IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 July 2023, accepted 28 August 2023, date of publication 4 September 2023, date of current version 7 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3312022

RESEARCH ARTICLE

A Binary Waterwheel Plant Optimization Algorithm for Feature Selection

AMEL ALI ALHUSSAN[1], ABDELAZIZ A. ABDELHAMID[2,3], EL-SAYED M. EL-KENAWY[4], (Senior Member, IEEE), ABDELHAMEED IBRAHIM[5], (Member, IEEE), MARWA METWALLY EID[4,6], DOAA SAMI KHAFAGA[1], AND AYMAN EM AHMED[7]

[1]Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh 11671, Saudi Arabia
[2]Department of Computer Science, College of Computing and Information Technology, Shaqra University, Shaqra 11961, Saudi Arabia
[3]Department of Computer Science, Faculty of Computer and Information Sciences, Ain Shams University, Cairo 11566, Egypt
[4]Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura 35111, Egypt
[5]Computer Engineering and Control Systems Department, Faculty of Engineering, Mansoura University, Mansoura 35516, Egypt
[6]Faculty of Artificial Intelligence, Delta University for Science and Technology, Mansoura 35712, Egypt
[7]Faculty of Engineering, King Salman International University, El Tor 11765, Egypt

Corresponding authors: Abdelaziz A. Abdelhamid (abdelaziz@su.edu.sa), El-Sayed M. El-Kenawy (skenawy@ieee.org), and Doaa Sami Khafaga (dskhafga@pnu.edu.sa)

This work was supported by the Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R308), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

ABSTRACT The vast majority of today's data is collected and stored in enormous databases with a wide range of characteristics that have little to do with the overarching goal concept. Feature selection is the process of choosing the best features for a classification problem, which improves the classification's accuracy. Feature selection is considered a multi-objective optimization problem with two objectives: boosting classification accuracy while decreasing the feature count. To efficiently handle the feature selection process, we propose in this paper a novel algorithm inspired by the behavior of waterwheel plants when hunting their prey and how they update their locations throughout exploration and exploitation processes. The proposed algorithm is referred to as the binary waterwheel plant algorithm (bWWPA). In this particular approach, the binary search space as well as the technique's mapping from the continuous to the discrete spaces are both represented in a new model. Specifically, the fitness and cost functions that are factored into the algorithm's evaluation are modeled mathematically. To assess the performance of the proposed algorithm, a set of extensive experiments were conducted and evaluated in terms of 30 benchmark datasets that include low, medium, and high dimensional features. In comparison to other recent binary optimization algorithms, the experimental findings demonstrate that the bWWPA performs better than the other competing algorithms. In addition, a statistical analysis is performed in terms of the one-way analysis-of-variance (ANOVA) and Wilcoxon signed-rank tests to examine the statistical differences between the proposed feature selection algorithm and compared algorithms. These experiments' results confirmed the proposed algorithm's superiority and effectiveness in handling the feature selection process.

INDEX TERMS Feature selection, waterwheel plant, meta-heuristic optimization, K-nearest neighbors, binary optimizer, bWWPA.

I. INTRODUCTION

We live in a decade where information is so valuable that it has been called the currency. The processing of data has become increasingly laborious due to several factors, including the explosion of data and the advancements in machine learning and data mining. To address these challenges, organizations are constantly working on improving data processing techniques and developing more efficient algorithms. Machine learning and data mining are expanding

The associate editor coordinating the review of this manuscript and approving it for publication was Claudia Raibulet.

fields of study and business practice because of the exponential growth of data that has to be processed and analyzed. Knowledge discovery from data heavily relies on the conversion procedure, which involves a series of cyclical tasks such as data translation, reduction, cleaning, and integration. These tasks are essential for transforming raw data into meaningful insights and actionable knowledge [1]. Pre-processing consists of these actions, the results of which immediately affect the efficacy of subsequent machine learning and data mining algorithms. Training machine learning and data mining algorithms take more time and effort as the number of dimensions grows, increasing the computational cost of these tasks. Consequently, it is crucial that data be appropriately handled. Researchers have come up with various approaches to solve the dimensionality problem. Feature selection is a strategy since it eliminates irrelevant information that might otherwise hinder the classification process, such as irrelevant features [2], [3], [4].

To boost the classifiers' efficiency, feature selection is a pre-processing stage that helps pick the most relevant features and filter out the rest. Through the elimination of superfluous features, this technique reduces the computational complexity [5]. Wrapper and filter techniques are the most common approaches to feature selection [6], [7]. In the wrapper approach, feature subsets are determined by one or more learning algorithms. When compared to the filter approach, this one yields superior performance but at a higher computational cost. Wrapper-based feature selection is typically framed as an optimization problem [8]. The filter-based method selects valuable features independently of a learning algorithm, taking advantage of information gain, mutual information, and so on [9], [10]. While it is computationally cheap, this approach's performance is subpar compared to wrapper-based methods. Discovering the most pertinent subset of features is difficult since the goal is to use as few features as feasible while achieving the highest possible accuracy. Feature selection is considered an NP-hard task [11] because choosing the best possible set of features takes a long time. If there are N features, it leads to checking $2^N - 1$ different permutations to find the optimal set of features [12]. For this reason, it is crucial to have access to a high-performing meta-heuristic for handling this kind of problem to speed up the computation time.

Meta-heuristic search procedures rely on a compromise between exploitation (intensification), which performs a comprehensive neighborhood search to discover better feasible solutions, and exploration (diversification), which evaluates the solution of candidates not inside the neighborhood. These two criteria determine one's solution-finding process. Recently, meta-heuristic algorithms have been used to handle the optimization problem of feature selection [13], [14], [15], [16], [17], since they provide superior performance than precise techniques. The necessity for new or improved techniques to produce high-quality solutions for the candidate problem arises from the no-free lunch (NFL) theorem, which states that no one algorithm can handle all optimization problems. Figure 1 shows the standard feature selection method. This figure extracts an initial random population from the dataset, with ones and zeros representing the selected and not-selected features. This random population is used to compute the fitness value for each solution and save the best values from this step. This step is performed in terms of the classification using the K-nearest neighbors (KNN) classifier. Based on the best fitness, the solution is updated. This process is repeated iteratively until a stopping criterion is met. This iterative procedure results in the optimal collection of features for evaluation by machine learning classifiers.

The algorithm's efficiency is a significant factor in using it as the foundation for the binary optimization strategy. When compared to other cutting-edge optimization techniques that took inspiration from similar areas, the proposed algorithm performed exceptionally well. Given the algorithm's success in solving highly challenging continuous optimization problems, we set out to see if its operators and optimization process might also yield optimum solutions to binary optimization problems. Therefore, this research examines the impact, benefit, and influence of creating and deploying a binary WWPA technique through extensive and rigorous testing on diverse, high-dimensional datasets. The performance of the approach inspired this binarization, although it has not been used to tackle the feature selection problem since its inception. We provide a binary variant of the WWPA to address the bounded-continuity nature of the feature selection problem. As a classifier, we use kNN to verify how well the chosen collection of features performs. What follows is a list of the work's most significant contributions:

- For solving the feature-selection problem, we propose a novel binary algorithm based on the waterwheel plant algorithm (bWWPA).
- The application of a set of computational analysis metrics for assessing the performance of the proposed bWWPA algorithm for feature selection.
- Extensive testing on 30 datasets of varying sizes and dimensionalities to demonstrate the efficacy of the proposed feature selection algorithm.
- A number of well-known feature selection approaches are compared to the bWWPA to determine its relative superiority.
- Two high-dimensional datasets are investigated in terms of statistical analysis to confirm the superiority and effectiveness of the proposed approach.

The remaining manuscript is organized as follows: Section II reviews the existing research on the feature selection problem. The proposed methodology is covered in further detail in Section III. In Section IV, we explain the experimental setup of the conducted experiments. Experiment findings and analysis are presented and discussed in Section V. The conclusion and future perspective of this work are presented in Section VI.

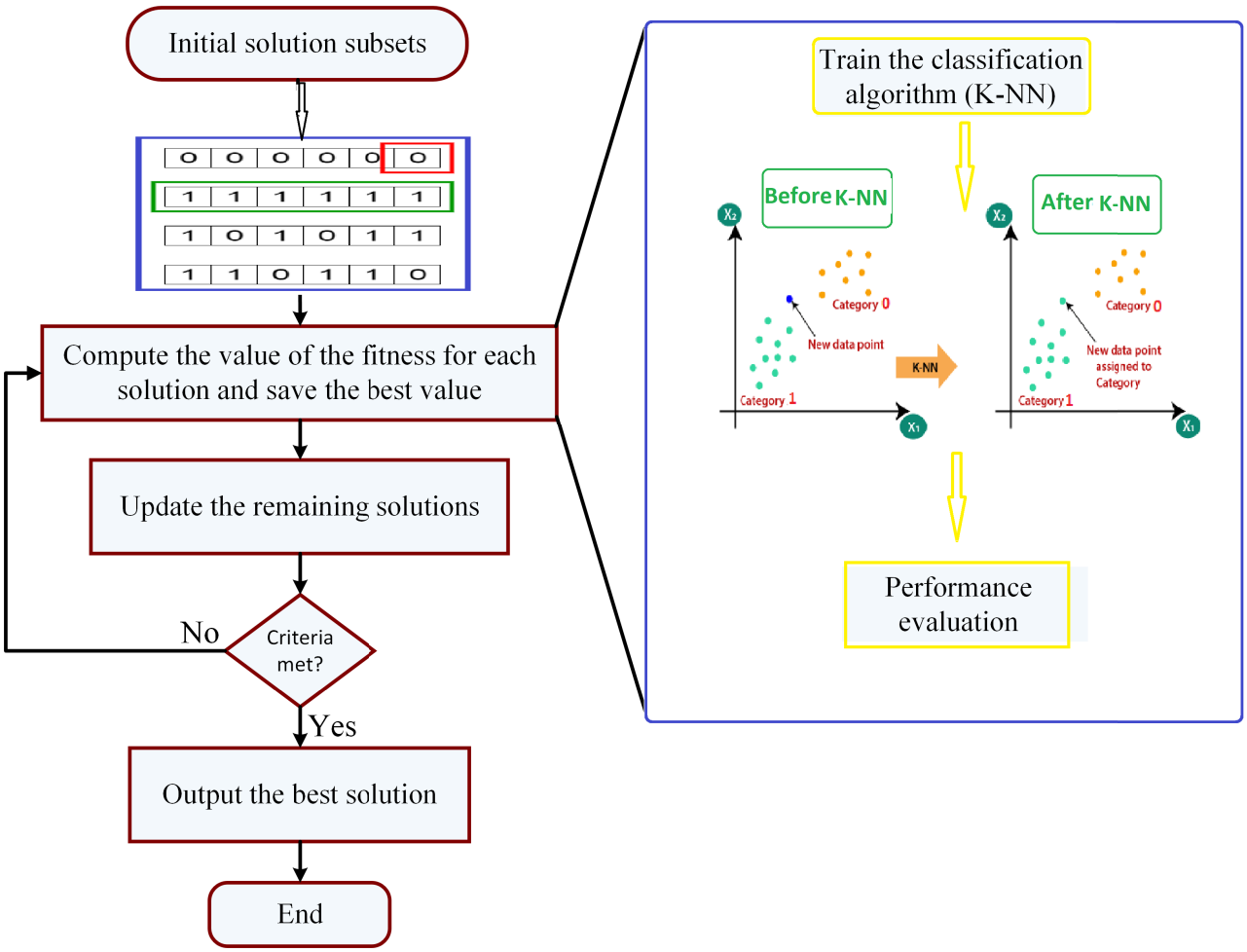

FIGURE 1. The typical process of feature selection.

II. LITERATURE REVIEW

In this part, we provide a comprehensive literature analysis of prior research dealing with the same general topic as the one under investigation in the current study. Several binary meta-heuristic methods for resolving the feature selection problem have been proposed in the literature. The wrapper-based feature selection method uses the binary search capacity of meta-heuristic algorithms. In feature selection, swarm- and evolutionary-based algorithms are becoming standard practices [18]. Due to its tried-and-true mathematical modeling, Particle Swarm Optimization (PSO) [19] has gained a lot of interest as a bio-inspired meta-heuristic approach. As a result of binarization and optimization, this approach can now be used to address problems in discrete search spaces. Authors in [20] provided a tweaked discrete PSO that implemented the logistic regression model in the feature selection setting. It wasn't until a year later that authors in [21] proposed "catfishBPSO," an enhanced version of BPSO that incorporated the catfish effect, to be used in feature selection. To solve the optimization problem of feature selection, the BPSO was additionally enhanced [22]. Authors developed an improved PSO (IPSO) in [17] to solve the feature selection problem; it uses the Levy flight local factor, the global factor's weighted inertia coefficient, and the method of mutation diversity's improvement factor. However, this enhancement was not without drawbacks, such as adding additional parameters compared to previous enhanced PSO versions, increasing computing time, and making tuning harder for different application problems. The main problem with BPSO is that it cannot progress since every particle in its approaches and recedes from the hypercube corner.

Another widely used bio-inspired feature selection approach is the genetic algorithm (GA), often implemented as a wrapper-based methodology. To address the feature selection problem, authors in [23] proposed a GA-based approach using the support vector machine (SVM) learning algorithm. Parameter and feature subset optimization for the SVM simultaneously without sacrificing classification accuracy was the primary focus of their research. The approach increased classification accuracy by decreasing the number of

feature subsets, but it still lagged behind the Grid algorithm. Authors in [24] later introduced a feature selection technique for predicting protein functions that combined GA with the ant colony optimizer (ACO). In addition, authors in [25] developed a modified GA (MGA), which is a feature selection technique employing a pre-trained deep neural network (DNN) for the prediction of demand for several essential resources in an outpatient setting. As a result of merging these two methods, we now have access to enhanced, quicker capabilities with almost little overhead in terms of processing power.

Many other strategies, some drawn directly from nature, have also been used to address feature selection problems, but these two are the most well-known. Authors in [26] created the bat algorithm using binary wrappers. The classifier optimum-path forest is used to find the optimal feature sets for classification. Using an evolutionary-inspired similarity search technique, authors in [27] presented a binary artificial bee colony (ABC) to tackle the feature selection problem. Authors in [15] introduced the binary ant lion optimizer (BALO), which uses the transfer function to relocate ant lions inside a discrete search space. The following year, it was proposed to use a binary grey wolf optimizer with two techniques to find a subset of features that simultaneously meets the two competing goals of the feature selection problem: improving classification accuracy while reducing the total number of features used in the analysis. Despite its superior performance relative to the approaches employed as a comparison in the study, this approach suffered from early convergence. The return-cost-based firefly algorithm (Rc-FFA) was developed by authors in [28] to improve the original binary firefly method that avoids convergence too soon. The exploitation and exploration stages of the binary dragonfly optimizer created by authors in [29] were enhanced using a time-varying transfer function. However, it did not even come close to performing optimally.

Two variations of the salp swarm algorithm (SSA) were presented by authors in [30] to address the feature selection problem. However, the study did not analyze the transfer functions used in the first method, which used eight transfer functions to transform a continuous search space into a binary one, or the crossover operator used to enhance the SSA's exploratory capabilities. Using the V-shaped transfer function and the sigmoid, authors in [31] created a binary grasshopper optimization algorithm (BGOA). To improve the BGOA's exploration phase, the mutation operator was introduced into this study. Specifically, two binary variants of the whale optimization technique were presented by authors in [32]. The first iteration relied on a random operator's roulette wheel and tournament selection processes throughout the search process, while the second used mutation and crossover to increase diversity. To address this feature selection problem, authors in [33] presented a binary seagull optimizer based on the baseline approach, which uses four transfer functions with S and V shapes to ''binarize'' the process. In addition to comparing their method's performance to that of others, they also tested it on high-dimensional datasets.

Bio-inspired meta-heuristics were created by the authors of [34] and [35] utilizing three different approaches to the problem. Both methods classified medical diagnoses using a backpropagation neural network and the AddaBoostSVM classifier. While the former utilized a mixture of the glow world swarm optimization method, the lion optimization algorithm, and differential evolution [36], [37], the latter utilized a mixture of the krill herd optimization algorithm, the cat swarm optimization algorithm, the bacteria foraging optimizers, and the carnivorous plant algorithm [38]. The results from using these strategies were better than those from using alternatives. The proposed approaches included many meta-heuristic techniques, which resulted in high computing costs. Authors in [39] proposed a bio-inspired technique (salp swarm) with kernel-ELM as a classifier to identify glaucoma from medical photos. In comparison to previous approaches, the outcomes achieved by this strategy were superior. However, huge real-time dataset collections were not used to evaluate the method due to the difficulty they presented. Some of the problems with feature selection were improved upon by the aforementioned methods [40]. Several techniques failed to provide a desirable feature set when applied to high-dimensional datasets. The NFL theorem concludes that no universal optimization problem solution applies to feature selection. The optimization challenge of feature selection necessitates the creation of a novel binary approach.

The susceptible infectious recovery (SIR) model [41] is one example of a strategy developed to address the detection and classification challenges. Based on sample pathways, this method was used to identify the original data sources in a network. All network nodes were assumed to be in their vulnerable beginning condition in that study, with the exception of a single infectious source. After that, the diseased node, which may no longer be infectious, might spread the infection to the vulnerable nodes. This simulated exercise demonstrated that the estimate generated by the tree network's reverse infection technique was more aligned with the actual source. Many real-world networks underwent additional performance studies with encouraging results. The limitation of this model was that it relied on just one source node, which is usually not the case in practice. Authors in [42] developed a divide-and-conquer strategy to tackle this problem by employing the SIR model to locate multiple sources in social networks. Results from using the method were encouraging, with estimates quite close to the mark. These techniques have not, however, been applied directly to the problem of optimizing feature selection.

More SIR model-based approaches have been developed to detect or diagnose coronavirus infection in individuals after the 2020 COVID-19 virus epidemic. To prevent the spread of the virus across society, authors in [43] proposed a novel coronavirus herd immunity optimizer (CHIO) based on herd immunity and the social distance strategy. The herd

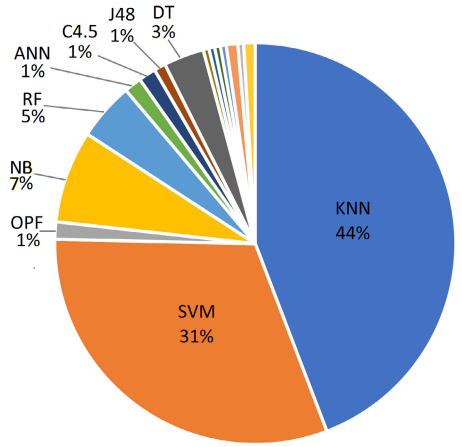

FIGURE 2. The most common classifiers used in feature selection. The percentages refer to the number of feature selection algorithms that use each classifier.

immunity method was used to solve an optimization problem in engineering by combining the expertise of three types of individuals: those vulnerable to infection, those who had contracted it, and those who had been immunized. Recently, a unique COVID-19 diagnostic technique, patient detection strategy (CPDS) [44], was introduced that leveraged this algorithm by combining the wrapper and filter approaches to feature selection. Using chest CT scans of patients infected with and uninfected with COVID-19, the wrapper approach implemented EKNN. The findings demonstrated that the proposed technique outperformed recently established alternatives regarding accuracy, sensitivity, precision, and running time. Additionally, two wrapper-based approaches were developed by combining the greedy search operator with and without the CHIO and then tested on 23 benchmark datasets and the real-world COVID-19 dataset [45].

The success of any feature selection technique relies on selecting a suitable classifier. Classifiers such as SVM, KNN, artificial neural networks (ANN), naive Bayesian (NB), kernel extreme learning machines (KELM), random forests (RF), fuzzy rules-based systems (FR), C4.5, and optimum-path forests (OPF) are used with meta-heuristic algorithms to solve feature selection problems. Figure 2 shows that KNN is the most often used classifier in the literature because of its versatility and suitability for high-dimensional datasets. We adopted the KNN classifier with the proposed feature selection algorithm in this work.

III. THE PROPOSED METHODOLOGY

In this section, we detail the procedure of the binarization strategy we propose to use with the WWPA algorithm. Once the search space has been generated, the technique for creating the binary representation of it will be described. Next, we develop the binary version of WWPA and add it to the binary search matrix. The provided transformation functions enable the variation to create a discrete map from the continuous space.

A. INSPIRATION OF THE WATERWHEEL PLANT ALGORITHM (WWPA)

Waterwheel plants (*Aldrovanda vesiculosa*) have traps that look like tiny, see-through flytraps and are carried on broad petioles [46]. A ring of bristles that mimic hair surrounds the trap to prevent harm or false triggers from other water plants. The trap's outside edges are covered in many hook-like teeth that interlock as the trap closes around its prey, like the teeth of a flytrap. A ''flytrap'' is a type of carnivorous plant; it has evolved to be a cunning and highly specialized predator of the insect world. When the clamshell is closed, it is due to the action of roughly forty long trigger hairs (there are only about 6-8 trigger hairs within a Venus flytrap trap). Predators are equipped with both trigger hairs and acid-secreting glands to aid in the digestion of meat. The victim is sucked into the trap and pinned to the floor at the hinge by the interlocking teeth and a mucus sealant. The trap has forced out most of the water, which is being replaced by digestive fluids. Like a flytrap, an Aldrovanda trap may only capture two to four meals before it quits up. The waterwheel facility is seen in Figure 3.

B. THE MATHEMATICAL MODEL OF WWPA

Using a model of the waterwheel's real behavior, this part first discusses how to set up WWPA [47], then outlines how to update the location of the waterwheel throughout exploration and exploitation.

1) INITIALIZATION

WWPA is an approach that uses a group of individuals to find a good solution to a problem through repeated attempts in the search space. The WWPA population has different values for the problem variables due to the different locations of the waterwheels inside the search space. A vector can be seen as a graphical depiction of various solutions to the problem, with each waterwheel representing a distinct vector. It is conceivable that a matrix might be employed to portray the entire WWPA population, encompassing all the waterwheel variations. In the first stage of a WWPA implementation, the starting positions of the waterwheels in the search area are determined randomly.

$$P = \begin{bmatrix} P_1 \\ \vdots \\ P_i \\ \vdots \\ P_N \end{bmatrix} = \begin{bmatrix} p_{1,1} & \cdots & p_{1,j} & \cdots & p_{1,m} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ p_{i,1} & \cdots & p_{i,j} & \cdots & p_{i,m} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ p_{N,1} & \cdots & p_{N,j} & \cdots & p_{N,M} \end{bmatrix} \tag{1}$$

$$p_{i,j} = lb_j + r_{i,j}.(ub_j - lb_j), i = 1, 2, \ldots, N, j = 1, 2, \ldots, m \tag{2}$$

where N and m represent the number of waterwheels and the number of variables, respectively; $r_{i,j}$ is a random number in the interval [0, 1]; lb_j and ub_j represent the lower bound and upper bound of the j-th problem variable; P represents the

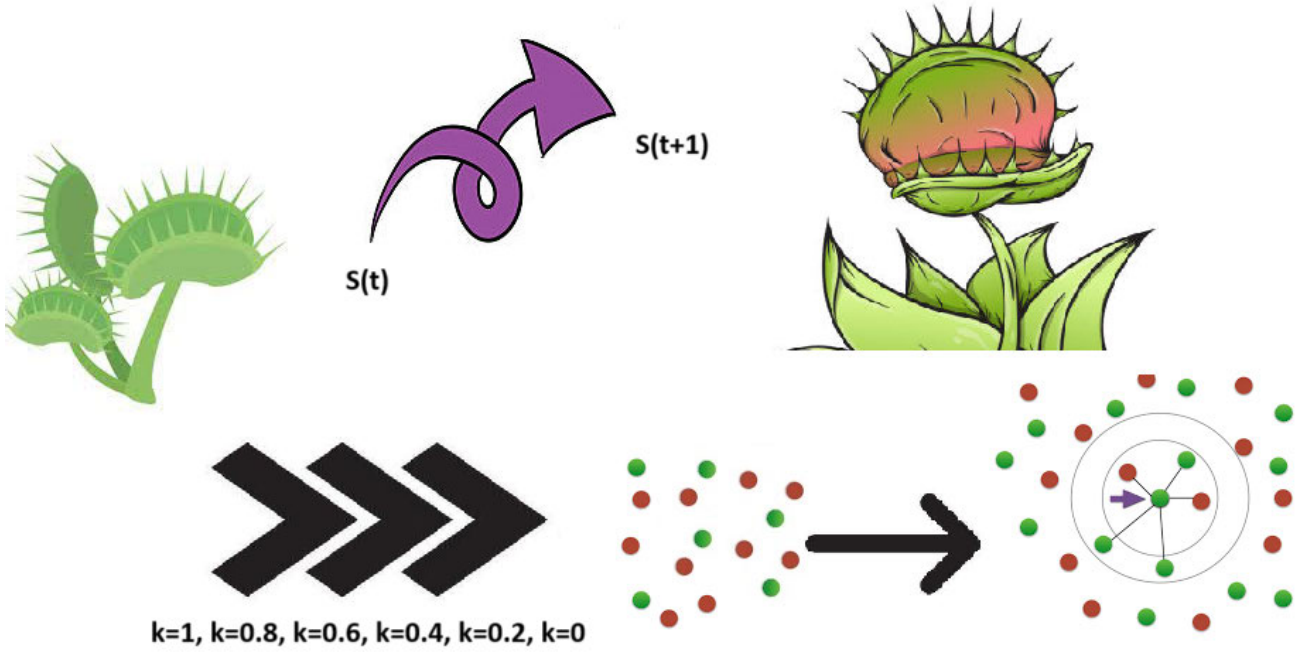

FIGURE 3. The facility of the WWPA algorithm.

population matrix of waterwheel locations; *Pi* represents the *i*-th waterwheel (a candidate solution), and $p_{i,j}$ (problem variable). Since each waterwheel stands for a distinct approach to the problem, its objective function can be determined independently. The values that make up the problem's objective function can be represented effectively in the form of a vector in equation (3).

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix} \tag{3}$$

where F is a vector containing all the objective function values, and F_i is the predicted value for the i-th waterwheel. The assessments of objective functions are utilized as the primary yardstick for choosing the best solutions. This means that the greatest value of the objective function corresponds to the best candidate solution (i.e., the best member). In contrast, the lowest value corresponds to the worst candidate solution (i.e., the worst member). The current optima will change over time due to the waterwheels' random movement throughout the search space at each iteration.

2) PHASE 1: POSITION IDENTIFICATION AND HUNTING OF INSECTS (EXPLORATION)

Waterwheels are powerful predators because of their keen sense of smell, allowing them to pinpoint pests' origins. A waterwheel will begin attacking any insect that gets within its range. After locating its prey, it launches an assault and continues its pursuit. WWPA models the first part of its population update process by simulating this waterwheel activity. By simulating the waterwheel's attack on the insect, which results in large variations in the waterwheel's location in the search space, we may improve WWPA's exploration capability in locating the optimal region and escaping from local optima. Using a simulation of the waterwheel's approach to the bug, equation (4) is used to obtain the waterwheel's new position. The previous site will be abandoned in favor of the one mentioned below if the value of the objective function is enhanced by shifting the waterwheel there.

$$W = r_1.(P(t) + 2K) \tag{4}$$

$$P(t+1) = P(t) + W.(2K + r_2) \tag{5}$$

where r_1 and r_2 are random variables having values between [0, 2] and [0, 1]. In addition, W is a vector that shows the circle's diameter in which the waterwheel plant will look for the potential locations, and K is an exponential variable with values in the range [0, 1]. Equation (6) can adjust the waterwheel's location if the solution remains unchanged after three iterations.

$$P(t+1) = Gaussian(\mu_P, \sigma) + r_1 \left(\frac{P(t) + 2K}{W} \right) \tag{6}$$

3) PHASE 2: CARRYING THE INSECT IN THE SUITABLE TUBE (EXPLOITATION)

An insect is sucked into a waterwheel and transferred to a feeding tube. This simulated waterwheel activity informs WWPA's second-stage population update. The model of

transporting the insect to the appropriate tube leading to the creation of small changes in the position of the waterwheel in the search space increases the WWPA's exploitation power during the local search, and better solutions are converged upon near the ones that have already been discovered. WWPA's designers emulate the natural behavior of waterwheels by determining a new random location as an ''excellent position for devouring insects'' for each waterwheel in the population. The following equations demonstrate that the waterwheel is relocated to the new position if the target function's value is more significant than the original location.

$$W = r_3.(KP_{best}(t) + r_3P(t)) \tag{7}$$

$$P(t+1) = P(t) + KW \tag{8}$$

where r_3 is a random variable with values in the range [0, 2], $P(t)$ is the current solution at iteration t, and P_{best} is the best solution.

Similar to the exploration phase, if the solution does not improve for three iterations, the following mutation is applied to avoid getting stuck into particular local minima.

$$P(t+1) = (r_1 + K)sin(\frac{F}{C}\theta) \tag{9}$$

where F and C are random variables with values in the range $[-5, 5]$. In addition, the value of K decreases exponentially using equation 10.

$$K = \left(1 + \frac{2 * t^2}{T_{max}} + F\right) \tag{10}$$

C. SEARCH SPACE BINARIZATION

Individuals whose representations form a binary search space make up the bWWPA search space. Binary numbers represent all individuals in the search space. This representation is necessary because it facilitates distinguishing desirable and undesirable features. Figure 4 shows the representation of the full search space available to the bWWPA algorithm. In the first step, the number of individuals in the search space is set by the population size *popsize* and the dataset dimension D. When applied to the search space, the WWPA $\bigoplus$ operation is supposed to yield optimal solutions with a new internal representation that ranges from 0 to 1 along all of D in ind_i. Each ind_i will receive results when the full optimization process is finished, which is expected to take several rounds. Cells with values of 1 s are expected to represent the chosen features. The number of features $|F|$ in the dataset X is roughly equivalent to the dimension of D for arbitrary solution ind_i. Therefore, for each ind_i that stands for an instance in dataset X, we tally the number of 1's in dimension D. Binarization of WWPA is an approach to feature selection problems that benefit from formalizing the search space. A breakdown of the proposed bWWPA approach and how its components work together follows.

D. BINARIZATION OF WWPA

By combining the original WWPA's functionality with several additional operators, a new variation of the algorithm has been designed to optimize solutions in a discrete solution space. The solution representation and optimization process can be transformed from a continuous to a discrete form through the first step, which defines transformation functions. This is essential so that the novel approach may tackle problems specific to feature selection. Altering the fitness function is the second modeled procedure for achieving the new variation of bWWPA. Finding the optimal answer overall means computing the fitness of all possible options. To address the specifics of the situation at hand, we provide a specification of the fitness function. The algorithmic structure of the bWWPA is also shown and explored, along with a diagram of its operation. The continuous solution resulting from the WWPA algorithm is converted to binary using the following sigmoid function, where S_{best} is the best continuous solution retrieved by the continuous WWPA. The representation of this sigmoid function is shown in Figure 5.

$$Binary = \begin{cases} 1 & \text{if } Sigmoid(S_{best}) \geq 0.5 \\ 0 & otherwise \end{cases},$$

$$Sigmoid(S_{Best}) = \frac{1}{1 + e^{-10(S_{Best} - 0.5)}} \tag{11}$$

E. FITNESS AND COST FUNCTIONS

Finding the most effective answer to the feature selection problem required a hybrid approach considering both fitness function assessment and cost function evaluation. As shown in the following equation, the solution is rated according to how well it performs using the classifier *clf* based on the application of control parameter Ω and a subset of the dataset $X[:\ 1^{ind_i}]$ while the tuning parameter is in effect. In the equation, the notation 1^{ind_i} yields the total number of 1's in the array standing in for the ind_i themselves.

$$fit = \Omega * (1 - clf\left(X[:\ 1^{ind_i}]\right)) + \left((1 - \Omega)\frac{|F|}{D}\right) \tag{12}$$

By subtracting the value returned by *fit* from 1, as shown in the following equation, the cost function is assessed based on the output of the fitness function. Each optimal solution for a given dataset is visually analyzed and interpreted using the fitness and cost function values.

$$cost = 1 - fit \tag{13}$$

F. PROPOSED bWWPA ALGORITHM

We provide the formalization of the proposed algorithm in Algorithm 1 and in the flowchart depicted in Figure 6. In this algorithm, the values for the initial parameters required for input and the global best solution are initialized, and the value of the cost function is calculated at each iteration. Lines 8-12 detail exploring the search space to find the best solution. On the other hand, lines 13–16 explain the exploitation in which the best solutions are exploited to find new regions in the search space for further investigation. Whether the procedure does a local or global search depends on the result of the examination of the conditions, we estimate the total number

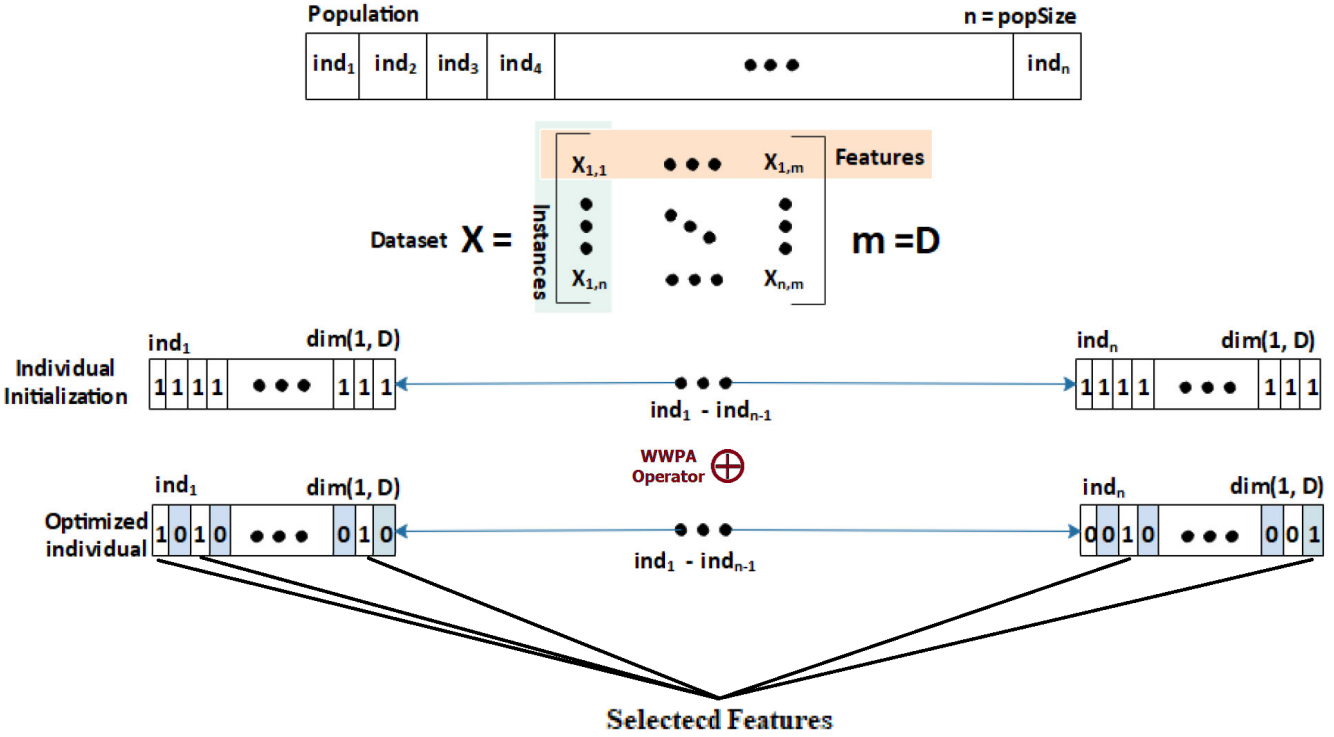

FIGURE 4. The search space representation of the feature selection process.

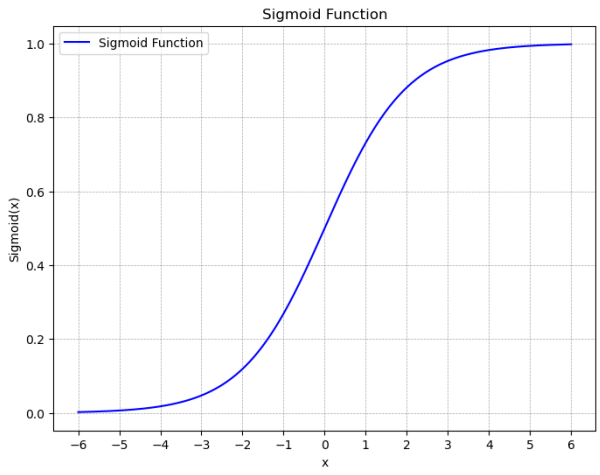

FIGURE 5. Sigmoid function corresponding to equation (11).

of open positions in both the exploitation and exploration scenarios. At last, the compartments are brought up to date, and the best global solution is established before the next iteration is carried out.

G. FEATURE SELECTION

The accuracy of a classifier applied to a subset of the dataset determines how much of the dataset to use in calculating the fitness and cost functions. KNN is served as the foundational classifier. In this research, we investigate the impact of many well-known classifiers on the feature selection problem and report the findings. The following equation is used to determine the number of selected features for a given individual ind_i, where D is the dimension of the feature size in the dataset, and $1^{ind_i^k}$ is the number of feature locations that have 1^s in them.

$$fs_i = \frac{\sum_{k=0}^{D}(1^{ind_i^k})}{D} \tag{14}$$

The KNN model is used to find groups of things that are related and thereby solves the classification problem. The most successful conditions were found by testing k-fold values of 5, 3, and 2. We found that a k-fold of 5 provided the best results across the majority of the datasets we tested, while a k-fold of 2 was best for the Iris and Lung datasets when utilizing the bWWPA algorithm. The experimental conditions and computing infrastructure utilized to evaluate this methodology will be discussed in depth in the next section.

H. FITNESS FUNCTION

The effectiveness of the proposed feature selection algorithm is evaluated in terms of the quality of its solutions using a fitness function. The two main variables influencing the fitness function are the number of features used for classification and the misclassification rate. It is deemed a good solution if it narrows down the range of features to choose from to reduce the classification error rate and the number of selected features. The following equation is used to determine how

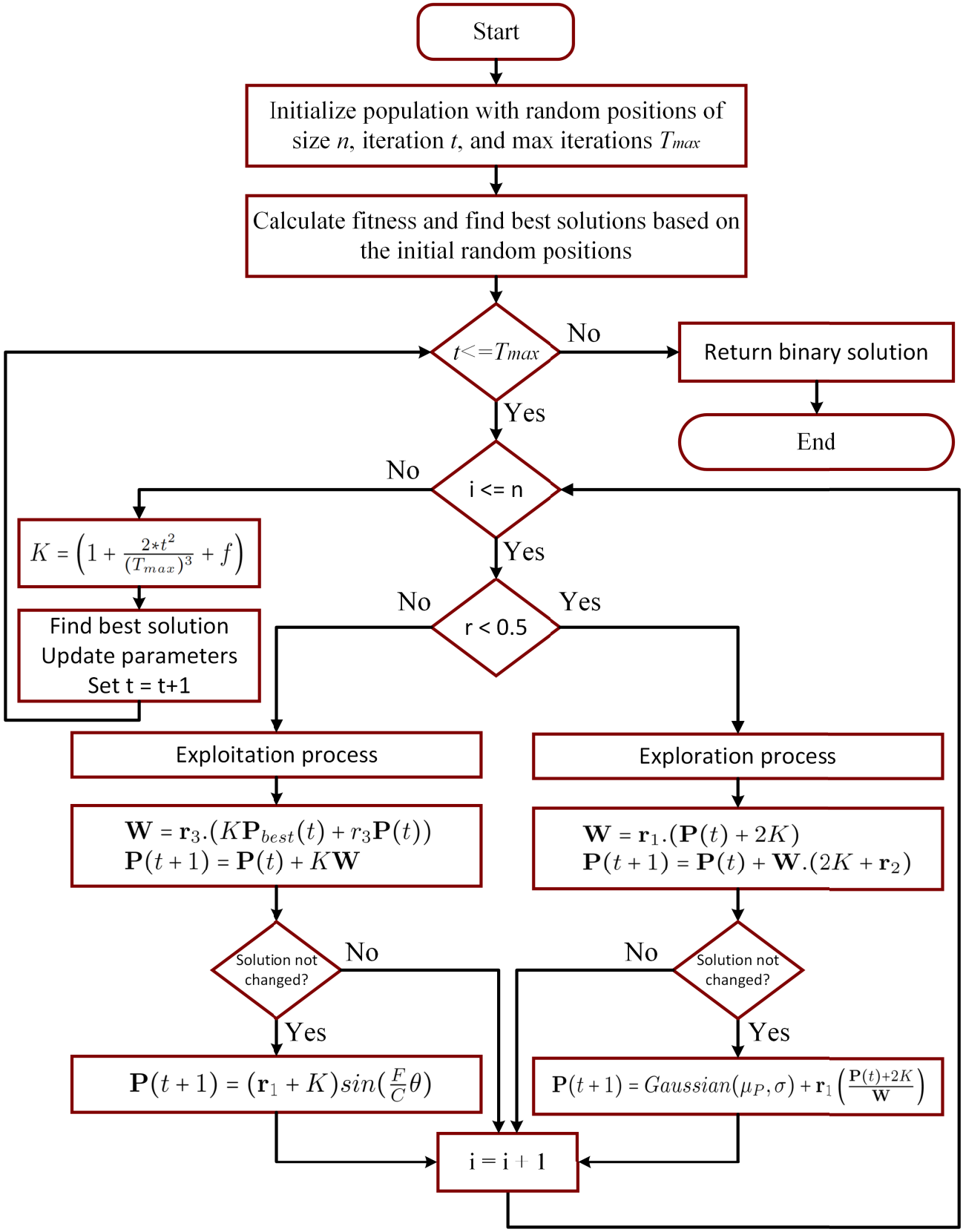

FIGURE 6. Flowchart of the feature selection process based on the proposed bWWPA.

well each solution performs.

$$F = h_1 \frac{|s|}{|f|} + h_2 Error(D) \tag{15}$$

where the total number of features and the number of selected features are denoted by $|f|$ and $|s|$, respectively. The classifier error is denoted by $Error(.)$ and h_1 and h_2 are parameters used to manage the importance of the selected features with $h_1 \in [0, 1]$ and $h_2 = 1 - h_1$.

Algorithm 1 : The Proposed Binary bWWPA Algorithm

Initialize waterwheel plants' positions $P_i(i = 1, 2, \ldots, n)$ for n plants, objective function f_n, iterations t, T_{max}, parameters of r, r_1, r_2, r_3, f, c, and K
Binarize the solution space
Calculate fitness of f_n for each position P_i
Find best plant position P_{best}
Set $t = 1$
while $t \leq T_{max}$ **do**
  for $(i = 1 : i < n + 1)$ **do**
    if $(r < 0.5)$ **then**
      Explore the waterwheel plant search space using:
      $W = r_1.(P(t) + 2K)$
      $P(t + 1) = P(t) + W.(2K + r_2)$
      if Solution does not change for three iterations **then**
        $P(t + 1) = Gaussian(\mu_P, \sigma) + r_1 \left(\frac{P(t)+2K}{W}\right)$
      end if
    else
      Exploit the current solutions to get best solution using:
      $W = r_3.(KP_{best}(t) + r_3 P(t))$
      $P(t + 1) = P(t) + KW$
      if Solution does not change for three iterations **then**
        $P(t + 1) = (r_1 + K) sin(\frac{F}{C}\theta)$
      end if
    end if
  end for
  Decrease the value of K exponentially using:
  $K = \left(1 + \frac{2*t^2}{(T_{max})^3} + f\right)$
  Update r, r_1, r_2, r_3, f, c
  Calculate objective function f_n for each position P_i
  Find the best position P_{best}
  Set $t = t + 1$
end while
Convert best solution to binary
Return best solution, cost of best solution

IV. EXPERIMENTAL SETUP

In this part, we present the experimental setup in detail. As a preliminary matter, we should mention that the PC we utilized for the studies had the following specifications: Processing unit, Intel® Core i5-4210U CPU 1.70 GHz, 2.40 GHz; Memory, 8 GB; Operating System, Windows 10. We also tried out various configurations on a number of other computers: a 1.70 GHz or 2.40 GHz Intel® Core i5-4200 processor, 16 GB of RAM, and a 64-bit version of Windows 10. Python version 3.9 and its supplementary libraries, including Numpy and others, were used to create the binary meta-heuristic algorithms. The computing environment is described here, and in the following sections, the input parameters and types are explained, as well as the results of the experiments. In this part, we also introduce the assessment criteria we used to compare the outcomes and explain why we chose these particular metrics.

TABLE 1. The UCI datasets employed in this research.

No.	Dataset	No. Attr.	No. Inst.	No. Classes
1	Zoo	17	101	7
2	Breast cancer tissue	9	106	2
3	Breast cancer Coimbra	9	116	2
4	Lymphography	18	148	4
5	Hepatitis	10	155	2
6	WineEW	13	178	3
7	Parkinson	22	195	2
8	SonarEW	60	208	2
9	Seeds	7	210	2
10	Glass	9	214	2
11	Lung cancer	21	226	5
12	SpectEW	22	267	2
13	HeartEW	13	270	2
14	Vertebral	6	310	2
15	Ionosphere	34	351	2
16	IonosphereEW	34	351	2
17	Fri_c0_500_10	10	500	2
18	Kc2	21	522	2
19	Climate	20	540	2
20	WDBC	30	569	2
21	Australian	14	690	2
22	Breast_Cancer	8	700	2
23	Blood	4	744	2
24	Segment	19	2310	2
25	Space-ga	6	3207	2
26	WaveformEW	21	5000	3
27	Diabetes	8	768	2
28	M-of-n	10	1324	2
29	HAR	561	10299	6
30	ISOLET	617	7797	26

A. DATASET

Extensive testing of bWWPA was conducted on 30 prominent and standard-setting datasets [48]. As such, the efficiency and performance of the approach described in this paper were evaluated using these datasets, which have been extensively utilized for comparative evaluations of binary meta-heuristic algorithms. Dataset details that were used can be found in Table 1.

This table has datasets with high, moderate, and low dimensions, all amenable to exploration using the bWWPA technique. This was required because testing an algorithm's performance on several datasets, especially high-dimensional ones equivalent to practical binary optimization problems, is crucial. High-dimensional features created due to the proliferation of biomedical datasets have a detrimental

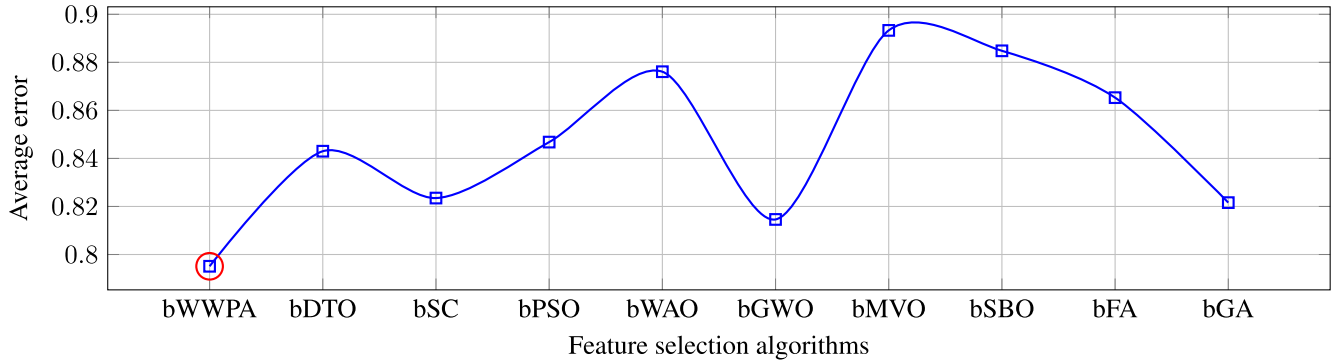

FIGURE 7. The average error over thirty UCI datasets.

TABLE 2. Configuration of compared algorithms with 100 iterations and ten agents for each.

Algorithm	Parameter (s)	Value (s)
GWO	a	2 to 0
PSO	Inertia W_{max}, W_{min}	[0.9,0.6]
	Acceleration constants C_1, C_2	[2,2]
DTO	Exploration percentage	70
WOA	a	2 to 0
	r	[0,1]
MVO	Wormhole existence probability	[0.2,1]
GA	Mutation ratio	0.1
	Crossover	0.9
	Selection mechanism	Roulette wheel
SBO	Step size	0.94
	Mutation probability	0.05
	Upper and lower limit difference	0.02
FA	# fireflies	10
SCA	Parameters r_2, r_3, r_4,	[0,1]

TABLE 3. Configuration parameters of the proposed bWWPA algorithm.

Parameter (s)	Value (s)
Number of iterations	100
Number of search agents	10
Search domain	[0, 1]
Problem dimension	number of features in the data
Number of runs	20
r_2, r_3, r_4	[0, 1]
h_1	0.99
h_2	0.01

TABLE 4. Evaluation metrics used in assessing the proposed feature selection method.

Metric	Value
Average fitness	$\frac{1}{M}\sum_{i=1}^{M} S_i^*$
Best fitness	$min_{i=1}^{M} S_i^*$
Worst fitness	$max_{i=1}^{M} S_i^*$
Average fitness size	$\frac{1}{M}\sum_{i=1}^{M} size(S_i^*)$
Average error	$\frac{1}{M}\sum_{j=1}^{M}\frac{1}{N}\sum_{i=1}^{N} MSE(\hat{V}_i - V_i)$
Standard deviation	$\sqrt{\frac{1}{M-1}\sum_{i=1}^{M}\left(S_i^* - Mean\right)^2}$
Average Run time	$\frac{1}{M}\sum_{i=1}^{M} RunTime_{o,i}$

impact on machine learning classifiers. When tested on high-dimensional datasets like the ever-expanding collection of biomedical datasets, many feature selection methods reported in the literature run into problems with a variety of population and local optima. Since feature selection aims to pick the most valuable features from a pool that may have included noise, we must demonstrate the bWWPA's effectiveness with high-dimensional datasets. This research investigates high-dimensional datasets with feature sizes between 4 to 617. The majority of these data sets include classification problems, either binary or multi-classification problems, such as in the Lung and WaveformEW datasets. Some datasets with a medium number of dimensions include BreastEW, Kc2, Blood, M-of-n, and Tic-tac-toe. While the Blood dataset has approximately four features, the other datasets in this group typically contain between 6 and 617 features.

B. PARAMETER CONFIGURATION AND SETTINGS

The proposed method was compared to nine binary variants of meta-heuristic algorithms: the binary dipper throated optimizer (bDTO) [49], the binary sine cosine optimizer (bSC) [50], the binary particle swarm optimizer (bPSO) [51], [52], the binary whale optimization algorithm (bWOA) [53], the binary grey wolf optimizer (bGWO) [54], [55], the binary multiverse optimizer (bMVO) [56], and the binary satin bowerbird optimizer (bSBO) [57], the binary firefly algorithm (bFA) [28], the binary genetic algorithm (bGA) [58]. In Table 2, the algorithm-specific parameter settings that were used are presented. The algorithms were trained using 100 iterative procedures, with each experiment generally being conducted 10 times to get an average performance. The next section details the equations used to determine these averages and the similar measures we compared.

TABLE 5. Average error results of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	**0.5517**	0.5654	0.5585	0.5557	0.5526	0.5621	0.5526	0.5703	0.5595	0.5526
Breast cancer tissue	**0.4890**	0.5020	0.5031	0.5298	0.5149	0.4906	0.5238	0.5102	0.5204	0.5097
Breast cancer Coimbra	**0.5609**	0.5740	0.5721	0.5794	0.5711	0.5627	0.5871	0.5934	0.5857	0.5750
Lymphography	0.6417	0.6517	**0.6051**	0.6353	0.6467	0.6117	0.6339	0.6132	0.6446	0.6346
Hepatitis	0.4901	0.5080	0.5084	0.5061	0.4969	0.5061	0.5085	**0.4838**	0.4961	0.5015
WineEW	**0.5048**	0.5198	0.5076	0.5116	0.5140	0.5363	0.5153	0.5189	0.5222	0.5098
Parkinsons	**0.5947**	0.6038	0.6092	0.6110	0.6042	0.6040	0.6032	0.6036	0.5978	0.6098
SonarEW	**0.3972**	0.4031	0.4044	0.3992	0.3982	0.4044	0.3988	0.4018	0.4010	0.4014
Seeds	0.6257	0.6230	**0.6109**	0.6302	0.6296	0.6304	0.6257	0.6135	0.6290	0.6376
Glass	3.6202	3.7832	3.6700	3.8515	3.9526	3.6104	4.1168	3.9110	3.8454	**3.5454**
Lung cancer	**0.4885**	0.5008	0.4931	0.5015	0.5215	0.5008	0.5000	0.5115	0.5123	0.5069
SpectEW	**0.5118**	0.5137	0.5181	0.5194	0.5179	0.5149	0.5151	0.5203	0.5176	0.5197
HeartEW	0.5768	0.6002	0.5863	0.5865	0.5841	0.5847	0.5790	**0.5693**	0.5862	0.5807
Vertebral	**0.3868**	0.4039	0.3974	0.4210	0.3890	0.3878	0.4006	0.4070	0.4115	0.4236
Ionosphere	**0.7472**	0.7578	0.7832	0.7808	0.7506	0.7912	0.7706	0.7694	0.7777	0.7975
IonosphereEW	0.6117	0.6270	0.6292	0.6240	**0.6110**	0.6174	0.6146	0.6549	0.6302	0.6276
Fri_c0_500_10	**0.6197**	0.6334	0.6265	0.6237	0.6206	0.6301	0.6206	0.6383	0.6275	0.6206
Kc2	**0.5570**	0.5700	0.5711	0.5978	0.5829	0.5586	0.5918	0.5782	0.5884	0.5777
Climate	**0.6289**	0.6420	0.6401	0.6474	0.6391	0.6307	0.6551	0.6614	0.6537	0.6430
WDBC	**0.5728**	0.5878	0.5756	0.5796	0.5820	0.6043	0.5833	0.5869	0.5902	0.5778
Australian	**0.4652**	0.4711	0.4724	0.4672	0.4662	0.4724	0.4668	0.4698	0.4690	0.4694
Breast_Cancer	**0.5565**	0.5688	0.5611	0.5695	0.5895	0.5688	0.5680	0.5795	0.5803	0.5749
Blood	**0.5798**	0.5817	0.5861	0.5874	0.5859	0.5829	0.5831	0.5883	0.5856	0.5877
Segment	**0.4548**	0.4719	0.4654	0.4890	0.4570	0.4558	0.4686	0.4750	0.4795	0.4916
Space-ga	0.8152	0.8258	**0.8512**	0.8488	0.8186	0.8592	0.8386	0.8374	0.8457	0.8655
WaveformEW	0.6937	0.6910	**0.6789**	0.6982	0.6976	0.6984	0.6937	0.6815	0.6970	0.7056
Diabetes	3.6882	3.8512	3.7380	3.9195	4.0206	3.6784	4.1848	3.9790	3.9134	**3.6134**
M-of-n	**0.4838**	0.5389	0.5602	0.5555	0.5489	0.5196	0.5406	0.5305	0.5579	0.5642
HAR Using Smartphones	**0.8362**	1.4100	1.3100	1.2417	1.9966	1.0878	2.1680	2.2666	1.7466	1.3105
ISOLET	**1.1015**	1.3082	1.1122	1.3349	1.4224	1.1767	1.3907	1.4191	1.3877	1.1121

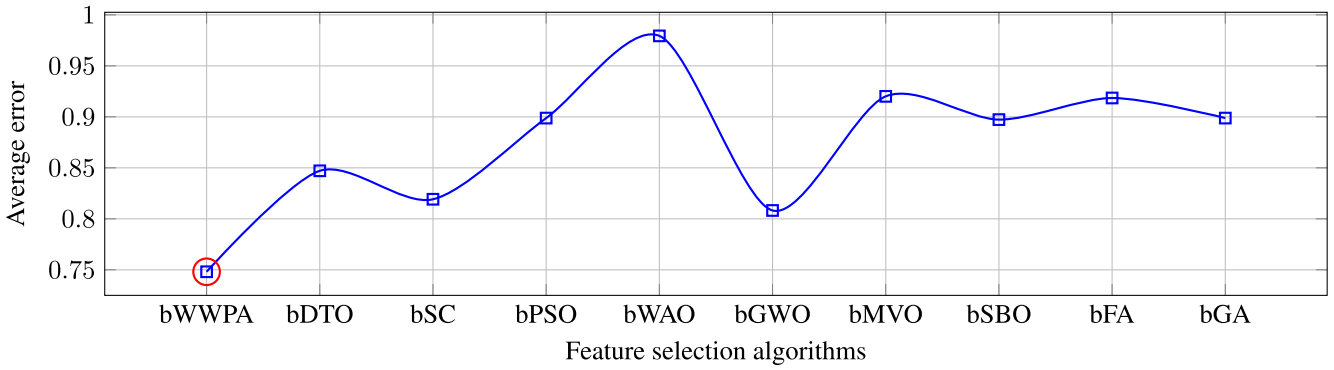

FIGURE 8. The average select size over thirty UCI datasets.

C. EVALUATION METRICS

In the following, the evaluation metrics employed to assess the performance of the proposed methodology are presented. These metrics include classification accuracy, average accuracy, maximum accuracy, and standard deviation fitness. Table 3 contains the formulas used to calculate these metrics [59], [60]. This table denotes the number of runs of the proposed and other competing optimizers by M. The best solution at the run number j is denoted by S_j^*, $size(S_j^*)$ refers to the length of the best solution vector. N denotes the number of points in the test set. $\hat{V}_n$ and V_n refer to the predicted and actual values, respectively. The calculation of the average runtime is also presented in this table, where M is the number of iterations for optimization technique o, and runtime is the actual computation time for technique o at run i [61], [62].

V. RESULTS AND DISCUSSION OF FINDINGS

In this section, we provide results comparing the efficacy of the proposed bWWPA to that of the other nine binary optimizers. These algorithms were chosen because they are widely considered the best of the best when it comes to binary

TABLE 6. Average select size results of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	**0.6059**	0.6699	0.6649	0.7999	0.8949	0.7161	0.7699	0.7849	0.8049	0.7649
Breast cancer tissue	**0.4013**	0.5316	0.5558	0.7558	0.5967	0.5190	0.7361	0.6952	0.7528	0.6558
Breast cancer Coimbra	**0.6715**	0.7649	0.7649	0.7732	0.7732	0.7536	0.7816	0.6979	0.7732	0.7732
Lymphography	0.7935	**0.7720**	0.8069	0.9649	0.9506	0.8261	0.7935	0.8649	0.8506	0.8149
Hepatitis	**0.5211**	0.6763	0.6558	0.7331	0.7490	0.5739	0.8013	0.7549	0.7354	0.7126
WineEW	**0.5370**	0.6935	0.7220	0.7970	0.9435	0.6363	0.7792	0.8363	0.7899	0.7720
Parkinsons	0.9074	0.9649	0.8649	0.9149	1.0399	**0.8085**	1.0024	0.9149	1.0274	1.0399
SonarEW	**0.7414**	0.7899	0.7649	0.8587	0.9024	0.7863	0.8587	0.9149	0.9024	0.8462
Seeds	**0.6082**	0.7774	0.7899	0.8712	0.9024	0.6899	0.8524	0.7899	0.8399	0.8274
Glass	0.7849	0.6399	**0.5427**	0.7482	0.7732	0.7860	0.7427	0.7982	0.7232	0.6955
Lung cancer	**0.5240**	0.6263	0.6467	0.7581	0.7649	0.6558	0.7399	0.7558	0.7558	0.7240
SpectEW	**0.5649**	0.5849	0.6049	0.6249	0.5899	0.5860	0.5999	0.6049	0.6199	0.5999
HeartEW	**1.0649**	1.0649	1.1316	1.0816	1.1149	1.0960	1.1482	1.0789	1.1482	1.1149
Vertebral	**0.9049**	1.1174	1.1149	0.9549	1.2424	0.9949	1.1149	1.0549	1.0124	1.1349
Ionosphere	**0.9133**	0.9967	0.9824	1.0491	1.3633	1.0681	1.1086	1.0681	1.0752	1.1205
IonosphereEW	**0.5918**	0.7470	0.7265	0.8038	0.8197	0.6446	0.8720	0.8256	0.8061	0.7833
Fri_c0_500_10	**0.6077**	0.7642	0.7927	0.8677	1.0142	0.7070	0.8499	0.9070	0.8606	0.8427
Kc2	0.9781	1.0356	0.9356	0.9856	1.1106	**0.8792**	1.0731	0.9856	1.0981	1.1106
Climate	**0.8121**	0.8606	0.8356	0.9294	0.9731	0.8570	0.9294	0.9856	0.9731	0.9169
WDBC	**0.6789**	0.8481	0.8606	0.9419	0.9731	0.7606	0.9231	0.8606	0.9106	0.8981
Australian	0.8556	0.7106	**0.6134**	0.8189	0.8439	0.8567	0.8134	0.8689	0.7939	0.7662
Breast_Cancer	**0.5947**	0.6970	0.7174	0.8288	0.8356	0.7265	0.8106	0.8265	0.8265	0.7947
Blood	**0.6356**	0.6556	0.6756	0.6956	0.6606	0.6567	0.6706	0.6756	0.6906	0.6706
Segment	**1.1356**	1.1356	1.2023	1.1523	1.1856	1.1667	1.2189	1.1496	1.2189	1.1856
Space-ga	**0.9756**	1.1881	1.1856	1.0256	1.3131	1.0656	1.1856	1.1256	1.0831	1.2056
WaveformEW	**0.7808**	0.8642	0.8499	0.9166	1.2308	0.9356	0.9761	0.9356	0.9427	0.9880
Diabetes	**0.7967**	0.8689	0.8102	0.9467	1.0856	0.8090	0.9467	0.8059	0.9689	0.9467
M-of-n	**0.5173**	0.9356	0.5332	1.0006	1.1956	0.5426	0.9806	0.7435	1.0106	1.0206
HAR Using Smartphones	**0.9027**	1.2201	1.1108	1.2150	1.2391	1.0343	1.2521	1.2906	1.2917	1.1070
ISOLET	**1.0343**	1.2127	1.1141	1.1497	1.3007	1.1069	1.2712	1.3170	1.2688	1.1308

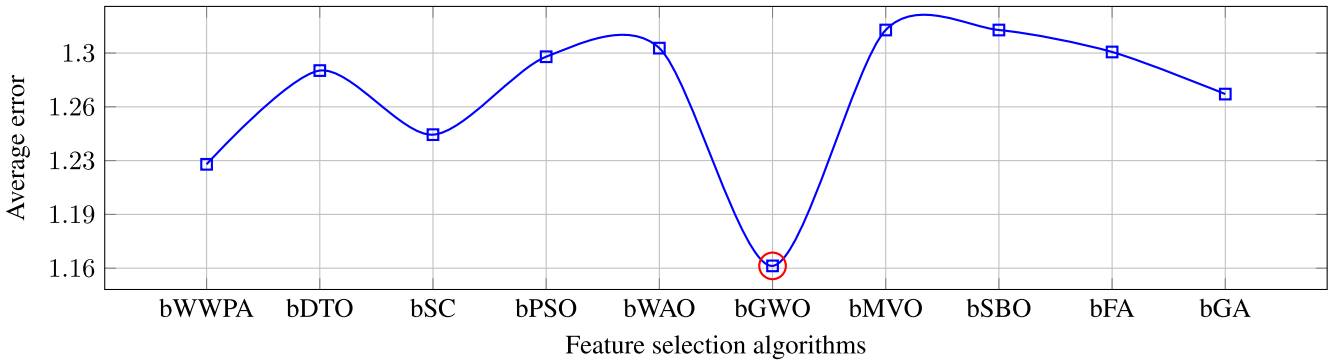

FIGURE 9. The average fitness over thirty UCI datasets.

optimization and have shown exceptional performance in previous studies. We point out that we used the same parameterization, such as the number of iterations and parameter values, in the proposed analyses of most of these methods. The parts that follow are structured as shown below. We begin by comparing and contrasting each available approach's fitness performance and feature selection. We then compare the success rates of several methods for classifying data. We then examine how different approaches' cost functions stack up against one another and demonstrate how the classifiers we choose affect the feature categorization process overall. At last, we detail the time required by each technique and analyze the results. The following sections present the achieved results using tables and graphs with more explanation.

The proposed bWWPA algorithm is evaluated using 30 datasets drawn from the UCI machine learning library to gauge its quality and efficacy. To test the proposed approach

TABLE 7. Average fitness results of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	**0.5140**	0.5499	0.5322	0.5363	0.5373	0.5157	0.5373	0.5442	0.5441	0.5373
Breast cancer tissue	**0.4357**	0.4706	0.4372	0.4981	0.4833	0.4673	0.4922	0.4442	0.4888	0.4782
Breast cancer Coimbra	0.6318	0.6798	**0.5062**	0.6850	0.6769	0.5122	0.6927	0.6272	0.6913	0.6807
Lymphography	**0.6535**	0.6834	0.7432	0.6671	0.6784	0.6769	0.6657	0.7172	0.6763	0.6664
Hepatitis	0.4596	0.4520	**0.4422**	0.4688	0.4596	0.4605	0.4711	0.4782	0.4589	0.4642
WineEW	0.6022	0.6171	**0.4422**	0.6089	0.6112	0.4709	0.6125	0.4535	0.6194	0.6072
Parkinsons	1.1411	1.1609	0.5438	1.1681	1.1613	0.5386	1.1603	**0.5382**	1.1530	1.1669
SonarEW	**0.6126**	0.6285	0.6390	0.6246	0.6236	0.6360	0.6242	0.8364	0.6263	0.6268
Seeds	**0.8659**	0.8749	0.8755	0.8820	0.8815	0.8750	0.8776	0.8826	0.8809	0.8894
Glass	**3.3628**	3.7113	4.6770	3.7790	3.8790	3.7790	4.0416	5.0280	3.7729	3.4759
Lung cancer	**0.4513**	0.4634	0.4677	0.4642	0.4840	0.4554	0.4627	0.4661	0.4749	0.4695
SpectEW	1.6784	1.6803	0.4527	1.6859	1.6844	**0.4295**	1.6816	0.4549	1.6841	1.6862
HeartEW	**2.9558**	2.9790	3.0065	2.9655	2.9630	3.0084	2.9580	3.2365	2.9651	2.9597
Vertebral	**1.4650**	1.5715	1.7320	1.5884	1.5568	1.6375	1.5683	1.8416	1.5791	1.5910
Ionosphere	**1.4717**	1.4822	1.7178	1.5050	1.4750	1.6128	1.4948	1.7040	1.5019	1.5215
IonosphereEW	**0.8992**	0.9145	1.0792	0.9115	0.8986	0.9330	0.9022	0.9095	0.9176	0.9151
Fri_c0_500_10	0.6022	0.6171	**0.4422**	0.6089	0.6112	0.4709	0.6125	0.4535	0.6194	0.6072
Kc2	0.9843	1.0041	0.3870	1.0113	1.0045	**0.3818**	1.0035	0.3814	0.9962	1.0101
Climate	**0.6573**	0.6732	0.6837	0.6693	0.6683	0.6807	0.6689	0.8811	0.6710	0.6715
WDBC	**0.9106**	0.9196	0.9202	0.9267	0.9262	0.9197	0.9223	0.9273	0.9256	0.9341
Australian	**3.4075**	3.7560	4.7217	3.8237	3.9237	3.8237	4.0863	5.0727	3.8176	3.5206
Breast_Cancer	**0.4960**	0.5081	0.5124	0.5089	0.5287	0.5001	0.5074	0.5108	0.5196	0.5142
Blood	1.7231	1.7250	0.4974	1.7306	1.7291	**0.4742**	1.7263	0.4996	1.7288	1.7309
Segment	**3.0005**	3.0237	3.0512	3.0102	3.0077	3.0531	3.0027	3.2812	3.0098	3.0044
Space-ga	**1.5097**	1.6162	1.7767	1.6331	1.6015	1.6822	1.6130	1.8863	1.6238	1.6357
WaveformEW	**1.5164**	1.5269	1.7625	1.5497	1.5197	1.6575	1.5395	1.7487	1.5466	1.5662
Diabetes	**0.9439**	0.9592	1.1239	0.9562	0.9433	0.9777	0.9469	0.9542	0.9623	0.9598
M-of-n	**0.8375**	0.8911	0.8518	0.9075	0.9009	0.8553	0.8928	0.9133	0.9098	0.9161
HAR Using Smartphones	**0.8782**	1.1906	1.1230	1.1895	1.2474	1.0993	1.2687	1.3084	1.2564	1.1080
ISOLET	**1.0110**	1.1782	1.1095	1.2151	1.2795	1.1104	1.2651	1.3195	1.2487	1.1351

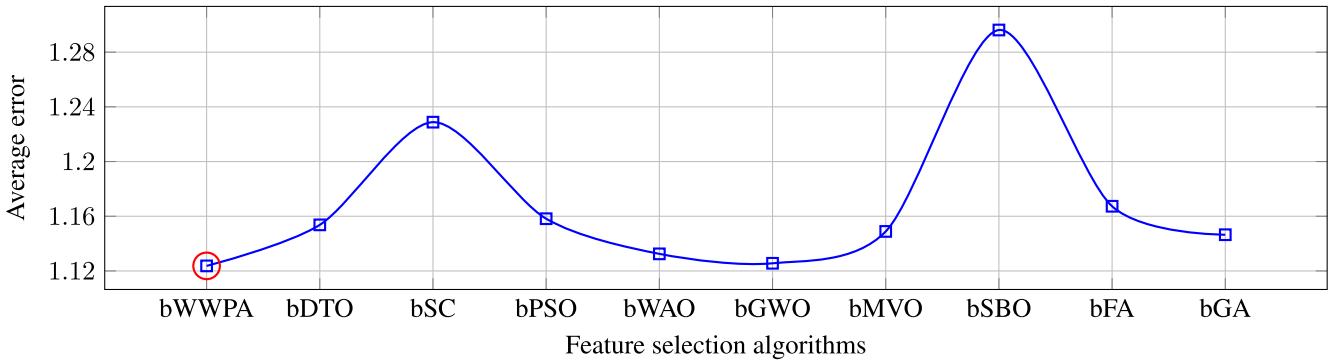

FIGURE 10. The average best fitness over thirty UCI datasets.

on a wide range of problems, many datasets were chosen with varying numbers of features, instances, and classes; two of these datasets contain more than 500 attributes each. A dataset's training, validation, and testing segments are all represented equally. During its development, the KNN classifier relies on the training component. For a given solution, the fitness function can be calculated with the help of validation, and the effectiveness of the proposed model can be tested with the help of testing. Using 10 search agents, each optimizer is executed 20 times for 100 iterations. The k-fold cross-validation value is set to 10, and the k-neighbors parameter for the KNN classifier is 5. We set h_1 to a value of 0.99 and h_2 to a value of 0.01.

Table 5, Table 6, and Table 7 display the results of the proposed optimization algorithm in terms of average error, average select size, and average fitness (Mean), respectively. The optimizer first picks the optimal collection of features to train the classifier and achieve a smaller error on

TABLE 8. Best fitness results of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	**0.4090**	0.4284	0.5060	0.4284	0.4672	0.4566	0.4866	0.4866	0.4672	0.4284
Breast cancer tissue	**0.3740**	0.3909	0.4163	0.4248	0.3994	0.4241	0.3825	0.4078	0.3740	0.4078
Breast cancer Coimbra	**0.5872**	0.6064	0.6545	0.5968	0.5968	0.5956	0.6160	0.6545	0.6160	0.5968
Lymphography	**0.3793**	0.4642	0.5207	0.4359	0.4359	0.4460	0.5066	0.5066	0.5207	0.4642
Hepatitis	**0.3398**	0.3545	0.3850	0.4002	0.3698	0.4459	0.3698	0.3698	0.3545	0.3698
WineEW	**0.5555**	0.5684	0.5770	0.5641	0.5598	0.5727	0.5770	0.5856	0.5727	0.5684
Parkinsons	1.1111	1.1111	1.1150	1.1190	1.1111	1.1360	1.1190	1.1171	**1.1071**	**1.1071**
SonarEW	0.5880	0.5918	0.6088	0.5960	0.5876	0.5943	0.5918	0.6045	**0.5833**	0.5960
Seeds	**0.8082**	0.8275	0.8429	0.8352	0.8159	0.8335	0.8352	0.8275	0.8236	0.8236
Glass	2.2467	2.1861	3.1356	2.1861	1.7618	**1.7012**	1.9436	4.1054	2.2467	2.0244
Lung cancer	**0.3698**	0.3698	0.4002	0.3698	0.3850	0.4135	0.3698	0.3850	0.3698	0.3850
SpectEW	**1.6485**	1.6497	1.6642	1.6634	1.6613	1.6642	1.6597	1.6637	1.6497	1.6650
HeartEW	2.9232	2.9232	**2.9151**	2.9232	2.9232	2.9159	2.9164	2.9251	2.9232	2.9232
Vertebral	**1.4725**	1.5501	1.5465	1.5525	1.5409	1.4911	1.5433	1.5598	1.5578	1.5630
Ionosphere	**1.3964**	1.4338	1.4808	1.4279	1.4071	1.4872	1.4469	1.4433	1.4201	1.4528
IonosphereEW	**0.9293**	0.9293	0.9479	0.9324	0.9448	0.9545	0.9293	0.9510	0.9386	0.9417
Fri_c0_500_10	**0.6420**	0.6549	0.6635	0.6506	0.6463	0.6592	0.6635	0.6721	0.6592	0.6549
Kc2	1.1976	1.1976	1.2015	1.2055	1.1976	1.2225	1.2055	1.2036	**1.1936**	**1.1936**
Climate	0.6745	0.6783	0.6953	0.6825	0.6741	0.6808	0.6783	0.6910	**0.6698**	0.6825
WDBC	**0.8947**	0.9140	0.9294	0.9217	0.9024	0.9200	0.9217	0.9140	0.9101	0.9101
Australian	2.3332	2.2726	3.2221	2.2726	1.8483	**1.7877**	2.0301	4.1919	2.3332	2.1109
Breast_Cancer	**0.4563**	0.4563	0.4867	0.4563	0.4715	0.5000	0.4563	0.4715	0.4563	0.4715
Blood	**1.7350**	1.7362	1.7507	1.7499	1.7478	1.7507	1.7462	1.7502	1.7362	1.7515
Segment	3.0097	3.0097	**3.0016**	3.0097	3.0097	3.0024	3.0029	3.0116	3.0097	3.0097
Space-ga	**1.5590**	1.6366	1.6330	1.6390	1.6274	1.5776	1.6298	1.6463	1.6443	1.6495
WaveformEW	**1.4829**	1.5203	1.5673	1.5144	1.4936	1.5737	1.5334	1.5298	1.5066	1.5393
Diabetes	**0.9293**	0.9293	0.9479	0.9324	0.9448	0.9545	0.9293	0.9510	0.9386	0.9417
M-of-n	**0.8168**	0.8505	0.8437	0.8639	0.8886	0.8337	0.8505	0.8280	0.9043	0.8796
HAR Using Smartphones	**0.8659**	1.1990	1.1281	1.1801	1.2762	1.0724	1.2465	1.2665	1.2773	1.1277
ISOLET	**0.9761**	1.1703	1.0769	1.2114	1.2781	1.0992	1.2785	1.1654	1.2551	1.1553

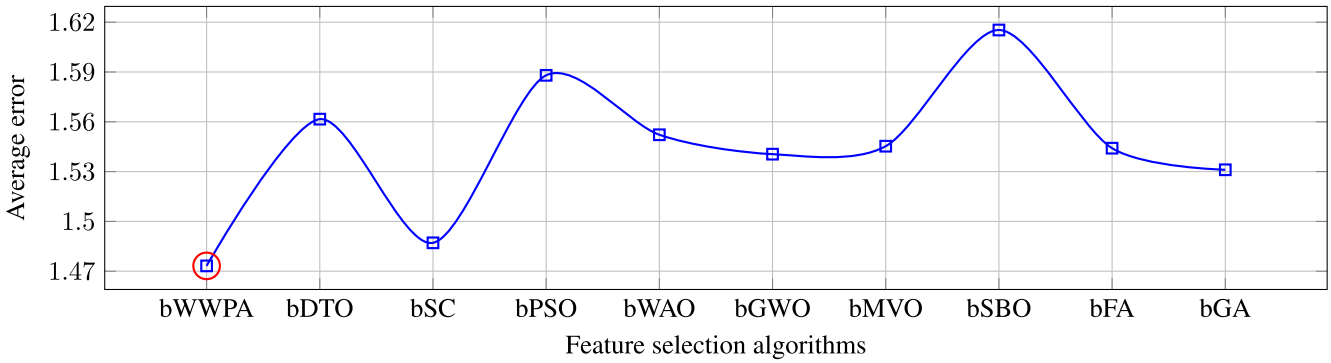

FIGURE 11. The average worst fitness over thirty UCI datasets.

the concealed test data. The proposed (bWWPA) algorithm achieves the lowest overall average error for all the datasets compared to the other optimization algorithms, as shown in Figure 7.

The efficacy of the proposed method is demonstrated by the average selected attributes in Table 6. Maintaining a low error rate is crucial, even when selecting a smaller number of features suggests that the optimizer is engaging in feature selection. In this way, the fitness function pushes the optimizer to pick fewer features by giving the classification error a larger weight. Most datasets may be classified to a lower level with the help of the bWWPA algorithm, which uses the fewest possible channels. Figure 8 shows the average select size over the 30 datasets. The figure shows that the proposed bWWPA picks the smallest average select size compared to the other binary optimization algorithms. This reveals a significant advantage of the proposed algorithm.

TABLE 9. Worst fitness results of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	0.6440	0.6630	**0.6242**	0.6824	0.6824	0.6630	0.6436	0.6436	0.7019	0.6436
Breast cancer tissue	0.5658	0.5753	**0.5245**	0.5838	0.6007	0.5838	0.5668	0.5499	0.6176	0.6261
Breast cancer Coimbra	0.7695	0.7527	**0.7143**	0.7815	0.7815	0.7911	0.8200	0.8200	0.9257	0.8200
Lymphography	0.8298	0.9714	0.7593	0.9290	0.9290	**0.7393**	0.8583	0.7593	0.8724	0.9431
Hepatitis	0.5610	**0.5474**	0.5626	0.5931	0.5779	0.5779	0.5779	0.5626	0.5626	0.5626
WineEW	0.6667	0.7166	0.6606	0.6994	0.6821	0.7553	0.6649	**0.6563**	0.8156	0.6735
Parkinsons	1.2550	1.2550	1.2391	1.2550	1.2351	1.2391	1.2550	1.2391	**1.2152**	1.2550
SonarEW	0.6800	0.6961	0.6791	0.6748	**0.6706**	0.6876	0.6748	0.6748	0.6833	0.6876
Seeds	0.9478	0.9763	**0.8990**	0.9415	0.9608	0.9454	0.9647	0.9067	0.9531	1.0266
Glass	5.6024	5.5603	**5.3582**	6.2472	5.7825	5.9472	5.8027	6.9746	5.4189	5.4391
Lung cancer	0.5487	0.5931	**0.5474**	0.6083	0.6388	0.5879	0.5626	0.5779	0.5626	0.6083
SpectEW	1.7216	1.7236	1.7304	1.7348	1.7308	**1.7078**	1.7280	1.7328	1.7360	1.7320
HeartEW	**2.8056**	3.4283	2.9975	3.0529	3.1001	2.9718	3.0380	3.0069	3.1812	3.0515
Vertebral	**1.4856**	1.6184	1.6027	1.6601	1.6076	1.6156	1.6156	1.6124	1.6236	1.6601
Ionosphere	**1.5085**	1.6104	1.5700	1.5885	1.5308	1.5647	1.5754	1.5795	1.5968	1.6116
IonosphereEW	**0.9789**	1.0137	1.0044	1.0695	0.9795	1.0075	1.0013	1.0261	1.0013	0.9889
Fri_c0_500_10	1.3720	1.3720	1.3561	1.3720	1.3521	1.3561	1.3720	1.3561	**1.3322**	1.3720
Kc2	0.7970	0.8131	0.7961	0.7919	**0.7876**	0.8046	0.7919	0.7919	0.8004	0.8046
Climate	1.0648	1.0933	**1.0160**	1.0585	1.0779	1.0624	1.0817	1.0237	1.0701	1.1436
WDBC	5.7194	5.6773	**5.4753**	6.3642	5.8996	6.0642	5.9198	7.0916	5.5359	5.5561
Australian	0.6658	0.7101	**0.6644**	0.7253	0.7558	0.7049	0.6797	0.6949	0.6797	0.7253
Breast_Cancer	1.8386	1.8406	1.8474	1.8518	1.8478	**1.8248**	1.8450	1.8498	1.8530	1.8490
Blood	**2.9226**	3.5453	3.1145	3.1699	3.2171	3.0888	3.1550	3.1239	3.2982	3.1685
Segment	**1.6026**	1.7354	1.7197	1.7772	1.7246	1.7326	1.7326	1.7294	1.7406	1.7772
Space-ga	**1.6255**	1.7275	1.6871	1.7055	1.6478	1.6817	1.6924	1.6966	1.7138	1.7287
WaveformEW	**1.0959**	1.1307	1.1214	1.1866	1.0966	1.1245	1.1183	1.1431	1.1183	1.1059
Diabetes	0.7837	0.8336	0.7776	0.8164	0.7992	0.8723	0.7819	**0.7733**	0.9326	0.7906
M-of-n	1.0171	1.0652	**1.0091**	1.0764	1.0428	1.0376	1.0652	1.0203	1.0383	1.0652
HAR Using Smartphones	**1.0045**	1.3033	1.3047	1.3145	1.4246	1.2250	1.3824	1.2825	1.3693	1.2469
ISOLET	1.1168	**1.2980**	1.2502	1.3268	1.4014	1.2515	1.3913	1.5609	1.3740	1.2702

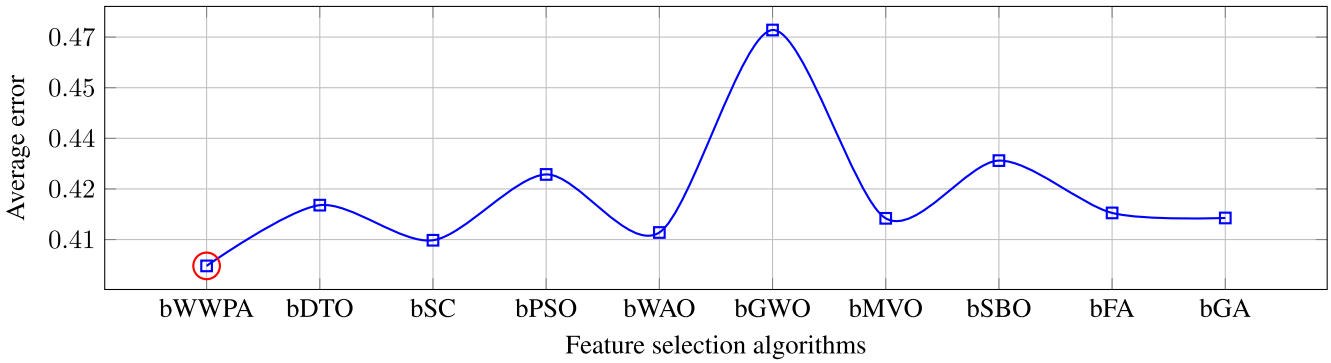

FIGURE 12. The average standard deviation fitness over thirty UCI datasets.

Table 7 presents the recorded values of the average fitness achieved by the proposed bWWPA and the other competing feature selection algorithms. The recorded values show the superiority of the proposed approach in achieving the lowest average fitness for several datasets such as Zoo, Beast cancer tissue,.., and other datasets. However, the plot shown in Figure 7 depicts an overall assessment of the average fitness. In this figure, it can be noted that the proposed bWWPA algorithm achieves the second-best algorithm compared to the other algorithms. These results emphasize the superiority of the proposed algorithm when compared to eight different algorithms included in the conducted experiments.

Table 8 and Table 9 displays the best and worst fitness scores obtained from a number of distinct optimization methods. The results in these tables show that the proposed bWWPA algorithm outperforms existing optimization methods across several iterations in terms of fitness. The average best fitness and the average worst fitness over all the datasets are shown in Figure 10 and Figure 11, respectively. These figures show that the proposed method achieves the

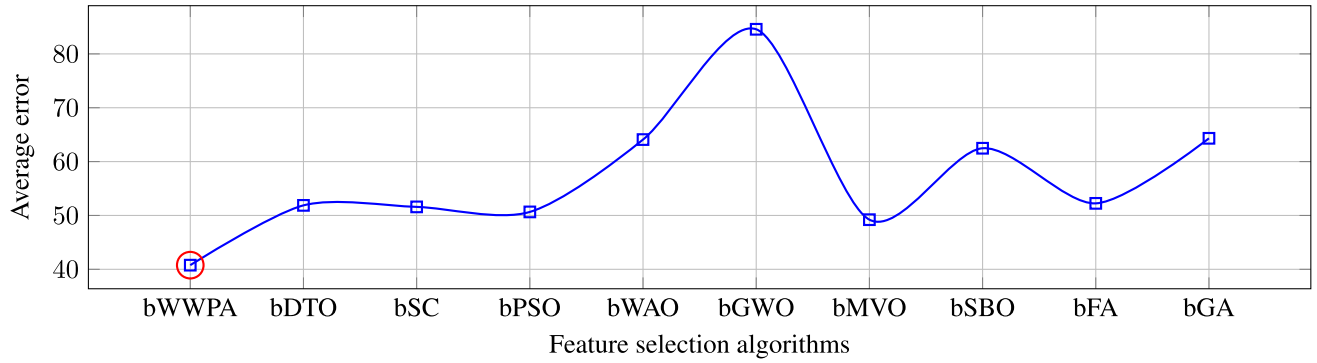

FIGURE 13. The average time profile (in seconds) over thirty UCI datasets.

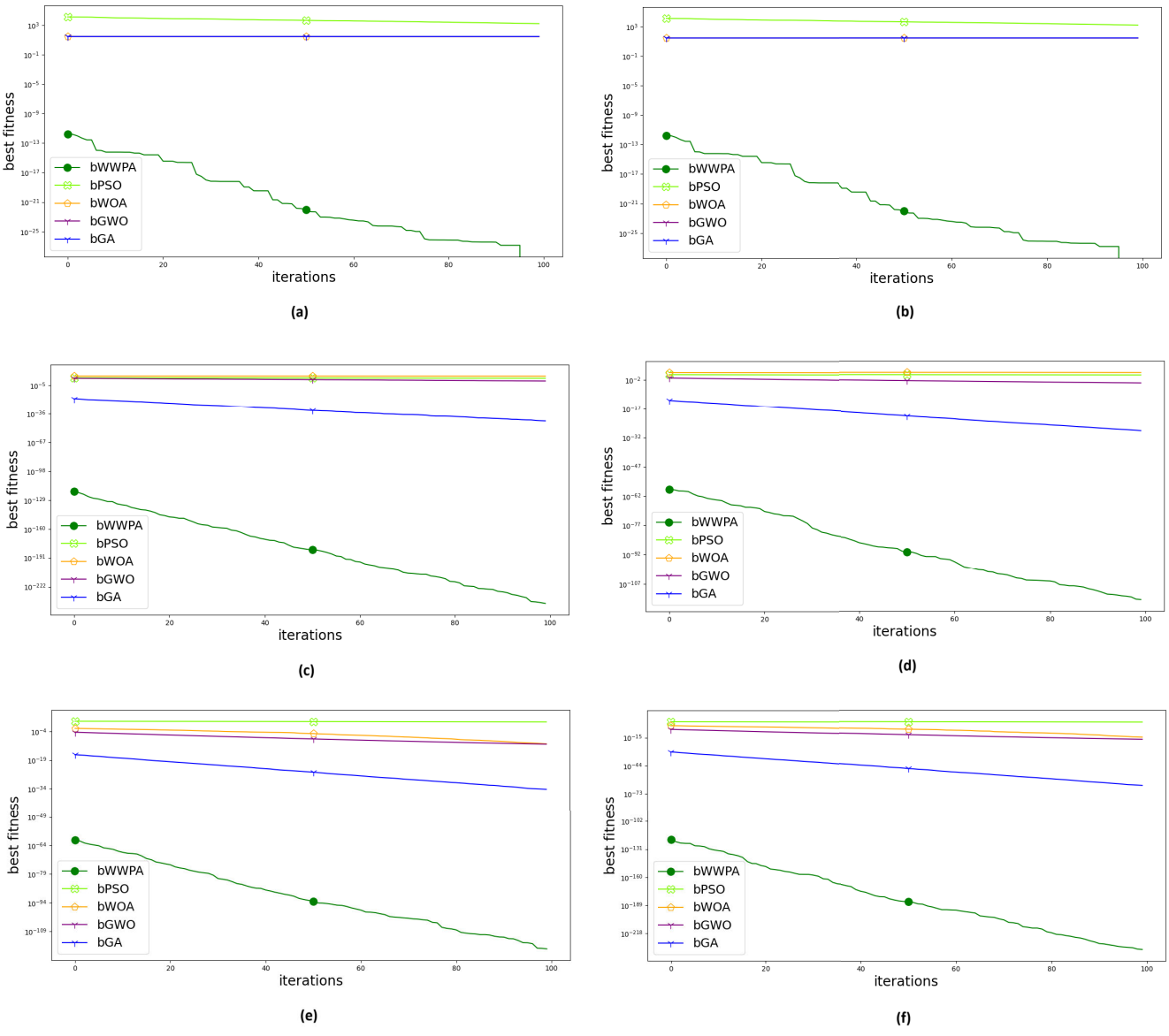

FIGURE 14. The convergence curves of the proposed bWWPA compared to other techniques. These convergence curves are for sample datasets, which are (a) Zoo, (b) Parkinson, (c) Wine, (d) Climate, (e) HAR, and (f) ISOLET.

lowest fitness, outperforming the average best fitness using the other optimization methods. Table 10 also includes the statistical findings standard deviation. Compared to other algorithms, the proposed bWWPA method has the lowest

TABLE 10. Standard deviation fitness results of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	**0.3109**	0.3299	0.3135	0.3376	0.3242	0.3204	0.3178	0.3255	0.3286	0.3283
Breast cancer tissue	**0.3110**	0.3205	0.3163	0.3113	0.3203	0.3157	0.3177	0.3156	0.3260	0.3255
Breast cancer Coimbra	**0.3040**	0.3068	0.3066	0.3180	0.3157	0.3156	0.3184	0.3291	0.3420	0.3230
Lymphography	0.3710	0.3942	0.3595	0.3818	0.3852	0.3731	**0.3574**	0.3774	0.3611	0.3837
Hepatitis	**0.3099**	0.3193	0.3297	0.3209	0.3305	0.3134	0.3308	0.3356	0.3224	0.3155
WineEW	**0.2934**	0.3100	0.2969	0.2999	0.3016	0.3121	0.2934	0.2947	0.3235	0.2959
Parkinsons	**0.2983**	0.3031	0.3118	0.3065	0.3028	0.3006	0.3030	0.3183	0.2989	0.3094
SonarEW	0.2907	0.2922	0.2906	**0.2873**	0.2891	0.2887	0.2895	0.2896	0.2912	0.2913
Seeds	**0.2938**	0.3082	0.2955	0.2950	0.3005	0.3096	0.3047	0.2945	0.3035	0.3131
Glass	**1.1188**	1.1794	1.1584	1.4144	1.1252	2.0405	1.2501	1.4102	1.1794	1.2160
Lung cancer	**0.3187**	0.3234	0.3224	0.3275	0.3338	0.3226	0.3245	0.3412	0.3234	0.3256
SpectEW	0.2847	0.2864	0.2879	0.2858	0.2874	0.2847	0.2858	0.2888	0.2897	**0.2843**
HeartEW	**0.2928**	0.3777	0.2934	0.3050	0.3101	0.3228	0.2973	0.2967	0.3241	0.2995
Vertebral	**0.2840**	0.2867	0.2845	0.2943	0.2846	0.2887	0.2863	0.2864	0.2855	0.2927
Ionosphere	0.2992	0.3096	**0.2986**	0.3074	0.3035	0.3094	0.3093	0.3205	0.3114	0.3117
IonosphereEW	**0.2984**	0.3125	0.3176	0.3017	0.3074	0.3001	0.2984	0.3214	0.3076	0.3079
Fri_c0_500_10	**0.3834**	0.4000	0.3869	0.3899	0.3916	0.4021	0.3834	0.3847	0.4135	0.3859
Kc2	**0.3883**	0.3931	0.4018	0.3965	0.3928	0.3906	0.3930	0.4083	0.3889	0.3994
Climate	0.3807	0.3822	0.3806	**0.3773**	0.3791	0.3787	0.3795	0.3796	0.3812	0.3813
WDBC	**0.3838**	0.3982	0.3855	0.3850	0.3905	0.3996	0.3947	0.3845	0.3935	0.4031
Australian	**1.2088**	1.2694	1.2484	1.5044	1.2152	2.1305	1.3401	1.5002	1.2694	1.3060
Breast_Cancer	**0.4087**	0.4134	0.4124	0.4175	0.4238	0.4126	0.4145	0.4312	0.4134	0.4156
Blood	0.3747	0.3764	0.3779	0.3758	0.3774	0.3747	0.3758	0.3788	0.3797	**0.3743**
Segment	**0.3828**	0.4677	0.3834	0.3950	0.4001	0.4128	0.3873	0.3867	0.4141	0.3895
Space-ga	**0.3740**	0.3767	0.3745	0.3843	0.3746	0.3787	0.3763	0.3764	0.3755	0.3827
WaveformEW	0.3892	0.3996	**0.3886**	0.3974	0.3935	0.3994	0.3993	0.4105	0.4014	0.4017
Diabetes	**0.3884**	0.4025	0.4076	0.3917	0.3974	0.3901	0.3884	0.4114	0.3976	0.3979
M-of-n	0.3979	0.4064	0.4088	0.4142	0.4015	0.4371	0.3891	0.4171	0.4076	**0.3846**
HAR Using Smartphones	**0.3849**	0.4014	0.3979	0.3992	0.4224	0.3902	0.4136	0.4181	0.4125	0.3981
ISOLET	**0.3913**	0.4094	0.4058	0.4079	0.4303	0.3981	0.4192	0.4203	0.4204	0.3982

TABLE 11. Time profile in seconds of the proposed bWWPA algorithm and compared binary algorithms.

Dataset	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	**7.0384**	7.6824	7.2744	7.3644	8.7644	7.1714	7.4544	7.6254	7.8144	8.3924
Breast cancer tissue	**7.4564**	9.3874	8.5244	8.9594	8.7144	8.2924	7.9944	8.8994	9.1944	8.9764
Breast cancer Coimbra	**6.7084**	7.9044	8.1444	7.9834	8.0744	6.7194	7.4054	7.6764	8.1844	8.2324
Lymphography	**6.7094**	7.3754	7.5044	8.2204	7.7344	6.8114	3.0.2	7.9884	7.3544	7.3124
Hepatitis	**6.7124**	7.6554	7.1044	7.4344	7.9444	7.3874	7.4454	7.5584	7.7144	7.6464
WineEW	**8.7774**	10.0464	10.1494	9.9604	11.2944	10.1324	9.1044	9.9444	10.2044	10.1514
Parkinsons	8.7874	8.5104	9.0934	**6.9884**	9.6144	8.9064	9.2044	10.0424	9.9444	10.3564
SonarEW	**8.5914**	9.8984	9.0934	10.2104	9.5344	9.5824	9.4144	10.2544	9.9744	9.7944
Seeds	**11.4554**	12.4654	13.0844	11.9474	13.7744	12.0864	12.4544	12.4364	12.9644	12.0214
Glass	**8.7524**	9.0734	9.4344	9.7584	10.1644	10.6314	9.6344	10.9614	9.7444	11.6164
Lung cancer	**9.2274**	9.8714	9.4634	9.5534	10.9534	9.3604	9.6434	9.8144	10.0034	10.5814
SpectEW	**9.6454**	11.5764	10.7134	11.1484	10.9034	10.4814	10.1834	11.0884	11.3834	11.1654
HeartEW	**8.8974**	10.0934	10.3334	10.1724	10.2634	8.9084	9.5944	9.8654	10.3734	10.4214
Vertebral	**8.8984**	9.5644	9.6934	10.4094	9.9234	9.0004	3.0.2	10.1774	9.5434	9.5014
Ionosphere	**8.9014**	9.8444	9.2934	9.6234	10.1334	9.5764	9.6344	9.7474	9.9034	9.8354
IonosphereEW	**10.9664**	12.2354	12.3384	12.1494	13.4834	12.3214	11.2934	12.1334	12.3934	12.3404
Fri_c0_500_10	10.9764	10.6994	11.2824	**9.1774**	11.8034	11.0954	11.3934	12.2314	12.1334	12.5454
Kc2	**10.7804**	12.0874	11.2824	12.3994	11.7234	11.7714	11.6034	12.4434	12.1634	11.9834
Climate	**13.6444**	14.6544	15.2734	14.1364	15.9634	14.2754	14.6434	14.6254	15.1534	14.2104
WDBC	**10.9414**	11.2624	11.6234	11.9474	12.3534	12.8204	11.8234	13.1504	11.9334	13.8054
Australian	**13.6444**	14.6544	15.2734	14.1364	15.9634	14.2754	14.6434	14.6254	15.1534	14.2104
Breast_Cancer	**9.5604**	10.1054	10.4834	11.2524	12.2824	9.7624	10.4034	10.2784	11.0634	10.8284
Blood	**13.6914**	14.9374	15.1434	13.7384	16.2434	14.8464	15.4734	14.1944	15.4834	14.1924
Segment	**57.7054**	111.6004	122.0634	86.5414	141.9434	79.3094	65.7134	86.6514	89.2834	87.5174
Space-ga	**17.6404**	22.4144	20.4434	20.9764	25.3934	21.8294	22.1934	23.8794	37.9584	23.4424
WaveformEW	**105.9144**	138.3534	142.9234	147.8404	156.3734	733.2374	110.4634	168.5144	146.2914	175.4664
Diabetes	**38.9954**	58.6444	69.2234	65.0484	90.4834	74.9404	77.1934	60.4314	56.5734	62.6504
M-of-n	**16.0474**	17.5394	17.7134	17.5274	16.5834	17.0304	18.2834	17.2934	18.2534	17.5604
HAR Using Smartphones	**329.8234**	466.3934	446.2034	477.1734	609.9534	634.1834	426.5834	623.4734	469.9234	610.4734
ISOLET	**436.1434**	499.5834	487.4934	465.8934	624.3934	740.3434	456.2034	646.3634	499.2934	702.4634

standard deviation, demonstrating its consistency and durability across various datasets. In addition, the plot in Figure 12 depicts the average standard deviation fitness across the complete set of datasets employed in the conducted experiments.

TABLE 12. p-values using Wilcoxon's rank-sum ($p > 0.05$ are underlined) of bWWPA in comparison to other algorithms.

Dataset	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Zoo	9.47E-04	9.47E-04	7.58E-05	**2.38E-01**	6.69E-05	**6.63E-02**	9.56E-04	6.69E-05	**8.86E-02**
Breast cancer tissue	9.47E-04	9.47E-04	7.58E-05	6.69E-05	3.76E-02	6.69E-05	9.56E-04	6.69E-05	6.69E-05
Breast cancer Coimbra	9.47E-04	9.47E-04	7.58E-05	6.69E-05	4.78E-02	6.69E-05	9.47E-04	6.69E-05	6.69E-05
Lymphography	9.47E-04	9.47E-04	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
Hepatitis	9.47E-04	9.47E-04	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
WineEW	9.47E-04	9.47E-04	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.59E-05
Parkinsons	9.47E-04	9.47E-04	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
SonarEW	9.47E-04	9.47E-04	7.58E-05	**7.44E-02**	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.59E-05
Seeds	9.47E-04	9.47E-04	7.58E-05	6.69E-05	6.69E-05	**8.82E-02**	9.56E-04	6.69E-05	6.69E-05
Glass	9.47E-04	9.47E-04	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	**7.06E-02**
Lung cancer	1.25E-03	7.58E-05	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
SpectEW	1.25E-03	7.58E-05	7.58E-05	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
HeartEW	1.25E-03	7.58E-05	**6.30E-02**	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
Vertebral	1.25E-03	7.58E-05	6.69E-05	6.69E-05	**5.88E-02**	6.69E-05	9.56E-04	7.58E-05	7.58E-05
Ionosphere	1.25E-03	7.58E-05	**7.20E-02**	6.69E-05	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
IonosphereEW	1.25E-03	7.58E-05	6.69E-05	**6.15E-02**	6.69E-05	6.69E-05	9.56E-04	6.69E-05	6.69E-05
Fri_c0_500_10	1.25E-03	7.58E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	7.58E-05	6.69E-05	6.69E-05
Kc2	1.25E-03	7.58E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	7.58E-05	6.69E-05	6.69E-05
Climate	9.24E-02	9.16E-02	6.69E-05	6.69E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	**7.33E-02**
WDBC	1.25E-03	7.58E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
Australian	1.25E-03	7.58E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
Breast_Cancer	1.25E-03	7.58E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
Blood	1.25E-03	7.58E-05	7.58E-05	7.58E-05	7.58E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
Segment	1.25E-03	7.58E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
Space-ga	1.25E-03	7.58E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
WaveformEW	1.25E-03	7.58E-05	7.58E-05	7.58E-05	7.58E-05	6.69E-05	6.69E-05	7.58E-05	6.69E-05
Diabetes	1.25E-03	6.59E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
M-of-n	1.25E-03	6.59E-05	6.59E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
HAR Using Smartphones	1.25E-03	6.59E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05	6.69E-05
ISOLET	1.25E-03	6.59E-05	6.59E-05	6.69E-05	6.69E-05	6.69E-05	7.58E-05	6.69E-05	6.69E-05

TABLE 13. Statistical analysis of the results achieved by bWWPA when applied to the HAR-Smartphones dataset.

	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Number of values	20	20	20	20	20	20	20	20	20	20
Minimum	0.8162	1.21	1.11	1.142	1.697	1.088	1.992	2.167	1.347	1.011
25% Percentile	0.8362	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311
Median	0.8362	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311
75% Percentile	0.8362	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311
Maximum	0.8362	1.71	1.41	1.442	2	1.288	2.368	2.567	1.875	1.611
Range	0.02	0.5	0.3	0.3	0.3033	0.2	0.3763	0.4	0.5281	0.6
10% Percentile	0.8172	1.23	1.22	1.242	1.907	1.088	2.122	2.267	1.657	1.311
90% Percentile	0.8362	1.59	1.31	1.332	1.997	1.188	2.348	2.457	1.837	1.501
Actual confidence level	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%
Lower confidence limit	0.8362	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311
Upper confidence limit	0.8362	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311
Mean	0.8337	1.415	1.3	1.252	1.977	1.108	2.177	2.292	1.733	1.326
Std. Deviation	0.006387	0.105	0.05525	0.05525	0.06964	0.05231	0.07685	0.08507	0.1007	0.1089
Std. Error of Mean	0.001428	0.02348	0.01235	0.01235	0.01557	0.0117	0.01718	0.01902	0.02252	0.02436
Lower 95% CI of mean	0.8307	1.366	1.274	1.226	1.944	1.083	2.141	2.252	1.686	1.275
Upper 95% CI of mean	0.8367	1.464	1.326	1.278	2.009	1.132	2.213	2.331	1.78	1.376
Coefficient of variation	0.7661%	7.421%	4.250%	4.414%	3.523%	4.722%	3.531%	3.712%	5.811%	8.219%
Geometric mean	0.8337	1.411	1.299	1.251	1.975	1.107	2.175	2.29	1.73	1.321
Geometric SD factor	1.008	1.076	1.046	1.043	1.038	1.046	1.035	1.036	1.066	1.088
Lower 95% CI of geo. mean	0.8307	1.364	1.272	1.226	1.941	1.084	2.14	2.252	1.679	1.27
Upper 95% CI of geo. mean	0.8367	1.461	1.326	1.276	2.011	1.13	2.211	2.329	1.782	1.374
Harmonic mean	0.8337	1.408	1.298	1.25	1.974	1.106	2.174	2.289	1.726	1.317
Lower 95% CI of harm. mean	0.8306	1.361	1.27	1.226	1.938	1.084	2.14	2.252	1.671	1.265
Upper 95% CI of harm. mean	0.8367	1.457	1.327	1.274	2.012	1.128	2.21	2.326	1.785	1.373
Quadratic mean	0.8337	1.419	1.301	1.253	1.978	1.109	2.178	2.293	1.736	1.33
Lower 95% CI of quad. mean	0.8308	1.367	1.276	1.226	1.947	1.083	2.141	2.251	1.692	1.278
Upper 95% CI of quad. mean	0.8367	1.468	1.326	1.28	2.008	1.135	2.214	2.334	1.779	1.38
Skewness	-2.441	0.8008	-2.164	2.164	-3.872	2.745	0.9621	2.315	-3.091	-0.05495
Kurtosis	4.771	3.919	8.208	8.208	15.52	7.401	4.465	6.032	12.56	5.522
Sum	16.67	28.3	26	25.03	39.54	22.16	43.53	45.83	34.66	26.51

The plots emphasize the superiority of the proposed methods as they achieve the lowest standard deviation fitness compared to other methods.

As indicated in Table 11, the last experiment examines how long various optimization strategies process. A faster elapsed time suggests that the optimizer can more quickly

TABLE 14. Analysis of variance (ANOVA) test applied to the results of the HAR-Smartphones dataset using bWWPA feature selection method.

	SS	DF	MS	F(DFn, DFd)	P-value
Treatment	41.72	9	4.636	F (9, 190) = 773.8	P<0.0001
Residual	1.138	190	0.005991		
Total	42.86	199			

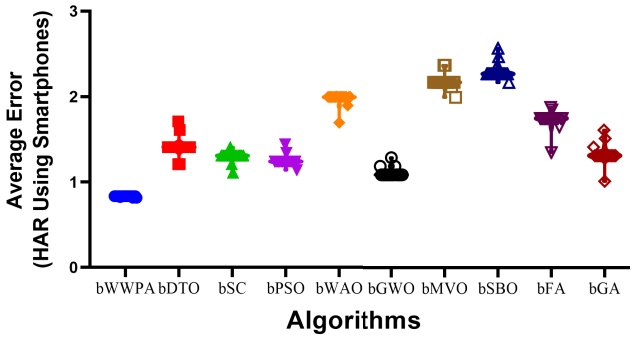

FIGURE 15. Average error based on the HAR dataset.

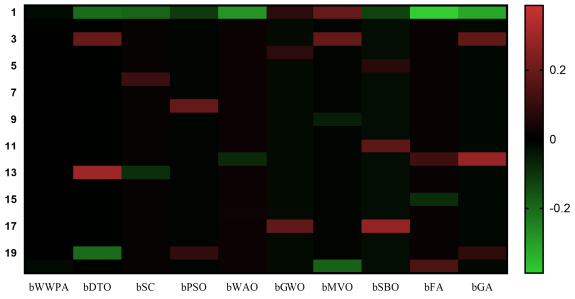

FIGURE 16. The heatmap of the ANOVA test using the HAR dataset.

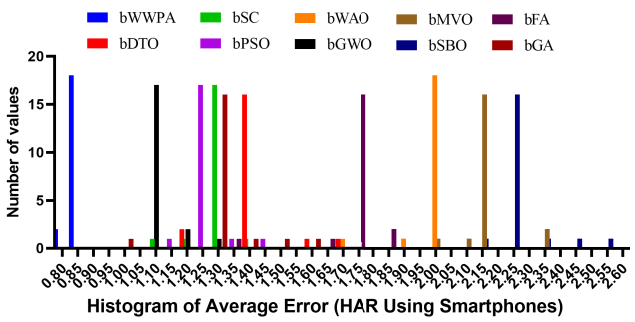

FIGURE 17. Histogram of average error for the HAR dataset.

identify the features that will yield the best results. For the higher-dimensional HAR Using Smartphones and ISOLET datasets, the suggested optimizer achieves results that are competitive with those of previous techniques. Figure 13 shows that the suggested optimizer can avoid local optima and has a rapid convergence time, proving its strong exploitation potential. This demonstrates that the bWWPA method is trustworthy and dependable in locating the best possible subset of features in a practical period of time.

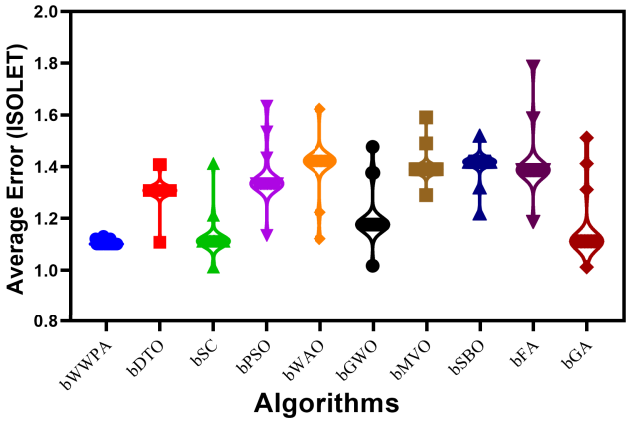

FIGURE 18. Average error based on the ISOLET dataset.

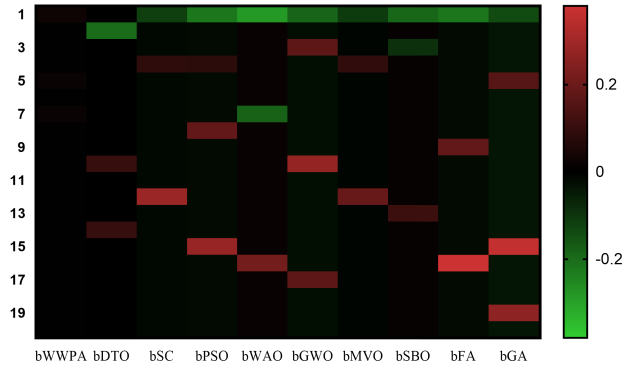

FIGURE 19. The heatmap of the ANOVA test using the ISOLET dataset.

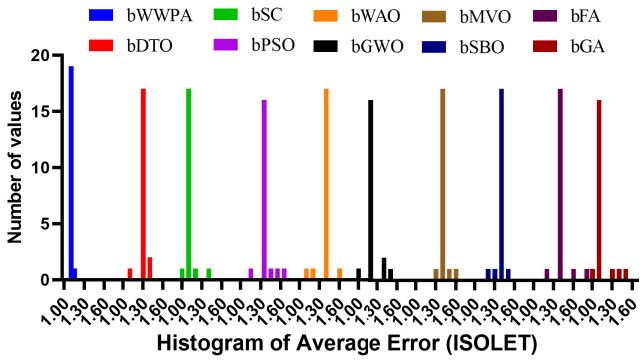

FIGURE 20. Histogram of average error for the ISOLET dataset.

On the other hand, we compare the proposed bWWPA method to various meta-heuristic algorithms and calculate their p-values using Wilcoxon's rank-sum test. Using this metric, we can see if there is a substantial difference between the outcomes of our suggested algorithm and those of competing algorithms. The results of the proposed algorithm are substantially different from those of the compared methods if the p-value is less than 0.05. A p-value greater than 0.05 indicates no statistically significant difference between the groups. The worst p-values in Table 12 are those larger than 0.05. As seen in the table below, the

TABLE 15. Wilcoxon signed rank test of the results achieved by applying bWWPA feature selection to the HAR-Smartphones dataset.

	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Theoretical median	0	0	0	0	0	0	0	0	0
Actual median	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311
Number of values	20	20	20	20	20	20	20	20	20
Sum of signed ranks (W)	210	210	210	210	210	210	210	210	210
Sum of positive ranks	210	210	210	210	210	210	210	210	210
Sum of negative ranks	0	0	0	0	0	0	0	0	0
P-value (two tailed)	0.0012468	0.0000659	0.0000659	0.0000669	0.0000669	0.0000669	0.0000669	0.0000669	0.0000669
Exact or estimate?	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact
Significant (alpha=0.05)?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Discrepancy	1.41	1.31	1.242	1.997	1.088	2.168	2.267	1.747	1.311

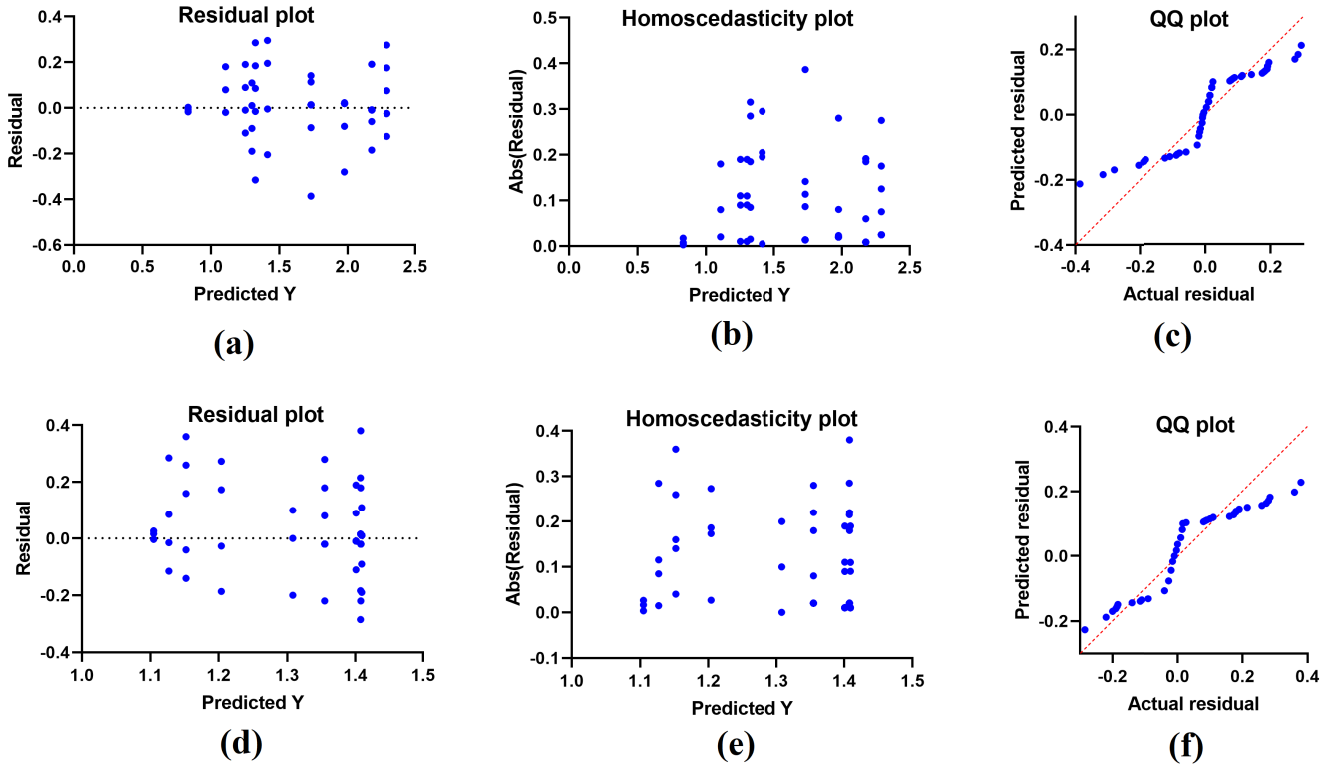

FIGURE 21. Analysis plots of the ANOVA test results based on the HAR dataset in the top row plots and based on the ISOLET dataset in the bottom row plots.

TABLE 16. Analysis of variance test of the results achieved by applying bWWPA feature selection to the ISOLET-Smartphones dataset.

	SS	DF	MS	F(DFn, DFd)	P-value
Treatment	2.915	9	0.3239	F (9, 190) = 47.32	P<0.0001
Residual	1.301	190	0.006845		
Total	4.216	199			

p-values obtained using this test are less than 0.05 when comparing the suggested approach to others. This demonstrates the efficacy and statistical significance of the bWWPA algorithm.

Figure 14 displays plots of the cost function values that were calculated for each method and dataset combination. Inspecting the graphs for Zoo, Parkinson, Wine, Climate, HAR, and ISOLET datasets reveals that bWWPA produced excellent cost values during the iterative phase. Performance on the cost function was inconsistent for the bGWO technique across all six datasets, whereas bPSO, bWOA, and bGA performed consistently well. For each of the six data sets, the bWWPA curve was lower than those obtained using the other approaches. Across all six datasets, the bWWPA showed the highest performance. The bWWPA outperformed the other feature selection techniques on high-dimensional datasets like HAR and ISOLET.

A. STATISTICAL ANALYSIS OF TWO CASES STUDIES

In this section, we investigate the performance of the proposed feature selection algorithm in terms of two datasets with a large set of features. These datasets are Har with

TABLE 17. Wilcoxon signed rank test of the results achieved by applying bWWPA feature selection to the ISOLET dataset.

	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Theoretical median	0	0	0	0	0	0	0	0	0
Actual median	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112
Number of values	20	20	20	20	20	20	20	20	20
Sum of signed ranks (W)	210	210	210	210	210	210	210	210	210
Sum of positive ranks	210	210	210	210	210	210	210	210	210
Sum of negative ranks	0	0	0	0	0	0	0	0	0
P value (two tailed)	0.0012468	0.0000659	0.0000659	0.0000669	0.0000669	0.0000669	0.0000758	0.0000669	0.0000669
Exact or estimate?	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact	Exact
Significant (alpha=0.05)?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Discrepancy	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112

TABLE 18. Statistical analysis of the results achieved by applying feature selection using bWWPA to the ISOLET dataset.

	bWWPA	bDTO	bSC	bPSO	bWAO	bGWO	bMVO	bSBO	bFA	bGA
Number of values	20	20	20	20	20	20	20	20	20	20
Minimum	1.102	1.108	1.012	1.135	1.122	1.018	1.291	1.219	1.188	1.012
25% Percentile	1.102	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112
Median	1.102	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112
75% Percentile	1.102	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112
Maximum	1.132	1.408	1.412	1.635	1.622	1.477	1.591	1.519	1.788	1.512
Range	0.03	0.3	0.4	0.5	0.5	0.459	0.3	0.3	0.6	0.5
10% Percentile	1.102	1.308	1.112	1.335	1.244	1.177	1.391	1.329	1.388	1.112
90% Percentile	1.122	1.398	1.202	1.525	1.422	1.377	1.481	1.419	1.568	1.402
Actual confidence level	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%	95.86%
Lower confidence limit	1.102	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112
Upper confidence limit	1.102	1.308	1.112	1.335	1.422	1.177	1.391	1.419	1.388	1.112
Mean	1.105	1.308	1.127	1.355	1.407	1.204	1.401	1.409	1.408	1.152
Std. Deviation	0.008751	0.0562	0.07452	0.09515	0.09316	0.09751	0.05525	0.05525	0.1105	0.1188
Std. Error of Mean	0.001957	0.01257	0.01666	0.02128	0.02083	0.0218	0.01235	0.01235	0.02471	0.02656
Lower 95% CI of mean	1.101	1.282	1.092	1.31	1.364	1.158	1.375	1.383	1.356	1.097
Upper 95% CI of mean	1.109	1.335	1.162	1.399	1.451	1.249	1.427	1.435	1.459	1.208
Coefficient of variation	0.7919%	4.296%	6.611%	7.022%	6.619%	8.101%	3.945%	3.921%	7.850%	10.31%
Geometric mean	1.105	1.307	1.125	1.352	1.404	1.2	1.4	1.408	1.404	1.147
Geometric SD factor	1.008	1.046	1.063	1.071	1.072	1.081	1.039	1.042	1.077	1.099
Lower 95% CI of geo. mean	1.101	1.28	1.093	1.309	1.359	1.157	1.375	1.381	1.356	1.097
Upper 95% CI of geo. mean	1.109	1.335	1.158	1.396	1.451	1.245	1.425	1.435	1.453	1.199
Harmonic mean	1.105	1.306	1.123	1.349	1.401	1.197	1.399	1.407	1.4	1.142
Lower 95% CI of harm. mean	1.101	1.277	1.094	1.308	1.354	1.157	1.375	1.379	1.356	1.098
Upper 95% CI of harm. mean	1.109	1.335	1.154	1.393	1.451	1.24	1.423	1.436	1.448	1.19
Quadratic mean	1.105	1.309	1.13	1.358	1.41	1.207	1.402	1.41	1.412	1.158
Lower 95% CI of quad. mean	1.101	1.284	1.091	1.311	1.368	1.158	1.375	1.385	1.355	1.095
Upper 95% CI of quad. mean	1.109	1.334	1.167	1.404	1.452	1.255	1.428	1.435	1.466	1.218
Skewness	2.344	-1.977	3.136	1.19	-1.4	1.509	2.164	-2.164	2.164	2.254
Kurtosis	4.393	9.5	12.45	4.908	5.928	3.291	8.208	8.208	8.208	4.611
Sum	22.1	26.16	22.54	27.1	28.15	24.07	28.01	28.18	28.15	23.04

smartphones and the ISOLET. The number of features of these datasets is 561 for HAR and 617 for the ISOLET dataset. In addition, the number of instances in the HAR dataset is 10299, and 7797 for the ISOLET dataset. The large size of these datasets represents a challenge in performing well for the proposed approach compared to the other approach. Thus, we adopted these datasets to emphasize the capability of the proposed approach to handle high-dimensional datasets with a large number of samples. The statistical analysis is performed in terms of a set of experiments including ANOVA, Wilcoxon, heatmap of ANOVA test, average error, and histogram of average error in addition to a set of plots representing the outputs of the ANOVA test.

The average error, heatmap of the ANOVA test, and the histogram of the average error measured from the results achieved by the proposed algorithm when applied to the HAR dataset are shown in Figure 15, Figure 16, and Figure 17, respectively. Similarly, the corresponding plots for the ISOLET dataset are shown in Figure 18, Figure 19, and Figure 20, respectively. The results depicted in these figures emphasize the superiority of the proposed algorithm when compared to the competitor feature selection methods.

In addition, the analysis plots shown in Figure 21 represented the ANOVA test results for HAR and ISOLET datasets and represented in terms of the Residual plot, Homoscedasticity plot, and QQ plot. These plots show a promising

performance of the proposed methodology in feature selection when applied to a large dataset.

On the other hand, the statistical analysis of the results recorded based on the HAR dataset is shown in Table 13. This table presents the percentiles, means, variances, and other measuring criteria to prove the effectiveness of the proposed methodology. The presented results using the proposed bWWPA are better than the other methods, which confirms the expected findings. In addition, the ANOVA and Wilcoxon results are presented in Table 14 and 15, respectively. The p-values in these tables are less than 0.05, which refers to the statistical difference between the proposed method and the other feature selection methods. These tests are performed using 20 random samples from the HAR dataset.

Similarly, the statistical analysis of the results recorded using the ISOLET dataset is shown in Table 18. In addition, the ANOVA and Wilcoxon results are presented in Table 14 and 15, respectively. The results in these tables emphasize the statistical difference of the proposed approach when tested on a dataset with a large number of features and samples.

VI. CONCLUSION

This work presented a novel feature selection algorithm that takes cues from the waterwheel plant's method of prey selection. By striking a good balance between exploration and exploitation, the proposed method (bWWPA) is utilized in conjunction with the KNN classifier to determine the best possible combination of features to apply to each unique situation. A Sigmoid function was used to transform the continuous values into binary ones to apply the proposed approach to the feature selection problem. Experiments were conducted with 30 datasets from the UCI machine learning repository to examine the stability and robustness of the proposed bWWPA method. The results were compared to bDTO, bSC, bPSO, bWOA, bGWO, bMVO, bSBO, bFA, and bGA optimization techniques. The results showed the superiority of the proposed bWWPA algorithm. In addition, extensive statistical analysis is performed in terms of two datasets with a large number of features to confirm the proposed approach's effectiveness and its superiority compared to other feature selection methods. This analysis includes ANOVA and the Wilcoxon rank test, and the results are visualized to clearly show the proposed algorithm's promising performance. The proposed approach will be evaluated in further work on continuous problems, restricted engineering challenges, and additional binary problems, such as the EEG problem and binary problems with a larger number of features.

REFERENCES

[1] A. Hatamlou, "Black hole: A new heuristic optimization approach for data clustering," *Inf. Sci.*, vol. 222, pp. 175–184, Feb. 2013.

[2] E.-S.-M. El-Kenawy, S. Mirjalili, A. A. Abdelhamid, A. Ibrahim, N. Khodadadi, and M. M. Eid, "Meta-heuristic optimization and keystroke dynamics for authentication of smartphone users," *Mathematics*, vol. 10, no. 16, p. 2912, Aug. 2022.

[3] E.-S.-M. El-kenawy, F. Albalawi, S. A. Ward, S. S. M. Ghoneim, M. M. Eid, A. A. Abdelhamid, N. Bailek, and A. Ibrahim, "Feature selection and classification of transformer faults based on novel meta-heuristic algorithm," *Mathematics*, vol. 10, no. 17, p. 3144, Sep. 2022.

[4] H. A. Alsayadi, N. Khodadadi, and S. Kumar, "Improving the regression of communities and crime using ensemble of machine learning models," *J. Artif. Intell. Metaheuristics*, vol. 1, no. 1, pp. 27–34, 2022.

[5] O. A. Akinola, J. O. Agushaka, and A. E. Ezugwu, "Binary dwarf mongoose optimizer for solving high-dimensional feature selection problems," *PLoS One*, vol. 17, no. 10, Oct. 2022, Art. no. e0274850.

[6] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*. Cham, Switzerland: Springer, Dec. 2012.

[7] A. A. Abdelhamid et al., "Innovative feature selection method based on hybrid sine cosine and dipper throated optimization algorithms," *IEEE Access*, vol. 11, pp. 79750–79776, 2023, doi: 10.1109/ACCESS.2023.3298955.

[8] Y. Li, T. Li, and H. Liu, "Recent advances in feature selection and its applications," *Knowl. Inf. Syst.*, vol. 53, no. 3, pp. 551–577, Dec. 2017.

[9] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003.

[10] I. J. Mohammed, B. T. Al-Nuaimi, and T. I. Baker, "Weather forecasting over Iraq using machine learning," *J. Artif. Intell. Metaheuristics*, vol. 2, no. 2, pp. 39–45, 2022.

[11] J. Žerovnik, "Heuristics for NP-hard optimization problems—Simpler is better!" *Logistics Sustain. Transp.*, vol. 6, no. 1, pp. 1–10, Nov. 2015.

[12] A. I. Hammouri, M. Mafarja, M. A. Al-Betar, M. A. Awadallah, and I. Abu-Doush, "An improved dragonfly algorithm for feature selection," *Knowl.-Based Syst.*, vol. 203, Sep. 2020, Art. no. 106131.

[13] S. Ahmed, K. H. Sheikh, S. Mirjalili, and R. Sarkar, "Binary simulated normal distribution optimizer for feature selection: Theory and application in COVID-19 datasets," *Expert Syst. Appl.*, vol. 200, Aug. 2022, Art. no. 116834.

[14] H. Banka and S. Dara, "A Hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification and validation," *Pattern Recognit. Lett.*, vol. 52, pp. 94–100, Jan. 2015.

[15] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary ant lion approaches for feature selection," *Neurocomputing*, vol. 213, pp. 54–65, Nov. 2016.

[16] E. Emary and H. M. Zawbaa, "Feature selection via Lèvy antlion optimization," *Pattern Anal. Appl.*, vol. 22, no. 3, pp. 857–876, Aug. 2019.

[17] B. Ji, X. Lu, G. Sun, W. Zhang, J. Li, and Y. Xiao, "Bio-inspired feature selection: An improved binary particle swarm optimization approach," *IEEE Access*, vol. 8, pp. 85989–86002, 2020.

[18] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, Aug. 2016.

[19] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. Comput. Cybern. Simul.*, vol. 5, Oct. 1997, pp. 4104–4108.

[20] A. Unler and A. Murat, "A discrete particle swarm optimization method for feature selection in binary classification problems," *Eur. J. Oper. Res.*, vol. 206, no. 3, pp. 528–539, Nov. 2010.

[21] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, "Improved binary particle swarm optimization using catfish effect for feature selection," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 12699–12707, Sep. 2011.

[22] M. Mafarja, R. Jarrar, S. Ahmad, and A. A. Abusnaina, "Feature selection using binary particle swarm optimization with time varying inertia weight strategies," in *Proc. 2nd Int. Conf. Future Netw. Distrib. Syst.* New York, NY, USA: Association for Computing Machinery, Jun. 2018, pp. 1–9.

[23] C.-L. Huang and C.-J. Wang, "A GA-based feature selection and parameters optimizationfor support vector machines," *Expert Syst. Appl.*, vol. 31, no. 2, pp. 231–240, Aug. 2006.

[24] S. Nemati, M. E. Basiri, N. Ghasem-Aghaee, and M. H. Aghdam, "A novel ACO-GA hybrid algorithm for feature selection in protein function prediction," *Expert Syst. Appl.*, vol. 36, no. 10, pp. 12086–12094, Dec. 2009.

[25] S. Jiang, K.-S. Chin, L. Wang, G. Qu, and K. L. Tsui, "Modified genetic algorithm-based feature selection combined with pre-trained deep neural network for demand forecasting in outpatient department," *Expert Syst. Appl.*, vol. 82, pp. 216–230, Oct. 2017.

[26] R. Y. M. Nakamura, L. A. M. Pereira, K. A. Costa, D. Rodrigues, J. P. Papa, and X.-S. Yang, "BBA: A binary bat algorithm for feature selection," in *Proc. 25th SIBGRAPI Conf. Graph., Patterns Images*, Aug. 2012, pp. 291–297.

[27] E. Hancer, B. Xue, D. Karaboga, and M. Zhang, "A binary ABC algorithm based on advanced similarity scheme for feature selection," *Appl. Soft Comput.*, vol. 36, pp. 334–348, Nov. 2015.
[28] Y. Zhang, X.-F. Song, and D.-W. Gong, "A return-cost-based binary firefly algorithm for feature selection," *Inf. Sci.*, vols. 418–419, pp. 561–574, Dec. 2017.
[29] M. Mafarja, I. Aljarah, A. A. Heidari, H. Faris, P. Fournier-Viger, X. Li, and S. Mirjalili, "Binary dragonfly optimization for feature selection using time-varying transfer functions," *Knowl.-Based Syst.*, vol. 161, pp. 185–204, Dec. 2018.
[30] H. Faris, M. M. Mafarja, A. A. Heidari, I. Aljarah, A. M. Al-Zoubi, S. Mirjalili, and H. Fujita, "An efficient binary Salp Swarm Algorithm with crossover scheme for feature selection problems," *Knowl.-Based Syst.*, vol. 154, pp. 43–67, Aug. 2018.
[31] M. Mafarja, I. Aljarah, H. Faris, A. I. Hammouri, A. M. Al-Zoubi, and S. Mirjalili, "Binary grasshopper optimisation algorithm approaches for feature selection problems," *Expert Syst. Appl.*, vol. 117, pp. 267–286, Mar. 2019.
[32] M. Mafarja and S. Mirjalili, "Whale optimization approaches for wrapper feature selection," *Appl. Soft Comput.*, vol. 62, pp. 441–453, Jan. 2018.
[33] V. Kumar, D. Kumar, M. Kaur, D. Singh, S. A. Idris, and H. Alshazly, "A novel binary seagull optimizer and its application to feature selection problem," *IEEE Access*, vol. 9, pp. 103481–103496, 2021.
[34] V. R. E. Christo, H. K. Nehemiah, B. Minu, and A. Kannan, "Correlation-based ensemble feature selection using bioinspired algorithms and classification using backpropagation neural network," *Comput. Math. Methods Med.*, vol. 2019, Sep. 2019, Art. no. e7398307.
[35] S. Murugesan, R. S. Bhuvaneswaran, H. K. Nehemiah, S. K. Sankari, and Y. N. Jane, "Feature selection and classification of clinical datasets using bioinspired algorithms and super learner," *Comput. Math. Methods Med.*, vol. 2021, May 2021, Art. no. e6662420.
[36] Bilal, M. Pant, H. Zaheer, L. Garcia-Hernandez, and A. Abraham, "Differential evolution: A review of more than two decades of research," *Eng. Appl. Artif. Intell.*, vol. 90, Apr. 2020, Art. no. 103479.
[37] Bilal, M. Pant, M. Stanko, and L. Sales, "Differential evolution for early-phase offshore oilfield design considering uncertainties in initial oil-in-place and well productivity," *Upstream Oil Gas Technol.*, vol. 7, Sep. 2021, Art. no. 100055.
[38] K. M. Ong, P. Ong, and C. K. Sia, "A carnivorous plant algorithm for solving global optimization problems," *Appl. Soft Comput.*, vol. 98, Jan. 2021, Art. no. 106833.
[39] K. Balasubramanian and N. P. Ananthamoorthy, "Correlation-based feature selection using bio-inspired algorithms and optimized KELM classifier for glaucoma diagnosis," *Appl. Soft Comput.*, vol. 128, Oct. 2022, Art. no. 109432.
[40] P. Agrawal, H. F. Abutarboush, T. Ganesh, and A. W. Mohamed, "Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019)," *IEEE Access*, vol. 9, pp. 26766–26791, 2021.
[41] Z. Chen, K. Zhu, and L. Ying, "Detecting multiple information sources in networks under the SIR model," *IEEE Trans. Netw. Sci. Eng.*, vol. 3, no. 1, pp. 17–31, Jan. 2016.
[42] W. Zang, P. Zhang, C. Zhou, and L. Guo, "Locating multiple sources in social networks under the SIR model: A divide-and-conquer approach," *J. Comput. Sci.*, vol. 10, pp. 278–287, Sep. 2015.
[43] M. A. Al-Betar, Z. A. A. Alyasseri, M. A. Awadallah, and I. A. Doush, "Coronavirus herd immunity optimizer (CHIO)," *Neural Comput. Appl.*, vol. 33, no. 10, pp. 5011–5042, May 2021.
[44] W. M. Shaban, A. H. Rabie, A. I. Saleh, and M. A. Abo-Elsoud, "A new COVID-19 patients detection strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier," *Knowl.-Based Syst.*, vol. 205, Oct. 2020, Art. no. 106270.
[45] M. Alweshah, "Coronavirus herd immunity optimizer to solve classification problems," *Soft Comput.*, vol. 27, no. 6, pp. 3509–3529, Mar. 2022.
[46] A. S. Westermeier, R. Sachse, S. Poppinga, P. Vögele, L. Adamec, T. Speck, and M. Bischoff, "How the carnivorous waterwheel plant *(Aldrovanda vesiculosa)* snaps," *Proc. Roy. Soc. B, Biol. Sci.*, vol. 285, no. 1878, May 2018, Art. no. 20180012.
[47] A. A. Abdelhamid, S. K. Towfek, N. Khodadadi, A. A. Alhussan, D. S. Khafaga, M. M. Eid, and A. Ibrahim, "Waterwheel plant algorithm: A novel metaheuristic optimization method," *Processes*, vol. 11, no. 5, p. 1502, May 2023.
[48] C. L. Blake and C. J. Merz. (1998). *UCI Repository of Machine Learning Databases*. Accessed: Mar. 3, 2023. [Online]. Available: https://archive.ics.uci.edu/ml/index.php
[49] A. E. Takieldeen, E.-S. M. El-kenawy, M. Hadwan, and R. M. Zaki, "Dipper throated optimization algorithm for unconstrained function and feature selection," *Comput., Mater. Continua*, vol. 72, no. 1, pp. 1465–1481, 2022.
[50] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowl.-Based Syst.*, vol. 96, pp. 120–133, Mar. 2016.
[51] J. L. Awange, B. Paláncz, R. H. Lewis, and L. Völgyesi, "Particle swarm optimization," in *Mathematical Geosciences: Hybrid Symbolic-Numeric Methods*. Cham, Switzerland: Springer, 2018, pp. 167–184.
[52] D. Martínez-Rodríguez, R. Novella, G. Bracho, J. Gomez-Soriano, C. Fernandes, T. Lucchini, A. D. Torre, R.-J. Villanueva, and J. I. Hidalgo, "A particle swarm optimization algorithm with novelty search for combustion systems with ultra-low emissions and minimum fuel consumption," *Appl. Soft Comput.*, vol. 143, Aug. 2023, Art. no. 110401.
[53] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016.
[54] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.
[55] Y. Liu, Y. Jiang, X. Zhang, Y. Pan, and J. Wang, "An improved grey wolf optimizer algorithm for identification and location of gas emission," *J. Loss Prevention Process Industries*, vol. 82, Apr. 2023, Art. no. 105003.
[56] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: A nature-inspired algorithm for global optimization," *Neural Comput. Appl.*, vol. 27, no. 2, pp. 495–513, Feb. 2016.
[57] S. H. S. Moosavi and V. K. Bardsiri, "Satin bowerbird optimizer: A new optimization algorithm to optimize ANFIS for software development effort estimation," *Eng. Appl. Artif. Intell.*, vol. 60, pp. 1–15, Apr. 2017.
[58] S. D. Immanuel and U. K. Chakraborty, "Genetic algorithm: An approach on optimization," in *Proc. Int. Conf. Commun. Electron. Syst. (ICCES)*, Jul. 2019, pp. 701–708.
[59] D. S. Khafaga, A. A. Alhussan, E. M. El-Kenawy, A. Ibrahim, M. M. Eid, and A. A. Abdelhamid, "Solving optimization problems of metamaterial and double T-shape antennas using advanced meta-heuristics algorithms," *IEEE Access*, vol. 10, pp. 74449–74471, 2022.
[60] A. A. Alhussan, D. S. Khafaga, E. M. El-Kenawy, A. Ibrahim, M. M. Eid, and A. A. Abdelhamid, "Pothole and plain road classification using adaptive mutation dipper throated optimization and transfer learning for self driving cars," *IEEE Access*, vol. 10, pp. 84188–84211, 2022.
[61] A. A. Abdelhamid, E.-S.-M. El-Kenawy, N. Khodadadi, S. Mirjalili, D. S. Khafaga, A. H. Alharbi, A. Ibrahim, M. M. Eid, and M. Saber, "Classification of monkeypox images based on transfer learning and the Al-Biruni Earth radius optimization algorithm," *Mathematics*, vol. 10, no. 19, p. 3614, Oct. 2022.
[62] M. M. Eid, E.-S.-M. El-Kenawy, N. Khodadadi, S. Mirjalili, E. Khodadadi, M. Abotaleb, A. H. Alharbi, A. A. Abdelhamid, A. Ibrahim, G. M. Amer, A. Kadi, and D. S. Khafaga, "Meta-heuristic optimization of LSTM-based deep network for boosting the prediction of monkeypox cases," *Mathematics*, vol. 10, no. 20, p. 3845, Oct. 2022.

AMEL ALI ALHUSSAN received the B.Sc., M.Sc., and Ph.D. degrees in computer and information sciences from King Saud University, Saudi Arabia. Her M.Sc. thesis was in software engineering and the Ph.D. thesis was in artificial intelligence. She is currently an Assistant Professor with the Computer Sciences Department, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University (PNU), Saudi Arabia. She was with the college in various administrative and academic positions. Her research interests include machine learning, networking, and software engineering.

ABDELAZIZ A. ABDELHAMID received the M.Sc. degree in computer science from the Faculty of Computer and Information Sciences, Ain Shams University, and the Ph.D. degree in computer engineering from the Faculty of Engineering, Auckland University, New Zealand. He is an Assistant Professor with the Department of Computer Science, Faculty of Computer and Information Sciences, Ain Shams University. He is also an Assistant Professor with the Computer Science Department, College of Computing and Information Technology, Shaqra University. His research interests include speech and image processing and machine learning-based intelligent systems.

EL-SAYED M. EL-KENAWY (Senior Member, IEEE) is an Assistant Professor with the Delta Higher Institute for Engineering and Technology (DHIET), Mansoura, Egypt. He has inspired and motivated the students by providing a thorough understanding of a variety of computer concepts. He has pioneered and launched independent research programs. He has interested in computer science and the machine learning field. He is adept at explaining sometimes complex concepts in an easy-to-understand manner.

ABDELHAMEED IBRAHIM (Member, IEEE) received the bachelor's and master's degrees in engineering from the Computer Engineering and Systems Department, in 2001 and 2005, respectively, and the Ph.D. degree in engineering from Chiba University, Japan, in 2011. Since 2001, he has been with the Faculty of Engineering, Mansoura University, Egypt, where he is currently an Associate Professor of computer engineering. He has published over 110 articles and has over 3300 citations. His H-index is 31. His research pursuits encompass machine learning, optimization, swarm intelligence, meta-heuristic techniques, and pattern recognition. He actively contributes as a Reviewer for various esteemed journals, including *Engineering Applications of Artificial Intelligence*, IEEE Access, IEEE Journal of Biomedical and Health Informatics, *Expert Systems with Applications*, *Biomedical Signal Processing and Control*, *Sensors*, *PLOS ONE*, *Alexandria Engineering Journal*, *Scientific Reports*, and *Renewable and Sustainable Energy Reviews*.

MARWA METWALLY EID received the Ph.D. degree in electronics and communications engineering from the Faculty of Engineering, Mansoura University, Egypt, in 2015. She has been an Assistant Professor with the Delta Higher Institute for Engineering and Technology, since 2011. Her current research interests are in image processing, encryption, wireless communication systems, and field programmable gate array (FPGA) applications.

DOAA SAMI KHAFAGA received the B.Sc. degree (Hons.) in computer and information sciences in the field of computer science and the M.Sc. and Ph.D. degrees in computer science from the College of Faculty of Computers and Artificial Intelligence, Helwan University, Egypt, in 2003, 2008, and 2013, respectively. She has 18 years of academic experience. She was with the Computer Science Department, College of Information Technology and Artificial Intelligence, Misr University for Science and Technology, Egypt; the Computer Science Department, Institute of Public Administration, Saudi Arabia; and the Computer Science Department, Faculty of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University. Her main research interests include data science, artificial intelligence, machine learning, data mining, and software engineering. She has FHEA, a fellow recognition from the U.K. Higher Education Academy. Currently, she is a reviewer of some journals.

AYMAN EM AHMED received the B.Sc., M.Sc., and Ph.D. degrees in computer and information sciences from Mansoura University, Egypt. His M.Sc. thesis was in software engineering and the Ph.D. thesis was in artificial intelligence. He is currently an Assistant Professor with the Computer Engineering Department, Faculty of Engineering, King Salman International University, Egypt. His research interests include machine learning, networking, and software engineering.