0.998	0.985	0.988	0.977	0.969	0.984	0.977	0.978	0.978	0.979	0.976	0.968
	STDE	0.008	0.012	0.013	0.006	0.011	0.012	0.006	0.008	0.009	0.009	0.005	0.011
Ranking	W T L	4 18 0	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4

Boldface values denote the best results.

4.4.2. Results Employing the SVM Model

Table 5 displays the outcomes of the proposed RBAVO-DE and various optimization techniques employing the SVM model concerning classification accuracy results assessed under identical running conditions. The experimental outcomes demonstrate that the proposed RBAVO-DE algorithm outperformed the other approaches by obtaining the most promising values for four datasets. All competitive methods, including the proposed RBAVO-DE employing SVM, obtained equivalent values across 18 benchmarks.

The rest of the metrics utilizing SVM are presented in Appendix A.2. Regarding the mean fitness values of the proposed RBAVO-DE and its counterparts employing the SVM model under equivalent running conditions, the proposed RBAVO-DE proved to be more efficient than the other methods. The proposed RBAVO-DE employing the SVM model yielded the lowest fitness values and the most competitive STDE across all benchmarks. Also, all utilized benchmarks are high-dimensional, demonstrating that the proposed RBAVO-DE can run effectively on all benchmarks regardless of size. The RBAVO-DE algorithm shows promise by effectively balancing exploitation and exploration of the search space, avoiding becoming trapped in local optima during iterations. Unlike many other algorithms that tend to become trapped, it demonstrated the ability to escape such traps.

Regarding the mean results for the genes selected by the proposed RBAVO-DE and its counterparts employing SVM, the proposed RBAVO-DE achieved more promising results than other techniques across all benchmarks utilized in this study. Also, the superiority of the proposed RBAVO-DE employing SVM in this context demonstrates its capability to effectively explore valuable regions of the search space while avoiding regions with non-feasible solutions.

With regard to the average precision values of the proposed RBAVO-DE and its counterparts employing SVM, the proposed RBAVO-DE outperformed the alternative approaches for three of the twenty-two datasets. On the other hand, for eighteen datasets, BABC, BGWO, BWOA, BBSA, and BASO yielded comparable results. For seventeen datasets, BSSA, BPSO, BGOA, and BHHO yielded results similar to those of RBAVO-DE, thereby ranking fourth. Ultimately, BBA ranked lowest among all approaches, producing similar outcomes to those of the proposed RBAVO-DE for twelve datasets.

In terms of the mean recall of the proposed RBAVO-DE method and its counterparts using SVM, the proposed RBAVO-DE outperformed the alternative approaches for three of the twenty-two datasets. In contrast, BPSO and BHHO performed similarly for eighteen datasets, while BSSA, BABC, BGWO, BWOA, BGOA, and BBSA produced the same results as the proposed RBAVO-DE for nineteen datasets. BHGSO, which obtained the same outcomes as RBAVO-DE for seventeen datasets, ranked fourth. In the end, BASO and BBA ranked lowest among all approaches, producing outcomes similar to those of RBAVO-DE for sixteen datasets.

Regarding the mean F1-score of the proposed RBAVO-DE algorithm and its counterparts employing SVM, the proposed RBAVO-DE outperformed the alternative approaches for five of the twenty-two datasets. On the other hand, BWOA produced outcomes similar to those of the proposed RBAVO-DE for seventeen datasets, while BSSA, BABC, BPSO, BGWO, BGOA, and BBSA produced comparable results for sixteen datasets. BHHO, which obtained the same outcomes as the proposed RBAVO-DE for fifteen datasets, ranked fourth. Ultimately, BBA ranked lowest among all approaches, producing similar outcomes to those of RBAVO-DE for 12 datasets.

Table 5. The results of the proposed RBAVO-DE algorithm and its counterparts employing SVM regarding average classification accuracy values.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
CESC	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
CHOL	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
COAD	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
ESCA	Average	0.986	0.975	0.977	0.974	0.974	0.977	0.974	0.975	0.975	0.975	0.974	0.974
	STDE	0.013	0.004	0.008	0.000	0.000	0.008	0.000	0.005	0.005	0.005	0.000	0.000
GBM	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
HNSC	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
KICH	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
KIRC	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
KIRP	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LIHC	Average	1.000	1.000	1.000	1.000	0.9992	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.0029	0.000	0.000	0.000	0.000	0.000	0.000	0.000
LUAD	Average	0.993	0.991	0.991	0.991	0.991	0.991	0.991	0.991	0.991	0.991	0.992	0.991
	STDE	0.003	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.002	0.000
LUSC	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PAAD	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
PCPG	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
READ	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
SARC	Average	0.994	0.986	0.989	0.986	0.981	0.989	0.987	0.986	0.984	0.986	0.984	0.981
	STDE	0.008	0.008	0.009	0.008	0.000	0.009	0.009	0.008	0.006	0.008	0.007	0.000
SKCM	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
STAD	Average	0.990	0.989	0.989	0.989	0.989	0.989	0.989	0.989	0.989	0.989	0.989	0.989
	STDE	0.003	0.000	0.000	0.000	0.002	0.000	0.000	0.000	0.002	0.000	0.000	0.000
THCA	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
THYM	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
UCEC	Average	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	STDE	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
Ranking	W T L	4 18 0	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4	0 18 4

Boldface numbers indicate the best results.

4.5. Convergence Analysis

It is evident in Appendix B [57] that the proposed RBAVO-DE employing the SVM and k -NN models attained optimum convergence behavior across all benchmarks. Therefore, the convergence performance of the proposed RBAVO-DE employing the SVM and k -NN models demonstrates its capability to reach optimal outcomes promptly while maintaining an efficient equilibrium between search and exploitation.

4.6. Wilcoxon's Rank-Sum Test

This paper compares the fitness values achieved by the proposed RBAVO-DE and its counterparts pairwise using the Wilcoxon rank-sum test [60], which aims to determine whether there is a statistically significant difference between the different approaches. This test is a crucial tool for evaluating the success of the proposed algorithm. The Wilcoxon test is employed in hypothesis testing to compare matched data. It involves sorting the absolute differences in the outcomes of paired procedures on the j^{th} of N issues. Next, the lowest value is found by summing the positive (R^+) and negative (R^-) rankings. The null hypothesis is rejected if the resultant significance level is less than 5%, and not rejected otherwise.

From examining the results of the comparison made between the RBAVO-DE and its counterparts pairwise using the Wilcoxon test, we found that all R^+ values equal 253, all R^- values equal 0, and all p -values equal 4.768×10^{-7} which are less than 0.05 (significance level of 5%). Hence, it can be concluded that the proposed RBAVO-DE employing SVM and k -NN performs better than any other algorithm in all cases. Therefore, all p -values less than 0.05 (significance level of 5%) offer compelling evidence that the outcomes of the proposed strategy are statistically significant and not just coincidental.

4.7. Computational Complexity of the RBAVO-DE Algorithm and Various Meta-Heuristic Algorithms

4.7.1. Computational Complexity of the Execution Time of the RBAVO-DE Algorithm

Each of RBAVO-DE's five core steps can be analyzed separately to determine its computational complexity. These steps include filtering features, initializing the population, boosting and amending the position, appraising the fitness function, and using DE. Then, $O_{execute_time}(RBAVO - DE)$ can be utilized to represent the overall computational complexity of the proposed RBAVO-DE algorithm. This can be computed using the following big-O notation formulas:

$$\begin{aligned} O_{execute_time}(RBAVO - DE) = & O_{execute_time}(\text{Filtering features}) + \\ & O_{execute_time}(\text{Initializing population}) + \\ & O_{execute_time}(\text{Boosting and amending position}) + \\ & O_{execute_time}(\text{Appraising fitness function}) + \\ & O_{execute_time}(\text{Incorporating DE technique}). \end{aligned} \quad (27)$$

Since N determines the population size, G_{max} is the maximum number of iterations allowed, and D is the problem's dimensional space. Hence,

$$\begin{aligned} O_{execute_time}(\text{Filtering features}) &= O_{execute_time}(D). \\ O_{execute_time}(\text{Initializing population}) &= O_{execute_time}(N). \\ O_{execute_time}(\text{Boosting and amending position}) &= O_{execute_time}(G_{max} \times N \times D). \\ O_{execute_time}(\text{Appraising fitness function}) &= O_{execute_time}(G_{max} \times N). \\ O_{execute_time}(\text{Incorporating DE technique}) &= O_{execute_time}(N \times D). \end{aligned}$$

For that,

$$\begin{aligned} O_{execute_time}(RBAVO - DE) &= O_{execute_time}(D) + \\ & O_{execute_time}(N) + O_{execute_time}(G_{max} \times N \times D) + \\ & O_{execute_time}(G_{max} \times N) + O_{execute_time}(N \times D) = \\ & O_{execute_time}(G_{max} \times N \times D). \end{aligned}$$

4.7.2. Computational Complexity of the Memory Usage of the RBAVO-DE Algorithm

This involves measuring the amount of memory required by an algorithm to tackle a problem as the quantity of the input increases. It is frequently expressed as the additional memory needed by the algorithm beyond the input. It entails merging the following two primary elements:

1. **Memory usage of input variables:** This is the amount of memory required for the algorithm to store the input data. There are 13 input variables related to the proposed RBAVO-DE algorithm, as follows: N , G_{max} , D , LB , UB , C_R , W_M , L_1 , L_2 , w , P_1 , P_2 , and P_3 . Since each variable stores numerical values, 4 bytes of memory are utilized by each one. Consequently, the total memory usage complexity of these 13 input variables is 52 bytes ($13 \times 4 \text{ bytes} = 52 \text{ bytes}$). The memory usage complexity of the input values is constant.
2. **Additional memory usage:** This shows how much more memory the algorithm needs in addition to the input. It includes the memory required for data structures, internal variables, and other components of the algorithm. Regardless of the size of the input, the RBAVO-DE algorithm requires a specific amount of extra memory. The following variables are involved:
 - The memory usage complexity consumed by $X_{initial}$ is $(4 \times N \times D)$ bytes. This is because each position in the positions vector $X_{initial}$ requires 4 bytes of memory, and its size is $(N \times D)$, proportionate to the initial population of N positions with dimension size D . Because the amount of memory required rises linearly with the value $(N \times D)$, its memory use complexity is linear.
 - The variables pr , g , F_i , t , R , D_i , d_t , S_1 , S_2 , A_1 , A_2 , σ , $fit(X_i)$, $fit(X_{Best_1})$, $fit(X_{SecondBest_2})$, and $fit(X_{Best})$ require 4 bytes of memory space each since they only represent numerical values. As a result, the total memory usage complexity of these 16 variables is 64 bytes ($16 \times 4 \text{ bytes} = 64 \text{ bytes}$). This memory usage complexity is constant.
 - The position vectors $BestVulture_1$, $SecondBestVulture_2$, X_i , $Levy_d$, μ , σ , X_i^{bin} , v_i , X_{r_1} , X_{r_2} , X_{r_3} , u_i , and X_{Best} each require 4 bytes of memory space, and the size of each vector is D , proportionate to the dimension size of the acquired positions. As a result, the vectors at these 13 points have a total memory usage complexity of $(52 \times D)$ bytes ($13 \times 4 \times D \text{ bytes}$). Since the memory needed grows linearly with the value D , its memory consumption complexity is linear.

As a result, the overall memory usage complexity for all of the additional variables listed above is $(4 \times N \times D) + 64 + (52 \times D)$ bytes.

Lastly, the following formula can be used to determine the computational complexity of the overall memory usage of the proposed RBAVO-DE algorithm:

$$\begin{aligned} \text{Memory usage complexity(RBAVO-DE)} &= \text{Input values memory usage} + \\ &\quad \text{Additional memory usage} = 52 + ((4 \times N \times D) + 64 + (52 \times D)) \text{ bytes.} \end{aligned}$$

Keep in mind that there are constant bytes that are not taken into account. In big-O notation, the computational complexity of the total memory consumption of RBAVO-DE can be represented as $O_{memory_usage}(RBAVO - DE)$. This can then be calculated in big-O notation after eliminating all constants in the following way:

$$\begin{aligned} O_{memory_usage}(RBAVO - DE) &= O_{memory_usage}(\text{Input values memory usage}) + \\ &\quad O_{memory_usage}(\text{Additional memory usage}) = \\ O_{memory_usage}(1) &+ (O_{memory_usage}(N \times D) + O_{memory_usage}(1) + O_{memory_usage}(D)) = \\ &\quad O_{memory_usage}(N \times D). \end{aligned}$$

It can be difficult to compile a thorough comparison of the memory usage and execution time complexity of various meta-heuristic algorithms since these complexities might differ based on the particular implementation, size of the problem, and other operators. Furthermore, not all of the algorithms listed have comprehensive evaluations of the execution time and memory usage complexity available, and their properties may vary depending on the problem at hand.

5. Benefits and Drawbacks of the RBAVO-DE Algorithm

This part presents a balanced discussion of the benefits and drawbacks of the RBAVO-DE algorithm. The benefits of the RBAVO-DE algorithm can be listed as follows:

- **High Accuracy in Classification:** The RBAVO-DE algorithm has demonstrated high classification accuracy, achieving up to 100% in some cases. This is a noteworthy achievement, particularly regarding cancer datasets, as accurate gene selection can directly impact diagnostic and therapeutic outcomes.
- **Effective Feature Size Reduction:** The algorithm has shown remarkable capability in reducing the feature size by up to 98% while maintaining or improving classification accuracy. This dimensionality reduction is crucial for processing high-dimensional datasets efficiently.
- **Robustness Across Diverse Datasets:** The effectiveness of the RBAVO-DE algorithm across twenty-two cancer datasets indicates its robustness and adaptability to various genetic data characteristics. This versatility is beneficial for broader applications in genomics research.
- **Superior Performance Over Competitors:** When compared with binary variants of widely recognized meta-heuristic algorithms, RBAVO-DE stands out for its outstanding performance in accuracy and feature reduction, highlighting its innovative approach to gene selection.

On the other hand, the drawbacks of the proposed methodology can be listed as follows:

- **Computational Complexity:** The high dimensionality of the datasets and the iterative nature of meta-heuristic algorithms suggest that RBAVO-DE may have a significant computational cost. This aspect could limit its applicability in environments with constrained computational resources.
- **Generalizability Concerns:** Despite its success across various cancer datasets, the algorithm's generalizability to other types of biological data or diseases remains to be thoroughly investigated. It is crucial to test RBAVO-DE in broader contexts to confirm its applicability beyond the datasets examined.
- **Parameter Sensitivity:** Like many meta-heuristic algorithms, RBAVO-DE might be sensitive to its parameter settings, affecting performance and efficiency. Detailed studies on the impact of different parameter configurations and strategies for their optimization could enhance the algorithm's utility.

In summary, the RBAVO-DE algorithm represents a significant advancement in gene selection for cancer classification, with notable strengths in accuracy and efficiency. However, addressing its potential drawbacks through further research could broaden its applicability and improve its performance, making it an even more valuable tool in genomics and personalized medicine.

6. Conclusions and Future Directions

The RBAVO-DE approach proposed in this paper was the first to be implemented for handling GS issues in RNA-Seq gene expression data and determining potential biomarkers for different tumor classes to enhance the best solution discovered. The outcomes were promising, revealing that the proposed RBAVO-DE algorithm's effectiveness and capabilities were significantly improved. SVM and k -NN, two widely used classification models, were employed to evaluate the efficacy of each set of selected genes. The proposed RBAVO-DE algorithm's capability was compared to binary variants of eleven widely used

meta-heuristic techniques to assess it on different tumor classes with various instances. The assessment was executed employing a combination of evaluation measures, such as average fitness, classification accuracy, the number of selected genes, precision, recall, and F1-score. The proposed RBAVO-DE algorithm using the SVM and k -NN classifiers achieved more promising results than other optimizers in handling GS issues. Despite the promising outcomes, this research opens several avenues for future exploration to further advance the field:

- **Algorithmic Enhancements:** Improving the algorithm's efficiency and reducing computational complexity.
- **Cross-disease Applicability:** Testing the proposed algorithm on genetic data from various diseases beyond cancer.
- **Comparative Analyses with Deep Learning Models:** Evaluating RBAVO-DE against advanced deep learning models for genetic data analysis.
- **Real-world Clinical Validation:** Collaborating with clinical experts to validate the practical utility of selected genes in cancer treatment.
- **Scalability and Parallelization:** Enhancing the algorithm's scalability through parallel computing to handle larger genetic datasets efficiently.
- **Interdisciplinary Applications:** Exploring the algorithm's potential in other fields dealing with high-dimensional data, such as finance and environmental modeling.

In conclusion, while the RBAVO-DE algorithm represents a significant step forward in the field of gene selection for cancer classification, the paths outlined for future research highlight the potential for further advancements and broader applications of this work. Continued interdisciplinary collaboration and innovation will be crucial for unlocking the full capacity of gene selection methodologies to enhance healthcare outcomes and advance our understanding of complex diseases.

Author Contributions: M.G.G.: Resources, Validation, and Writing—review and editing. A.A.A.: Resources, Formal analysis, Data Curation, Validation, Writing—original draft, and Writing—review and editing. A.E.E.: Investigation, Visualization, Data Curation, Validation, Writing—original draft, and Writing—review and editing. A.A.A.E.-M.: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data Curation, Validation, Writing—original draft, and Writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: Prince Sattam bin Abdulaziz University funded this research work under project number PSAU/2023/01/25612.

Data Availability Statement: For enhancing the applicability and reproducibility of this research, the developed software and relevant Python code of this research are publicly available and obtainable in [61].

Acknowledgments: The authors express their gratitude to Prince Sattam bin Abdulaziz University for providing financial support for this research project.

Conflicts of 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.

Abbreviations

The following abbreviations are used in this research:

AVO	African Vultures Optimization
HNSC	Head and Neck Squamous Cell Carcinoma
DE	Differential Evolution
KICH	Kidney Chromophobe
RBAVO-DE	Relief Binary African Vultures Optimization based on Differential Evolution
KIRP	Kidney Renal Papillary
GS	Gene Selection
LIHC	Liver Hepatocellular Carcinoma
RNA-Seq	RNA Sequencing

PAAD	Pancreatic Adenocarcinoma
DNA	Deoxyribonucleic Acid
PCPG	Pheochromocytoma and Paraganglioma
SVM	Support Vector Machine
READ	Rectum adenocarcinoma
<i>k</i> -NN	<i>k</i> -Nearest Neighbor
SARC	Sarcoma
FS	Feature Selection
SKCM	Skin Cutaneous Melanoma
MHMs	Meta-heuristic methods
$Mean_{Fi}$	Mean fitness
ML	Machine Learning
$Mean_{Ge}$	Mean size of chosen genes
DL	Deep learning
<i>STDE</i>	Standard Deviation
SA	Simulated Annealing
RF	Random Forest
FOIL	First-Order Inductive Learner
GSO	Gravitational Optimizer
SCCSA	Sine-Cosine-Cuckoo Search Algorithm
SMO	Spider Monkey Optimization
BHGSO	Binary Henry Gas Solubility Optimization
BPSO	Binary Particle Swarm Optimization
BABC	Binary Artificial Bee Colony
BSSA	Binary Salp Swarm Algorithm
BBA	Binary Bat Algorithm
BGWO	Binary Grey-Wolf Optimization
BGOA	Binary Grasshopper Optimization Algorithm
BWOA	Binary Whale Optimization Algorithm
BASO	Binary Atom Search Optimization
BBSA	Binary Bird Swarm Algorithm
BHHO	Binary Harris Hawks Optimization
BBBO	Binary Brown-Bear Optimization
BAO	Binary Aquila Optimization
BMOA	Binary Meerkat Optimization Algorithm
LUAD	Lung Adenocarcinoma
LUSC	Lung Squamous Cell Carcinoma
BRCA	Breast Invasive Carcinoma
KIRC	Kidney Renal Clear Cell Carcinoma
UCEC	Uterine Corpus Endometrial Carcinoma
RPKM	Reads Per Kilobase per Million mapped reads
STAD	Stomach Adenocarcinoma
SDAE	Stacked Denoising Autoencoder
CGA	Cancer Genome Atlas
BLCA	Bladder Urothelial Carcinoma
THCA	Thyroid Cancer
CESC	Cervical and Endocervical Cancers
CHOL	Cholangiocarcinoma
COAD	Colon Adenocarcinoma
ESCA	Esophageal Cancer
GBM	Glioblastoma Multiforme
THYM	Thymoma

Appendix A. Comparison Results of the Proposed RBAVO-DE and Its Counterparts

Appendix A.1. Comparison Results Based on the k -NN Model

Table A1. The results of the proposed RBAVO-DE algorithm and its counterparts employing k -NN regarding average fitness values.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Average	0.00190	0.01110	0.00710	0.01120	0.01470	0.00940	0.01070	0.01140	0.01310	0.01030	0.01080	0.01590
	STDE	0.00430	0.00530	0.00440	0.00540	0.00240	0.00550	0.00540	0.00540	0.00440	0.00550	0.00540	0.00180
CESC	Average	0.01610	0.01990	0.02020	0.02050	0.01980	0.02000	0.02020	0.02020	0.01970	0.02020	0.02030	0.02060
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
CHOL	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
COAD	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
ESCA	Average	0.02560	0.02930	0.02960	0.02990	0.02930	0.02940	0.02970	0.02960	0.02910	0.02960	0.02980	0.03000
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00020
GBM	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
HNSC	Average	0.00710	0.01250	0.01290	0.01330	0.01280	0.01240	0.01310	0.01320	0.01270	0.01290	0.01320	0.01360
	STDE	0.00350	0.00160	0.00140	0.00010	0.00030	0.00210	0.00010	0.00010	0.00020	0.00150	0.00010	0.00010
KICH	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
KIRC	Average	0.00250	0.01110	0.01040	0.01220	0.01200	0.01020	0.01200	0.01070	0.01040	0.01120	0.01170	0.01290
	STDE	0.00380	0.00270	0.00340	0.00190	0.00020	0.00340	0.00190	0.00320	0.00300	0.00280	0.00220	0.00010
KIRP	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
LIHC	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00420	0.00430	0.00380	0.00430	0.00420	0.00460
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
LUAD	Average	0.00870	0.01260	0.01290	0.01310	0.01240	0.01270	0.01290	0.01290	0.01240	0.01290	0.01300	0.01340
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00010	0.00010	0.00020	0.00010	0.00010	0.00010
LUSC	Average	0.00000	0.00400	0.00430	0.00450	0.00380	0.00410	0.00430	0.00430	0.00390	0.00430	0.00430	0.00480
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00010	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
PAAD	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
PCPG	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
READ	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
SARC	Average	0.01890	0.02260	0.02290	0.02320	0.02260	0.02270	0.02300	0.02290	0.02240	0.02290	0.02300	0.02330
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
SKCM	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
STAD	Average	0.00070	0.00830	0.00620	0.00930	0.01470	0.00810	0.00950	0.00800	0.01170	0.00940	0.00840	0.01570
	STDE	0.00280	0.00500	0.00360	0.00510	0.00020	0.00500	0.00510	0.00480	0.00480	0.00520	0.00490	0.00010
THCA	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00460
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
THYM	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
UCEC	Average	0.00000	0.01920	0.01710	0.02780	0.03540	0.02010	0.02770	0.02690	0.02560	0.02520	0.02860	0.03670
	STDE	0.00250	0.01200	0.01220	0.00610	0.01000	0.01190	0.00620	0.00720	0.00830	0.00910	0.00450	0.01050
Ranking	W T L	22 0 0	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22

Boldface values denote the best results.

Table A2. The results of the proposed RBAVO-DE algorithm and its counterparts employing k -NN regarding average values of selected genes.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Average	142.03	210.17	237.70	235.00	196.03	218.17	228.50	226.30	195.23	228.27	232.03	237.13
	STDE	017.18	020.50	019.07	017.12	023.83	018.27	019.43	014.81	013.17	014.60	018.90	014.14
CESC	Average	122.50	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	213.33	215.90	233.43
	STDE	005.45	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	005.88	006.02	005.64
CHOL	Average	122.13	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	006.18	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
COAD	Average	120.30	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	005.82	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
ESCA	Average	120.17	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.57	218.87	230.07
	STDE	006.81	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.94	004.68	007.64
GBM	Average	120.73	195.13	212.77	225.30	194.10	199.73	213.17	212.80	185.43	211.00	211.83	232.63
	STDE	006.54	011.17	005.40	003.11	009.66	009.05	008.51	005.72	006.40	006.44	004.99	004.39
HNSC	Average	133.43	202.87	222.67	227.43	203.00	211.37	217.97	220.87	195.93	219.83	223.63	241.27
	STDE	017.63	009.26	011.96	003.91	013.24	007.65	007.41	005.32	008.26	007.71	005.68	006.21
KICH	Average	121.43	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	006.58	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36

Table A2. *Cont.*

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
KIRC	Average	137.57	200.83	220.17	227.70	192.17	209.87	216.13	222.30	193.27	218.73	218.60	234.10
	STDE	017.10	009.63	011.74	008.27	008.73	014.08	010.11	015.18	017.36	013.82	012.78	004.21
KIRP	Average	122.33	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	009.08	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
LIHC	Average	123.40	197.00	214.03	225.67	193.33	200.63	212.33	213.63	189.30	213.10	212.03	232.43
	STDE	006.59	007.78	005.22	003.36	009.17	010.02	009.35	004.89	006.89	005.69	004.81	004.97
LUAD	Average	127.03	197.47	216.13	222.93	190.83	205.33	212.57	215.67	191.80	215.27	217.83	237.90
	STDE	008.62	008.98	004.74	006.16	008.61	010.57	006.15	004.43	008.50	006.13	004.71	004.94
LUSC	Average	125.73	198.87	215.63	223.00	189.13	203.60	213.23	216.37	194.67	215.53	215.23	241.23
	STDE	006.35	008.77	003.54	004.89	010.69	007.20	009.70	003.44	006.40	005.54	006.16	004.66
PAAD	Average	121.10	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.71	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
PCPG	Average	122.17	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	006.42	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
READ	Average	120.13	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.72	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
SARC	Average	122.47	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	212.90	216.43	231.80
	STDE	005.79	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.30	004.98	003.40
SKCM	Average	123.03	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	008.83	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
STAD	Average	142.60	215.00	236.17	243.33	187.50	221.17	234.87	232.77	201.73	231.13	237.10	236.97
	STDE	013.31	018.43	013.94	016.29	010.27	019.24	022.63	014.25	014.00	014.04	015.35	005.86
THCA	Average	117.97	195.43	213.40	225.13	194.10	199.63	214.13	212.80	186.53	211.00	211.57	231.90
	STDE	006.67	011.20	005.29	003.10	009.66	009.81	008.59	005.69	005.88	006.44	005.08	004.78
THYM	Average	120.37	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	004.89	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
UCEC	Average	143.93	216.33	237.87	236.00	242.37	220.07	230.80	229.17	206.77	228.40	235.00	266.67
	STDE	013.18	015.41	015.82	010.11	034.13	011.26	011.27	011.66	010.84	010.32	006.14	030.55
Ranking	W T L	22 0 0	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22

Boldface values denote the best results.

Table A3. The results of the proposed RBAVO-DE algorithm and its counterparts employing k -NN regarding average precision values.

Dataset	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	1.00000	1.00000	1.00000	1.00000	0.99960	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CESC	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390	0.98390
CHOL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
COAD	1.00000	1.00000	1.00000	1.00000	0.99890	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
ESCA	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370	0.97370
GBM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
HNSC	0.99180	0.96470	0.96540	0.96500	0.96350	0.96690	0.96660	0.96570	0.96540	0.96540	0.96100	0.96290
KICH	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRP	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LIHC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LUAD	1.00000	0.99770	0.99740	0.99840	0.99870	0.99770	0.99800	0.99540	0.99710	0.99740	0.99670	0.99800
LUSC	1.00000	1.00000	1.00000	1.00000	0.99790	1.00000	0.99970	1.00000	1.00000	1.00000	0.97900	0.99930
PAAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
PCPG	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
READ	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SARC	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110	0.98110
SKCM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
STAD	0.99920	0.95180	0.95180	0.95180	0.94870	0.95180	0.95180	0.95180	0.95180	0.95180	0.99000	0.95090
THCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THYM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
UCEC	1.00000	1.00000	1.00000	1.00000	0.99910	1.00000	1.00000	1.00000	1.00000	1.00000	0.93330	0.99910
Ranking (W T L)	3 19 0	0 19 3	0 19 3	0 19 3	0 15 7	0 19 3	0 18 4	0 19 3	0 19 3	0 19 3	0 17 5	0 17 5

Boldface values denote the best results.

Table A4. The results of the proposed RBAVO-DE algorithm and its counterparts employing k -NN regarding average recall values.

[illegible]

Table A4. Cont.

Dataset	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
HNSC	1.00000	0.99520	0.99450	0.99480	0.99590	0.99280	0.99310	0.99420	0.99450	0.99450	0.99790	0.99730
KICH	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRC	0.99710	0.99040	0.99040	0.99040	0.98970	0.99040	0.99040	0.99040	0.99040	0.99040	0.99010	0.99040
KIRP	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LIHC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LUAD	1.00000	0.99240	0.99270	0.99170	0.99140	0.99240	0.99210	0.99470	0.99310	0.99270	0.99340	0.99210
LUSC	1.00000	1.00000	1.00000	1.00000	0.99580	1.00000	1.00000	1.00000	0.99930	1.00000	0.99310	0.99270
PAAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
PCPG	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
READ	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SARC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SKCM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
STAD	1.00000	1.00000	1.00000	1.00000	0.99750	1.00000	1.00000	1.00000	1.00000	1.00000	0.99620	0.99750
THCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THYM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
UCEC	1.00000	1.00000	1.00000	1.00000	0.99370	1.00000	1.00000	1.00000	1.00000	1.00000	0.99640	1.00000
Ranking (W T L)	3 19 0	0 19 3	0 19 3	0 19 3	0 16 6	0 19 3	0 19 3	0 19 3	0 18 4	0 19 3	0 16 6	0 17 5

Boldface values denote the best results.

Table A5. The results of the proposed RBAVO-DE algorithm and its counterparts employing k -NN regarding average F1-score values.

Dataset	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	1.00000	1.00000	1.00000	1.00000	0.99980	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CESC	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190	0.99190
CHOL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
COAD	1.00000	1.00000	1.00000	1.00000	0.99940	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
ESCA	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670	0.98670
GBM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
HNSC	0.99590	0.97970	0.97970	0.97970	0.97940	0.97970	0.97970	0.97970	0.97970	0.97970	0.97980	0.97970
KICH	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRC	0.99860	0.99520	0.99520	0.99520	0.99480	0.99520	0.99520	0.99520	0.99520	0.99520	0.99500	0.99520
KIRP	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LIHC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LUAD	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500	0.99500
LUSC	1.00000	1.00000	1.00000	1.00000	0.99690	1.00000	0.99980	1.00000	0.99970	1.00000	0.99630	0.99600
PAAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
PCPG	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
READ	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SARC	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050	0.99050
SKCM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
STAD	0.99960	0.97530	0.97530	0.97530	0.97240	0.97530	0.97530	0.97530	0.97530	0.97530	0.97180	0.97360
THCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THYM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
UCEC	1.00000	1.00000	1.00000	1.00000	0.99640	1.00000	1.00000	1.00000	1.00000	1.00000	0.99640	0.99960
Ranking (W T L)	3 19 0	0 18 4	0 18 4	0 18 4	0 14 8	0 18 4	0 17 5	0 18 4	0 17 5	0 18 4	0 16 6	0 16 6

Boldface values denote the best results.

Appendix A.2. Comparison Results Based on the SVM Model

Table A6. The results of the proposed RBAVO-DE algorithm and its counterparts employing SVM regarding average fitness values.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00460
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
CESC	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
CHOL	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
COAD	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
ESCA	Average	0.01370	0.02840	0.02720	0.02990	0.02930	0.02690	0.02970	0.02880	0.02830	0.02880	0.02980	0.03000
	STDE	0.01280	0.00460	0.00750	0.00010	0.00020	0.00750	0.00020	0.00450	0.00460	0.00440	0.00010	0.00020
GBM	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00460
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
HNSC	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00460
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010

Table A6. Cont.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
KICH	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
KIRC	Average	0.00000	0.00390	0.00430	0.00450	0.00380	0.00400	0.00430	0.00430	0.00380	0.00430	0.00430	0.00460
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00020
KIRP	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
LIHC	Average	0.00000	0.00410	0.00440	0.00450	0.00500	0.00430	0.00440	0.00450	0.00400	0.00450	0.00450	0.00500
	STDE	0.00000	0.00010	0.00010	0.00010	0.00280	0.00010	0.00020	0.00010	0.00020	0.00010	0.00010	0.00020
LUAD	Average	0.00840	0.01250	0.01290	0.01310	0.01250	0.01260	0.01290	0.01290	0.01230	0.01290	0.01260	0.01330
	STDE	0.00160	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00150	0.00010
LUSC	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
PAAD	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
PCPG	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
READ	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
SARC	Average	0.00570	0.01780	0.01570	0.01840	0.02260	0.01480	0.01760	0.01870	0.02000	0.01870	0.02000	0.02330
	STDE	0.00860	0.00800	0.00880	0.00800	0.00020	0.00900	0.00820	0.00770	0.00610	0.00770	0.00670	0.00010
SKCM	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
STAD	Average	0.01040	0.01490	0.01530	0.01550	0.01450	0.01500	0.01530	0.01530	0.01470	0.01520	0.01530	0.01560
	STDE	0.00280	0.00020	0.00010	0.00010	0.00200	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
THCA	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
THYM	Average	0.00000	0.00390	0.00430	0.00450	0.00390	0.00400	0.00430	0.00430	0.00370	0.00420	0.00420	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00010	0.00010	0.00010	0.00010
UCEC	Average	0.00000	0.00390	0.00430	0.00450	0.00380	0.00400	0.00430	0.00430	0.00380	0.00420	0.00430	0.00470
	STDE	0.00000	0.00020	0.00010	0.00010	0.00020	0.00020	0.00020	0.00010	0.00020	0.00010	0.00010	0.00010
Ranking	W T L	22 0 0	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22

Boldface values denote the best results.

Table A7. The results of the proposed RBAVO-DE algorithm and its counterparts employing SVM regarding average chosen gene values.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	Average	121.37	194.70	213.03	225.30	194.10	199.50	213.40	212.67	186.27	211.00	211.83	232.37
	STDE	007.49	011.44	005.18	003.11	009.66	009.30	008.44	005.66	005.77	006.44	005.25	004.42
CESC	Average	119.83	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.02	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
CHOL	Average	125.57	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.14	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
COAD	Average	122.63	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.73	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
ESCA	Average	125.50	195.13	215.40	225.30	194.10	201.77	213.63	213.13	185.70	212.40	218.87	230.07
	STDE	011.18	011.17	009.58	003.11	009.66	012.33	008.35	006.44	006.47	008.98	004.68	007.64
GBM	Average	119.73	194.93	213.03	225.30	193.90	198.77	213.70	212.83	185.50	211.00	211.00	232.33
	STDE	008.07	011.27	005.31	003.11	009.47	010.28	008.36	005.69	006.51	006.44	004.92	004.15
HNSC	Average	122.17	196.40	213.90	225.13	194.03	202.10	213.83	213.50	186.93	211.47	210.87	232.17
	STDE	008.29	008.49	005.22	003.19	009.56	010.76	009.34	005.22	005.74	005.66	004.79	005.02
KICH	Average	121.73	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	008.49	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
KIRC	Average	123.87	197.10	214.70	225.00	191.93	200.30	212.97	213.13	187.57	213.83	214.63	231.27
	STDE	007.57	009.42	005.09	004.23	008.70	011.59	007.64	004.94	005.31	005.81	004.20	008.02
KIRP	Average	120.07	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.87	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
LIHC	Average	122.63	204.27	220.37	226.40	212.83	214.07	219.07	223.70	200.60	223.03	225.17	249.90
	STDE	008.01	006.99	006.29	006.80	017.60	007.06	007.88	004.13	007.66	006.48	004.81	007.97
LUAD	Average	121.17	195.13	213.03	225.30	194.10	199.43	213.63	212.67	185.87	212.40	214.50	233.00
	STDE	007.56	011.17	005.31	003.11	009.66	009.37	008.35	005.66	006.40	006.10	005.98	004.34
LUSC	Average	122.67	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	006.28	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
PAAD	Average	119.23	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	006.90	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
PCPG	Average	122.43	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	006.68	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
READ	Average	123.93	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.45	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
SARC	Average	135.27	202.90	226.90	233.77	194.10	210.53	223.87	217.80	189.93	220.57	220.87	231.53
	STDE	018.57	020.86	020.25	014.44	009.66	017.74	017.39	013.90	013.41	016.40	011.76	003.70
SKCM	Average	122.47	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.66	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36

Table A7. Cont.

Dataset	Metric	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
STAD	Average	123.53	194.60	213.40	225.03	194.30	198.80	214.10	212.70	185.70	211.97	215.07	231.77
	STDE	010.65	010.22	005.38	003.29	009.79	009.09	008.68	005.54	006.92	005.88	005.12	004.58
THCA	Average	119.47	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	005.58	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
THYM	Average	121.80	195.13	213.03	225.30	194.10	199.50	213.63	212.67	185.60	211.00	211.57	232.63
	STDE	007.21	011.17	005.31	003.11	009.66	009.30	008.35	005.66	006.49	006.44	005.08	004.36
UCEC	Average	122.23	196.00	213.27	224.83	192.03	200.97	212.60	213.03	189.23	212.23	212.93	233.93
	STDE	005.93	009.06	006.31	004.27	007.85	008.24	007.74	005.67	008.78	004.98	004.97	003.97
Ranking	W T L	22 0 0	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22	0 0 22

Boldface values denote the best results.

Table A8. The results of the proposed RBAVO-DE algorithm and its counterparts employing SVM regarding average precision values.

Dataset	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CESC	1.00000	1.00000	1.00000	1.00000	1.00000	0.99890	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CHOL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
COAD	1.00000	1.00000	1.00000	1.00000	0.99890	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
ESCA	1.00000	1.00000	1.00000	1.00000	0.99820	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.99210
GBM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
HNSC	1.00000	0.94140	0.94170	0.93750	0.92840	0.94140	0.93990	0.94170	0.94110	0.94050	0.84550	0.93150
KICH	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRC	1.00000	1.00000	1.00000	1.00000	0.99940	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRP	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LIHC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LUAD	1.00000	1.00000	1.00000	1.00000	0.99870	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	0.99930
LUSC	1.00000	0.99860	0.99860	0.99860	0.99830	0.99660	0.99860	0.99900	0.99930	0.99860	0.81400	0.99970
PAAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
PCPG	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
READ	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SARC	1.00000	0.99940	1.00000	0.99940	0.99500	0.99940	1.00000	0.99940	0.99620	0.99940	1.00000	0.99120
SKCM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
STAD	0.98830	0.94840	0.95110	0.94170	0.93610	0.94730	0.94690	0.94690	0.93910	0.94650	0.95200	0.93500
THCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THYM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
UCEC	1.00000	0.99910	0.99820	0.99820	0.99300	1.00000	0.99820	0.99820	0.99910	1.00000	1.00000	0.99390
Ranking (W T L)	3 19 0	0 17 5	0 18 4	0 17 5	0 12 10	0 18 4	0 18 4	0 17 5	0 17 5	0 18 4	0 19 3	0 15 7

Boldface values denote the best results.

Table A9. The results of the proposed RBAVO-DE algorithm and its counterparts employing SVM regarding average recall values.

Dataset	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CESC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CHOL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
COAD	1.00000	1.00000	1.00000	1.00000	0.99940	1.00000	1.00000	1.00000	1.00000	1.00000	0.99940	1.00000
ESCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
GBM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
HNSC	1.00000	1.00000	1.00000	0.99970	0.99930	1.00000	1.00000	1.00000	1.00000	1.00000	0.99930	0.99930
KICH	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRC	1.00000	0.99040	0.99040	0.99040	0.99010	0.99040	0.99040	0.99040	0.99040	0.99040	0.98780	0.99040
KIRP	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LIHC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LUAD	1.00000	1.00000	1.00000	1.00000	0.99500	1.00000	1.00000	1.00000	0.99930	1.00000	0.99830	0.99770
LUSC	1.00000	0.99100	0.99100	0.99100	0.99130	0.98960	0.99100	0.99060	0.99030	0.99100	0.99100	0.98990
PAAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
PCPG	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
READ	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SARC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SKCM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
STAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THYM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
UCEC	1.00000	0.99550	0.99640	0.98380	0.98200	0.99460	0.99100	0.99010	0.98470	0.99370	0.97840	0.97930
Ranking (W T L)	3 19 0	0 19 3	0 19 3	0 18 4	0 16 6	0 19 3	0 19 3	0 19 3	0 18 4	0 19 3	0 16 6	0 17 5

Boldface values denote the best results.

Table A10. The results of the proposed RBAVO-DE algorithm and its counterparts employing SVM regarding average F1-score values.

Dataset	RBAVO-DE	BSSA	BABC	BPSO	BBA	BGWO	BWOA	BGOA	BHHO	BBSA	BASO	BHGSO
BLCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CESC	1.00000	1.00000	1.00000	1.00000	0.99950	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
CHOL	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
COAD	1.00000	1.00000	1.00000	1.00000	0.99910	1.00000	1.00000	1.00000	1.00000	1.00000	0.99970	1.00000
ESCA	1.00000	1.00000	1.00000	1.00000	0.99910	1.00000	1.00000	1.00000	1.00000	1.00000	0.99820	0.99600
GBM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
HNSC	1.00000	0.96980	0.97000	0.96760	0.96250	0.96980	0.96900	0.96970	0.96970	0.96940	0.95730	0.96420
KICH	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
KIRC	1.00000	0.99520	0.99520	0.99520	0.99470	0.99520	0.99520	0.99520	0.99520	0.99520	0.99320	0.99520
KIRP	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LIHC	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
LUAD	1.00000	1.00000	1.00000	1.00000	0.99690	1.00000	1.00000	1.00000	0.99970	1.00000	0.99880	0.99850
LUSC	1.00000	0.99480	0.99480	0.99480	0.99480	0.99480	0.99480	0.99480	0.99480	0.99480	0.99480	0.99480
PAAD	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
PCPG	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
READ	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
SARC	0.99900	0.99970	0.99810	0.99970	0.99750	0.99970	1.00000	0.99970	0.99810	0.99970	0.99710	0.99560
SKCM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
STAD	0.99410	0.97350	0.97490	0.96990	0.96700	0.97290	0.97270	0.97270	0.96860	0.97250	0.96560	0.96640
THCA	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
THYM	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000	1.00000
UCEC	1.00000	0.99730	0.99730	0.99090	0.98730	0.99730	0.99450	0.99410	0.99180	0.99680	0.98680	0.98640
Ranking (W T L)	5 17 0	0 16 6	0 16 6	0 16 6	0 12 10	0 16 6	0 17 5	0 16 6	0 15 7	0 16 6	0 13 9	0 14 8

Boldface values denote the best results.

Appendix B. Convergence Graphs of the Proposed RBAVO-DE and Competitive Methods

Appendix B.1. Convergence Graphs Employing the SVM Model

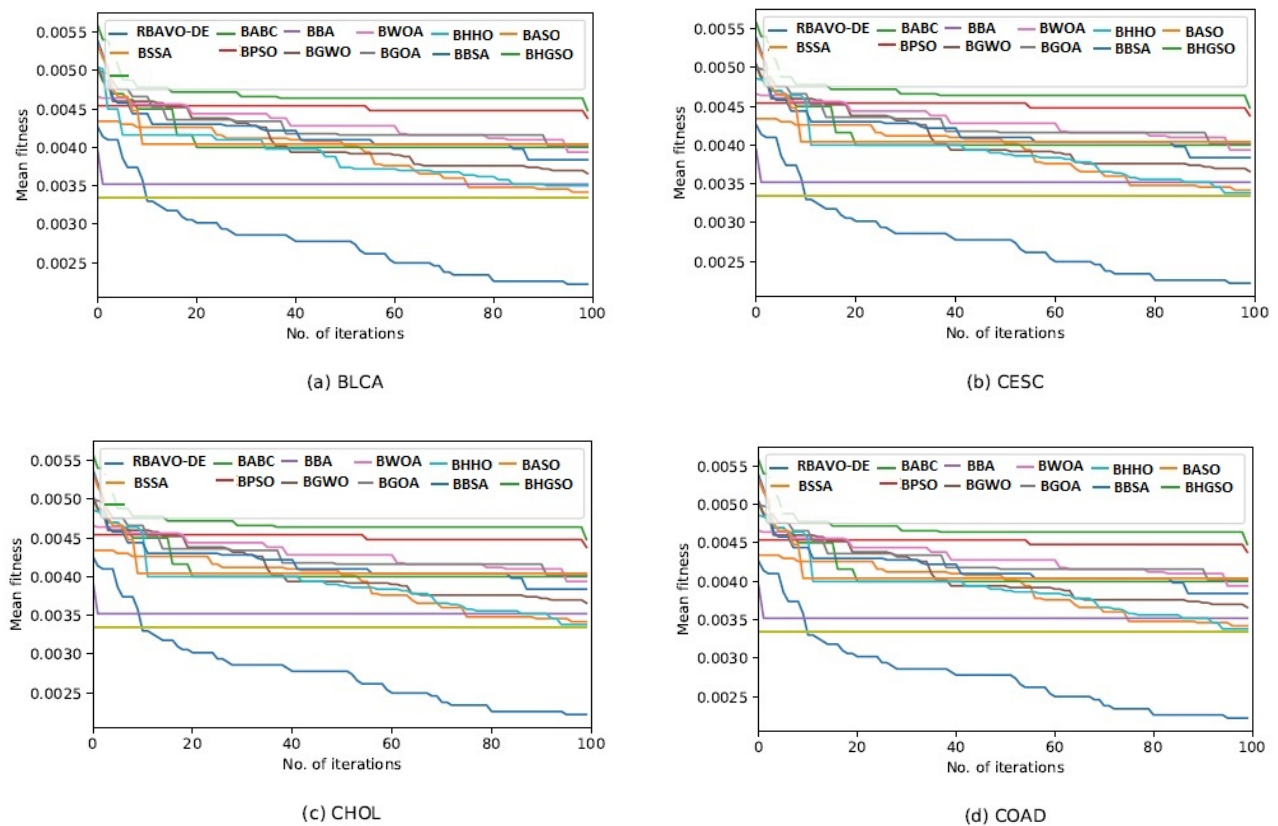
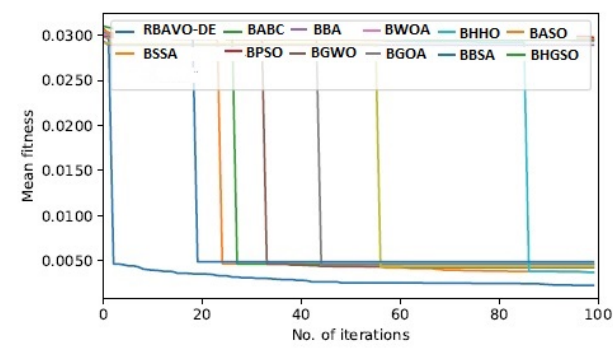
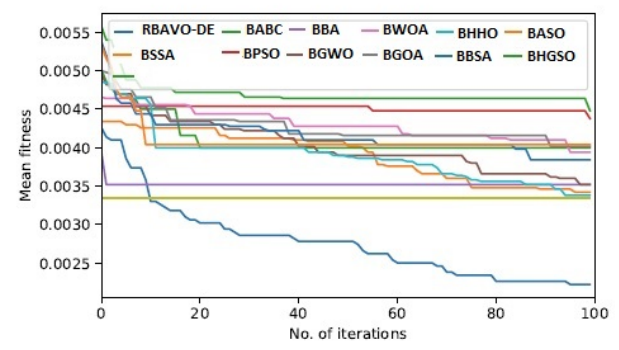


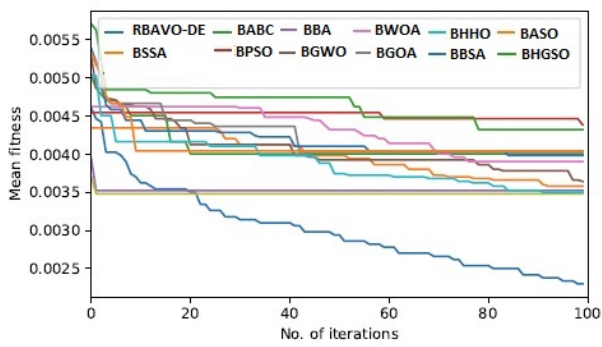
Figure A1. Cont.



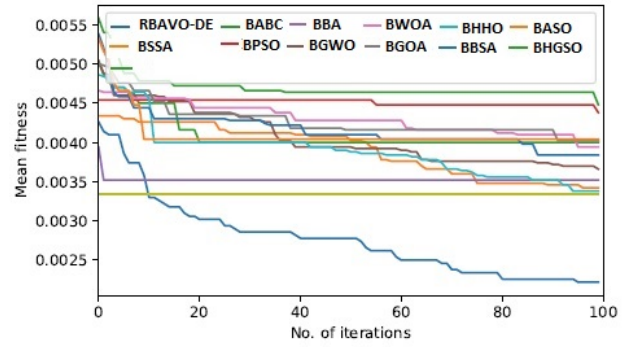
(e) ESCA



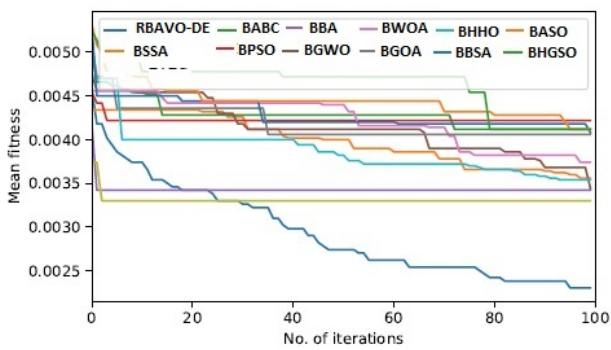
(f) GBM



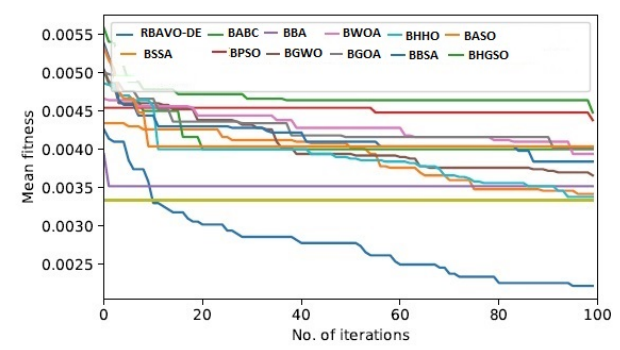
(g) HNSC



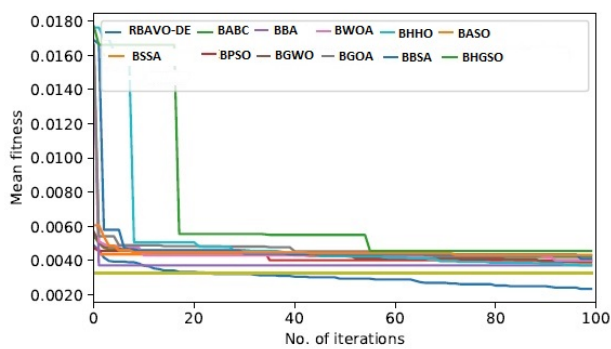
(h) KICH



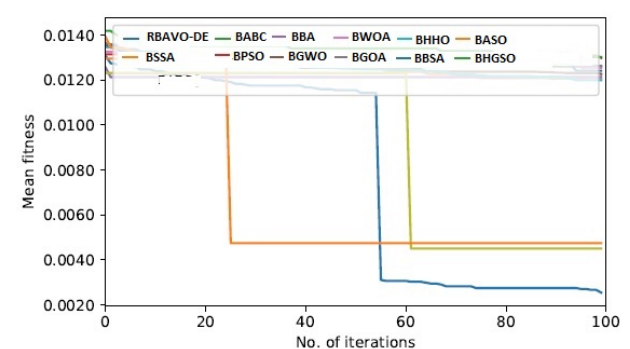
(i) KIRC



(j) KIRP

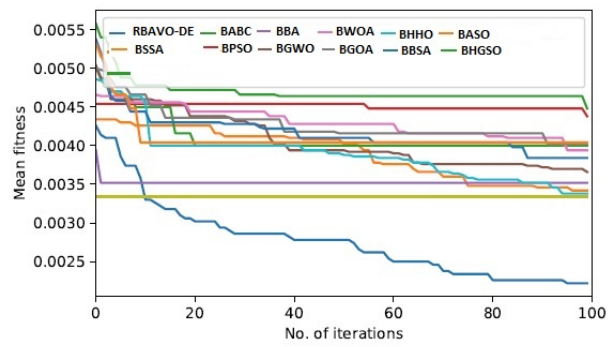


(k) LIHC

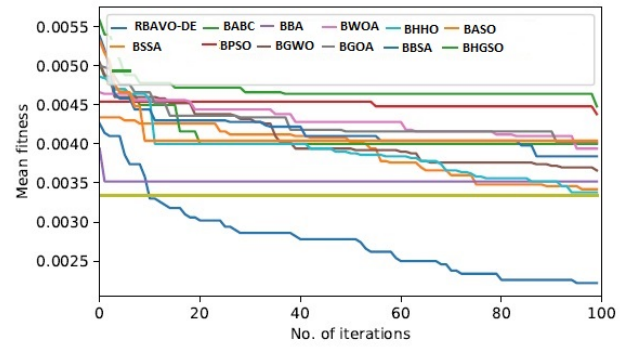


(l) LUAD

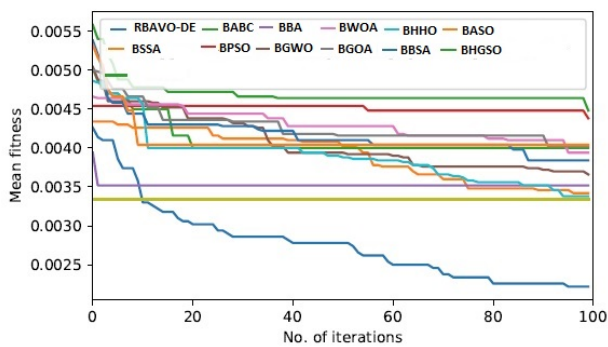
Figure A1. Cont.



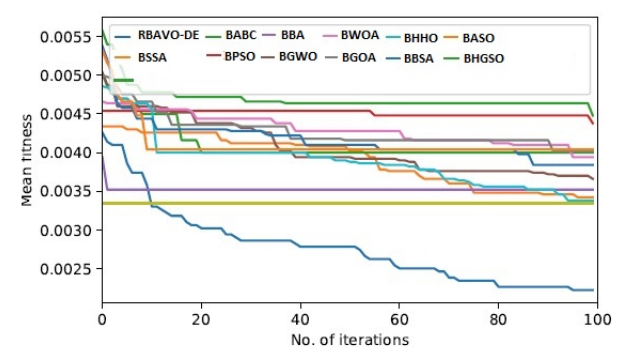
(m) LUSC



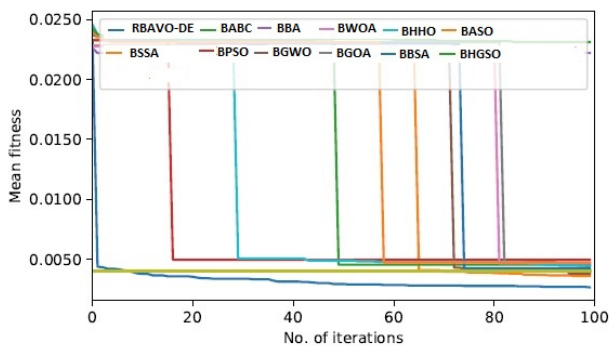
(n) PAAD



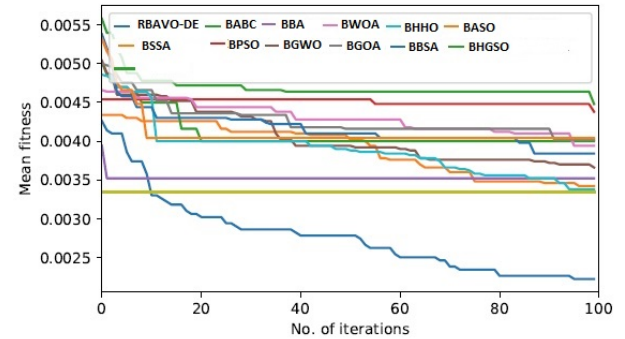
(o) PCPG



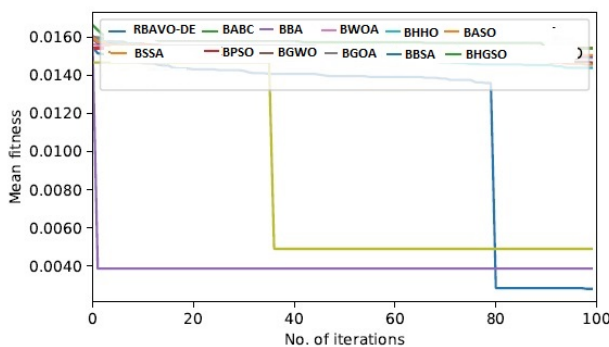
(p) READ



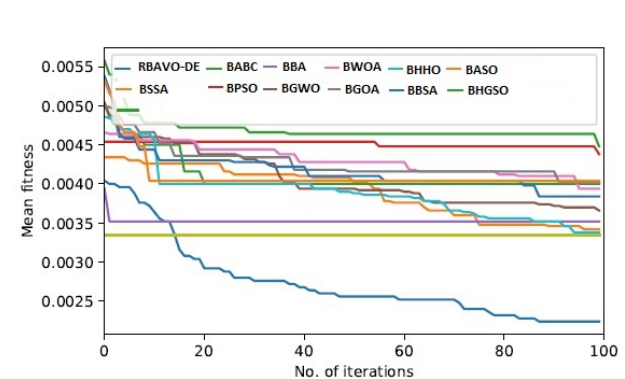
(q) SARC



(r) SKCM

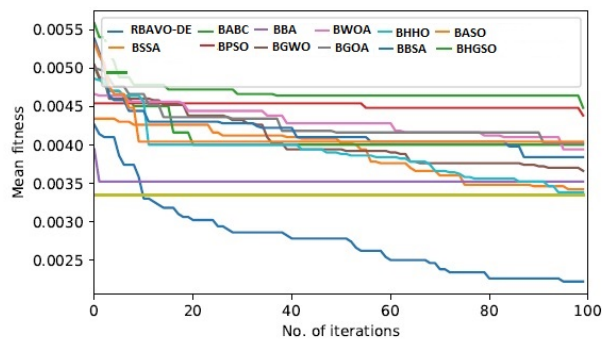


(s) STAD

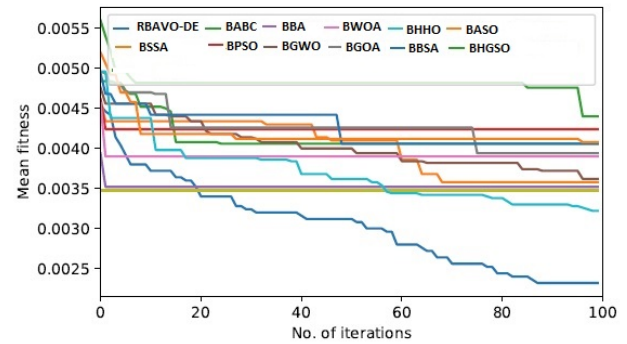


(t) THCA

Figure A1. Cont.



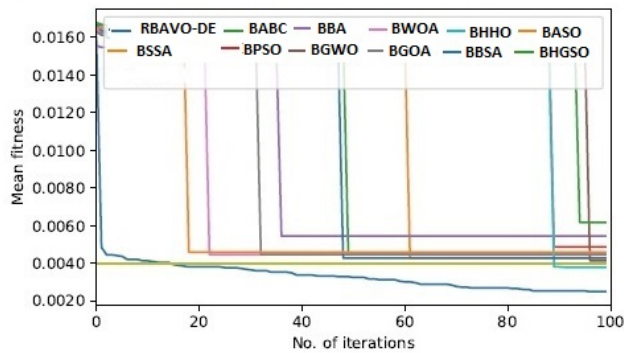
(u) THYM



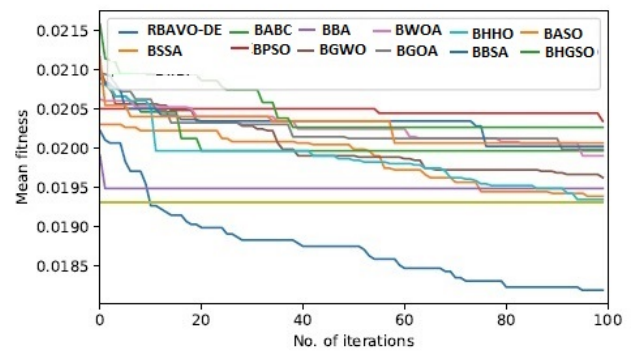
(v) UCEC

Figure A1. Convergence graphs of the proposed RBAVO-DE algorithm and competitive methods employing the SVM model on the overall benchmarks [57].

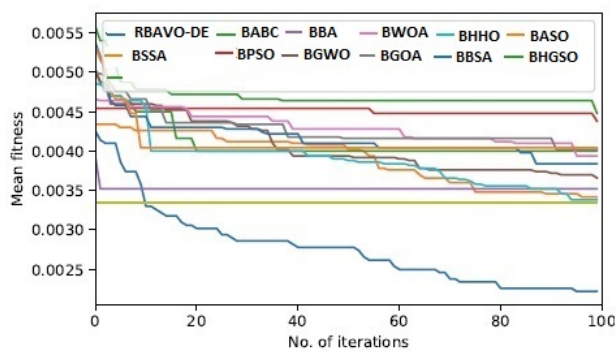
Appendix B.2. Convergence Graphs Employing the k-NN Model



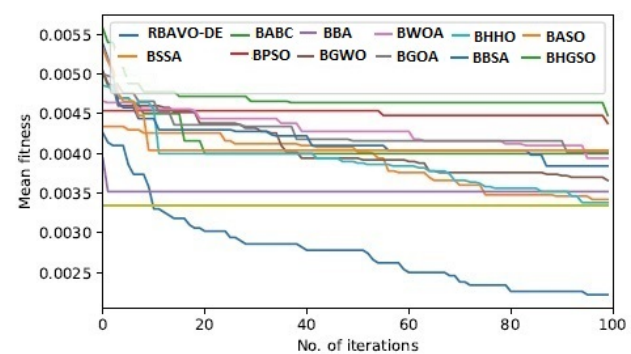
(a) BLCA



(b) CESC

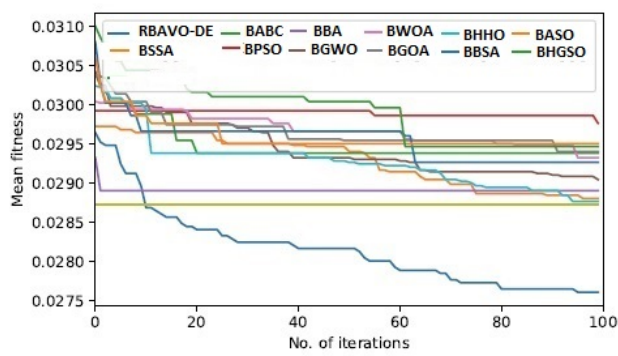


(c) CHOL

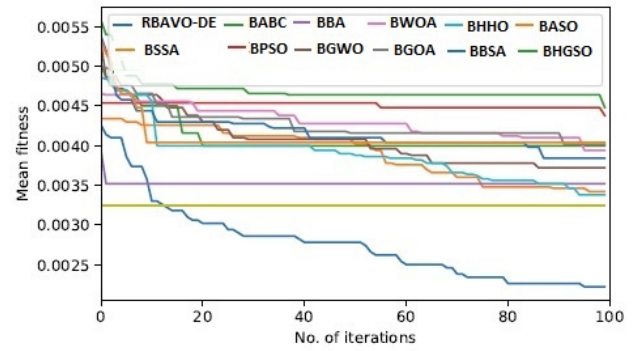


(d) COAD

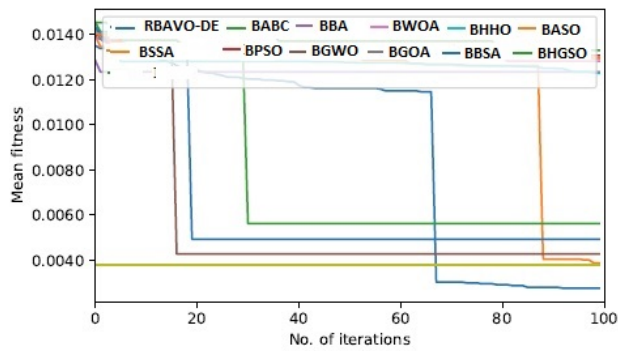
Figure A2. Cont.



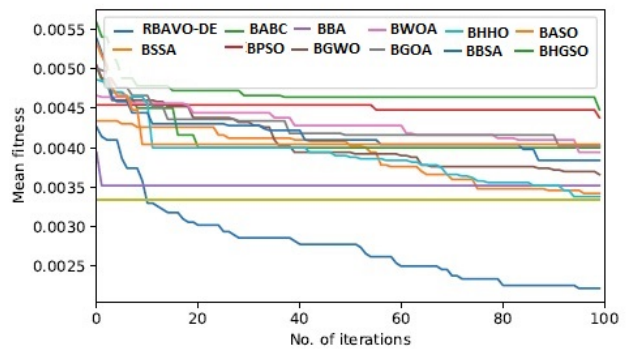
(e) ESCA



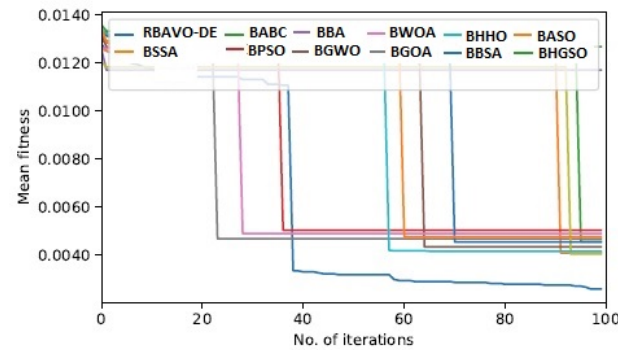
(f) GBM



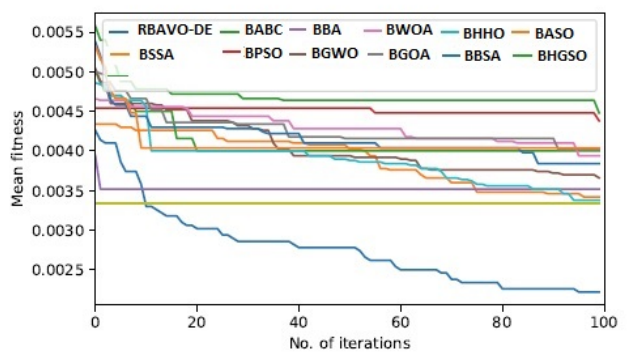
(g) HNSC



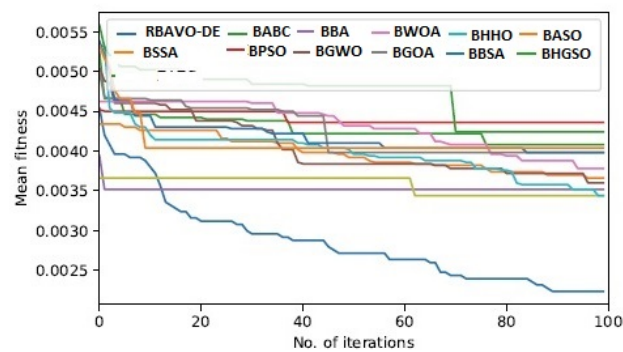
(h) KICH



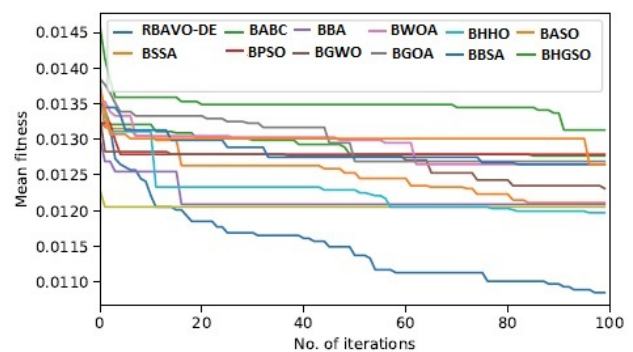
(i) KIRC



(j) KIRP

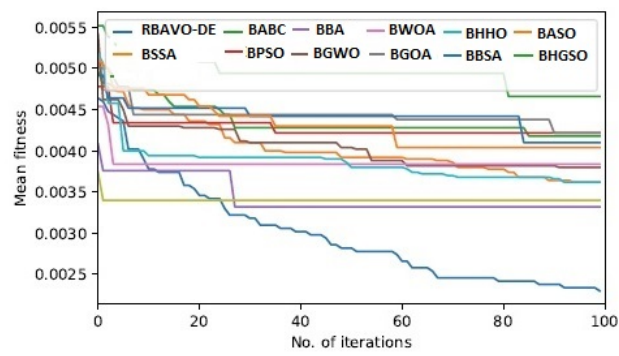


(k) LIHC

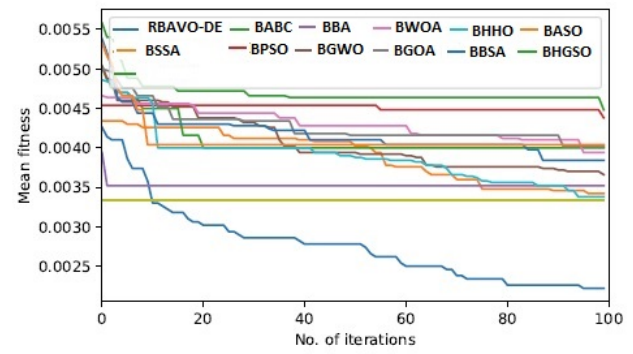


(l) LUAD

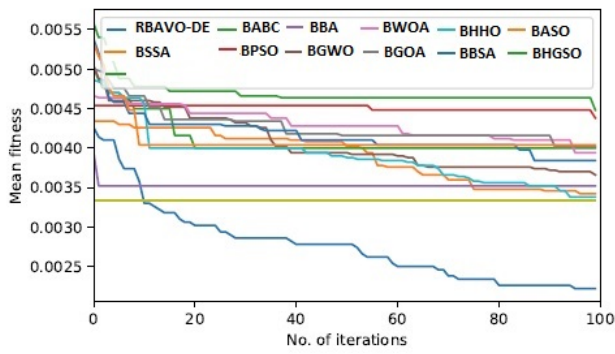
Figure A2. Cont.



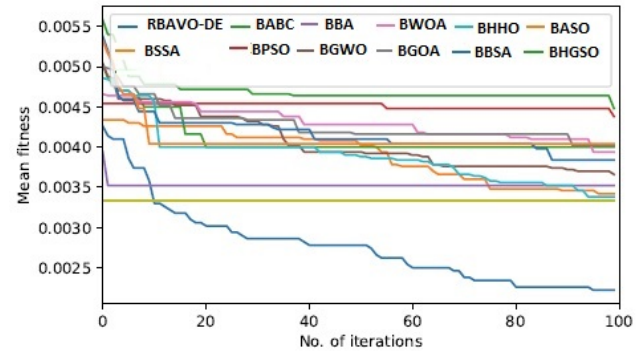
(m) LUSC



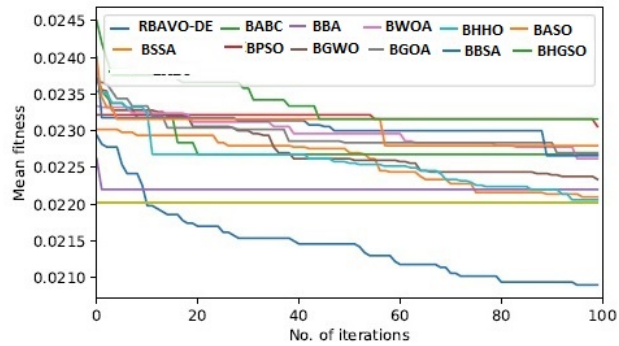
(n) PAAD



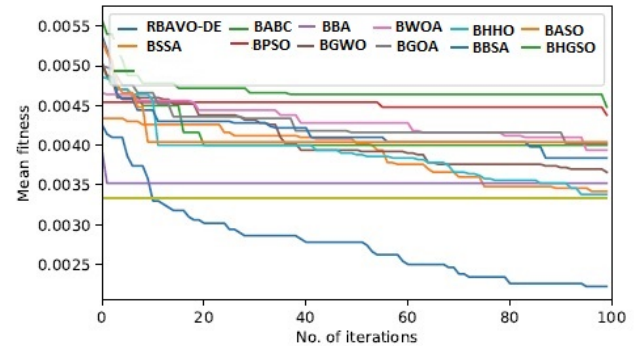
(o) PCPG



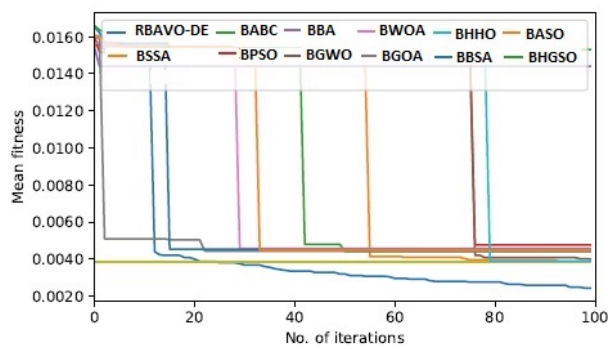
(p) READ



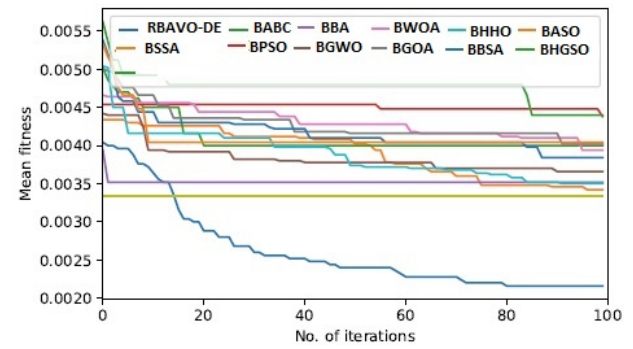
(q) SARC



(r) SKCM



(s) STAD



(t) THCA

Figure A2. Cont.

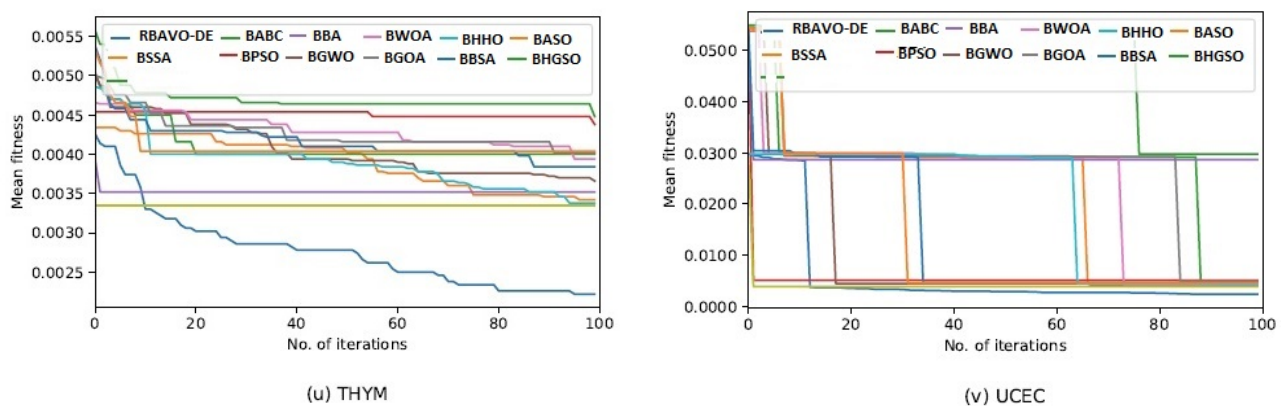


Figure A2. Convergence graphs of the proposed RBAVO-DE algorithm and competitive methods employing the k -NN model on the overall benchmarks (Cont.) [57].

References

- Estrada-Meza, C.; Torres-Copado, A.; Loreti González-Melgoza, L.; Ruiz-Manriquez, L.M.; De Donato, M.; Sharma, A.; Pathak, S.; Banerjee, A.; Paul, S. Recent insights into the microRNA and long non-coding RNA-mediated regulation of stem cell populations. *3 Biotech* **2022**, *12*, 270. [\[CrossRef\]](#) [\[PubMed\]](#)
- Kakati, T.; Bhattacharyya, D.K.; Kalita, J.K.; Norden-Krichmar, T.M. DEGnext: Classification of differentially expressed genes from RNA-seq data using a convolutional neural network with transfer learning. *BMC Bioinform.* **2022**, *23*, 17. [\[CrossRef\]](#)
- Zhao, S.; Fung-Leung, W.P.; Bittner, A.; Ngo, K.; Liu, X. Comparison of RNA-Seq and microarray in transcriptome profiling of activated T cells. *PLoS ONE* **2014**, *9*, e78644. [\[CrossRef\]](#)
- Chen, Z.; Luo, Z.; Zhang, D.; Li, H.; Liu, X.; Zhu, K.; Zhang, H.; Wang, Z.; Zhou, P.; Ren, J.; et al. TIGER: A web portal of tumor immunotherapy gene expression resource. *Genom. Proteom. Bioinform.* **2023**, *21*, 337–348. [\[CrossRef\]](#) [\[PubMed\]](#)
- Nunez-Garcia, J.; AbuOun, M.; Storey, N.; Brouwer, M.; Delgado-Blas, J.; Mo, S.S.; Ellaby, N.; Veldman, K.; Haenni, M.; Châtre, P.; et al. Harmonisation of in-silico next-generation sequencing based methods for diagnostics and surveillance. *Sci. Rep.* **2022**, *12*, 14372. [\[CrossRef\]](#)
- Wang, Z.; Gerstein, M.; Snyder, M. RNA-Seq: A revolutionary tool for transcriptomics. *Nat. Rev. Genet.* **2009**, *10*, 57–63. [\[CrossRef\]](#)
- Kim, W.J.; Choi, B.R.; Noh, J.J.; Lee, Y.Y.; Kim, T.J.; Lee, J.W.; Kim, B.G.; Choi, C.H. Comparison of RNA-Seq and microarray in the prediction of protein expression and survival prediction. *Front. Genet.* **2024**, *15*, 1342021. [\[CrossRef\]](#)
- Wang, M.; Chen, X.; Dai, Y.; Wu, D.; Liu, F.; Yang, Z.; Song, B.; Xie, L.; Yang, L.; Zhao, W.; et al. Concordance study of a 520-gene next-generation sequencing-based genomic profiling assay of tissue and plasma samples. *Mol. Diagn. Ther.* **2022**, *26*, 309–322. [\[CrossRef\]](#)
- Metzker, M.L. Sequencing technologies—The next generation. *Nat. Rev. Genet.* **2010**, *11*, 31–46. [\[CrossRef\]](#) [\[PubMed\]](#)
- Pandey, D.; Onkara Perumal, P. A scoping review on deep learning for next-generation RNA-Seq. data analysis. *Funct. Integr. Genom.* **2023**, *23*, 134. [\[CrossRef\]](#)
- Liu, S.; Yao, W. Prediction of lung cancer using gene expression and deep learning with KL divergence gene selection. *BMC Bioinform.* **2022**, *23*, 175. [\[CrossRef\]](#) [\[PubMed\]](#)
- Houssein, E.H.; Oliva, D.; Celik, E.; Emam, M.M.; Ghoniem, R.M. Boosted sooty tern optimization algorithm for global optimization and feature selection. *Expert Syst. Appl.* **2023**, *213*, 119015. [\[CrossRef\]](#)
- Joshi, A.A.; Aziz, R.M. A two-phase cuckoo search based approach for gene selection and deep learning classification of cancer disease using gene expression data with a novel fitness function. *Multimed. Tools Appl.* **2024**, *83*, 71721–71752. [\[CrossRef\]](#)
- Ramaswamy, R.; Kandhasamy, P.; Palaniswamy, S. Feature selection for Alzheimer's gene expression data using modified binary particle swarm optimization. *IETE J. Res.* **2023**, *69*, 9–20. [\[CrossRef\]](#)
- Cui, X.; Li, Y.; Fan, J.; Wang, T. A novel filter feature selection algorithm based on relief. *Appl. Intell.* **2022**, *52*, 5063–5081. [\[CrossRef\]](#)
- Alhenawi, E.; Al-Sayyed, R.; Hudaib, A.; Mirjalili, S. Feature selection methods on gene expression microarray data for cancer classification: A systematic review. *Comput. Biol. Med.* **2022**, *140*, 105051. [\[CrossRef\]](#) [\[PubMed\]](#)
- Parlak, B.; Uysal, A.K. A novel filter feature selection method for text classification: Extensive Feature Selector. *J. Inf. Sci.* **2023**, *49*, 59–78. [\[CrossRef\]](#)
- Albulayhi, K.; Abu Al-Haija, Q.; Alsuhbany, S.A.; Jillepalli, A.A.; Ashrafuzzaman, M.; Sheldon, F.T. IoT intrusion detection using machine learning with a novel high performing feature selection method. *Appl. Sci.* **2022**, *12*, 5015. [\[CrossRef\]](#)
- Fatima, A.; Nazir, T.; Nazir, A.K.; Din, A.M.U. An efficient Incremental Wrapper-based Information Gain Gene Subset Selection (IG based on IWSSr) method for Tumor Discernment. *Multimed. Tools Appl.* **2024**, *83*, 64741–64766. [\[CrossRef\]](#)
- Kaur, S.; Kumar, Y.; Koul, A.; Kumar Kamboj, S. A systematic review on metaheuristic optimization techniques for feature selections in disease diagnosis: Open issues and challenges. *Arch. Comput. Methods Eng.* **2023**, *30*, 1863–1895. [\[CrossRef\]](#)

21. Abd El-Mageed, A.A.; Gad, A.G.; Sallam, K.M.; Munasinghe, K.; Abohany, A.A. Improved Binary Adaptive Wind Driven Optimization Algorithm-Based Dimensionality Reduction for Supervised Classification. *Comput. Ind. Eng.* **2022**, *167*, 107904. [\[CrossRef\]](#)
22. Gad, A.G.; Sallam, K.M.; Chakraborty, R.K.; Ryan, M.J.; Abohany, A.A. An improved binary sparrow search algorithm for feature selection in data classification. *Neural Comput. Appl.* **2022**, *34*, 15705–15752. [\[CrossRef\]](#)
23. Hussien, R.M.; Abohany, A.A.; Abd El-Mageed, A.A.; Hosny, K.M. Improved Binary Meerkat Optimization Algorithm for efficient feature selection of supervised learning classification. *Knowl.-Based Syst.* **2024**, *292*, 111616. [\[CrossRef\]](#)
24. Abd El-Mageed, A.A.; Abohany, A.A.; Elashry, A. Effective Feature Selection Strategy for Supervised Classification based on an Improved Binary Aquila Optimization Algorithm. *Comput. Ind. Eng.* **2023**, *181*, 109300. [\[CrossRef\]](#)
25. Yin, Y.; Jang-Jaccard, J.; Xu, W.; Singh, A.; Zhu, J.; Sabrina, F.; Kwak, J. IGRF-RFE: A hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset. *J. Big Data* **2023**, *10*, 15. [\[CrossRef\]](#)
26. Nakao, H.; Imaoka, M.; Hida, M.; Imai, R.; Nakamura, M.; Matsumoto, K.; Kita, K. Determination of individual factors associated with hallux valgus using SVM-RFE. *BMC Musculoskelet. Disord.* **2023**, *24*, 534. [\[CrossRef\]](#) [\[PubMed\]](#)
27. Sarafrazi, S.; Nezamabadi-Pour, H. Facing the classification of binary problems with a GSA-SVM hybrid system. *Math. Comput. Model.* **2013**, *57*, 270–278. [\[CrossRef\]](#)
28. Cadenas, J.M.; Garrido, M.C.; MartíNez, R. Feature subset selection filter–wrapper based on low quality data. *Expert Syst. Appl.* **2013**, *40*, 6241–6252. [\[CrossRef\]](#)
29. Oh, I.S.; Lee, J.S.; Moon, B.R. Hybrid genetic algorithms for feature selection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 1424–1437.
30. Abdollahzadeh, B.; Gharehchopogh, F.S.; Mirjalili, S. African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Comput. Ind. Eng.* **2021**, *158*, 107408. [\[CrossRef\]](#)
31. El-Shafeiy, E.; Hassanien, A.E.; Sallam, K.M.; Abohany, A. Approach for training quantum neural network to predict severity of COVID-19 in patients. *Comput. Mater. Contin.* **2020**, *66*, 1745–1755. [\[CrossRef\]](#)
32. Yaqoob, A.; Verma, N.K.; Aziz, R.M. Optimizing gene selection and cancer classification with hybrid sine cosine and cuckoo search algorithm. *J. Med. Syst.* **2024**, *48*, 10. [\[CrossRef\]](#)
33. Joshi, A.A.; Aziz, R.M. Deep learning approach for brain tumor classification using metaheuristic optimization with gene expression data. *Int. J. Imaging Syst. Technol.* **2023**, *34*, e23007. [\[CrossRef\]](#)
34. Mahto, R.; Ahmed, S.U.; Rahman, R.U.; Aziz, R.M.; Roy, P.; Mallik, S.; Li, A.; Shah, M.A. A novel and innovative cancer classification framework through a consecutive utilization of hybrid feature selection. *BMC Bioinform.* **2023**, *24*, 479. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Neggaz, N.; Neggaz, I.; Abd Elaziz, M.; Hussien, A.G.; Abulaigh, L.; Damaševičius, R.; Hu, G. Boosting manta rays foraging optimizer by trigonometry operators: A case study on medical dataset. *Neural Comput. Appl.* **2024**, *36*, 9405–9436. [\[CrossRef\]](#)
36. Lyu, B.; Haque, A. Deep learning based tumor type classification using gene expression data. In Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, Washington, DC, USA, 29 August–1 September 2018; pp. 89–96.
37. Khalifa, N.E.M.; Taha, M.H.N.; Ali, D.E.; Slowik, A.; Hassanien, A.E. Artificial intelligence technique for gene expression by tumor RNA-Seq data: A novel optimized deep learning approach. *IEEE Access* **2020**, *8*, 22874–22883. [\[CrossRef\]](#)
38. Dillies, M.A.; Rau, A.; Aubert, J.; Hennequet-Antier, C.; Jeanmougin, M.; Servant, N.; Keime, C.; Marot, G.; Castel, D.; Estelle, J.; et al. A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. *Briefings Bioinform.* **2013**, *14*, 671–683. [\[CrossRef\]](#)
39. Xiao, Y.; Wu, J.; Lin, Z.; Zhao, X. A deep learning-based multi-model ensemble method for cancer prediction. *Comput. Methods Programs Biomed.* **2018**, *153*, 1–9. [\[CrossRef\]](#)
40. Liu, M.; Xu, L.; Yi, J.; Huang, J. A feature gene selection method based on ReliefF and PSO. In Proceedings of the 2018 10th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), Changsha, China, 7–8 January 2018; IEEE: New York, NY, USA, 2018; pp. 298–301.
41. Kononenko, I. Estimating attributes: Analysis and extensions of RELIEF. In Proceedings of the European Conference on Machine Learning, Catania, Italy, 6–8 April 1994; Springer: Berlin/Heidelberg, Germany, 1994; pp. 171–182.
42. Faris, H.; Mafarja, M.M.; Heidari, A.A.; Aljarah, I.; Ala'M, A.Z.; Mirjalili, S.; Fujita, H. An efficient binary salp swarm algorithm with crossover scheme for feature selection problems. *Knowl.-Based Syst.* **2018**, *154*, 43–67. [\[CrossRef\]](#)
43. Abdel-Basset, M.; Ding, W.; El-Shahat, D. A hybrid Harris Hawks optimization algorithm with simulated annealing for feature selection. *Artif. Intell. Rev.* **2020**, *54*, 593–637. [\[CrossRef\]](#)
44. Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [\[CrossRef\]](#)
45. Karaboga, D.; Basturk, B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **2007**, *39*, 459–471. [\[CrossRef\]](#)
46. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.; Mirjalili, S.M. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems. *Adv. Eng. Softw.* **2017**, *114*, 163–191. [\[CrossRef\]](#)
47. Khanesar, M.A.; Teshnehlab, M.; Shoorehdeli, M.A. A novel binary particle swarm optimization. In Proceedings of the 2007 Mediterranean Conference on Control & Automation, Athens, Greece, 27–29 June 2007; IEEE: New York, NY, USA, 2007; pp. 1–6.

48. Mirjalili, S.; Mirjalili, S.M.; Yang, X.S. Binary bat algorithm. *Neural Comput. Appl.* **2014**, *25*, 663–681. [[CrossRef](#)]
49. Emary, E.; Zawbaa, H.M.; Hassanien, A.E. Binary grey wolf optimization approaches for feature selection. *Neurocomputing* **2016**, *172*, 371–381. [[CrossRef](#)]
50. Hichem, H.; Elkamel, M.; Rafik, M.; Mesaaoud, M.T.; Ouahiba, C. A new binary grasshopper optimization algorithm for feature selection problem. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 316–328. [[CrossRef](#)]
51. Hussien, A.G.; Hassanien, A.E.; Houssein, E.H.; Bhattacharyya, S.; Amin, M. S-shaped binary whale optimization algorithm for feature selection. In *Recent Trends in Signal and Image Processing: ISSIP 2017*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 79–87.
52. Zhao, W.; Wang, L.; Zhang, Z. Atom search optimization and its application to solve a hydrogeologic parameter estimation problem. *Knowl.-Based Syst.* **2019**, *163*, 283–304. [[CrossRef](#)]
53. Meng, X.B.; Gao, X.Z.; Lu, L.; Liu, Y.; Zhang, H. A new bio-inspired optimisation algorithm: Bird Swarm Algorithm. *J. Exp. Theor. Artif. Intell.* **2016**, *28*, 673–687. [[CrossRef](#)]
54. Hashim, F.A.; Houssein, E.H.; Mabrouk, M.S.; Al-Atabany, W.; Mirjalili, S. Henry gas solubility optimization: A novel physics-based algorithm. *Future Gener. Comput. Syst.* **2019**, *101*, 646–667. [[CrossRef](#)]
55. Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. *Future Gener. Comput. Syst.* **2019**, *97*, 849–872. [[CrossRef](#)]
56. Normalized-level3 RNA-Seq Gene Expression Dataset. Available online: <https://gdac.broadinstitute.org/> (accessed on 20 December 2023).
57. El-Mageed, A.A.A.; Elkhoul, A.E.; Abohany, A.A.; Gafar, M. Gene selection via improved nuclear reaction optimization algorithm for cancer classification in high-dimensional data. *J. Big Data* **2024**, *11*, 46. [[CrossRef](#)]
58. Mafarja, M.; Mirjalili, S. Whale optimization approaches for wrapper feature selection. *Appl. Soft Comput.* **2018**, *62*, 441–453. [[CrossRef](#)]
59. Thaher, T.; Heidari, A.A.; Mafarja, M.; Dong, J.S.; Mirjalili, S. Binary Harris Hawks optimizer for high-dimensional, low sample size feature selection. In *Evolutionary Machine Learning Techniques*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 251–272.
60. Derrac, J.; García, S.; Molina, D.; Herrera, F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol. Comput.* **2011**, *1*, 3–18. [[CrossRef](#)]
61. Python Code for Gene Selection via Relief Binary African Vultures Optimization Integrated with Differential Evolution. Available online: https://github.com/D-Amr-Atef/Gene_Selection_RBAVO-DE.git (accessed on 18 May 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.