Article

Dynamic Candidate Solution Boosted Beluga Whale Optimization Algorithm for Biomedical Classification

Essam H. Houssein [1,*] and **Awny Sayed** [2]

1   Faculty of Computers and Information, Minia University, Minia 61519, Egypt
2   Information Technology Department, Faculty of Computing and Information Technology,
    King Abdulaziz University, Jeddah 21589, Saudi Arabia
*   Correspondence: essam.halim@mu.edu.eg

Abstract: In many fields, complicated issues can now be solved with the help of Artificial Intelligence (AI) and Machine Learning (ML). One of the more modern Metaheuristic (MH) algorithms used to tackle numerous issues in various fields is the Beluga Whale Optimization (BWO) method. However, BWO has a lack of diversity, which could lead to being trapped in local optimaand premature convergence. This study presents two stages for enhancing the fundamental BWO algorithm. The initial stage of BWO's Opposition-Based Learning (OBL), also known as OBWO,helps to expedite the search process and enhance the learning methodology to choose a better generation of candidate solutions for the fundamental BWO. The second step, referred to as OBWOD, combines the Dynamic Candidate Solution (DCS) and OBWO based on the k-Nearest Neighbor (kNN) classifier to boost variety and improve the consistency of the selected solution by giving potential candidates a chance to solve the given problem with a high fitness value. A comparison study with present optimization algorithms for single-objective bound-constraint optimization problems was conducted to evaluate the performance of the OBWOD algorithm on issues from the 2022 IEEE Congress on Evolutionary Computation (CEC'22) benchmark test suite with a range of dimension sizes. The results of the statistical significance test confirmed that the proposed algorithm is competitive with the optimization algorithms. In addition, the OBWOD algorithm surpassed the performance of seven other algorithms with an overall classification accuracy of 85.17% for classifying 10 medical datasets with different dimension sizes according to the performance evaluation matrix.

Keywords: Artificial Intelligence (AI); Beluga Whale Optimization (BWO); Dynamic Candidate Solution (DCS); Opposition-Based Learning (OBL); k-Nearest Neighbor (kNN)

MSC: 68Txx

1. Introduction

The availability of various medical data leads us to wonder if there are any effective and efficient ways to analyze these data and derive possibly novel and practical knowledge. Diagnosing various diseases presents one of the most-severe challenges for data analytics. The researchers focus their work in various ways, including creating high-accuracy prediction models, extracting if–then rules, and experimenting with new cut-off values for significant input variables. Diagnostic errors can result from a lack of sufficient information for an accurate diagnosis, a disruption in patient–clinician communication, a delayed or inaccurate diagnosis, or even a high degree of diagnostic complexity in a constrained amount of time. Clinicians may misdiagnose a patient, which may impact the treatment's result. The surroundings can also cause diagnostic mistakes and affect the equipment employed for the diagnosis [1].

A new trend has emerged in recent years: using computational intelligence to diagnose medical conditions [2]. Intelligent data classification activities can be used to categorize

various medical condition diagnosis techniques. The categorization strategies can be split into two groups based on the number of continually distributed groups. Binary classification (two-class task) is the initial classification distribution, which only distinguishes the data between the two classes. The second classification separates data from more than two classes and is called multi-classification (multi-class task) [3].

Machine Learning (ML) techniques are being used to help clinicians make decisions to improve healthcare services, such as reducing diagnostic errors and giving patients the right treatments [4]. Different ML techniques have been employed to diagnose various illnesses, such as Artificial Neural Networks (ANNs), which have achieved an accuracy of 95.63% in preventing diarrhea. In contrast, diarrhea is one of the leading causes of death worldwide [5]. Using clinical data, a Decision Tree (DT) accurately diagnoses and categorizes the thyroid with a 97.35% accuracy rate [6]. Random Forest (RF) has achieved 82.7% accuracy in identifying multiple indicators for the breast cancer survival rate, including the stage of the disease and the size of the tumor [7], and Support Vector Machine (SVM) achieves an accuracy rate of more than 90% on the prediction of influenza [8]. A hybrid algorithm enhances two or more ML algorithms. These hybrid models can achieve remarkable results and address issues that an individual algorithm cannot [9]. Better results may also be obtained using other ML techniques such as Latent Dirichlet Allocation (LDA) [10], Recurrent Neural Networks (RNNs) [11], Naive Bayes (NB) [12], and Logistic Regression (LR) [13].

Medical disease identification is a rapidly expanding area of research in the science of AI. Many efforts have been made in recent years to improve medical disease identification since mistakes and issues with disease diagnosis can result in seriously incorrect medical therapy. In the biomedical field, MH algorithms have been used frequently to diagnose medical disorders and promise improved perception and disease prediction accuracy [14,15].

Multiple medical tasks can now be improved with AI [16]. AI algorithms can find patterns in a dataset to achieve diagnostic tools. AI has already demonstrated good diagnostic accuracy in other medical specialties. In some specialized domains, it can match present diagnostic capabilities. AI could be employed in psychiatry for diagnostic purposes to support ongoing patient assessments or medication recommendations. AI has also been investigated for enhancing categorization and diagnosis capacities. It has also been employed in identifying suicide risk and the diagnosis of mood disorders [17].

AI can be employed for a variety of purposes. Algorithms for ML are often categorized as either supervised or unsupervised learning. Supervised learning was the primary class of ML algorithms employed in the included studies. Patterns in a dataset connected to a result can be found using supervised learning techniques. Regression and classification are the two divisions of supervised algorithms. Data can be categorized into many groups using classification techniques. Patients can be categorized into many groups using patterns. Classification tasks can be carried out using DT, SVM, and RF. To forecast quantitative data, regression methods are used. This class includes regression methods such as the most Least Absolute Shrinkage and Selection Operator (LASSO) and LR [16].

Overfitting refers to fitting an AI model based on data noise or inaccuracy rather than the true relationship. It is one of AI's drawbacks. Small data samples or having too many parameters for the data can contribute to overfitting. One method for reducing overfitting is cross-validation. Using this method, the dataset is divided into several groups, then into training and validation data. Because of this, a different dataset is used to train and evaluate the statistical model for each group. This method lessens the possibility of having an overly optimistic estimate [18].

Other strategies, such as the dropout rate, are also applied to reduce overfitting. Dropout is a regularization method for Neural Networks (NNs) to lessen overfitting and enhance generalization [19]. During training, it is intended to arbitrarily ignore the neurons (and connections they form) in the NN. The same input data can result in various outputs because the NN architecture changes at every inference. The idea is that it forces the units to be stronger and less codependent. Cross-validation and dropout differ primarily in the

source of randomization. In cross-validation, the data are randomly split into training and validation sets, whereas in dropout, the neural units are randomly removed. The majority of recent research demonstrating AI's ability to detect mental health disorders on social media has been published.

In this paper, we developed an improved BWO called OBWOD by integrating the Opposition-Based Learning (OBL) strategy and Dynamic Candidate Solution (DCS) with the original BWO. Based on the proposed OBWO, the classifier kNN was established for medical diagnosis. OBWOD was compared with the Whale Optimization Algorithm (WOA) [20], the Hunger Games Search (HGS) [21], the Sine Cosine Algorithm (SCA) [22], Harris Hawks Optimization (HHO) [23], the weIghted meaN oF vectOrs (INFO) [24] algorithm, Moth–Flame Optimization (MFO) [25], and the original BWO [26] algorithm based on the kNN classifier. The performance of the proposed OBWOD method was evaluated using the CEC'22 test suite. Then, OBWOD was combined with the kNN classifier to choose the best features in 10 biomedical datasets with varying dimensions. Performance measures assessed how well the developed model performed. The contributions of this paper are summarized as follows:

- BWO has a lack of diversity, which could lead to being trapped in local optima.
- OBL can enhance the exploration of BWO by exploring more potential regions that produce BWO.
- DCS and OBWO boost variety and improve the consistency of the selected solution by giving potential candidates a chance to solve the given problem with a high fitness value, which produces the OBWOD approach.
- In comparison to the other seven MH algorithms, OBWOD is used to resolve global optimization issues based on the CEC'22 test suite.
- The OBWOD approach based on the kNN classifier was built for biomedical classification tasks and estimated on ten disease datasets with different dimension sizes extracted from the UCI repository.
- The experimental results demonstrated the superiority of OBWOD over seven other competitors according to the performance evaluation matrix.

The remainder of this paper is organized as follows. A summary of relevant work is included in Section 2, while Section 3 discusses the BWO algorithm and other methodologies. The proposed OBWOD approach is explained in Section 4, and the experimental assessment and discussion are presented in Section 5. The conclusions and future research are covered in Section 6.

2. Literature Reviews

In [27], Hameed et al. proposed an ML technique to support the initial clinical diagnosis by using 3473 registers of patients older than 6 years who were treated for possible influenza and then submitted a sample to a PCRRT test to confirm the diagnosis. Support vector machine was the superior ML technique, with a sensitivity of 0.9715 and a specificity of 0.9285.

In addition, in [28], Bhattacharya et al. proposed the first use of ML for identifying Hypertrophic Cardiomyopathy (HC) patients with Ventricular Arrhythmias (VArs) using clinical variables, employing the LR and NB classifiers. While resolving the imbalance in the clinical data, the proposed model outperformed other Sudden Cardiac Death (SCD) prediction algorithms already in use (C-index).

In [29], Ouyang et al. proposed a unique online attention module using a $3D$ Convolutional Neural Network (CNN) to concentrate on the infection regions in the lungs. The uneven distribution of the infection zones' diameters between COVID-19 and CAP should be noted; this is partially a result of the quick progression of COVID-19 following the beginning of the symptoms. The results demonstrated that the COVID-19 images can be recognized by this method with an AUC of 0.944 and an accuracy of 87.5%. With this performance, the proposed algorithm may help radiologists diagnose COVID-19 from CAP, especially at the beginning of the COVID-19 outbreak.

Mohan et al. [30] proposed a new technique for predicting cardiovascular diseases: Hybrid RF with a Linear Model (HRFLM). The approach uses a hybrid model made up of RF and a linear model based on clinical data. The model's performance was better than previous ML and soft-learning-based models. It had an 88.7% accuracy rate.

For the dimension reduction of datasets, Ghazal and Taleb [31] proposed that Advanced Harmony Searching Optimization (AHSO) can improve the ovarian cancer diagnosis method. The model put out in this study is also capable of an early cancer diagnosis with high accuracy and a low root-mean-squared error (RMSE). The RMSE, SOM, and NN methods demonstrated detection and precision of ovarian cancer of 94% and 0.029, respectively. An effective classification method with a lower failure rate was developed using optimization (AHSO).

In addition, Calp and Hanefi proposed SVM combined with a Cognitive Development Optimization Algorithm (SVM-CoDOA) for general medical diagnosis in [32]. An SVM, trained by the CoDOA, makes up the system. Optimization algorithms must be used to train and enhance ML approaches and demonstrate the efficacy of the SVM-CoDOA hybrid formation.

For concentrating on features, in [33], Rahman et al. proposed the LASSO, mutual information, and recursive feature elimination techniques in addition to three classification algorithms (kNN, RF, and NB). These were carried out in this study using a 10-fold cross-validation approach. They also used RF and the bagging ensemble technique to enhance the result. Through recursive feature elimination with bagging, the proposed model achieved an accuracy of 85.18%.

It is critical to have a lightweight and reliable ML solution for categorization because there is a vast amount of labeled or unlabeled data. Several optimizers have been implemented to improve the inclusive performance of ML models [34,35]. ML algorithms have used an increasing amount of medically unstructured data to anticipate intuitions. However, drawing much intuition from that data is difficult. ML researchers have therefore used modern optimizers and cutting-edge feature selection strategies to overcome and improve the performance accuracy [36]. This study aimed to determine how ML and optimization algorithms affect medical diagnosis by providing an overview of the usage of ML and optimization algorithms to diagnose various diseases through a thorough evaluation of research papers, as shown in Table 1.

Table 1. Related works in the literature based on ML and optimization algorithms.

Ref.	Datasets	Used Techniques	Results
[37]	Dental disease	Using OBL, Cauchy mutation operators, and a modified version of Gaining– Sharing Knowledge (GSK).	Average accuracy = 88%.
[38]	Human heart diseases	GridSearchCV based on extreme gradient boosting classifier	Accuracy = 99.03%.
[39]	COVID-19	Tsukamoto Fuzzy inference system with PSO	Accuracy = 0.9914, precision = 0.9935, recall = 0.9814, F-measure = 0.9896, and AUC = 0.9903.
[40]	Breast, cervical, and lung cancer	A machine-learning-based feature modeling	Breast accuracy = 99.62%, cervical accuracy = 96.88%, and lung accuracy = 98.21%.
[41]	Breast cancer	Convolutional neural networks and advanced thermal exchange optimization algorithm	Accuracy = 93.79%, specificity = 67.7%, and recall = 96.89%.
[42]	Heart disease	kNN with ANN	Accuracy = 94.2%.
[43]	Breast cancer	Visual Geometry Group (VGG)-16 and VGG-19 pre-trained CNNs	Accuracy = 97.1%, specificity = 97.9%, and recall = 96.3%.
[44]	Cardiovascular disease	REP tree, M5P tree, random tree, linear regression, NB, J48, and JRIP	Random tree's accuracy = 100%.

Ref.	Datasets	Used Techniques	Results
[45]	Breast cancer	Optimized deep residual learning model, Improved Marine Predators Algorithm (IMPA), IMPA-ResNet50	Accuracy = 98.32%.
[46]	Skin cancer	Grasshopper Optimization Algorithm (GOA) with AlexNet and extreme learning machine network	Accuracy = 98% and sensitivity = 93%.
[47]	Heart disease	SVM algorithm with X^2 statistical FS	Accuracy = 89.7%.
[48]	Glaucoma clinical data	DL model built on CFP classified	Accuracy = 97%.
[49]	Lymphography	Principle Component Analysis (PCA), fuzzy weighting pre-processing, and Adaptive Neuro-Fuzzy Inference System (ANFIS)	Accuracy = 88.83%.

3. Preliminaries

3.1. K-Nearest Neighbor

KNN, another name for nearest neighbor classification, is based on the idea that the patterns closest to a target pattern X', for which we are looking for a label, provide important information. KNN gives the most K-nearest ways in the data space of a class label. We must be able to define a similarity measure in the data space [50,51]. The calculation is given by Equation (1).

$$\left| X' - X_j \right|^p = \left(\sum_{i=1}^{q} \left| (X_i)' - (X_i)_j \right|^p \right)^{\frac{1}{p}} \tag{1}$$

This, for $p = 2$, corresponds to the Euclidean distance. The issue is how to select K or for what neighborhood size the best categorization result is achieved. This issue is sometimes referred to as a model. Selection and various methods, such as cross-validation, pick the optimal model and parameters. We set $k = 5$ after looking through the literature [52,53].

3.2. Opposition-Based Learning

OBL's idea was first presented by Tizhoosh [54]. The primary concept behind this optimization is to simultaneously analyze an estimate and its corresponding opposing estimate closer to the overall optimum to discover a better candidate solution. It has been used in numerous soft computing fields relatively quickly. Consider the points $x \in [y, z]$ and $y, z \in R$. It is possible to find the opposite point of x represented by x_{op} indicated by (2).

$$x_{op} = y + z - x \tag{2}$$

Many researchers use the idea of opposing numbers to improve their understanding of, ability to use, and optimization of MH algorithms [55].

3.3. Dynamic Candidate Solution

This section presents the dynamic aspects of potential dynamic arithmetic solutions. Exploration and exploitation are the two critical stages of MH algorithms, and maintaining a healthy balance between the two is crucial for the algorithm. Each solution renews its positions dynamically from the best-obtained solution during the optimization process in the proposed dynamic version to emphasize exploration and exploitation [56], where, as shown by Equations (3) and (4), the introduction of the DCS function results from the influence of the candidate solution's lowering percentage and where its value fell during each iteration.

$$DCS(0) = 1 - \sqrt{\frac{it}{Max_{it}}} \tag{3}$$

$$DCS(it + 1) = DCS(it) * 0.99 \tag{4}$$

where it is the current iteration and Max_{it} is the maximum number of iterations.

3.4. Beluga Whale Optimization Algorithm

BWO [26] is a swarm-based algorithm for solving optimization problems that draws inspiration from beluga whales' activities such as swimming, hunting for prey, and whale fall. To improve BWO's capability of convergence, the exploitative phase uses the Levy flight function. We first created BWO motivated by beluga whales' behaviors such as swimming, hunting, and whale fall. The following establishes the BWO mathematical model. Beluga whales are considered the search agents because of the population-based process of BWO, and each beluga whale is a candidate solution that is updated during optimization. The matrix of search agent positions is represented by Equation (5).

$$X = \begin{bmatrix} X_{11} & X_{12} & ... & X_{1D} \\ X_{21} & X_{22} & ... & X_{2D} \\ X_{31} & X_{32} & ... & X_{3D} \\ .. & .. & .. \\ .. & .. & .. \\ X_{N1} & X_{N2} & ... & X_{ND} \end{bmatrix} \tag{5}$$

where N is the beluga whale population size and D is the dimension. The balancing factor B_f described by Equation (6) determines whether the BWO algorithm switches from exploration to exploitation.

$$B_f = B_0(1 - T/2 * T_{max}) \tag{6}$$

1. Exploration phase: Beluga whales' swimming behavior is considered when establishing the BWO exploring phase. Beluga whales can engage in social–sexual behaviors in various postures, as evidenced by the behaviors observed in beluga whales kept in human care, such as a pair of closely spaced beluga whales swimming in a coordinated manner. As a result, beluga whales' positions are updated as shown in Equation (7).

$$\begin{pmatrix} X_{ij}^{T+1} = X_{i,p_j}^T + \left(X_{r,p_1}^T - X_{i,p_j}^T\right)(1 + r_1) * sin(2\pi r_2), j = even \\ X_{ij}^{T+1} = X_{i,p_j}^T + \left(X_{r,p_1}^T - X_{i,p_j}^T\right)(1 + r_1) * cos(2\pi r_2), j = odd \end{pmatrix} \tag{7}$$

where T is the current iteration, X_{ij}^{T+1} denotes the next iteration, r_1 and r_2 denote random numbers between (0, 1), and j indicates the new position of the ith beluga whale in the jth dimension.

2. Exploitation phase: The Levy flight strategy is employed in BWO's exploitative phase to improve convergence. With the Levy flight technique as our supposition, the mathematical model is shown in Equation (8).

$$X_i^{T+1} = r_3 * X_{best}^T - r_4 * X_i^T + C_1 * L_F * \left(X_r^T - X_i^T\right) \tag{8}$$

where C_1 is calculated by Equation (9), Levy flight function L_F is calculated by Equation (10), and σ is calculated by Equation (11).

$$C_1 = 2r_4 * \left(\frac{1 - T}{T_{max}}\right) \tag{9}$$

$$L_F = 0.05 * \left(\frac{u * \sigma}{|v|^{\frac{1}{\beta}}}\right) \tag{10}$$

$$\sigma = \left(\frac{(1+\beta) * sin\left(\frac{\pi*\beta}{2}\right)}{((1+\beta)/2) * \beta * 2^{\left(\frac{\beta-1}{2}\right)}} \right) \tag{11}$$

where the default constant was set to 1.5 and u and v are random values with a normal distribution calculated by Equations (12) and (13) [57].

$$u = randn(1, dim) * \sigma \tag{12}$$

$$v = randn(1, dim) \tag{13}$$

3. Whale fall phase: We used the whale fall probability from the population's individuals as our arbitrary premise to simulate slight changes in the groups to represent the behavior of whale fall in each iteration. We presumed that these beluga whales have relocated or have been fired at and dropped into the deep ocean. The locations of beluga whales and the magnitude of a whale fall are used to determine the updated position to maintain a steady population size, as calculated by Equation (14).

$$X_i^{T+1} = r_5 * X_i^T - r_6 * X_r^T + r_7 * X_{step}^T \tag{14}$$

where X_{step} is calculated by Equation (15).

$$X_{step} = (u_b - l_b) * exp(-C_2 * T/T_{max}) \tag{15}$$

where C_2 is calculated by Equation (16).

$$C_2 = 2 * W_f * n \tag{16}$$

where W_f is calculated by Equation (17).

$$W_f = 0.1 - 0.05 * T/T_{max} \tag{17}$$

4. The Proposed OBWOD Approach

The OBWOD optimization algorithm is explained in this section. Firstly, a discussion of the BWO method's drawbacks is provided. Then, numerous crucial concepts including initialization, updating, and evaluation, which are all components of the OBWOD algorithm, are next covered. After that, the OBWOD termination process is described.

4.1. Drawbacks of the Original BWO

The BWO method has a drawback called premature convergence, which prevents the algorithm from discovering the best solution since it traps the program in a local optimum. To choose a better generation of candidate solutions for the core BWO, thereby accelerating the search process and improving the learning approach and to increase the variety and strengthen the consistency of the chosen solution, the OBWOD combines DCS and OBWO based on the kNN classifier. The improvement is made by giving potential candidates a chance to solve the problem with a high fitness value. The three phases of the proposed method, including the exploration, exploitation, and whale fall phases, are shown in a flowchart of the OBWOD algorithm in Figure 1. The process of classifying 10 medical datasets using the proposed OBWOD algorithm is shown in Figure 2.
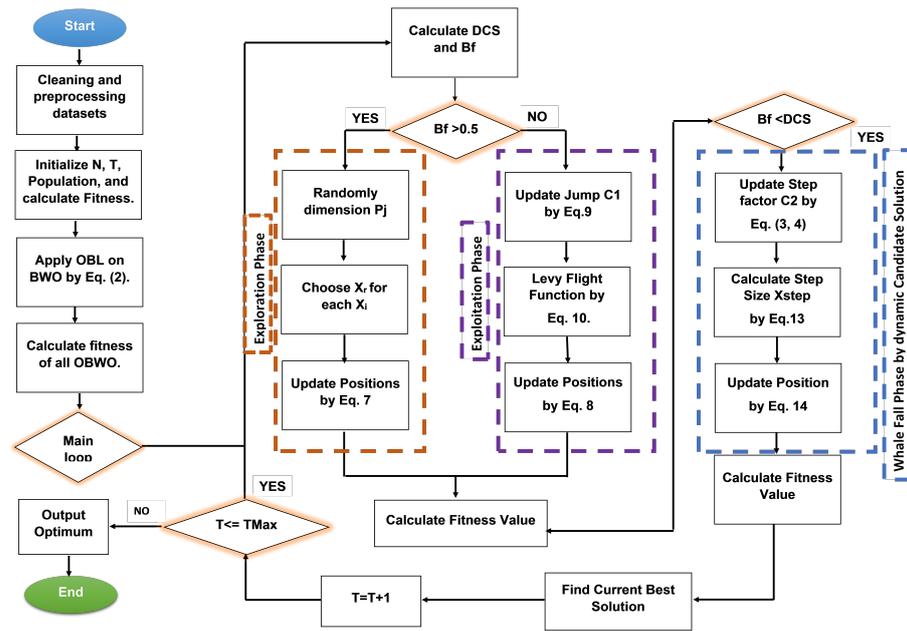
Figure 1. Flowchart of proposed OBWOD algorithm.
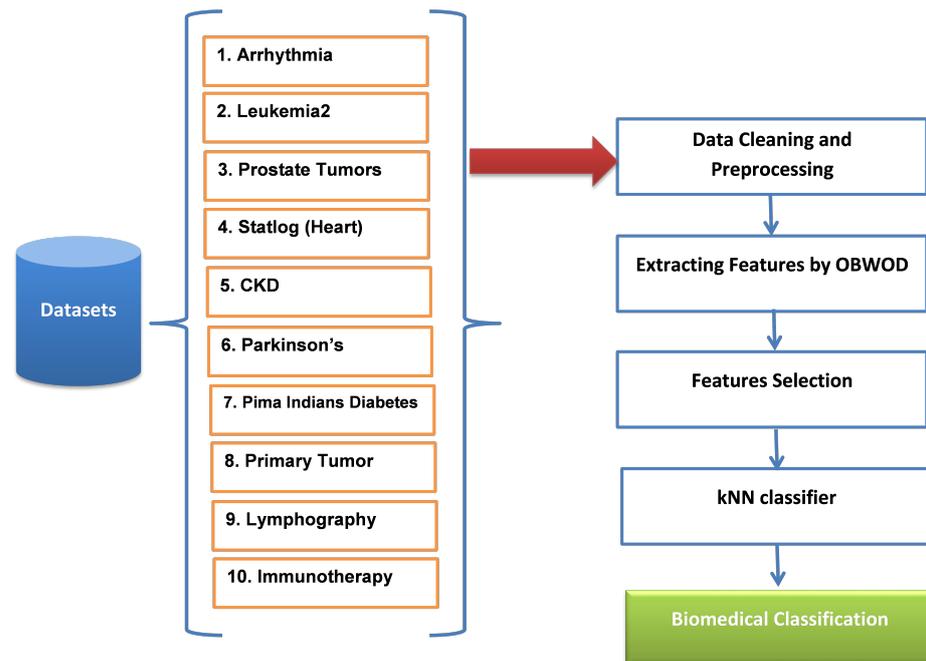
Figure 2. The process of classifying medical datasets.

4.2. Fitness Function

The fitness function (f_{obj}) evaluates how closely a specific solution adheres to the ideal solution to the desired problem. It determines how appropriate a solution is, as shown in Equation (18).

$$Fit_i = \alpha * Err_i + \beta * d_i / D \qquad (18)$$

whereas *beta* = 1 − *alpha* and *alpha* = 0.7. The ratio of the number of chosen features (d_i) to the classification error rate (Err_i) is balanced by the factor α.

4.3. The Major Stages of the OBWOD

We will now demonstrate how the OBWOD algorithm works. The premature convergence issue is resolved by OBWOD, which also enhances BWO's capacity for local search. Algorithm 1 provides the pseudo-code for the OBWOD algorithm.

The following three steps comprise the proposed OBWOD solution, as follows.

4.3.1. Initialization Phase

Due to the population-based nature of BWO, beluga whales are regarded as the search agents, and each one is a potential solution that is modified throughout the optimization. The fitness function (f_{obj}) is computed by Equation (18), and the initialization process (beluga whales X) is generated by Equation (5).

4.3.2. Solution Update Phase

The solution update phase contains two major steps, as shown:

The first step: To select a better generation of candidate solutions for the basic BWO, OBL helps to speed up the search process and improve the learning approach by using Equation (2). We allowed each population to evolve following an OBWOD position update for all individuals to increase the solution accuracy, accelerate convergence, and avoid becoming stuck in the local optimum. The exploratory sub-population increases its location to enlarge the search area and increase its global exploration potential. The exploitative sub-population performs a deep local search close to the current best solution to speed up convergence and improve the quality of the solution. The tiny population revises its position in response to OBL, which produces new individuals to replace those who are less fit and further broadens the population's genetic variation.

The second step: We used DCS to increase the variety and improve the consistency of the chosen solution by giving potential candidates a chance to solve the given problem with a high fitness value by using Equations (3) and (4) in the whale fall phase instead of using Equation (14). The fitness values of the new population are then calculated to find the optimum solution. This process is repeated until the termination condition (i.e., the maximum number of iterations).

4.3.3. Classification Phase

The OBWOD approach returns the best candidate solution in the prior stage. The original data are only used to keep features that are 1 in P^*. We used the *hold-out* classification approach, which randomly divides the dataset into two parts: 20% for the testing set and 80% for the training set. It is critical to emphasize that each experiment was run 20 times in isolation to obtain relevant results. After searching the literature for a plausible parallel, the number of classifiers ($k = 5$) was decided [52,53].

4.4. The Computational Complexity of OBWOD

The initialization, fitness assessment, and updating of the beluga whales are the three processes that comprise the computational complexity of BWO, which is a crucial parameter to evaluate its effectiveness. Remember that the initialization process's computational complexity for beluga whales is $O(n)$ and using OBL does not increase the computational complexity. The computational cost of the exploration and exploitation phase was estimated to be $O(n.T_{max})$, where n is the number of iterations and T_{max} is the maximum number of iterations. The whale fall probability Wf and balance factor Bf, which have an approximate value of $O(0.1.n.T_{max})$, have an impact on the computational complexity during the whale fall phase. As a result, the computational complexity of OBWOD was estimated to be around $O(n.(1 + 1.1.T_{max}))$.

Algorithm 1 The pseudo-code of the proposed OBWOD algorithm.

Inputs: Define the algorithm parameters, including population N and maximum iteration T_{Max}.

Outputs: Best solution P^*.

Initialize T_{Max} and N by applying OBL using Equation (2) and evaluating the fitness values.

while $(T < T_{Max})$ **do**

    Obtain the probability of DCS by Equations (3) and (4), and balance factor B_f by Equation (6)

    for (each beluga whale (X_i)) **do**

        if $B_f(i) > 0.5$ **then**

            Generate p_j $(j = 1,2,\ldots,d)$ randomly from the dimension

            Choose a beluga whale X_r randomly

            Update the new position of i by Equation (7)

        elseIf $B_f(i) <= 0.5$

            Update $C1$, and calculate the Levy flight function.

            Update the new position of i by Equation (8)

        end if
        Check the constraints of new positions, and evaluate f_{obj}.

    end for
    for (each beluga whale (X_i)) **do**

        if $B_f(i) <= DCS$ **then**

            Update the step factor $C2$ by Equation (16)

            Calculate X_{step}

            Update the new position of i by Equation (14)

            Check the constraints of new positions, and evaluate f_{obj}.

        end if

    end for

    Find the current best position, and update P^*.

    T++.

end while

Return P^*.

---

5. Experimental Evaluation and Discussion

The numerical data produced by the OBWOD approach and other competing algorithms based on the kNN classifier were statistically examined using the Friedman test. The performance of the OBWOD algorithm and other optimization algorithms was reasonably compared using the Friedman test for the *mean* and *STD* of the best solutions. The effectiveness of the OBWOD algorithm was evaluated using the CEC'22 single-objective benchmark functions and 10 disease datasets with various feature sizes taken from the UCI repository [58]. The final row is ranked in each of the results tables using Friedman's rank. In the following tables, the labels μ and σ refer to the mean and standard deviation of the function values, respectively. The bold type represents the best values.

5.1. Algorithm Configurations and Datasets

5.1.1. Parameter Settings

Table 2 demonstrates the parameter sets employed in constrained optimization algorithms. Numerous researchers have used these factors [59]. Table 3 summarizes our experiments. To find all possible super-quality solutions, OBWOD, BWO, HHO, HGS, SCA, WOA, MFO, and INFO have to be adequately repeated. A total of 20 runs were performed for this purpose, with each run having a maximum of 100 iterations.

Table 2. All used algorithms' parameter settings.

Algorithm	Parameter	Value
INFO [24]	c	2
	d	4
WOA [20]	$\overrightarrow{a_1}$	[2, 0]
	$\overrightarrow{a_2}$	[−1, 1]
MFO [25]	$\overrightarrow{a}$	[−1, −2]
SCA [22]	$\overrightarrow{b}$	[2, 0]
HHO [23]	escaping energy	[0.5, 0.5]
BWO [26]	whale fall W_f	[0.1, 0.05]
	α	0.99
	β	0.01
HGS [21]	R	$[-\infty, \infty]$
	r_1, r_2	[0, 1]

Table 3. Experimental runs' details.

Item	Setting
lb	0
ub	1
max_{it}	100
N	20
M	20
k	5

5.1.2. Dataset Description

An error in diagnostics may occasionally result from a human error or an incorrect interpretation of the data. This article offers a practical computer-aided diagnosis technique with intelligent learning models to avoid these issues. A computer-dependent functional

simulation is proposed to improve predictive effectiveness. The University of California, Irvine, repository (UCI) [58] is conducting experimental research using 10 disease datasets with different dimension sizes, as shown in Figure 2 with more detail in Table 4.

The training dataset had data cleaning performed on it. We substituted missing values for improved computation, whether null, nil, or NA [60]. The null data were removed for such attributes that were found. An attribute is eliminated from the dataset if most of its values are missing, as shown in Figure 2.

Table 4 contains comprehensive details about the used datasets.

Table 4. Descriptions of disease datasets.

Dataset	Total Features	Total Patients	Category	Feature Types	Size
Arrhythmia	279	452	Classification	Categorical, Integer, Real	High
Leukemia2	11,226	72	Classification	Integer, Real	High
Prostate Tumors	10,510	102	Classification	Integer, Real	High
Statlog (Heart)	13	270	Classification	Categorical, Real	Medium
Chronic Kidney Disease (CKD)	25	400	Classification	Real	Medium
Parkinson's	23	197	Classification	Real	Medium
Pima Indians Diabetes (Pima)	8	768	Classification	Integer, Real	Medium
Primary Tumor	17	339	Classification	Categorical	low
Lymphography	18	148	Classification	Categorical	low
Immunotherapy	8	90	Classification	Integer, Real	low

5.1.3. Performance Matrix

To determine which optimization approach was the most effective, we ran each under identical conditions, as given in Table 3.

It is the most straightforward approach to gauge how well a classification problem is performing when the result can include two or more different types of classes. A performance matrix is just a table having two dimensions, actual and predicted, as well as True Positives (TPs), True Negatives (TNs), False Positives (FPs), and False Negatives (FNs) for each dimension. Any project must have an ML algorithm evaluation. The model might achieve satisfactory results when measured against one metric, such as the accuracy score, but unsatisfactory results when measured against another, such as the specificity score. Most of the time, we used classification accuracy to gauge the effectiveness of our model, but more is needed to evaluate it fairly. This paper discusses the many evaluation measures that are accessible, as follows:

1. Mean accuracy (μ_{Acc}): For classification algorithms, the accuracy metric is the most-typical performance metric. One way to describe it is the proportion of correct predictions to all predictions made, as shown in Equation (19).

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)} \tag{19}$$

μ_{Acc} is calculated by Equation (20).

$$\mu_{Acc} = \frac{1}{M} \sum_{j=1}^{M} Acc_*^j \tag{20}$$

2. Mean best fitness (μ_{Fit}): The fitness metric measures the algorithm's effectiveness and, as stated in Equation (18), connects the decrease in the FS ratio to the reduction

in the classification error rate. Equation (21) shows that the lower value represents the best fitness.

$$\mu_{Fitness} = \frac{1}{M} \sum_{j=1}^{M} Fitness_*^j \tag{21}$$

3. Mean feature selection size (μ_{FS}): This metric (μ_{FS}), which is denoted by Equation (22), indicates the average size of FS.

$$\mu_{FS} = \frac{1}{M} \sum_{j=1}^{M} f_*^j \tag{22}$$

According to Equation (23), we determined the overall FS ratio by dividing the FS size f_* by the total size of the features F in the original dataset.

$$Overall_{FS} = \frac{1}{M} \sum_{j=1}^{M} \frac{f_*^j}{F} \tag{23}$$

4. Mean sensitivity (μ_{SE}): The proposed model's sensitivity gauges how well it can identify positive events. Another name for it is the recall or True Positive Rate (TPR). Sensitivity (SE) is used to assess the performance of models since it enables us to count the number of occurrences that the model was able to classify as accurately as positive, as shown in Equation (24).

$$Sensitivity = \frac{TP}{(TP + FN)} \tag{24}$$

The μ_{SE} metric is calculated as shown in Equation (25).

$$\mu_{SE} = \frac{1}{M} \sum_{j=1}^{M} SE_*^j \tag{25}$$

5. Mean specificity (μ_{SP}): As indicated in Equation (26), specificity indicates the percentage of TNs that the model accurately detects. This implies that an additional percentage of TNs was predicted to be positive and may be referred to as FPs. The True Negative Rate (TNR) is another name for this percentage. Specificity (actual negative rate)would always equal one. Low specificity indicates that the model mislabels as many negative findings as positive, whereas high specificity means that the model accurately detects the most unfavorable results.

$$Specificity = \frac{TN}{(TN + FP)} \tag{26}$$

The μ_{SP} metric is calculated as shown in Equation (27).

$$\mu_{SP} = \frac{1}{M} \sum_{j=1}^{M} SP_*^j \tag{27}$$

6. Mean precision (μ_{PPV}): The precision is measured as the proportion of categorized TPs to all positive samples (either correctly or incorrectly). The precision gauges how accurately the model classifies a sample as positive, as shown in Equation (28).

$$Precision = \frac{TP}{(TP + FP)} \tag{28}$$

Equation (29) is used to calculate the μ_{PPV} metric.

$$\mu_{PPV} = \frac{1}{M} \sum_{j=1}^{M} PPV_*^j \tag{29}$$

7. Standard Deviation (STD): The STD σ over the many executions determines the result variances for each optimization algorithm. It is calculated using Equation (30).

$$\sigma_x = \sqrt{\frac{1}{M} \sum_{j=1}^{M} (S_*^j - \mu_x)^2} \tag{30}$$

With the aid of σ_x, all measurements in the performance matrix were calculated.

8. Mean time consumption (μ_{Time}): Using μ_{Time}, each optimization algorithm's average consumption time (in seconds) was determined, as indicated in Equation (31).

$$\mu_{Time} = \frac{1}{M} \sum_{j=1}^{M} Time_*^j \tag{31}$$

5.2. Experimental Series 1: Global Optimization Using CEC'22 Benchmark Test Functions

Quantitative measures from the CEC'22 benchmark set [61] were used to assess the performance of the proposed OBWOD method. The proposed OBWOD algorithm and the other optimization algorithms obtained optimal solutions, and μ and σ were the quantitative measures applied to those solutions. To ensure a fair assessment, the proposed OBWOD outputs were contrasted with those of seven other algorithms, including the original BWO algorithm, HHO, HGS, SCA, WOA, MFO, and INFO.

5.2.1. Description of the CEC'22 Test Suite

Every year, the CEC competitions are held to evaluate optimization stochastic search techniques in custom-built test environments. These benchmark functions were created from a collection of widely used benchmark functions, including those from Ackley, Griewank, Rastrigin, Rosenbrock, Schwefel, and many more [61]. Single-objective optimization algorithms are the foundation for more complex methods such as multi-objective, niching, and constrained optimization algorithms. Therefore, improvements to single-objective optimization techniques are essential since they have the potential to affect other domains as well. One source of motivation for these algorithmic breakthroughs is the feedback from experiments with single-objective benchmark functions, which act as the basic building blocks for more complex tasks. As algorithms advance, more challenging functions must be created. The interaction between approaches and problems drives innovation; hence, we made the CEC'22 Special Session on Real-Parameter Optimization to strengthen this symbiosis. The same search ranges were set for all test functions with the wording "Search range: $[-100, 100]^D$", as seen in Table 5.

5.2.2. Statistical Results' Analysis

The evaluation criteria for the CEC'22 were used to evaluate OBWOD on complex problems and further demonstrate the proposed algorithm's efficacy. The performance of OBWOD was compared to that of seven optimization algorithms: INFO, HHO, HGS, SCA, MFO, WOA, and the original BWO. The results of OBWOD and the other algorithms for CEC'22 test functions from CEC-01 to CEC-10 are shown in Table 6. According to Friedman's rank, the OBWOD received the best average values for all functions, while SCA and WOA obtained the best STD values for the CEC-02 function. WOA also obtained the best STD values for the CEC-01, CEC-08, and CEC-10 functions.

Table 5. The description of the CEC'22 test suite functions [61].

Type	Function	Description	Search Range	F_i^*
Unimodal Function	CEC-01	Shifted and fully rotated Zakharov function.	$[-100, 100]^D$	300
Basic Functions	CEC-02	Shifted and fully rotated Rosenbrock function.	$[-100, 100]^D$	400
Basic Functions	CEC-03	Shifted and fully rotated expanded Schaffer f6 Function.	$[-100, 100]^D$	600
basic Functions	CEC-04	Shifted and fully rotated non-continuous Rastrigin function.	$[-100, 100]^D$	800
Basic Functions	CEC-05	Shifted and fully rotated Levy function.	$[-100, 100]^D$	900
Hybrid Functions	CEC-06	Hybrid Function 1 ($N = 3$)	$[-100, 100]^D$	1800
Hybrid Functions	CEC-07	Hybrid Function 2 ($N = 6$).	$[-100, 100]^D$	2000
Hybrid Functions	CEC-08	Hybrid Function 3 ($N = 5$).	$[-100, 100]^D$	2200
Composition Functions	CEC-09	Composition Function 1 ($N = 5$)	$[-100, 100]^D$	2300
Composition Functions	CEC-10	Composition Function 2 ($N = 4$)	$[-100, 100]^D$	2400

Table 6. Comparison of the CEC'22 between OBWOD and other optimization algorithms based on the kNN classifier.

CEC Functions	Measure	BWO	OBWOD	INFO	HHO	HGS	SCA	WOA	MFO
CEC-01	μ	2.494610×10^4	$\mathbf{2.152710 \times 10^4}$	5.979706×10^4	7.571684×10^4	9.463143×10^{12}	9.558730×10^{12}	8.52330×10^{12}	3.087340×10^4
	σ	1.953938×10^4	1.064252×10^4	2.311781×10^4	2.539657×10^4	9.558730×10^{11}	7.5746858×10^{-2}	$\mathbf{5.851858 \times 10^{-3}}$	1.947247×10^4
CEC-02	μ	6.564376×10^2	$\mathbf{5.086089 \times 10^2}$	1.508763×10^3	7.756663×10^2	7.433591×10^3	7.508678×10^2	7.69828×10^3	6.151887×10^2
	σ	9.464901×10^2	1.439867×10^2	8.297016×10^2	1.040701×10^3	7.508678×10^2	$\mathbf{9.14076 \times 10^{-13}}$	9.5896×10^{-12}	3.057235×10^2
CEC-03	μ	6.303864×10^2	$\mathbf{6.287108 \times 10^2}$	6.668264×10^2	6.408925×10^2	7.527101×10^2	7.603132×10^2	7.569872×10^2	6.421989×10^2
	σ	1.900605×10^1	$\mathbf{1.244823 \times 10^1}$	1.042105×10^1	2.797057×10^1	7.603132×10^1	4.570383×10^{-13}	3.72589×10^{-13}	2.024130×10^1
CEC-04	μ	8.965367×10^2	$\mathbf{8.449059 \times 10^2}$	9.676970×10^2	9.052959×10^2	1.066585×10^3	1.077359×10^3	1.188301×10^4	9.001926×10^2
	σ	4.459035×10^1	$\mathbf{2.369911 \times 10^1}$	3.657403×10^1	4.566980×10^1	1.077359×10^2	2.371115×10^3	4.371115×10^2	4.345011×10^1
CEC-05	μ	2.363152×10^3	$\mathbf{2.020499 \times 10^3}$	3.100224×10^3	3.369690×10^3	1.038756×10^4	1.049249×10^4	3.887064×10^3	3.666241×10^3
	σ	1.041466×10^3	$\mathbf{5.239663 \times 10^2}$	1.281234×10^3	1.946050×10^3	1.049249×10^3	7.312613×10^2	3.329509×10^3	8.850809×10^2
CEC-06	μ	6.790918×10^7	$\mathbf{1.521993 \times 10^7}$	1.513433×10^8	1.377625×10^8	8.770613×10^9	8.859205×10^9	1.263948×10^8	2.595407×10^7
	σ	4.237511×10^8	$\mathbf{1.003401 \times 10^8}$	8.375683×10^8	4.769067×10^8	8.859205×10^8	1.341870×10^9	1.541873×10^9	1.113193×10^8
CEC-07	μ	2.123416×10^3	$\mathbf{2.008917 \times 10^3}$	2.205811×10^3	2.094071×10^3	2.664960×10^3	2.691879×10^3	3.265476×10^3	3.452825×10^3
	σ	5.428055×10^1	$\mathbf{4.403649 \times 10^1}$	4.602230×10^1	7.801352×10^1	2.691879×10^2	2.691879×10^3	4.627033×10^2	4.452825×10^1
CEC-08	μ	2.251655×10^3	$\mathbf{2.208762 \times 10^3}$	2.280427×10^3	2.362581×10^3	2.230307×10^5	2.69876×10^5	2.252836×10^5	2.254398×10^3
	σ	6.435948×10^1	4.460133×10^1	1.178931×10^2	6.281607×10^1	2.252836×10^4	3.78027×10^{-8}	$\mathbf{1.755027 \times 10^{-10}}$	4.393503×10^1
CEC-09	μ	2.544037×10^3	$\mathbf{2.305232 \times 10^3}$	2.782409×10^3	2.533574×10^3	6.551957×10^3	6.618138×10^3	6.618138×10^3	2.509326×10^3
	σ	1.294362×10^2	$\mathbf{5.583183 \times 10^1}$	2.330702×10^2	1.580436×10^2	6.618138×10^2	6.618138×10^3	6.618138×10^3	7.913255×10^1
CEC-10	μ	4.133289×10^3	$\mathbf{2.803257 \times 10^3}$	6.313881×10^3	4.372988×10^3	1.081208×10^4	3.45629×10^3	1.092129×10^4	2.526238×10^3
	σ	1.211022×10^3	1.037671×10^3	9.866894×10^2	1.439390×10^3	1.092129×10^3	1.828153×10^{-8}	$\mathbf{1.828153 \times 10^{-11}}$	1.725860×10^2
Friedman Rank	μ	2.65	**1.10**	4.70	4.00	6.30	7.20	6.80	3.25
	σ	4.75	**2.40**	4.70	6.00	6.40	4.15	4.35	3.25

5.2.3. Convergence Behavior Analysis

The proposed algorithm OBWOD was evaluated on complex problems, specifically the CEC'22 evaluation criteria, to demonstrate the method's efficacy further. The convergence curves of the proposed OBWOD algorithm, HHO, HGS, SCA, WOA, MFO, INFO, and the original BWO for the CEC'22 test functions are shown in Figure 3. The proposed algorithm attained a stable position for all functions. This behavior demonstrates the convergence of the given algorithm. The proposed OBWOD algorithm also quickly achieved the lowest average of the best solutions for most functions. Because of its quick convergence to the near-optimal solution, the proposed OBWOD algorithm is a promising tool for solving problems that require fast computing, such as online optimization issues.
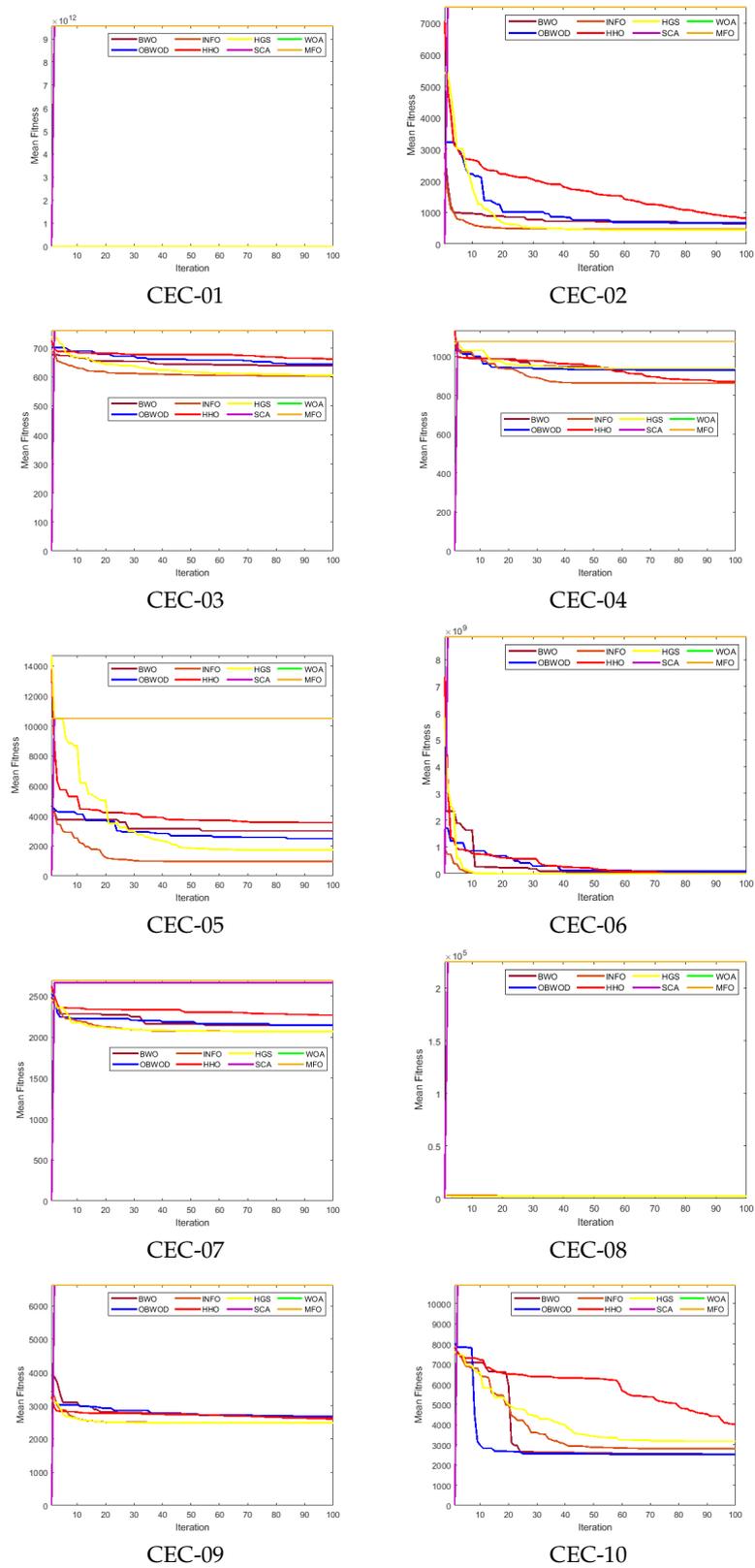
Figure 3. Convergence curves of the proposed OBWOD and the compared optimization algorithms obtained on CEC'22 test functions.

5.3. Experimental Series 2: Biomedical Classification Tasks

Eight optimization algorithms were executed in the same computing environment, as shown in Table 3. The results were validated by the Friedman test utilizing the performance assessment metrics for categorization. With the same parameter settings as those indicated in Table 2, the result of OBWOD was compared with those of the other seven optimization algorithms (BWO, HHO, HGS, SCA, WOA, MFO, and INF) based on the kNN classifier.

The following section compares the experimental results of the OBWOD-based kNN classifier with those of the seven other optimization algorithms (HHO, HGS, SCA, WOA, MFO, INFO, and the original BWO; see Table 2), based on the kNN classifier on ten disease datasets with various dimensionality sizes, which are presented in Table 4. The Friedman test was used to validate the results statistically. Remember that each run consisted of 100 iterations and 20 runs. The measures listed in Section 5.1.3 served as the foundation for the comparative assessment.

5.3.1. Best Fitness and Convergence Evaluation

A fitting function was used in each generation to assess every beluga whale. The beluga whale with the best fitness function value was then selected, as shown in Equation (21). After 100 generations and 20 runs, the value of the fitness function, as depicted in Table 7, decreased. It should be noted that the best fitness of the proposed OBWOD algorithm for classifying the 10 disease datasets with different dimension sizes was 1.40, with the mean and best fitness values converging. The OBWOD approach outperformed the other seven optimization algorithms with greater mean fitness values. In contrast, the second-best optimization technique (BWO) showed lower fitness values for the mean best fitness that competed. Convergence is a stable position at the end of the process when no more changes or improvements are expected if optimization is a process that creates candidate solutions. A failure mode for an optimization algorithm is known as premature convergence, where the process ends at a stable point, but does not necessarily represent the best answer. The OBWOD approach is superior to the other seven optimization algorithms based on the kNN classifier, as shown in Figure 4.

Table 7. Comparison of the best fitness between OBWOD and the other optimization algorithms based on the kNN classifier.

Datasets	BWO		OBWOD		INFO		HHO		HGS		SCA		WOA		MFO	
	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$	$\mu_{Fitness}$	$\sigma_{Fitness}$
Arrhythmia	0.2465	**0.0000**	**0.2038**	**0.0000**	0.0670	0.2455	0.0001	0.0013	0.2201	0.0021	0.4480	0.4982	0.4910	0.5008	0.4480	0.4982
Leukemia2	0.0601	0.0307	**0.0560**	0.0005	0.1399	0.0606	0.0977	0.0333	0.1099	8.0834	0.2440	0.0246	0.2448	0.0006	0.2454	**0.0000**
Prostate Tumors	0.0524	0.0538	**0.0011**	**0.0250**	0.2133	0.0828	0.1558	0.0402	0.1781	0.0918	0.2700	0.0272	0.2992	0.0039	0.3138	0.0061
Statlog (Heart)	0.2149	0.0095	**0.1854**	0.0005	0.2439	0.0422	0.2357	0.0238	0.2243	0.0496	0.2851	0.0288	0.2709	0.0061	0.3124	0.0007
CKD	0.2137	0.0266	**0.2037**	0.0014	0.2891	0.0563	0.3120	0.0268	0.2481	0.0685	0.3627	0.0366	0.3562	0.0067	0.3994	0.0025
Parkinson's	0.0926	0.0141	**0.0810**	0.0016	0.1375	0.0355	0.1312	0.0313	0.1327	0.0497	0.1517	0.0153	0.2025	0.0046	0.1876	0.0446
Pima	0.2567	0.0004	**0.2006**	**0000**	0.2988	0.1715	0.2657	0.0269	0.2649	0.0354	0.2972	0.0300	0.3182	0.0078	0.2911	0.0072
Primary Tumor	0.1891	0.0420	**0.1801**	0.0008	0.2251	0.0040	0.2219	0.0025	0.2275	0.5291	0.3112	0.0324	0.2790	0.0009	0.5793	0.0005
Lymphography	0.1227	0.0128	**0.1167**	0.0198	0.1290	**0.0070**	0.1802	0.0199	0.1318	0.0206	0.1451	0.0315	0.1849	0.0143	0.4516	0.0188
Immunotherapy	0.1533	0.0171	**0.1219**	0.0002	0.1404	0.0474	0.0129	0.0384	0.1602	0.0538	0.2234	0.0225	0.1907	0.0007	0.1198	0.0020
Friedman Rank	2.60	3.85	**1.40**	**1.85**	4.40	6.00	3.30	5.00	4.00	7.40	6.45	5.55	7.00	3.20	6.85	3.15
Rank	2		**1**		5		3		4		6		8		7	

5.3.2. Accuracy and Boxplots' Evaluation

The accuracy of OBWOD and the other seven optimization algorithms based on the kNN classifier is compared in Table 8 under a typical scenario. On all disease datasets, the OBWOD technique beat the other algorithms, reaching classification accuracy on the Prostate Tumors dataset of 100% and an accuracy on the Leukemia2, Parkinson's, and Immunotherapy datasets ranging from 90% to 99.8%. Meanwhile, INFO, HHO, HGS, and MFO achieved the best results on the Prostate Tumors dataset with an accuracy of 100%.

The Friedman rank accuracy of the OBWOD technique was the greatest (7.20), followed by that of the INFO approach, which was at (6.40). A single box that provides a visual representation of the five components of a dataset box and whiskers plot and just a boxplot are other names for it. Some of these are the minimum, first quartile (Q1), median, third

quartile (Q3), and maximum. This helps describe measurements of dispersion and central tendencies in a dataset. By using a boxplot analysis, the properties of the data distribution can be seen in Figure 5.
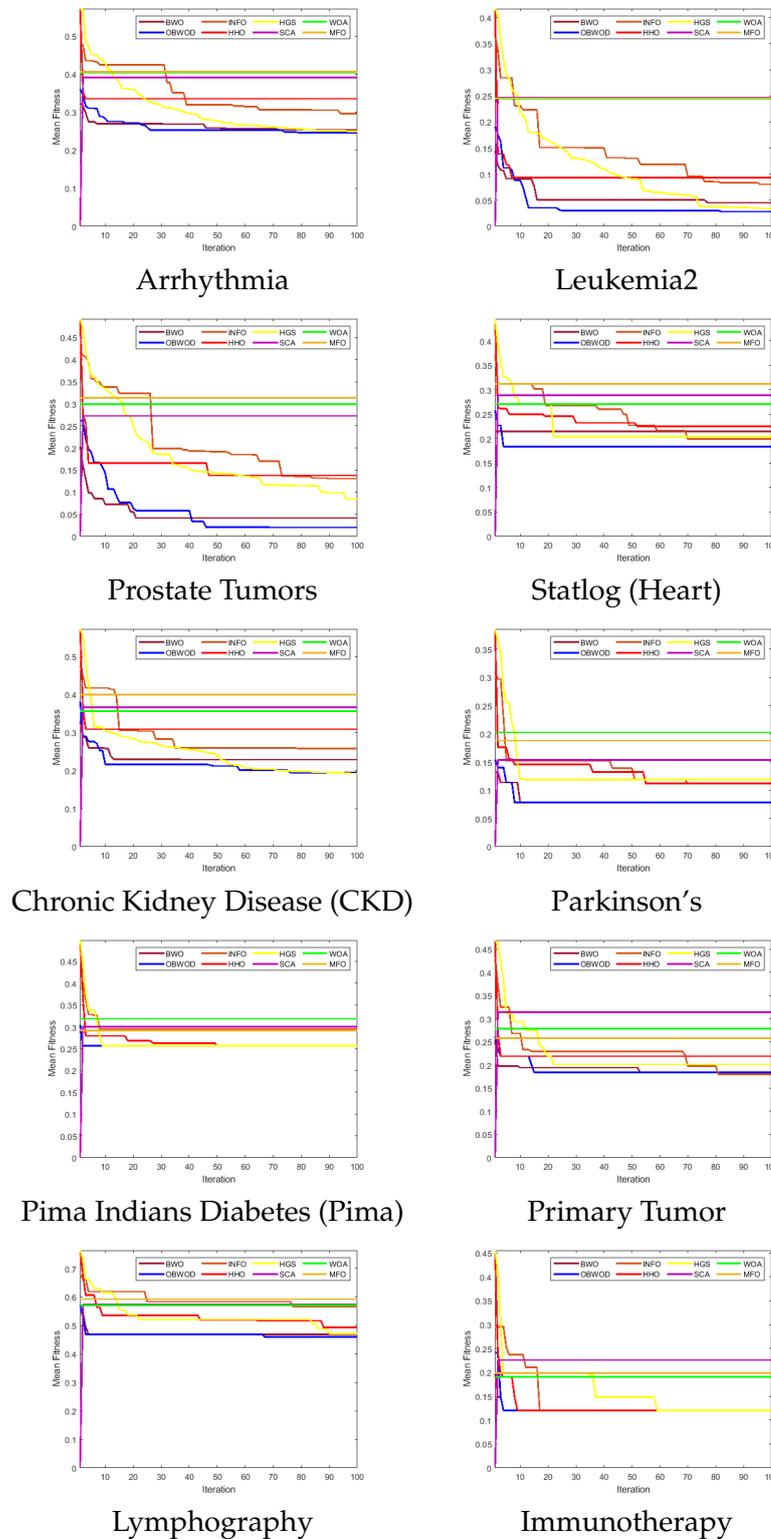

Figure 4. Convergence curves of OBWOD and other optimization algorithms based on the kNN classifier.

There are various distributions and datasets where you will require more information than just the measures of central tendency (median, mean, and mode). You must have information regarding the data's variability or dispersion. A boxplot is a graph that

effectively conveys the distribution of the values in the data. Despite their seemingly being simplistic compared to a histogram or density plot, boxplots have the benefit of taking up less room, which helps compare distributions over numerous groups or datasets. Figure 5 shows that the OBWOD approach produced the highest boxplots for all disease datasets with varying feature sizes.
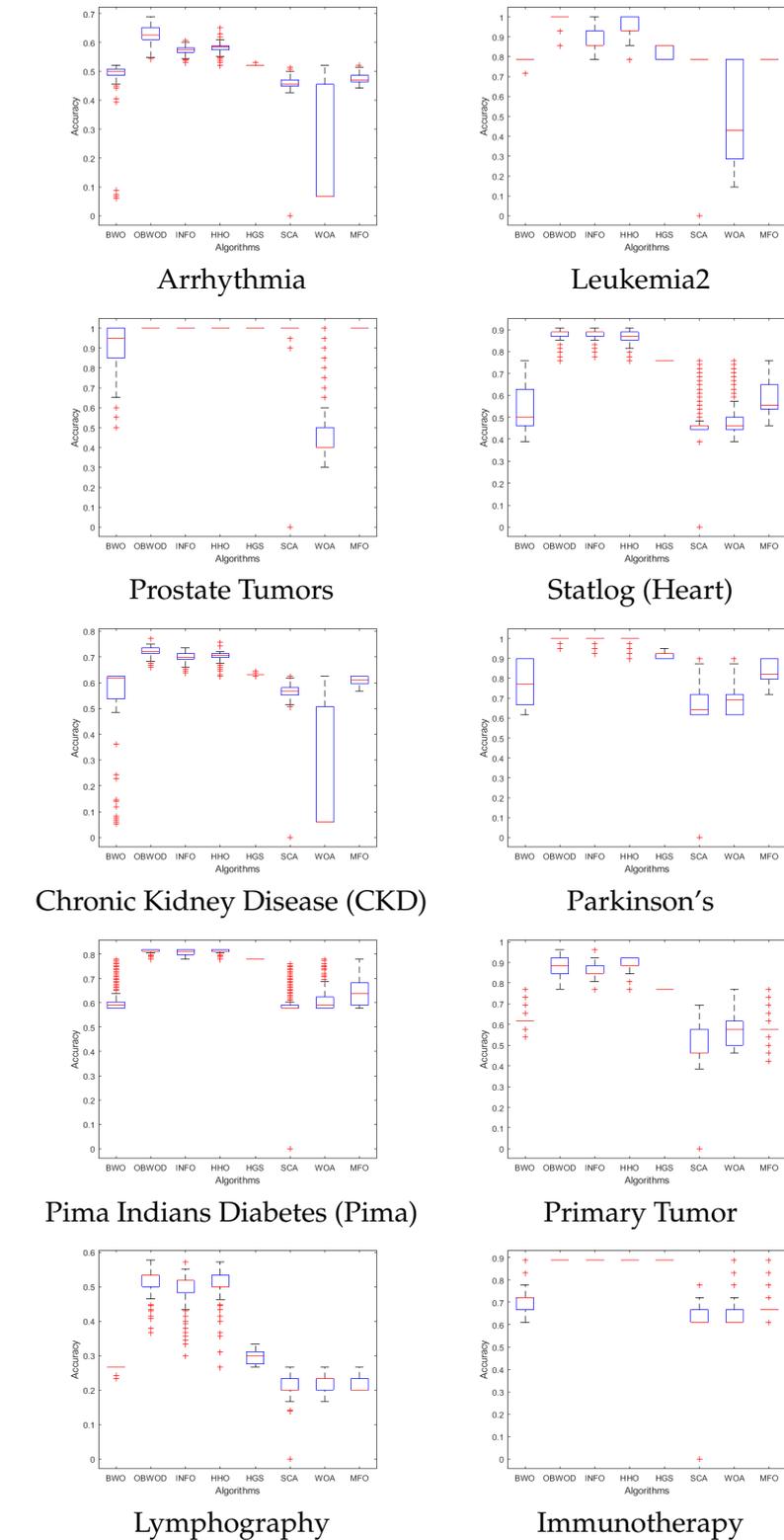

Figure 5. Boxplots of OBWOD and other optimization algorithms based on the kNN classifier.

Table 8. Comparison of the accuracy between OBWOD and other optimization algorithms based on the kNN classifier.

Datasets	BWO		OBWOD		INFO		HHO		HGS		SCA		WOA		MFO	
	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}	μ_{ACC}	σ_{ACC}
Arrhythmia	0.4816	0.0775	**0.6590**	0.0014	0.5731	0.0126	0.5865	0.0212	0.5222	0.0008	0.4551	0.0487	0.2263	0.1940	0.4778	0.01949
Leukemia2	0.7832	0.0132	**0.9883**	0.00294	0.8902	0.0460	0.9555	0.0442	0.8356	0.0328	0.7779	0.0782	0.4944	0.2184	0.7857	0.0320
Prostate Tumors	0.9091	0.0981	**1.0000**	**0.0000**	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000	0.9807	0.1015	0.5126	0.2067	**1.0000**	**0.0000**
Statlog (Heart)	0.5535	0.1065	**0.8813**	0.0004	0.8786	0.0238	0.8622	0.0279	0.7593	0.0050	0.4617	0.0706	0.4744	0.0649	0.5912	0.0891
CKD	0.5231	0.1914	**0.7817**	0.00171	0.7022	0.0149	0.7038	0.0175	0.6339	0.0056	0.5595	0.0617	0.2124	0.2207	0.6066	0.0164
Parkinson's	0.7684	0.1090	**0.9983**	0.0006	0.9956	0.0110	0.9944	0.0152	0.9113	0.0129	0.6764	0.0987	0.6827	0.0615	0.8284	0.0521
Pima	0.6109	0.0537	**0.8545**	0.0006	0.8079	0.0094	0.8113	0.0098	0.7792	0.0020	0.5816	0.0625	0.6005	0.0330	0.6516	0.0604
Primary Tumor	0.6171	0.0398	**0.8876**	0.0003	0.8574	0.0348	0.8816	0.0368	0.7692	0.0750	0.4943	0.0874	0.5638	0.0525	0.5936	0.0682
Lymphography	0.5269	0.0133	**0.5702**	0.0010	0.5889	0.0229	0.5288	0.0359	0.5325	0.0520	0.5672	0.0572	0.5707	0.0070	0.05918	0.0082
Immunotherapy	0.7037	0.0502	**0.9089**	0.0001	0.8889	0.0320	0.8889	0.0420	0.8889	0.0320	0.6317	0.0700	0.6336	0.0413	0.6908	0.0879
Friedman mean rank	3.10	5.80	**7.20**	**1.30**	6.40	3.25	6.30	4.40	5.20	3.35	2.10	7.00	2.10	5.90	3.60	5.00
Rank	6		**1**		2		3		4		7		8		5	

5.3.3. Analyzing Qualitative Information

- *Feature Selection (FS) evaluation:*

  FS, which involves reducing duplicate or unnecessary features from disease datasets, can make classifiers more efficient, quick, and accurate. Regarding medical decisions, FS has an obvious advantage in increasing understandability. The advantages of choosing a subset of all attributes are several; among them, it makes data understanding and visualization more accessible. It decreases the time needed for training and usage and the measurement and storage requirements. The OBWOD algorithm can be fed data with lower dimensionality and produce a more accurate result by applying FS. Table 9 shows that the OBWOD approach outperformed the other optimizers on seven datasets. Meanwhile, the WOA achieved the best results on the Arrhythmia dataset. The second-best optimization method (HHO) had a rank of 3.40, while the highest rank of the FS ratio in the OBWOD was 2.50.

Table 9. Comparison of the FS between OBWOD and other optimization algorithms based on the kNN classifier.

Datasets	BWO		OBWOD		INFO		HHO		HGS		SCA		WOA		MFO	
	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}	μ_{FS}	σ_{FS}
Arrhythmia	0.5100	0.4999	1.0000	**0.0000**	0.5035	0.5000	0.4956	0.4572	0.4996	0.5000	0.4973	0.5000	**0.4985**	0.5000	0.5210	0.4996
Leukemia2	0.0448	0.0030	**0.0280**	0.0000	0.0529	0.2161	0.0083	0.0014	NaN	NaN	0.4976	0.5000	0.4919	0.5000	0.4942	0.5000
Prostate Tumors	0.0402	**0.0000**	0.0215	0.0000	0.0052	0.0278	**0.0000**	0.0000	0.3214	0.1259	0.4976	0.5000	0.4942	0.5000	0.4970	0.5000
Statlog (Heart)	0.2838	0.0890	**0.2149**	0.0000	0.2949	0.2758	0.2910	0.0000	0.2321	0.0081	0.4615	0.5189	0.3846	0.5064	0.3846	0.5064
CKD	0.2949	0.0320	**0.2288**	0.0000	0.2882	0.2706	0.0000	0.0006	0.2723	0.0251	0.4086	0.4925	0.4875	0.5007	0.5018	0.5009
Parkinson's	0.1779	0.0090	**0.0779**	0.0000	0.1817	0.3945	0.1800	**0.0000**	0.1023	0.2769	0.2727	0.4558	0.3182	0.4767	0.3636	0.4924
Pima	0.2562	0.0200	**0.2062**	0.0000	0.2935	0.4198	0.3210	0.0800	0.2365	0.0021	0.3750	0.5175	0.1250	0.3536	0.3750	0.5175
Primary Tumor	0.1938	0.0200	**0.1728**	0.0000	0.1943	0.2961	0.0137	0.1966	0.1823	0.0020	0.4118	0.5073	0.4706	0.5145	0.2941	0.4697
Lymphography	0.4590	0.0300	**0.3685**	0.0000	0.3594	0.4486	0.46100	0.0230	0.4250	0.0230	0.3833	0.4851	0.3889	0.5016	0.3933	0.4851
Immunotherapy	0.1406	**0.0000**	0.1206	0.0000	0.3449	**0.0000**	0.3780	0.1303	0.3449	0.2857	0.4880	0.4286	0.5345	**0.0000**	0.1429	**0.0000**
Friedman Rank	4.20	3.40	**2.50**	**1.50**	4.15	4.95	3.40	2.85	3.35	3.60	6.25	6.95	5.65	6.50	6.50	6.25
Rank	5		**1**		4		3		2		7		6		8	

- *Sensitivity (SE) evaluation:*

  Sensitivity is a measure of the percentage of positive cases that were misclassified as positive TP. This suggests that a different proportion of positive cases will occur, but will be misdiagnosed as negative FPs. An FN rate can also be used to illustrate this. The sensitivity and FP rates added together would equal one.

  Table 10 shows that OBWOD achieved the best results on seven datasets. OBWOD achieved 100% on the Arrhythmia, Leukemia2, and Prostate Tumors datasets, ranging from 99.7% to 99.9% on the CKD, Parkinson's, and Lymphography datasets.

  Notably, the Leukemia2 and Prostate Tumors datasets were classified with 100% sensitivity by OBWOD, INFO, HHO, HGS, and MFO (Table 10). Regarding Friedman's rank sensitivity, OBWOD came in first with a score of 7.30, followed by HHO (6.50).

Table 10. Comparison of the sensitivity between OBWOD and other optimization algorithms based on the kNN classifier.

Datasets	BWO		OBWOD		INFO		HHO		HGS		SCA		WOA		MFO	
	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}	μ_{SE}	σ_{SE}
Arrhythmia	0.9178	0.1368	**1.0000**	**0.0000**	0.9178	0.1368	0.9998	0.0017	0.9848	**0.0000**	0.8974	0.0951	0.3851	0.4423	0.9292	0.0233
Leukemia2	0.9814	0.0093	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	0.9900	0.0995	0.7722	0.2715	**1.0000**	**0.0000**
Prostate Tumors	0.8614	0.1959	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	0.9900	0.0995	0.3840	0.2943	**1.0000**	**0.0000**
Statlog (Heart)	0.5432	0.2887	**0.9565**	0.0178	0.5432	0.2887	0.9545	0.0203	0.8519	**0.0000**	0.1629	0.2547	0.2866	0.2853	0.6453	0.1052
CKD	0.7915	0.3283	**0.9999**	**0.0014**	0.9915	0.3283	0.9996	0.0029	0.9771	0.0044	0.9060	0.0973	0.2236	0.3740	0.9479	0.0227
Parkinson's	0.6138	0.2166	**0.9992**	**0.0006**	0.9902	0.0092	0.9981	0.0138	0.9000	0.0050	0.4692	0.1913	0.3518	0.1520	0.6899	0.1290
Pima	**0.8995**	**0.0023**	0.8345	0.0370	0.8965	0.0073	0.7242	0.0119	0.7093	0.0714	0.7179	0.0088	0.7177	0.0066	0.8868	0.0156
Primary Tumor	0.5655	0.1354	**0.7995**	**0.0090**	0.6021	0.2031	0.7544	0.0856	0.5000	0.0750	0.0081	0.0418	0.0095	0.0617	0.1830	0.1446
Lymphography	0.6654	0.0116	**0.9977**	**0.0157**	0.7454	0.0326	0.9952	0.0355	0.7804	0.0508	0.5413	0.0862	0.4923	0.1694	0.6154	0.0554
Immunotherapy	0.5446	0.1418	**0.7959**	0.0283	0.5626	0.1418	0.7885	0.0466	0.6000	**0.0000**	0.5608	0.0141	0.0086	0.0688	0.1186	0.2228
Friedman Rank	3.60	5.60	**7.30**	**2.45**	5.40	5.00	6.50	3.60	5.10	3.15	2.40	5.00	1.30	6.20	4.40	5.00
Rank	6		**1**		3		2		4		7		8		5	

- *Specificity (SP) evaluation:*

  The percentage of *TN*s projected as negatives is known as specificity. This proposes that a different percentage of *TN* will occur. The specificity plus the *FP* rate would always add up to one.

  Table 11 shows that the OBWOD approach outperformed the other seven optimization algorithms on all datasets. The OBWOD achieved 100% on the Leukemia2 and Prostate Tumors datasets, ranging from 99.7% to 99.9% on the Primary Tumor, Parkinson's, and Immunotherapy datasets.

  Notably, the Leukemia2 and Prostate Tumors datasets were classified with 100% specificity by OBWOD, HHO, HGS, and MFO (Table 11). Regarding Friedman's rank specificity, the OBWOD came in first with a score of 7.30, followed by HHO (6.50).

Table 11. Comparison of the specificity between OBWOD and other optimization algorithms based on the kNN classifier.

Datasets	BWO		OBWOD		INFO		HHO		HGS		SCA		WOA		MFO	
	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}	μ_{SP}	σ_{SP}
Arrhythmia	0.1010	0.0194	**0.2734**	**0.0095**	0.1230	0.0175	0.6367	0.3527	0.1441	0.0047	0.0481	0.0210	0.0571	0.0328	0.0638	0.0281
Leukemia2	0.9873	0.0426	**1.0000**	**0.0000**	0.9763	0.0526	**1.0000**	**0.0000**	**1.0000**	**0.0000**	0.9822	0.1028	0.6197	0.2744	**1.0000**	**0.0000**
Prostate Tumors	0.9884	0.0464	**1.0000**	**0.0000**	0.9754	0.0454	**1.0000**	**0.0000**	**1.0000**	**0.0000**	0.9900	0.0995	0.6188	0.3106	**1.0000**	**0.0000**
Statlog (Heart)	0.4403	0.1805	**0.8327**	0.0724	0.4563	0.1725	0.8809	0.0806	0.6667	**0.0000**	0.2354	0.1136	0.3174	0.0939	0.4889	0.1096
CKD	0.1846	0.0175	**0.4247**	**0.0099**	0.1846	0.0175	0.5838	0.2678	0.2474	0.0152	0.1333	0.0332	0.1118	0.0535	0.1373	0.0262
Parkinson's	0.8130	0.0842	**0.9972**	**0.0096**	0.8330	0.0952	0.9971	0.0111	0.9392	0.0147	0.7092	0.0868	0.7256	0.0549	0.8776	0.0583
Pima	0.2778	0.0883	**0.7794**	**0.0003**	0.2638	0.0436	0.6736	0.0338	0.5105	0.0025	0.2130	0.0437	0.2170	0.0560	0.3430	0.0918
Primary Tumor	0.8667	0.0473	**0.9994**	**0.0003**	0.7452	0.0321	0.9992	0.0068	0.9458	0.0213	0.6710	0.1143	0.7680	0.0683	0.7619	0.0889
Lymphography	0.1380	0.1188	**0.6093**	0.0081	0.12410	0.3028	0.6621	0.0765	0.4286	**0.0000**	0.0512	0.0486	0.0191	0.0628	0.1142	0.1176
Immunotherapy	0.9129	0.0334	**0.9999**	0.0024	0.9129	0.0334	0.9992	0.0077	0.9261	0.0150	0.7287	0.1010	0.7640	0.0703	0.8808	0.0683
Friedman Rank	4.00	5.60	**7.30**	**1.60**	3.50	5.30	7.29	3.80	6.10	2.10	1.70	6.20	1.80	6.10	4.30	5.30
Rank	5		**1**		6		2		3		8		7		4	

- *Precision (PPV) evaluation:*

  To evaluate the effectiveness of ML models or overall AI solutions, precision is a frequently used statistic. It aids in understanding how accurate the predictions made by models are. The percentage of accurate positive predictions is how precision is calculated. Table 12 shows that the OBWOD approach outperformed the other seven algorithms on all datasets. The OBWOD achieved 100% on the Prostate Tumors dataset and achieved a range from 91.9% to 99.9% on the Primary Tumor, Parkinson's, and Immunotherapy datasets.

  Notably, the Prostate Tumors dataset was classified with 100% precision by OBWOD, HHO, HGS, and MFO (Table 12). Regarding Friedman's rank precision, OBWOD came in first with a score of 7.65, followed by HHO (7.15).

Table 12. Comparison of the precision between OBWOD and other optimization algorithms based on the kNN classifier

Datasets	BWO		OBWOD		INFO		HHO		HGS		SCA		WOA		MFO	
	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}	μ_{PPV}	σ_{PPV}
Arrhythmia	0.4043	0.1798	**0.6347**	**0.0090**	0.5243	0.2172	0.5442	0.0126	0.5039	0.0100	0.4632	0.0479	0.1922	0.2312	0.4806	0.0095
Leukemia2	0.6214	0.0173	**0.7765**	**0.0020**	0.6214	0.0173	0.8605	0.0770	0.6796	0.0435	0.6188	0.0622	0.3786	0.1708	0.6250	0.0034
Prostate Tumors	0.9830	0.0766	**1.0000**	**0.0000**	0.9740	0.0526	**1.0000**	**0.0000**	**1.0000**	**0.0000**	0.9900	0.0995	0.6746	0.2313	**1.0000**	**0.0000**
Statlog (Heart)	0.5508	0.0943	**0.8322**	**0.0015**	0.6608	0.0783	0.8514	0.0366	0.7195	0.0029	0.4619	0.0628	0.4965	0.0554	0.5808	0.0718
CKD	0.4851	0.2389	**0.6706**	**0.0080**	0.4851	0.2389	0.6659	0.0085	0.622	0.0032	0.5763	0.0598	0.2284	0.2786	0.6075	0.0078
Parkinson's	0.8279	0.0757	**0.9994**	**0.0004**	0.8529	0.0432	0.9992	0.0052	0.9637	0.0007	0.7934	0.0961	0.7930	0.0375	0.8833	0.0373
Pima	0.3637	0.1146	**0.7593**	**0.0004**	0.3637	0.1146	0.7561	0.0161	0.7273	0.0008	0.2768	0.0685	0.3172	0.0751	0.4515	0.1420
Primary Tumor	0.5431	0.1414	**0.9995**	**0.0091**	0.6241	0.2030	0.9971	0.0216	0.8535	0.0525	0.0585	0.1587	0.1385	0.2114	0.3940	0.2076
Lymphography	0.2397	0.0017	**0.3738**	**0.0014**	0.2397	0.0017	0.3703	0.0233	0.2595	0.0131	0.1940	0.0320	0.1783	0.0449	0.2166	0.0181
Immunotherapy	0.7239	0.0354	**0.9198**	**0.0004**	0.7239	0.0354	0.9191	0.0132	0.8667	0.0300	0.6908	0.0703	0.7041	0.0220	0.7290	0.0555
Friedman Rank	3.25	5.55	**7.65**	**1.35**	3.95	5.55	7.15	3.65	5.95	2.85	2.10	6.00	1.40	6.50	4.55	4.55
Rank	6		**1**		5		2		3		7		8		4	

- *Time consumption evaluation:*

  Making a precise estimate of the amount of time and money needed to train a proposed model is crucial. This is particularly true when you use a cloud environment to train your model on a sizable amount of data. Knowing the length of the training period will help you make crucial decisions if you are working on a proposed approach.

  As shown in Table 13, the OBWOD approach outperformed all experiments, including the other seven algorithms, except for the time consumption test.

  The statistics also showed that OBWOD did an excellent job balancing exploration and extraction. Due to its premature convergence issue, BWO outperformed all other methods in terms of time consumption. This prevented us from discovering the best candidate solution since the algorithms became stuck in a local optimum, as seen in Figure 3.

  BWO had the highest Friedman's rank of consumption time and was the fastest optimizer (1). The results also demonstrate that OBWOD ranked eighth.

Table 13. Comparison of the time consumption between OBWOD and the other optimization algorithms based on the kNN classifier.

Datasets	BWO	OBWOD	INFO	HHO	HGS	SCA	WOA	MFO
	μ_{Time}	μ_{Time}	μ_{Time}	μ_{Time}	μ_{Time}	μ_{Time}	μ_{Time}	μ_{Time}
Arrhythmia	**10.6230**	17.3304	11.0516	12.3240	10.1320	9.3268	10.6587	10.6861
Leukemia2	**22.5840**	51.7540	45.5153	32.6532	30.4671	46.0079	29.2826	51.6296
Prostate Tumors	42.1054	43.1311	62.3347	**32.3657**	42.6201	58.2549	44.8264	66.4606
Statlog (Heart)	**3.4428**	8.4855	7.6711	6.2339	7.0938	7.6333	7.6686	7.6215
CKD	**7.9837**	17.7139	11.1429	9.2948	8.8317	9.3986	8.8126	10.9263
Parkinson's	**7.3120**	8.2712	7.8990	7.2244	8.0921	7.5191	7.5082	7.3047
Pima	**7.1146**	9.3084	8.7200	9.2549	8.1086	8.1127	8.1264	8.3065
Primary Tumor	**6.0160**	8.2361	7.3495	0.2445	7.0904	7.4478	7.3979	7.1740
Lymphography	**6.3034**	8.4789	7.5684	7.2203	0.0926	7.5350	7.5396	7.3110
Immunotherapy	**6.0871**	8.1276	7.5058	7.2314	7.0917	7.2710	7.2860	7.2984
Friedman Rank	**1.70**	7.60	6.30	3.30	2.90	4.70	4.40	5.10
Rank	**1**	8	7	3	2	5	4	6

- Friedman test: To objectively compare the performance of the OBWOD algorithm with that of the other seven algorithms, Friedman's non-parametric test [62] was used for the mean and STD of the best solutions in Tables 6–12.

  In Table 6, for the Friedman test, the worst-performing algorithm is rated highest, and the one with the best performance is listed lowest. Conversely, in Tables 8 and 10–12, the algorithm that performs the best is rated first, and the algorithm that performs the worst is ranked last.

5.4. Discussion

The purpose of this section is to go over the various algorithms for biomedical classification. Table 14 summarizes the comparison of OBWOD and the MH algorithms on biomedical classification and proves its superiority. This study's goal was to suggest an effective search technique for the FS problem that takes into account both low- and high-dimensional datasets. The study suggested integrating OBL into the BWO exploration phase and enhancing the BWO exploitation strategy with DCS. The experimental analysis and comparative study proved that the proposed methodology is effective.

Table 14. Comparison of OBWOD and MH algorithms on biomedical classification.

Ref.	Datasets	Used Techniques	Comparative Accuracy	Algorithm	OBWOD Accuracy
[63]	Heart disease	SVM with PSO called (PSO-SVM)	88.13%		**88.13%**
[2]	Leukemia2	Centroid-mutation-based search-and-rescue optimization algorithm	95.60%		**98.83%**
[64]	Heart disease	Extreme Learning Machine (ELM) algorithm	80%		**88.13%**
[65]	Parkinson's disease	Minimum average maximum tree and SVD with kNN classifier	92.46%		**99.83%**
[66]	Parkinson's disease	Binary Rao1 with a kNN classifier	96.47%		**99.83%**
[2]	Primary Tumor	Centroid-mutation-based search-and-rescue optimization algorithm	44.46%		**88.76%**
[67]	Parkinson's disease	SMOTE + random forests	98.32%		**99.83%**
[68]	Leukemia2	Slime Mold Algorithm (SMA) integrated with OBL based on kNN classifier	90.59%		**98.83%**
[2]	Statlog (Heart)	Centroid-mutation-based search-and-rescue optimization algorithm	86.67%		**88.13%**
[69]	Primary Tumor	Tunicate Swarm Algorithm (TSA) integrated with OBL based on the kNN classifier	82.87%		**88.76%**

The proposed OBWOD approach provides the following advantages:

- OBWOD is well able to resolve global optimization issues based on the CEC'22 test suite. OBWOD generates optimization solutions with better fitness values than the other seven MH algorithms (see Table 6). The proposed hybridization also proves to enhance the convergence ability of the algorithm; see Figure 3.
- The datasets used for this study's analysis range in feature size from 8 to 10510 features, offering a sufficient testing environment for an optimization technique. Here, OBWOD has the highest rank of the FS ratio of 2.50, on average, on all datasets, which is better than the other seven MH algorithms; see Table 9.
- As for accuracy, Table 8 and Figure 5 show that OBWOD chose the subset of features that enabled the learning method k-NN to achieve an average accuracy of 85.17% across all classification datasets.
- Analyzing the qualitative information of the performance evaluation metrics produced proves the superiority of OBWOD (see Table 10 for the overall metrics measuring sensitivity (93.87%), Table 11 for specificity (78%), and Table 12 for precision (76%)).
- Any chance to improve OBWOD can be easily implemented because of its straightforward architecture.
- OBWOD proved its superiority compared to the other optimization algorithms on biomedical classification, as illustrated in Table 14.

Along with the advantages, the proposed OBWOD also has some limitations, which are detailed below:

- The features used by OBWOD may change each time it is run because it is an optimization strategy based on randomization. As a result, there is no assurance that the features subset chosen in one run will be present in another.
- Because OBWOD was developed from BWO, it is computationally more expensive than the other seven MH algorithms (see Table 13).
- Due to kNN's simplicity, it was used as a learning algorithm. However, kNN has several drawbacks, such as the fact that it is a slow learner and noisy data can make it vulnerable.

6. Conclusions and Future Work

OBWOD was proposed in the paper. The framework of OBWOD's search operations consists of two stages: Opposition-Based Learning (OBL) and Dynamic Candidate Solution (DCS) methodologies to balance exploration and exploitation properly. The proposed OBWOD departs from the local optimum. The exploratory population will improve its exploration capacity during the population evolution. The exploitative sub-population will hasten convergence and raise the precision of the solutions. Based on OBL, the tiny subpopulation will improve the population diversity. OBWOD resolved global optimization issues based on ten benchmark functions from CEC-01 to CEC-10. According to the Friedman rank, the CEC'22 test suite was used to gauge the program's capacity to resolve challenging global optimization issues and prove its superiority compared to the other seven optimization algorithms. For ten disease datasets with various feature sizes, we applied the proposed OBWOD approach. The results of OBWOD and other optimizers' classification accuracy tests were superior to those of the Friedman test. Compared to well-known optimization algorithms based on the kNN classifier, including the BWO, HHO, HGS, SCA, WOA, MFO, and INFO algorithms, the proposed OBWOD algorithm proved to be superior. OBWOD was proposed, which was better than BWO in terms of convergence speed and solution accuracy.

Future research can use the OBWOD algorithm to classify biomedical data using different classifiers, such as SVM and NNs. These studies can evaluate the classification precision and generality of the chosen features in various contexts. Other real-world issues can be addressed using the proposed OBWOD approach.

References

1. Houssein, E.H.; Mohamed, R.E.; Ali, A.A. Machine learning techniques for biomedical natural language processing: A comprehensive review. *IEEE Access* **2021**, *9*, 140628–140653. [CrossRef]
2. Houssein, E.H.; Saber, E.; Ali, A.A.; Wazery, Y.M. Centroid mutation-based Search and Rescue optimization algorithm for feature selection and classification. *Expert Syst. Appl.* **2022**, *191*, 116235. [CrossRef]

3.   Chang, P.C.; Lin, J.J.; Liu, C.H. An attribute weight assignment and particle swarm optimization algorithm for medical database classifications. *Comput. Methods Programs Biomed.* **2012**, *107*, 382–392. [CrossRef]

4.   Houssein, E.H.; Abdelminaam, D.S.; Hassan, H.N.; Al-Sayed, M.M.; Nabil, E. A hybrid barnacles mating optimizer algorithm with support vector machines for gene selection of microarray cancer classification. *IEEE Access* **2021**, *9*, 64895–64905. [CrossRef]

5.   Abubakar, I.R.; Olatunji, S.O. Computational intelligence-based model for diarrhea prediction using Demographic and Health Survey data. *Soft Comput.* **2020**, *24*, 5357–5366. [CrossRef]

6.   Ioniţă, I.; Ioniţă, L. Prediction of thyroid disease using data mining techniques. *BRAIN. Broad Res. Artif. Intell. Neurosci.* **2016**, *7*, 115–124.

7.   Ganggayah, M.D.; Taib, N.A.; Har, Y.C.; Lio, P.; Dhillon, S.K. Predicting factors for survival of breast cancer patients using machine learning techniques. *BMC Med. Inform. Decis. Mak.* **2019**, *19*, 48. [CrossRef]

8.   Marquez, E.; Barrón, V. Artificial intelligence system to support the clinical decision for influenza. In Proceedings of the 2019 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), Ixtapa, Mexico, 13–15 November 2019; pp. 1–5.

9.   Abdelrahim, M.; Merlos, C. Hybrid machine learning approaches: A method to improve expected output of semi-structured sequential data. In Proceedings of the 2016 IEEE Tenth International Conference on Semantic Computing (ICSC), Laguna Hills, CA, USA, 4–6 February 2016; pp. 342–345.

10.   Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.

11.   Mandic, D.; Chambers, J. *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*; Wiley: Hoboken, NJ, USA, 2001.

12.   Zhang, H. The optimality of naive Bayes. *Aa* **2004**, *1*, 3.

13.   Peng, C.Y.J.; Lee, K.L.; Ingersoll, G.M. An introduction to logistic regression analysis and reporting. *J. Educ. Res.* **2002**, *96*, 3–14. [CrossRef]

14.   Houssein, E.H.; Emam, M.M.; Ali, A.A.; Suganthan, P.N. Deep and machine learning techniques for medical imaging-based breast cancer: A comprehensive review. *Expert Syst. Appl.* **2021**, *167*, 114161. [CrossRef]

15.   Houssein, E.H.; Emam, M.M.; Ali, A.A. Improved manta ray foraging optimization for multi-level thresholding using COVID-19 CT images. *Neural Comput. Appl.* **2021**, *33*, 16899–16919. [CrossRef]

16.   Rajkomar, A.; Dean, J.; Kohane, I. Machine learning in medicine. *N. Engl. J. Med.* **2019**, *380*, 1347–1358. [CrossRef] [PubMed]

17.   Gao, S.; Calhoun, V.D.; Sui, J. Machine learning in major depression: From classification to treatment outcome prediction. *CNS Neurosci. Ther.* **2018**, *24*, 1037–1052. [CrossRef]

18.   Bey, R.; Goussault, R.; Grolleau, F.; Benchoufi, M.; Porcher, R. Fold-stratified cross-validation for unbiased and privacy-preserving federated learning. *J. Am. Med. Inform. Assoc.* **2020**, *27*, 1244–1251. [CrossRef] [PubMed]

19.   Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.

20.   Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]

21.   Yang, Y.; Chen, H.; Heidari, A.A.; Gandomi, A.H. Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Syst. Appl.* **2021**, *177*, 114864. [CrossRef]

22.   Mirjalili, S. SCA: A sine cosine algorithm for solving optimization problems. *Knowl.-Based Syst.* **2016**, *96*, 120–133. [CrossRef]

23.   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]

24.   Ahmadianfar, I.; Heidari, A.A.; Noshadian, S.; Chen, H.; Gandomi, A.H. INFO: An Efficient Optimization Algorithm based on Weighted Mean of Vectors. *Expert Syst. Appl.* **2022**, *195*, 116516. [CrossRef]

25.   Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl.-Based Syst.* **2015**, *89*, 228–249. [CrossRef]

26.   Zhong, C.; Li, G.; Meng, Z. Beluga whale optimization: A novel nature-inspired metaheuristic algorithm. *Knowl.-Based Syst.* **2022**, *251*, 109215. [CrossRef]

27.   Hameed, N.; Shabut, A.; Hossain, M.A. A Computer-aided diagnosis system for classifying prominent skin lesions using machine learning. In Proceedings of the 2018 10th Computer Science and Electronic Engineering (CEEC), Colchester, UK, 19–21 September 2018 ; pp. 186–191.

28.   Bhattacharya, M.; Lu, D.Y.; Kudchadkar, S.M.; Greenland, G.V.; Lingamaneni, P.; Corona-Villalobos, C.P.; Guan, Y.; Marine, J.E.; Olgin, J.E.; Zimmerman, S.; et al. Identifying ventricular arrhythmias and their predictors by applying machine learning methods to electronic health records in patients with hypertrophic cardiomyopathy (HCM-VAr-risk model). *Am. J. Cardiol.* **2019**, *123*, 1681–1689. [CrossRef] [PubMed]

29.   Ouyang, X.; Huo, J.; Xia, L.; Shan, F.; Liu, J.; Mo, Z.; Yan, F.; Ding, Z.; Yang, Q.; Song, B.; et al. Dual-sampling attention network for diagnosis of COVID-19 from community acquired pneumonia. *IEEE Trans. Med. Imaging* **2020**, *39*, 2595–2605. [CrossRef]

30.   Mohan, S.; Thirumalai, C.; Srivastava, G. Effective heart disease prediction using hybrid machine learning techniques. *IEEE Access* **2019**, *7*, 81542–81554. [CrossRef]

31.   Ghazal, T.M.; Taleb, N. Feature optimization and identification of ovarian cancer using internet of medical things. *Expert Syst.* **2022**, *39*, e12987. [CrossRef]

32.   Calp, M.H. Medical diagnosis with a novel SVM-CoDOA based hybrid approach. *arXiv* **2019**, arXiv:1902.00685.

33. Rahman, F.; Mahmood, M.; et al. A Dynamic Approach to Identify the Most Significant Biomarkers for Heart Disease Risk Prediction Utilizing Machine Learning Techniques. In Proceedings of the International Conference on Bangabandhu and Digital Bangladesh, Dhaka, Bangladesh, 30 December 2021; Springer: Berlin/Heidelberg, Germany, 2022; pp. 12–22.

34. Hashim, F.A.; Hussain, K.; Houssein, E.H.; Mabrouk, M.S.; Al-Atabany, W. Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems. *Appl. Intell.* **2021**, *51*, 1531–1551. [CrossRef]

35. Hashim, F.A.; Houssein, E.H.; Hussain, K.; Mabrouk, M.S.; Al-Atabany, W. Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems. *Math. Comput. Simul.* **2022**, *192*, 84–110. [CrossRef]

36. Houssein, E.H.; Hosney, M.E.; Mohamed, W.M.; Ali, A.A.; Younis, E.M. Fuzzy-based hunger games search algorithm for global optimization and feature selection using medical data. *Neural Comput. Appl.* **2022** , 1–25. [CrossRef]

37. Fajri, D.M.N.; Mahmudy, W.F.; Anggodo, Y.P. Optimization of FIS Tsukamoto using particle swarm optimization for dental disease identification. In Proceedings of the 2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Jakarta, Indonesia, 28–29 October 2017; pp. 261–268.

38. Ahmad, G.N.; Fatima, H.; Ullah, S.; Saidi, A.S.; et al. Efficient medical diagnosis of human heart diseases using machine learning techniques with and without GridSearchCV. *IEEE Access* **2022**, *10*, 80151–80173. [CrossRef]

39. Jalali, S.M.J.; Ahmadian, M.; Ahmadian, S.; Khosravi, A.; Alazab, M.; Nahavandi, S. An oppositional-Cauchy based GSK evolutionary algorithm with a novel deep ensemble reinforcement learning strategy for COVID-19 diagnosis. *Appl. Soft Comput.* **2021**, *111*, 107675. [CrossRef]

40. Hsu, C.H.; Chen, X.; Lin, W.; Jiang, C.; Zhang, Y.; Hao, Z.; Chung, Y.C. Effective multiple cancer disease diagnosis frameworks for improved healthcare using machine learning. *Measurement* **2021**, *175*, 109145. [CrossRef]

41. Cai, X.; Li, X.; Razmjooy, N.; Ghadimi, N. Breast cancer diagnosis by convolutional neural network and advanced thermal exchange optimization algorithm. *Comput. Math. Methods Med.* **2021**, *2021*, 5595180 . [CrossRef]

42. Bharti, R.; Khamparia, A.; Shabaz, M.; Dhiman, G.; Pande, S.; Singh, P. Prediction of heart disease using a combination of machine learning and deep learning. *Comput. Intell. Neurosci.* **2021**, *2021*, 8387680. [CrossRef]

43. Ghosh, D.; Ghosh, E. Breast Mammography-based Tumor Detection and Classification Using Fine-Tuned Convolutional Neural Networks. *Int. J. Adv. Res. Radiol. Imaging Sci.* **2022**, *1*, .

44. Nadakinamani, R.G.; Reyana, A.; Kautish, S.; Vibith, A.; Gupta, Y.; Abdelwahab, S.F.; Mohamed, A.W. Clinical Data Analysis for Prediction of Cardiovascular Disease Using Machine Learning Techniques. *Comput. Intell. Neurosci.* **2022**, *2022*, 2973324. [CrossRef]

45. Houssein, E.H.; Emam, M.M.; Ali, A.A. An optimized deep learning architecture for breast cancer diagnosis based on improved marine predators algorithm. *Neural Comput. Appl.* **2022**, *34*, 18015–18033. [CrossRef] [PubMed]

46. Li, G.; Jimenez, G. Optimal diagnosis of the skin cancer using a hybrid deep neural network and grasshopper optimization algorithm. *Open Med.* **2022**, *17*, 508–517. [CrossRef]

47. Reddy, K.V.V.; Elamvazuthi, I.; Abd Aziz, A.; Paramasivam, S.; Chua, H.N.; Pranavanand, S. Prediction of Heart Disease Risk Using Machine Learning with Correlation-based Feature Selection and Optimization Techniques. In Proceedings of the 2021 7th International Conference on Signal Processing and Communication (ICSC), Noida, India, 25–27 November 2021; pp. 228–233.

48. Mehta, P.; Petersen, C.A.; Wen, J.C.; Banitt, M.R.; Chen, P.P.; Bojikian, K.D.; Egan, C.; Lee, S.I.; Balazinska, M.; Lee, A.Y.; et al. Automated detection of glaucoma with interpretable machine learning using clinical data and multimodal retinal images. *Am. J. Ophthalmol.* **2021**, *231*, 154–169. [CrossRef] [PubMed]

49. Polat, K.; Güneş, S. Automatic determination of diseases related to lymph system from lymphography data using principles component analysis (PCA), fuzzy weighting pre-processing and ANFIS. *Expert Syst. Appl.* **2007**, *33*, 636–641. [CrossRef]

50. Zhang, S.; Li, X.; Zong, M.; Zhu, X.; Wang, R. Efficient knn classification with different numbers of nearest neighbors. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *29*, 1774–1785. [CrossRef] [PubMed]

51. El-Kenawy, E.S.; Eid, M. Hybrid gray wolf and particle swarm optimization for feature selection. *Int. J. Innov. Comput. Inf. Control* **2020**, *16*, 831–844.

52. Mafarja, M.; Aljarah, I.; Heidari, A.A.; Faris, H.; Fournier-Viger, P.; Li, X.; Mirjalili, S. Binary dragonfly optimization for feature selection using time-varying transfer functions. *Knowl.-Based Syst.* **2018**, *161*, 185–204. [CrossRef]

53. Subasi, A. Use of artificial intelligence in Alzheimer's disease detection. In *Artificial Intelligence in Precision Health*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 257–278.

54. Tizhoosh, H.R. Opposition-based learning: A new scheme for machine intelligence. In Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06), Vienna, Austria, 28–30 November 2005; Volume 1, pp. 695–701.

55. Houssein, E.H.; Ibrahim, I.E.; Neggaz, N.; Hassaballah, M.; Wazery, Y.M. An Efficient ECG Arrhythmia Classification Method Based on Manta Ray Foraging Optimization. *Expert Syst. Appl.* **2021**, *181*, 115131. [CrossRef]

56. Khodadadi, N.; Snasel, V.; Mirjalili, S. Dynamic arithmetic optimization algorithm for truss optimization under natural frequency constraints. *IEEE Access* **2022**, *10*, 16188–16208. [CrossRef]

57. Mantegna, R.N. Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes. *Phys. Rev. E* **1994**, *49*, 4677. [CrossRef]

58. Frank. Machine Learning Repository. 2010. Available online: https://archive.ics.uci.edu/ml/index.php (accessed on 25 December 2022).

59. Yan, F.; Xu, X.; Xu, J. Grey wolf optimizer with a novel weighted distance for global optimization. *IEEE Access* **2020**, *8*, 120173–120197. [CrossRef]

60. Gajendra, E.; Kumar, J. A novel approach of ECG classification for diagnosis of heart diseases: Review. *Int. J. Adv. Res. Comput. Eng. Technol. (IJARCET)* **2015**, *4*, 4096–4100.

61. Ahrari, A.; Elsayed, S.; Sarker, R.; Essam, D.; Coello, C.A.C. Problem Definition and Evaluation Criteria for the CEC'2022 Competition on Dynamic Multimodal Optimization. In Proceedings of the IEEE World Congress on Computational Intelligence (IEEE WCCI 2022), Padua, Italy, 18–23 July 2022; pp. 1–10.

62. Friedman, M. A comparison of alternative tests of significance for the problem of m rankings. *Ann. Math. Stat.* **1940**, *11*, 86–92. [CrossRef]

63. Vijayashree, J.; Sultana, H.P. A machine learning framework for feature selection in heart disease classification using improved particle swarm optimization with support vector machine classifier. *Program. Comput. Softw.* **2018**, *44*, 388–397. [CrossRef]

64. Ismaeel, S.; Miri, A.; Chourishi, D. Using the Extreme Learning Machine (ELM) technique for heart disease diagnosis. In Proceedings of the 2015 IEEE Canada International Humanitarian Technology Conference (IHTC2015), Ottawa, ON, Canada, 31 May–4 June 2015; pp. 1–3.

65. Tuncer, T.; Dogan, S.; Acharya, U.R. Automated detection of Parkinson's disease using minimum average maximum tree and singular value decomposition method with vowels. *Biocybern. Biomed. Eng.* **2020**, *40*, 211–220. [CrossRef]

66. Sharma, S.R.; Singh, B.; Kaur, M. Classification of Parkinson disease using binary Rao optimization algorithms. *Expert Syst.* **2021**, *38*, e12674. [CrossRef]

67. Polat, K. A hybrid approach to Parkinson disease classification using speech signal: The combination of smote and random forests. In Proceedings of the 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), Istanbul, Turkey, 24–26 April 2019; pp. 1–3.

68. Wazery, Y.M.; Saber, E.; Houssein, E.H.; Ali, A.A.; Amer, E. An efficient slime mold algorithm combined with k-nearest neighbor for medical classification tasks. *IEEE Access* **2021**, *9*, 113666–113682. [CrossRef]

69. Houssein, E.H.; Saber, E.; Ali, A.A.; Wazery, Y.M. Opposition-based learning tunicate swarm algorithm for biomedical classification. In Proceedings of the 2021 17th International Computer Engineering Conference (ICENCO), Cairo, Egypt, 29–30 December 2021; pp. 1–6.