IGWO-RF: A Hybrid Improved Gray Wolf Optimizer and Random Forest Wrapper for High-Dimensional Feature Selection

Zhichao Xu, Zaiyi Pu
School of Information and Business Management, Dalian Neusoft University of Information, Dalian 116023, Liaoning, China
Education and Information Technology Center, China West Normal University, Nanchong 637009, Sichuan China
E-mail: zzaiypu@126.com
*Corresponding author

Keywords: machine learning, feature selection, wrapper model, evolutionary techniques, gray wolf optimization algorithm, random forest

Received: August 7, 2025

Not all data features are crucial for uncovering hidden knowledge within various datasets, making feature selection a significant area of interest. This work proposes IGWO-RF, a new meta-heuristic algorithm that combines an improved gray wolf optimization (GWO) algorithm with random forest (RF) for feature selection. The improved GWO introduces a nonlinear convergence parameter for better exploration-exploitation balance and a GA-inspired crossover operation using alpha and beta wolves to accelerate convergence. The RF algorithm evaluates the fitness of feature subsets in each iteration. The proposed technique was evaluated on 10 benchmark UCI datasets (including Wine, Sonar, Vehicle, and Parkinson's) based on the average number of selected features, average classification accuracy, and best fitness. Comparative analysis with four popular wrapper-based methods (GWO-RF, ACO, PSO, ABC) demonstrated the superiority of IGWO-RF. Specifically, IGWO-RF achieved the highest average classification accuracy of 91.23% using the SVM classifier, outperforming GWO-RF (89.91%), PSO (89.12%), and ABC (87.94%). Furthermore, IGWO-RF obtained the most compact feature subsets, selecting on average only 31.22% of the original features in the Glass dataset and 20.89% in the Vehicle dataset—a significant reduction compared to other methods. The algorithm also showed faster convergence and reduced execution time. Therefore, IGWO-RF proves to be an effective approach for enhancing pattern classification performance through efficient feature selection.

Povzetek: Članek predstavlja novo metodo IGWO-RF za izbiro značilk, ki združuje izboljšan algoritem sivega volka in naključni gozd ter omogoča natančnejšo klasifikacijo z manjšim številom izbranih značilk in hitrejšo konvergenco v primerjavi z obstoječimi metodami.

1 Introduction

Databases with multiple dimensions present numerous computational challenges, despite the opportunities they offer. An inherent issue with high-dimensional data is the presence of redundant or irrelevant features, which can obscure the valuable insights hidden within the dataset w. As a result, dimensionality reduction remains a crucial concern across various fields, with feature selection emerging as a popular strategy to streamline data. Over the years, a variety of solutions and algorithms have been introduced to tackle the feature selection problem, with some methods dating back as far as thirty to forty years [1–3]. However, the computational burden associated with certain algorithms has posed challenges, although modern advancements in computing power and storage capacity have somewhat alleviated this issue. Nevertheless, the escalating prominence of big data applications emphasizes the ongoing need for efficient algorithms to address feature selection tasks swiftly. Traditional approaches often struggle to cope with the computational demands of feature selection and may fall short in identifying the optimal subset of features [4]. In contrast, evolutionary algorithms (EA) offer a viable solution to the feature selection conundrum, capable of identifying the most relevant set of features while maintaining a reasonable computational overhead [5,6].

Evolutionary feature selection algorithms offer several advantages over traditional methods. Firstly, they have the ability to explore a broader solution space efficiently, allowing for a more comprehensive search for the optimal subset of features [7]. This capability can help in identifying complex relationships and patterns within the data that may not be evident through manual or deterministic approaches. Secondly, EA can adapt and evolve over successive generations, gradually improving the quality of the feature subset through iterative refinement. This adaptability enables them to dynamically adjust to changing data characteristics and requirements, enhancing the robustness and flexibility of the feature selection process [8]. Moreover, evolutionary feature selection algorithms are capable of handling high-dimensional data effectively by automatically selecting the most relevant features while disregarding redundant or irrelevant ones. This targeted approach can lead to improved model performance, reduced overfitting, and enhanced interpretability of the results. Additionally, EA offers a scalable and parallelizable framework, making it suitable for processing large datasets efficiently [9]. By leveraging the power of parallel computing, these algorithms can expedite the feature selection process, enabling quicker decision-making in data-driven applications [10].

In this work, a new meta-heuristic algorithm IGWO-RF which is a combination of improved GWO algorithm and RF is proposed for feature selection problems, which has important advantages and innovations: (I) The proposed IGWO-RF is capable of exploring a large search space efficiently, which allows it to find good solutions across the entire space without getting stuck in local optima; (II) The proposed IGWO-RF is flexible and can be applied to a wide range of issues without needing problem-specific modifications. It can be adapted and customized for different types of feature selection tasks; (III) The proposed IGWO-RF can handle feature selection as either a standalone optimization task or as part of a broader optimization problem. It evaluates subsets of features based on accuracy, fitness, or other performance measures to select the most relevant features; (IV) The proposed IGWO-RF can provide near-optimal solutions within a reasonable amount of computation time. It is suitable for large datasets and high-dimensional feature spaces, where exhaustive search methods become impractical. In this article, a literature review of related works is presented in part 2. Then, in part 3, the basic concepts of the proposed method and its detailed framework are presented. In part 4, the numerical results of applying the proposed method on different datasets are presented. Finally, conclusions and suggestions for future work are presented in part 5.

2 Literature review

Feature selection has emerged as a key component in numerous real-world applications like medical diagnosis, face recognition, text processing, image retrieval, as well as bioinformatics [11–14]. Since the 1970s, it has become a significant focus for research and advancement in statistical pattern detection, data mining, and ML. Various methodologies for feature selection have been developed and categorized into 4 groups - filters, wrappers, hybrids, and embedded - based on their respective evaluation processes [15,16]. Numerous endeavors have been made to assess these feature selection methods thoroughly and their effectiveness in different application domains. In the context of feature selection methods, the filter approach involves procedures that carry out feature selection independently from any specific learning algorithm, essentially functioning as standalone preprocessors. This method relies on statistical analysis to analyze the feature set and address feature selection challenges without the direct involvement of a learning model. On the other side, the wrapper approach utilizes a designated learning algorithm to appraise the chosen subsets' quality. Wrappers can produce reliable results, but they need a lot of resources and may have trouble handling many features. A hybrid approach merges elements of both the filter and wrapper techniques to leverage the strengths of both methods. In contrast, embedded techniques integrate feature selection within the learning process itself, aligning closely with specific learning models for enhanced compatibility and performance [17].

2.1 Evolutionary and swarm intelligence algorithms for feature selection

The utilization of EA has increased during the last few years, including Particle Swarm Optimization (PSO), genetic algorithm, Artificial Bee Colony (ABC), and Ant Colony Optimization (ACO) for feature selection tasks. Genetic algorithms, in particular, have shown effectiveness in tackling high-dimensional datasets within the realm of feature selection. However, a limitation of this approach is its oversight of inter-feature relationships during the selection process, leading to an elevated risk of including redundant features in the final subset chosen [18]. To manage the computational complexity associated with high-dimensional datasets, numerous feature selection methods employ meta-heuristic techniques. These algorithms handle optimization problems and iteratively search for the most advantageous solution by utilizing fundamental mechanisms and procedures [19–21]. Typically, these algorithms begin with a population of random solutions and work toward improving the quality of those solutions with each iteration. In most meta-heuristic algorithms, a set of initial solutions is first produced at random and then their quality is assessed within the resulting population using a fitness function. The procedure proceeds to create a new population if any of the predetermined termination conditions are not met. Until one of the termination requirements is met, this iterative procedure is continued [22].

EA and Swarm Intelligence (SI) are the 2 primary categories into which meta-heuristic techniques fall [23]. EA mimics biological evolution processes like reproduction, mutation, recombination, and selection to generate solutions for optimization problems. In EA, candidate solutions are akin to individuals within a population, and their quality is assessed by a fitness function. The starting population steadily changes with each algorithm iteration in the direction of global optimization [22]. Conversely, SI algorithms usually incorporate a collection of low-level artificial agents interacting locally in a setting. Inspired by natural systems, each agent performs basic tasks, and through local and sometimes random interactions, collective intelligent behavior emerges, surpassing the capabilities of individual agents [24]. In [25], a genetic algorithm is utilized in conjunction with a k-Nearest-Neighbors technique to efficiently select features, lessen the dataset's size, as well as improve the classification accuracy to diagnose patients' illness stages. Furthermore, [26] introduces a novel 2-layer approach for feature selection, to construct an appropriate predictor subset by combining an embedding method and a wrapper. In order to find the best predictor subset and reduce the number of predictors and prediction errors, the first layer uses a Genetic Algorithm as a wrapper. The remaining redundant or irrelevant predictors are then removed with the addition of a second layer, improving prediction accuracy. Ratnoo and Rathee [27] provide a way for multi-objective feature selection based on a genetic algorithm, integrating non-dominated sorting concepts in order to produce a group of non-dominated solutions. Additionally, in [28], a method for selecting ensemble features using evolutionary algorithms and t-tests is presented. Using this strategy, data preprocessing based on t-tests is followed by a Nested Genetic Algorithm that combines data from two different datasets to produce the ideal feature subset. The Nested Genetic Algorithm comprises 2 separate instances running on different datasets. For the investigation of Laser-induced breakdown spectroscopy, [29]offers a novel hybrid feature selection method according to PSO. Using the advantages of both coating and filtering techniques at the same time is the goal of this approach. To improve classification accuracy and reduce computing complexity, a feature selection strategy that combines PSO with multiple classifiers is suggested in[30].

2.2 Recent advancements in hybrid and improved meta-heuristics

Recent research has focused on enhancing the performance of base algorithms through hybridization and modification. In the domain of improved GWO variants, [31] proposed an improved GWO for feature selection in electronic nose data, incorporating novel binary transform approaches and an adaptive restart mechanism to enhance search capability. Experimental results demonstrated its effectiveness in selecting optimal feature subsets. A binary hybrid of GWO and PSO was developed for big data feature selection, incorporating a tent chaotic map to avoid local optima. The method significantly outperformed standard GWO and PSO [32]. To address feature selection in high-dimensional data, [33] developed three progressively enhanced variants of the binary GWO. The final variant, which integrated a novel mutation strategy and simulated annealing, demonstrated superior performance over six other wrapper methods across 32 UCI datasets, highlighting the significant impact of algorithmic improvements on feature selection efficacy. For unsupervised text feature selection, a hybrid GWO-GOA was proposed to select optimal features, which were then clustered using Fuzzy C-Means. The method achieved 87.6% efficiency on eight text datasets, outperforming standalone GWO and GOA and demonstrating the viability of GWO hybrids in text mining [34].

A comparative analysis, summarized in Table 1, reveals several persistent challenges and trends in wrapper-based feature selection. While PSO and GA are well-established, they often suffer from parameter sensitivity and premature convergence [19, 30]. The standard GWO algorithm, though simpler, is hampered by its linear convergence parameter, which fails to provide an adaptive balance between exploration and exploitation [31, 32]. Recent attempts to improve GWO through external hybridization [31] or internal modifications have shown promise but often introduce new complexities or rely on multiple auxiliary mechanisms.

Table 1: Comparative summary of recent wrapper-based feature selection techniques.

Technique	Key Characteristics	Performance Highlights	Reported Limitations/ Challenges
Standard GWO	- SI-based, continuous optimizer - Linear convergence parameter - Social hierarchy (α, β, δ)	- Effective global search - Few parameters to tune	- Premature convergence on complex problems - Linear control parameter limits exploration-exploitation balance - Lacks mechanisms for fine-tuning solutions
PSO-based	- SI-based, continuous optimizer - Self-adaptive parameters - Used with multiple classifiers	- Improved search capability in high-dimensions - Good convergence speed	- Sensitive to parameter tuning - Can get trapped in local optima without specific mechanisms (e.g., chaos)
ACO-based	- SI-based, discrete optimizer - Constructs solutions via pheromone trails	- Naturally suited for discrete problems like FS - Robust performance	- Slow convergence speed - Computational intensity for large-scale problems
GA-based	- EA-based, discrete optimizer - Uses crossover and mutation	- Powerful global exploration - Effective on microarray data	- Can overlook inter-feature relationships - Risk of including redundant features - Computationally expensive for fitness evaluation
Hybrid GWO-GOA	- Hybrid SI (GWO + Grasshopper OA)	- Superior to standalone GWO and GOA	- External hybridization can be complex

	- Used for text feature selection	- 87.6% efficiency in text clustering	- Performance dependent on effective integration of both algorithms
Binary GWO Variants	- EA/SI hybrid, binary optimizer - Integrates transfer functions & mutation	- Outperformed 6 other wrappers on 32 UCI datasets - Showed impact of internal enhancements	- Performance heavily dependent on choice of transfer function - Requires additional mechanisms (mutation, SA) to avoid local optima
Proposed IGWO-RF	- SI-based with GA-inspired crossover (Internal Hybrid) - Nonlinear convergence parameter - RF classifier for robust fitness evaluation	- Aims for superior exploration-exploitation balance - Aims for faster convergence and higher accuracy - Aims for compact, discriminative feature subsets	- Addresses GWO's linear convergence flaw - Introduces intelligent reproduction beyond social hierarchy - Mitigates classifier overfitting in wrapper model

A critical analysis of the recent literature reveals several interconnected gaps. Firstly, while GWO improvements exist, many focus on hybridization with external algorithms rather than enhancing its internal social hierarchy. Although Random Forest is commonly used, its selection is often not justified, and its synergy with a specifically tailored optimizer is underexplored. To resolve feature selection issues in high-dimensional datasets, the paper presents a unique self-adaptive parameter and strategy. Results indicate that implementing these mechanisms greatly improves particle optimization techniques' search capability for high-dimensional datasets.

3 Methodology

In this work, a new meta-heuristic algorithm (IGWO-RF) which is a combination of an improved GWO algorithm and RF is proposed for feature selection problems. Therefore, the concepts related to the GWO algorithm and then the RF algorithm are presented first. The GWO algorithm is then improved to help expedite the procedure and increase accuracy. In the following, the proposed method will be described.

3.1 GWO algorithm

GWO algorithm is a nature-inspired metaheuristic algorithm that mimics the hunting behavior of gray wolves in nature. In GWO, the algorithm is based on the hierarchical structure of a wolf pack, where there are alpha, beta, delta, and omega wolves representing the best solutions found so far. The wolves collaborate to track and locate prey, which in the algorithm represents the optimal solution to the optimization problem. The alpha pair, sometimes referred to as the group leader in gray wolf social interactions, makes decisions regarding hunting, sleeping locations, waking times, and other matters. Beta wolves occupy the second position in a pack's hierarchy. Alphas receive help from beta wolves in decision-making and other pack-related tasks. The lowest-ranking wolves are omega wolves. Omega wolves are the lowest-ranking wolves. Alpha and beta wolves lead delta wolves, who are superior to omega wolves [34]. Alpha is considered the best answer in modeling the social hierarchy of gray wolves. Also, beta and delta are the second and third most appropriate answers after alpha. Other answers are placed in the omega group. In this optimization algorithm, hunting is guided by alpha, beta, and delta and omega wolves follow these 3 categories. The alpha wolf launches the attack when the prey stops moving and is encircled by wolves. To implement this pattern (hunting mechanism), the following relationships are used[35]:

$$\vec{D} = \left|\vec{C}\vec{X}_P(t) - \vec{X}(t)\right| \tag{1}$$

$$\vec{X}(t+1) = \vec{X}_P(t) - \vec{A}\vec{D} \tag{2}$$

Where t indicates the number of iterations, $\vec{X}(t)$ indicates the locations of gray wolves, $\vec{X}_P(t)$ indicates the locations of prey, A and C indicate the coefficient vectors, and D indicates the distance between the wolf and the prey. A and C vectors are calculated as follows:

$$\vec{A} = 2\vec{a}\vec{r}_1 - \vec{a} \tag{3}$$

$$\vec{C} = 2\vec{r}_2 \tag{4}$$

Where a indicates the convergence factor, and $\vec{r}_1$ and $\vec{r}_2$ indicate random vectors in the range of 0 and 1. A decreases linearly and during iterations from the value of 2 to 0. To simulate the hunting pattern of gray wolves, it is assumed that alpha, beta, and delta have better information about the location of the prey. As a result, the top 3 obtained answers are saved and other wolves have to update their position according to these top 3 answers. The following relationships are used for this:

$$\vec{D}_\alpha = \left|\vec{C}_1\vec{X}_\alpha - \vec{X}\right|, \quad \vec{D}_\beta = \left|\vec{C}_2\vec{X}_\beta - \vec{X}\right|, \quad \vec{D}_\delta = \left|\vec{C}_3\vec{X}_\delta - \vec{X}\right| \tag{5}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1\vec{D}_\alpha, \quad \vec{X}_2 = \vec{X}_\beta - \vec{A}_2\vec{D}_\beta, \quad \vec{X}_3 = \vec{X}_\delta - \vec{A}_3\vec{D}_\delta \tag{6}$$

$$\vec{X}(t+1) = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \tag{7}$$

Where Xα, Xβ, and Xδ indicate the locations of alpha, beta, and delta wolves, X indicates the current location of the gray wolf, and X(t+1) indicates the location of the gray wolf after updating. The social hierarchy in this method makes the algorithm store the best answers obtained during several iterations. The hunting mechanism allows for determining the possible position of the prey with superior responses. Search and extraction are guaranteed with respect to a and A. This method has only one parameter a to set and initialize. In fact, the balance between the process of exploration and extraction is controlled by a, thus, it significantly affects how well the algorithm performs.

3.2 Random Forest

RF is a powerful and versatile ML algorithm that is widely utilized for both regression and classification tasks. During training, it operates by constructing multiple decision trees, and for regression problems, it produces the average prediction of the individual trees, and for classification issues, the mode prediction. By employing a subset of characteristics at each node and bootstrapping samples from the training data, RF adds randomness to the tree-building process. This randomness helps to de-correlate the individual trees, leading to a strong ensemble model with improved generalization performance. RF is known for its robustness to overfitting, feature importance analysis, ease of use, and capability to handle large datasets with high dimensionality. It is a popular choice among data scientists and ML practitioners due to its effectiveness and scalability[36,37].

3.3 Proposed IGWO-RF algorithm

The suggested method is according to feature selection with the help of the combination of RF and GWO. This feature selection method is based on wrapper algorithms. The general process of the suggested method is displayed in Fig. 1. The RF algorithm was selected as the classifier of choice within the wrapper framework for several compelling reasons. Firstly, RF's inherent resistance to overfitting is crucial in a feature selection context, where the fitness evaluation is performed on numerous, potentially noisy feature subsets throughout the iterative optimization process. This robustness ensures that the fitness scores guiding the IGWO are reliable and not overly optimistic on the training data. Secondly, RF efficiently handles high-dimensional data and provides a stable performance across various datasets, which aligns with our goal of developing a general-purpose feature selection method. Furthermore, the ensemble nature of RF, which averages the results of multiple decision trees, reduces the variance of the fitness estimate, leading to a smoother and more consistent fitness landscape for the optimization algorithm to navigate. While other classifiers could be employed, RF's combination of robustness, efficiency, and stability makes it particularly well-suited for the computationally intensive and iterative fitness evaluation required by the wrapper-based IGWO-RF approach.

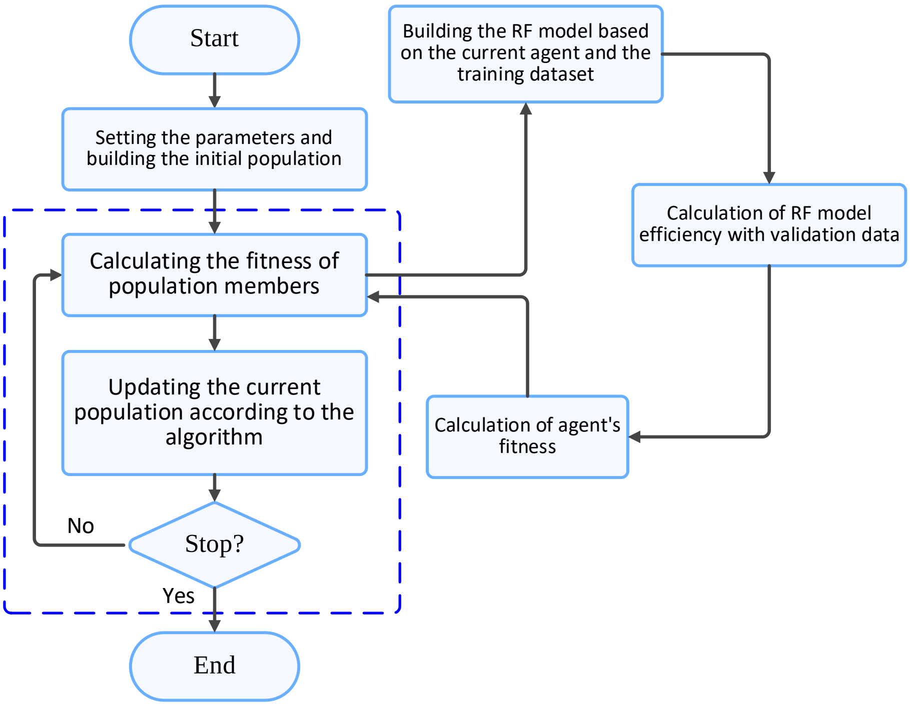

Figure 1: General process of the proposed IGWO-RF algorithm.

In feature selection, the objective is to lessen and eliminate irrelevant and redundant features, thereby constructing a precise pattern based on the selected feature set. Due to the fact that the model presented in this research uses the wrapper model for feature selection, the accuracy of the model should be checked in the evaluation of each agent. Therefore, the following relationship is used to calculate the fitness of the agents in the proposed method:

$$Fitness(Agent_i) = \frac{Number\ of\ correct\ sample\ detections}{Total\ samples} \quad (8)$$

The use of a stochastic classifier like Random Forest as a fitness function introduces a source of variance that must be carefully controlled. To ensure the reliability and fairness of the optimization process, we implemented two key precautions. First, a fixed random seed was used for the RF model during the entire feature selection process for a single experimental run. This creates a consistent and deterministic fitness landscape for the optimizer within that run. Second, each experiment was repeated 15 times with different initial populations, and for each run, a new fixed random seed was used for the RF evaluator. This practice allows us to report performance metrics as an average ± standard deviation, providing a statistically robust measure of the algorithm's performance. The agents of the problem, which are named wolves in the GWO algorithm, are in the form of vectors in Fig. 2.

f_1	f_2	f_3	…	f_i		f_N
0.5	0.2	0.3	…	0.75	…	0.6
w_1	w_2	w_3	…	w_i		w_N

Figure 2: gents in the proposed GWO.

According to the above representation, each agent has a length that corresponds to the quantity of features. Thus, the length of each agent is equal to N, and N is the number of features of each instance in the dataset. In this structure, the approach of weighting the features is used. That is, each part of a factor in the feature selection section indicates the degree of importance of that feature, and whatever this weight (wi, where $1 \leq i \leq N$) is more, the chance of choosing that feature increases. However, to extract useful features and remove non-useful features, the representation of Fig. 2 should be converted into a binary vector (one = feature selection, zero = no feature selection). For this, the following conversion function is used:

$$f(w_i) = \begin{cases} 1 & if \;\; w_i \geq 0.5 \\ 0 & Otherwise \end{cases} \tag{9}$$

where wi indicates the weight vector of the i-th feature and 0.5 is a predefined threshold. If the feature weight in the agent is greater than the threshold value, that feature is selected; otherwise, that feature will not be selected. Therefore, using Eq. (9) and a threshold value of 0.5, Fig. 2 is displayed as Fig. 3.

f_1	f_2	f_3	…	f_i		f_N
1	0	0	…	1	…	1
w_1	w_2	w_3	…	w_i		w_N

Figure 3: Representation of binary GWO for the feature selection process.

One of the primary issues of the GWO meta-heuristic algorithm is the use of a single parameter to balance exploration and mining. In this research, in the improved GWO algorithm (IGWO), 2 approaches are considered to improve the performance of the algorithm. Because of its linear behavior, the parameter a in the conventional technique tends to explore early in the algorithm's execution and mine toward the conclusion of its iterations. This parameter's behavior can be changed from linear to nonlinear to achieve a workable balance between exploration and mining. In the standard GWO algorithm, the control parameter a decreases linearly from 2 to 0 over the course of iterations. This linear decay assumes a constant and uniform shift from exploration to exploitation. However, this can be suboptimal for complex feature selection landscapes, often leading to premature convergence (insufficient exploration) or slow refinement (excessive late-stage exploration). To address this, we introduce a nonlinear convergence factor defined by:

$$a = 2 - 2\left(\frac{iteration}{iteration_{max}}\right)^2 \tag{10}$$

The selection of this specific quadratic form is motivated by three key principles:

1. Enhanced initial exploration: The quadratic term $\left(\frac{iteration}{iteration_{max}}\right)^2$ ensures that the value of a decreases very slowly during the initial stages of the optimization. This prolongs the period of high exploration (|A| > 1), allowing the algorithm to more thoroughly investigate the search space and reduce the probability of becoming trapped in local optima early on.
2. Accelerated final exploitation: In the later iterations, the quadratic decay causes a to drop more rapidly. This promotes a swift and intense phase of exploitation (|A| < 1), enabling the wolves to fine-tune their positions and converge precisely toward the global optimum.
3. Superior balance: This nonlinear strategy creates a more adaptive and dynamic balance between exploration and exploitation. It mimics a more natural search behavior—initially broad and inquisitive, followed by a decisive and focused conclusion—which is often more effective than a linear transition for navigating the discrete, high-dimensional search space of feature selection problems.

The next step makes more use of the beta wolves' position while determining how to go toward the objective. Thus, using inspiration from the genetic algorithm, alpha and beta wolves are considered as parents, and 2 children are produced using the crossover, which after checking their fitness is either added to the population and causes the delta wolves to be eliminated or does not affect the process. This accelerates convergence and a better search of the search space. In this study, uniform crossover is used. Intuitively, it is understandable that this method can be more effective than multiple-point and single-point crossovers. Every feature bit in the uniform crossover is distributed equally between the two parents (beta and delta wolves).

The decision to use the alpha (α) and beta (β) wolves as parents for the crossover operation is a deliberate design choice grounded in the social hierarchy and quality of solutions they represent. The rationale for this selection, over other potential pairings like alpha-delta (α-δ) or beta-delta (β-δ), is threefold. First, it allows for the exploitation of high-quality genetic material by recombining the two best solutions. Second, it provides a balanced diversity and convergence, as the alpha and beta are distinct yet elite points in the search space. Finally, it maintains biological and algorithmic fidelity with the core GWO metaphor, where the alpha and beta are the primary collaborators. The uniform crossover operator is applied with a probability of Pc = 1.0 to deterministically recombine the two elite solutions. Subsequently, a mutation operator with a probability of Pm = 1/N (where N is the total number of features) is applied to the offspring to facilitate local fine-tuning. The resulting offspring, after checking their fitness, are either added to the population or discarded if they do not improve the population's quality. The overall workflow is succinctly presented in Algorithm 1.

Algorithm 1. IGWO-RF algorithm.

Input:
Training data D_train, Test data D_test
Maximum iterations T_max
Population size N
Number of features F
Output:
Global best feature subset G_best
Best classification accuracy
1: Initialize the grey wolf population X_i (i = 1, 2, ..., N) randomly, where each wolf represents a continuous vector of length F.
2: Initialize X_α, X_β, X_δ ▷ Positions of alpha, beta, and delta wolves
3: t ← 1
4: while t ≤ T_max do
5: for each wolf X_i in the population do
6: ▷ Convert continuous position to binary feature subset
7: for j = 1 to F do
8: if X_i(j) ≥ 0.5 then
9: S_i(j) ← 1 ▷ Feature is selected
10: else
11: S_i(j) ← 0 ▷ Feature is not selected
12: end if
13: end for
14: ▷ Evaluate fitness using Random Forest
15: Fitness(i) ← RF_Accuracy(S_i, D_train) ▷ Eq. (8)
16: end for
17: ▷ Update alpha, beta, and delta positions
18: Update X_α, X_β, X_δ based on fitness
19: ▷ Update convergence parameter a (Nonlinear improvement)
20: a ← 2 - 2 × (t / T_max)^2 ▷ Eq. (10)
21: ▷ Update all wolves' positions
22: for each wolf X_i do
23: Update A, C using Eq. (3) and (4)
24: Calculate D_α = |C_1 × X_α - X_i|
25: Calculate D_β = |C_2 × X_β - X_i|
26: Calculate D_δ = |C_3 × X_δ - X_i|
27: Calculate X_1 = X_α - A_1 × D_α
28: Calculate X_2 = X_β - A_2 × D_β
29: Calculate X_3 = X_δ - A_3 × D_δ
30: X_i_new ← (X_1 + X_2 + X_3) / 3 ▷ Eq. (7)
31: end for
32: ▷ Crossover Operation (GA-inspired improvement)
33: Apply uniform crossover between X_α and X_β to produce two offspring, O_1 and O_2
34: Convert O_1 and O_2 to binary subsets S_O_1, S_O_2 (Lines 7-13)

```
35:   Evaluate Fitness(O₁), Fitness(O₂) using RF (Line 15)
36:   if Fitness(O₁) > Fitness(X_δ) then
37:     Replace X_δ with O₁ in the population
38:   end if
39:   if Fitness(O₂) > Fitness(X_δ) then
40:     Replace X_δ with O₂ in the population
41:   end if
42:   t ← t + 1
43: end while
44: ▷ Final Evaluation
45: Convert X_α to its binary representation G_best
46: Train final RF model on D_train using G_best
47: Report accuracy on D_test
```

3.4 Evaluation of the proposed algorithm

To have a fair assessment, the proposed IGWO-RF algorithm was compared with some wrapper-based approaches, including ACO-based, PSO-based, and ABC-based techniques. Moreover, various datasets were utilized to appraise the suggested algorithm and contrast its efficacy with alternative techniques. 10 popular benchmark datasets (Glass, Wine, Zoo, Vehicle, Soybean, Lung cancer, Sonar, Parkinson's, Yeast, and WiFi) with differences in the number of characteristics, the kind of data, and the number of samples were chosen from the UCI ML repository. Table 2 lists the features of these datasets in brief. Missing values of Soybean, as well as Lung cancer datasets, were substituted with the mean of the data. Furthermore, characteristics linked to an extensive array of values exert greater influence than those tied to narrower value ranges. To address this issue, a nonlinear normalization technique known as softmax scaling is implemented to evaluate the datasets effectively. During the search process, to determine fitness, 70% of the data were used as training data, as well as at the end, 30% of the data were used as test data to calculate the classification accuracy. Also, to report classification accuracy for all datasets, KNN (K = 3), SVM with RBF kernel ($\gamma = 2$, $C = 1$), and Naïve Bayes algorithms were used.

Table 2: Characteristics of UCI datasets utilized to appraise the suggested algorithm and compare its performance with other methods.

Dataset	Number of samples	Number of features	Number of classes	Type of data	Missing value
Glass	214	9	7	Real	No
Wine	178	13	3	Real and integer	No
Zoo	101	16	7	Integers and nominal	No
Vehicle	946	19	4	Integers	No
Soybean	307	35	19	Nominal	Yes
Lung cancer	32	56	2	Integers	Yes
Sonar	208	60	2	Real	No
Parkinson	195	22	2	Real	No
Yeast	1484	8	10	Real	No
WiFi	481	8	4	Real	No

In this article, to check the efficiency of the suggested method, an appropriate index was used to measure the classification performance. Therefore, the confusion matrix was used to calculate this evaluation measure that is described below.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{11}$$

4 Results and discussion

All experiments and simulations were conducted in MATLAB software, particularly through built-in MATLAB functions for SVM. The implementations were administered on a 2.20 GHz CPU and 16 GB RAM device. During the experiments, the evaluation of performance was performed based on the size of the feature subset and classification accuracy. Initially, the experiments involve assessing the performances of various wrapper feature selection methods with different classifiers. Each algorithm was implemented 15 times on each dataset and its average performance was reported. As mentioned, 3 separate classifiers were utilized to appraise the accuracy of the feature selection algorithms. Table 3 displays the averaged classification accuracy over 15 executions of the different wrapper search strategies using SVM, KNN, and Naïve Bayes classifiers.

Table 3: The results of selecting the best set of features using optimization algorithms and comparing them with the proposed algorithm based on the mean classification accuracy (%) obtained using the SVM, KNN, and Naïve Bayes classifiers. The accuracies are reported as mean ± standard deviation).

Dataset	Technique	Classifier		
		SVM	KNN	Naïve Bayes
Glass	IGWO-RF	91.34 ± 1.87	91.48 ± 1.54	90.41 ± 2.63
	GWO-RF	89.91 ± 2.56	89.93 ± 2.15	89.02 ± 2.50
	ACO	88.26 ± 2.21	88.79 ± 1.98	87.43 ± 2.59
	PSO	89.12 ± 2.01	89.37 ± 1.36	88.10 ± 2.96
	ABC	87.94 ± 2.70	88.13 ± 2.02	85.77 ± 3.05
Wine	IGWO-RF	81.47 ± 1.49	79.82 ± 1.32	77.89 ± 1.76
	GWO-RF	77.64 ± 1.85	74.19 ± 1.58	74.11 ± 1.35
	ACO	71.29 ± 2.11	70.29 ± 1.70	69.52 ± 1.69
	PSO	72.37 ± 1.67	71.16 ± 1.37	71.00 ± 1.19
	ABC	70.14 ± 2.34	68.56 ± 1.22	66.61 ± 1.90
Zoo	IGWO-RF	96.67 ± 2.06	95.43 ± 2.11	94.23 ± 2.41
	GWO-RF	95.22 ± 2.49	94.17 ± 2.33	93.61 ± 1.97
	ACO	95.82 ± 2.16	95.76 ± 2.24	94.19 ± 2.61
	PSO	97.00 ± 2.67	95.09 ± 2.34	94.13 ± 2.45
	ABC	96.20 ± 2.12	95.84 ± 2.40	94.78 ± 2.21
Vehicle	IGWO-RF	75.96 ± 1.97	74.05 ± 2.42	73.25 ± 2.26
	GWO-RF	74.14 ± 2.16	73.31 ± 2.24	72.19 ± 2.64
	ACO	72.74 ± 2.38	71.45 ± 1.84	70.55 ± 2.36
	PSO	73.72 ± 2.30	72.51 ± 1.91	72.23 ± 2.00
	ABC	73.06 ± 1.95	71.46 ± 2.15	70.67 ± 2.34
Soybean	IGWO-RF	100.00	100.00	100.00
	GWO-RF	100.00	100.00	100.00
	ACO	100.00	100.0	100.0
	PSO	100.00	100.00	100.00
	ABC	99.76 ± 1.23	99.71 ± 1.15	99.58 ± 1.32
Lung cancer	IGWO-RF	98.12 ± 2.53	98.23 ± 2.21	97.42 ± 2.45
	GWO-RF	96.05 ± 2.14	96.61 ± 2.62	95.17 ± 2.03
	ACO	97.18 ± 2.91	97.20 ± 1.94	96.39 ± 2.15
	PSO	98.33 ± 2.50	98.41 ± 2.43	97.28 ± 2.35
	ABC	95.49 ± 2.11	95.88 ± 1.99	95.14 ± 2.16
Sonar	IGWO-RF	88.39 ± 2.17	88.75 ± 2.48	87.01 ± 2.10
	GWO-RF	87.62 ± 3.11	88.12 ± 3.21	86.56 ± 3.33
	ACO	87.34 ± 3.30	88.05 ± 2.31	85.80 ± 2.61
	PSO	87.79 ± 2.28	88.17 ± 2.42	86.91 ± 3.11
	ABC	87.15 ± 2.73	87.24 ± 2.14	86.70 ± 1.99
Parkinson	IGWO-RF	98.89 ± 2.10	98.67 ± 1.68	97.57 ± 2.34
	GWO-RF	97.71 ± 3.15	97.50 ± 2.63	96.97 ± 2.45
	ACO	97.22 ± 2.20	96.17 ± 3.61	95.82 ± 1.97
	PSO	99.12 ± 3.14	98.75 ± 2.98	97.67 ± 2.47
	ABC	96.91 ± 1.43	96.55 ± 1.37	95.71 ± 3.30
Yeast	IGWO-RF	83.55 ± 3.43	84.62 ± 3.16	82.18 ± 3.34
	GWO-RF	82.27 ± 3.05	82.59 ± 2.60	81.64 ± 1.94
	ACO	79.20 ± 2.98	80.00 ± 3.16	78.72 ± 3.25
	PSO	80.91 ± 3.29	81.46 ± 2.69	80.02 ± 4.11
	ABC	80.11 ± 3.70	80.43 ± 3.51	79.07 ± 3.22
WiFi	IGWO-RF	98.81 ± 2.52	98.70 ± 3.29	97.65 ± 1.85
	GWO-RF	97.23 ± 3.09	97.25 ± 2.79	95.83 ± 2.13
	ACO	99.04 ± 3.11	98.76 ± 3.28	98.01 ± 2.70
	PSO	100.00	100.00	100.00
	ABC	98.54 ± 2.74	98.05 ± 1.66	96.53 ± 2.05

As displayed in Table 3, the suggested IGWO-RF algorithm outperforms other wrapper-based feature selection techniques in most datasets. For example, in the Wine dataset on the SVM classifier, the IGWO-RF achieved an 81.47% classification accuracy. However, for GWO-RF, PSO, ACO, and ABC techniques, these values were 77.64%, 72.37%, 71.29%, and 70.14%, respectively. The interesting point is that the IGWO-RF algorithm performs better than the GWO-RF algorithm in all datasets. Additionally, the mean classification accuracy for the SVM, KNN, and Naïve Bayes classifiers across all databases is displayed in Figs. 4, 5, and 6, in that order. As displayed in these figs, the proposed IGWO-RF technique produced the largest mean classification accuracy on all classifiers. For example, Fig. 3 shows that the IGWO-RF algorithm ranked first among all the investigated algorithms with an average accuracy of 91.23%, with a margin of about 1.4% compared to the second-ranked PSO algorithm. This trend can also be seen in Figs. 5 and 6.

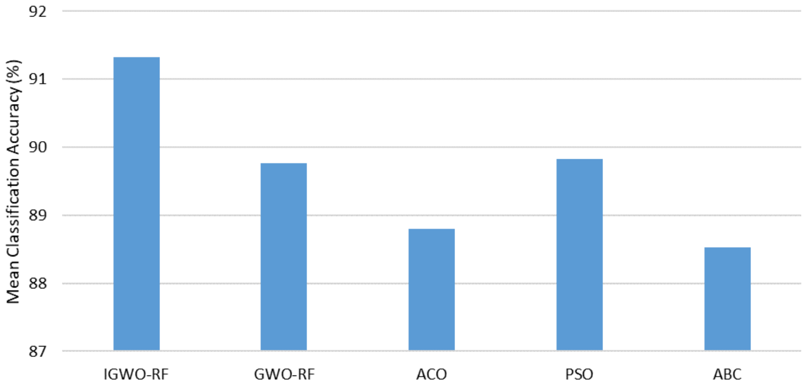

Figure 4: The SVM classifier's mean classification accuracy across all databases.

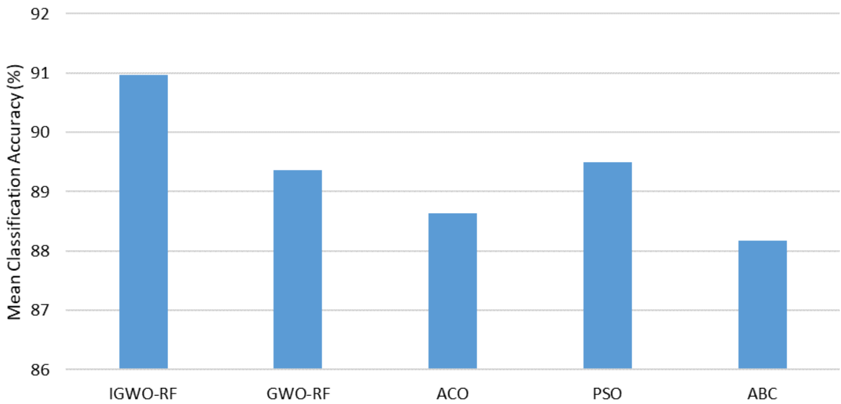

Figure 5: Mean classification accuracy on the KNN classifier across all databases.

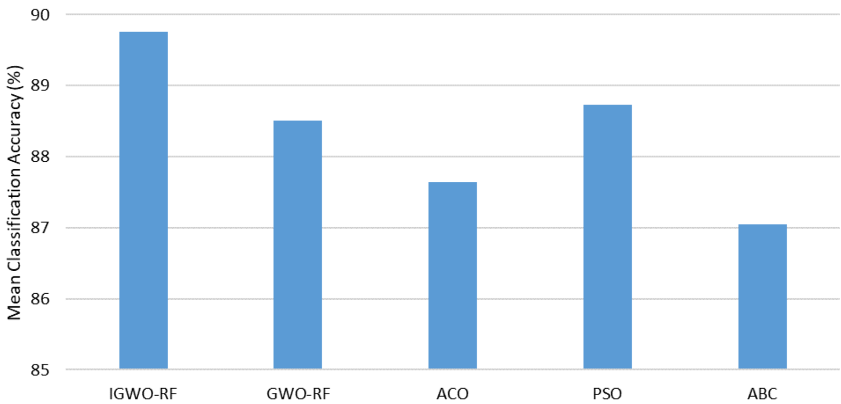

Figure 6: Mean classification accuracy using the Naïve Bayes classifier across all databases.

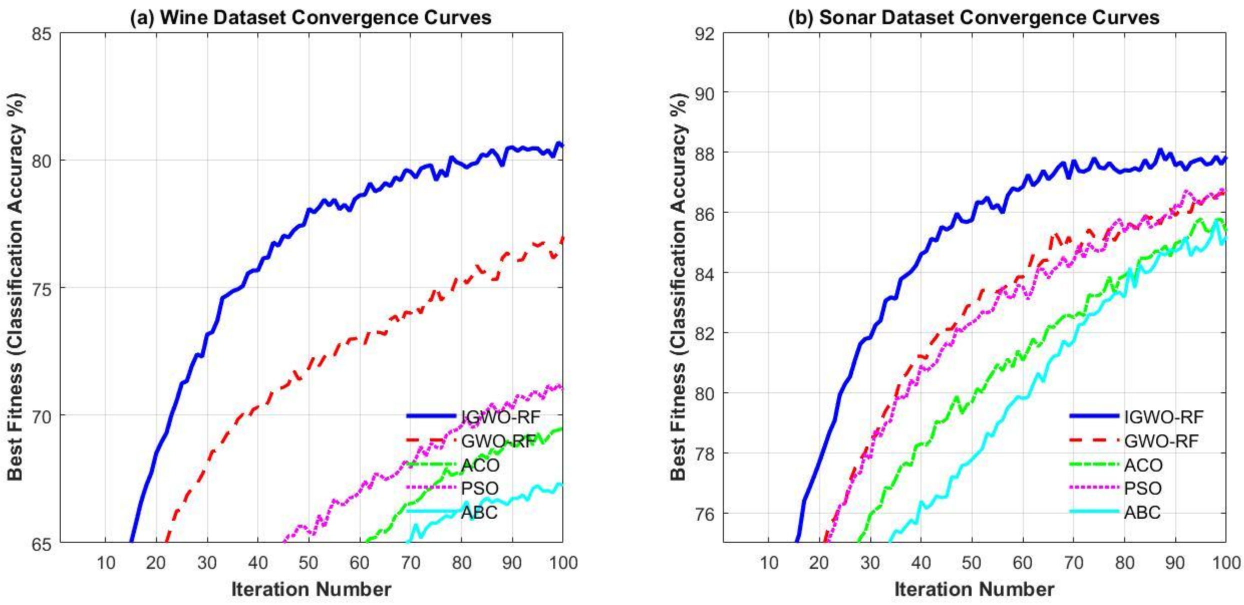

Figure 7: Convergence curves comparing the proposed IGWO-RF algorithm with other wrapper-based feature selection methods on (a) the Wine dataset and (b) the Sonar dataset. The plots depict the best fitness value (classification accuracy) achieved versus the number of iterations.

The convergence behavior of the proposed IGWO-RF algorithm compared to other wrapper-based feature selection techniques is illustrated in Figure 7. The plots clearly demonstrate that IGWO-RF not only achieves the highest final classification accuracy on both the Wine and Sonar datasets but also exhibits superior convergence characteristics. Specifically, IGWO-RF converges more rapidly during the initial iterations and maintains a steadier ascent toward the global optimum, with minimal oscillations. This enhanced performance can be attributed to the improved balance between exploration and exploitation facilitated by the nonlinear parameter update strategy and the effective use of beta wolves in the reproduction process. In contrast, other algorithms such as ABC and ACO display slower convergence rates and greater susceptibility to local optima, resulting in both lower final accuracy and less stable optimization trajectories.

For each database, Table 4 lists the total number of selected characteristics from the five-wrapper evolutionary-based feature selection methods. As shown, generally, all algorithms lead to a substantial dimensionality reduction by selecting a small portion of the original attributes. It can be displayed that the suggested IGWO-RF algorithm produces a smaller feature subset and outperforms other techniques on most databases.

Table 4: The five-wrapper evolutionary-based feature selection strategies' mean number of selected attributes.

Dataset	Number of all features	Technique	Number of selected features	Ration of selected features to all features (%)
Glass	9	IGWO-RF	2.81	31.22
		GWO-RF	3.12	34.66
		ACO	3.23	35.88
		PSO	3.31	36.78
		ABC	3.82	42.44
Wine	13	IGWO-RF	2.91	22.38
		GWO-RF	3.05	23.46
		ACO	3.77	29.00
		PSO	3.50	26.92
		ABC	3.98	30.61
Zoo	16	IGWO-RF	3.94	24.62
		GWO-RF	4.22	26.37
		ACO	4.82	30.12
		PSO	4.13	25.81
		ABC	4.91	30.68
Vehicle	19	IGWO-RF	3.97	20.89
		GWO-RF	4.68	24.63

		ACO	5.24	27.57
		PSO	4.60	24.21
		ABC	5.45	28.68
Soybean	35	IGWO-RF	1.95	5.57
		GWO-RF	2.08	5.94
		ACO	3.25	9.28
		PSO	2.00	5.71
		ABC	2.85	8.14
Lung cancer	56	IGWO-RF	8.53	15.23
		GWO-RF	8.89	15.87
		ACO	10.64	19.00
		PSO	7.98	14.25
		ABC	11.93	21.30
Sonar	60	IGWO-RF	6.43	10.71
		GWO-RF	7.10	11.83
		ACO	8.02	13.36
		PSO	8.14	13.57
		ABC	8.84	14.73
Parkinson	22	IGWO-RF	4.17	18.95
		GWO-RF	4.56	20.72
		ACO	4.97	22.59
		PSO	5.10	23.18
		ABC	5.81	26.40
Yeast	8	IGWO-RF	1.33	16.62
		GWO-RF	1.61	20.12
		ACO	2.05	25.62
		PSO	1.55	19.37
		ABC	1.69	21.12
WiFi	8	IGWO-RF	1.74	21.75
		GWO-RF	1.99	24.87
		ACO	1.85	23.12
		PSO	1.65	20.62
		ABC	2.10	26.25

For a better comparison, the statistical merit of the results of Tables 3 and 4 were determined according to Eq. (8), so that both classification accuracy and how many features are chosen influence how well-ranked the current algorithms are. For this purpose, the Friedman test with 6 degrees of freedom according to chi-square distribution was used. This statistical test was calculated for each execution of the algorithms and the average rank of each algorithm was obtained. Table 5 shows the average rank of each algorithm based on the number of selected features and classification accuracy obtained from the different classifiers. As shown, the proposed IGWO-RF algorithm is ranked better than other algorithms on average. The ranking in Fig. 4 shows that the IGWO-RF algorithm was able to find smaller feature subsets with higher classification accuracy than other wrapper evolutionary-based feature selection techniques.

Table 5: Average ranking of the five-wrapper evolutionary-based feature selection techniques on SVM, KNN, and Naïve Bayes classifiers.

Classifier	Wrapper evolutionary-based feature selection algorithm				
	IGWO-RF	GWO-RF	ACO	PSO	ABC
SVM	1.54	2.90	3.61	1.97	4.22
KNN	1.58	2.94	3.60	1.91	4.21
Naïve Bayes	1.55	2.90	3.65	1.99	4.34

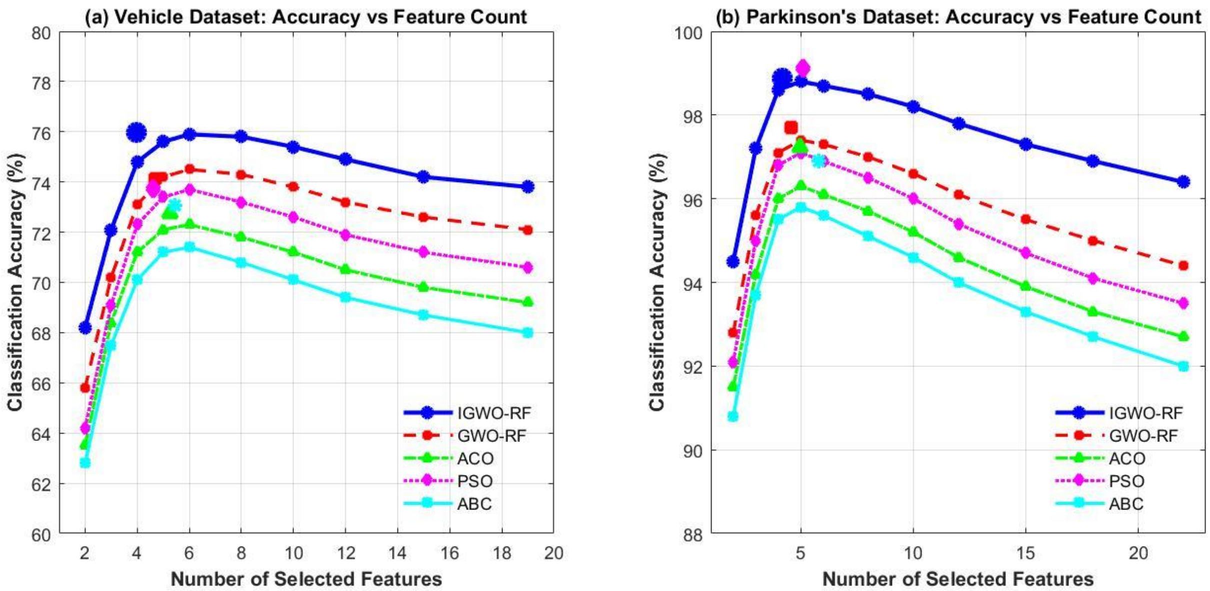

Figure 8: Comparative analysis of classification accuracy versus number of selected features for all wrapper-based feature selection techniques on (a) Vehicle and (b) Parkinson's datasets. The plots illustrate the trade-off between model complexity and performance across different algorithms.

Figure 8 presents a comparative analysis of the accuracy versus feature count relationship for all evaluated feature selection techniques on the Vehicle and Parkinson's datasets. The curves demonstrate that IGWO-RF consistently achieves higher classification accuracy across the entire spectrum of feature subset sizes, with its performance curve positioned above all other methods. Notably, IGWO-RF reaches near-optimal performance with fewer features than competing algorithms, indicating superior feature ranking and selection capability. The optimal operating points (marked on each curve) reveal that IGWO-RF attains the best accuracy (75.96% for Vehicle, 98.89% for Parkinson's) while selecting the most compact feature subsets. This superior performance-profile demonstrates IGWO-RF's enhanced ability to identify the most discriminative features early in the selection process, providing both computational efficiency and model interpretability advantages.

Multiple trials were carried out to examine the performance of various wrapper evolutionary-based feature selection techniques in terms of execution time. The corresponding time taken for each method to execute is documented in Table 6. As the feature selection and classification processes operate independently, solely the feature selection execution time is included in the table data. Findings from the analysis indicate that the IGWO-RF feature selection approach showcased the quickest average execution time across the entire dataset compared to the other methods. Following the IGWO-RF method, the GWO-RF and PSO methodologies achieved second and third place, respectively regarding execution time.

To critically evaluate the individual contributions of the IGWO and the RF within the proposed IGWO-RF framework, an ablation study was conducted. This study compares three configurations: (I) IGWO-RF (Proposed): The full proposed method; (II) RF-Only: Using Random Forest on the full feature set without any feature selection; and (III) IGWO-SVM: Replacing the RF classifier in the wrapper with a Support Vector Machine (SVM) to isolate the optimization performance from a specific classifier's inherent robustness. The results, averaged across all datasets, are summarized in Table 7. This analysis validates that the success of IGWO-RF is not merely due to the power of the RF rest classifier alone. Instead, it results from a synergistic combination where the IGWO algorithm effectively finds compact, discriminative feature subsets, and the RF classifier robustly evaluates them, creating a feedback loop that leads to superior performance compared to either component in isolation or when paired with a less suitable classifier.

Table 6: Mean run time (second) of the five-wrapper evolutionary-based feature selection techniques over 15 runs.

Dataset	IGWO-RF	GWO-RF	ACO	PSO	ABC
Glass	8.32	9.47	10.12	9.05	10.94
Wine	8.86	10.08	10.45	9.12	11.15
Zoo	6.71	7.50	7.85	6.92	8.44
Vehicle	8.76	9.00	9.63	8.97	10.26
Soybean	7.83	8.24	8.49	8.10	8.85
Lung cancer	10.81	11.20	11.56	11.15	12.21
Sonar	6.12	6.50	8.74	6.25	9.17
Parkinson	9.18	9.47	10.29	9.36	10.84

Yeast	4.21	4.39	4.73	4.30	4.91
WiFi	4.36	4.52	4.92	4.45	5.10

Table 7: Results of the ablation study comparing the proposed method with two ablated variants. Values represent averages across all datasets.

Configuration	Mean Accuracy (%)
IGWO-RF (Proposed)	91.23
RF-Only (no feature selection)	85.41
IGWO-SVM	88.95

Table 8: Analysis of computational cost and efficiency. Time per iteration is averaged across all datasets. Convergence is defined as reaching 99% of the final best fitness.

Algorithm	Time per Iteration (s)	Average Iterations to Converge	Total Time to Solution (s)
IGWO-RF	1.85	38	70.3
GWO-RF	1.72	52	89.4
PSO	1.65	61	100.7
ACO	2.10	58	121.8
ABC	1.91	66	126.1

The computational analysis in Table 8 reveals a critical insight into the efficiency of the proposed IGWO-RF algorithm. As expected, the per-iteration time for IGWO-RF (1.85s) is slightly higher than that of GWO-RF (1.72s) due to the overhead of the crossover operation and the two additional fitness evaluations. However, the enhanced search capability of IGWO-RF leads to significantly faster convergence, requiring only 38 iterations on average to reach 99% of the final fitness, compared to 52 for GWO-RF and 61 for PSO. This results in a superior performance-time trade-off. The Total Time to Solution for IGWO-RF (70.3s) is 21% faster than GWO-RF (89.4s) and 30% faster than PSO (100.7s), despite its higher per-iteration cost. This demonstrates that the computational overhead introduced by the crossover mechanism is a worthwhile investment, as it guides the population more efficiently toward the optimal feature subset, ultimately reducing the total computational effort required to find a high-quality solution.

5 Conclusions

In this research, a meta-heuristic wrapper evolutionary-based feature selection approach was proposed. This algorithm works based on an improved GWO (IGWO) algorithm and RF algorithm. Several experiments on various datasets proved the superiority of this algorithm compared to other wrapper evolutionary-based algorithms. The results showed that the IGWO-RF algorithm reduces the execution time of feature selection compared to the original GWO algorithm and other wrapper techniques, and finally produces a small optimal subset of the original features with high accuracy. Due to its rapid fitness evaluation process, the method suggested is well-suited for addressing feature selection challenges across various scales, from small to large. Although the introduced algorithm enhances the classifier's performance, the current experimental outcomes fall short of the intended objective, particularly when addressing the feature selection issue associated with class imbalance. Nevertheless, addressing this specific challenge remains a significant obstacle in the realm of feature selection for future works. While the current validation uses medium-scale UCI datasets, the algorithm's design principles make it a promising candidate for real-world, high-dimensional problems in genomics, text analytics, and medical imaging. Future work will involve applying IGWO-RF to these domains and developing a hybrid filter-wrapper variant to enhance its scalability for ultra-high-dimensional data. The manual parameter setting for both the IGWO and the RF classifier presents an opportunity for automation. Future work will involve employing advanced automated hyperparameter optimization techniques, such as Bayesian Optimization or Metaheuristic-based tuning, to systematically determine the optimal configuration (e.g., population size, crossover probability, number of trees in RF) for different dataset characteristics, thereby maximizing performance and robustness. Framing feature selection as a multi-objective optimization problem is a natural and valuable extension. Developing a multi-objective variant of IGWO (e.g., based on NSGA-II or MOEA/D frameworks) would allow for the simultaneous optimization of competing objectives, such as maximizing classification accuracy and minimizing the number of selected features.

Funding

This work was supported by the Excellent Talent Foundation of China West Normal University (No: 17YC497)

References

[1] Liu, X., S. Wang, S. Lu, Z. Yin, X. Li, L. Yin, J. Tian and W. Zheng (2023). Adapting feature selection algorithms for the classification of Chinese texts. *Systems*, 11(9): 483. https://doi.org/10.3390/systems11090483

[2] Rostami, M., K. Berahmand, E. Nasiri and S. Forouzandeh (2021). Review of swarm intelligence-based feature selection methods. *Engineering*

Applications of Artificial Intelligence, 100: 104210. https://doi.org/10.1016/j.engappai.2021.104210

[3] Solorio-Fernández, S., J.A. Carrasco-Ochoa and J.F. Martínez-Trinidad (2020). A review of unsupervised feature selection methods. *Artificial Intelligence Review*, 53(2): 907–948. https://doi.org/10.1007/s10462-019-09682-y

[4] Zebari, R., A. Abdulazeez, D. Zeebaree, D. Zebari and J. Saeed (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends*, 1(1): 56–70. https://doi.org/10.38094/jastt1224

[5] Chen, K., B. Xue, M. Zhang and F. Zhou (2020). An evolutionary multitasking-based feature selection method for high-dimensional classification. *IEEE Transactions on Cybernetics*, 52(7): 7172–7186. https://doi.org/10.1109/TCYB.2020.3042243

[6] Li, H., F. He, Y. Liang and Q. Quan (2020). A dividing-based many-objective evolutionary algorithm for large-scale feature selection. *Soft Computing*, 24: 6851–6870. https://doi.org/10.1007/s00500-019-04324-5

[7] Jiao, R., B.H. Nguyen, B. Xue and M. Zhang (2023). A survey on evolutionary multiobjective feature selection in classification: approaches, applications, and challenges. *IEEE Transactions on Evolutionary Computation*. https://doi.org/10.1109/TEVC.2023.3292527

[8] García-Torres, M., R. Ruiz and F. Divina (2023). Evolutionary feature selection on high dimensional data using a search space reduction approach. *Engineering Applications of Artificial Intelligence*, 117: 105556. https://doi.org/10.1016/j.engappai.2022.105556

[9] Saadatmand, H. and M.-R. Akbarzadeh-T (2023). Set-based integer-coded fuzzy granular evolutionary algorithms for high-dimensional feature selection. *Applied Soft Computing*, 142: 110240. https://doi.org/10.1016/j.asoc.2023.110240

[10] Al-Tashi, Q., H. Md Rais, S.J. Abdulkadir, S. Mirjalili and H. Alhussian (2020). A review of grey wolf optimizer-based feature selection methods for classification. *Evolutionary Machine Learning Techniques: Algorithms and* Applications, 273–286. https://doi.org/10.1007/978-981-32-9990-0_13

[11] Khaleghi, A., P.M. Birgani, M.F. Fooladi and M.R. Mohammadi (2020). Applicable features of electroencephalogram for ADHD diagnosis. *Research on Biomedical Engineering*, 36: 1–11. https://doi.org/10.1007/s13246-015-0375-0

[12] Khaleghi, A., M.R. Mohammadi, K. Shahi and A.M. Nasrabadi (2022). Computational neuroscience approach to psychiatry: a review on theory-driven approaches. *Clinical Psychopharmacology and Neuroscience*, 20(1): 26. https://doi.org/10.9758/cpn.2022.20.1.26

[13] Khaleghi, A., A. Sheikhani, M.R. Mohammadi, A.M. Nasrabadi, S.R. Vand, H. Zarafshan and M. Moeini (2015). EEG classification of adolescents with type I and type II of bipolar disorder. *Australasian Physical & Engineering Sciences in Medicine*, 38: 551–559. https://doi.org/10.1007/s13246-015-0375-0

[14] Mohammadi, M.R., A. Khaleghi, A.M. Nasrabadi, S. Rafieivand, M. Begol and H. Zarafshan (2016). EEG classification of ADHD and normal children using non-linear features and neural network. *Biomedical Engineering Letters*, 6: 66–73. https://doi.org/10.1007/s13534-016-0218-2

[15] Zhang, Y., D. Gong, X. Gao, T. Tian and X. Sun (2020). Binary differential evolution with self-learning for multi-objective feature selection. *Information Sciences*, 507: 67–85. https://doi.org/10.1016/j.ins.2019.08.040

[16] Neggaz, N., A.A. Ewees, M. Abd Elaziz and M. Mafarja (2020). Boosting salp swarm algorithm by sine cosine algorithm and disrupt operator for feature selection. *Expert Systems with Applications*, 145: 113103. https://doi.org/10.1016/j.eswa.2019.113103

[17] Arowolo, M.O., M.O. Adebiyi, A.A. Adebiyi and O.J. Okesola (2020). A hybrid heuristic dimensionality reduction methods for classifying malaria vector gene expression data. *IEEE Access*, 8: 182422–182430. https://doi.org/10.1109/ACCESS.2020.3029234

[18] Rostami, M., K. Berahmand and S. Forouzandeh (2021). A novel community detection based genetic algorithm for feature selection. *Journal of Big Data*, 8(1): 2. https://doi.org/10.1186/s40537-020-00398-3

[19] Singh, U. and S.N. Singh (2019). A new optimal feature selection scheme for classification of power quality disturbances based on ant colony framework. *Applied Soft Computing*, 74: 216–225.

[20] Karimi, F., M.B. Dowlatshahi and A. Hashemi (2023). SemiACO: A semi-supervised feature selection based on ant colony optimization. *Expert Systems with Applications*, 214: 119130. https://doi.org/10.1016/j.eswa.2022.119130

[21] Kaur, S., Y. Kumar, A. Koul and S. Kumar Kamboj (2023). A systematic review on metaheuristic optimization techniques for feature selections in disease diagnosis: open issues and challenges. *Archives of Computational Methods in Engineering*, 30(3): 1863–1895.

[22] Sadeghian, Z., E. Akbari, H. Nematzadeh and H. Motameni (2023). A review of feature selection methods based on meta-heuristic algorithms. *Journal of Experimental & Theoretical Artificial Intelligence*, 1–51. https://doi.org/10.1080/0952813X.2023.2183267

[23] Sharma, M. and P. Kaur (2021). A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem. *Archives of Computational Methods in Engineering*, 28: 1103–1127.

[24] Gong, D., B. Xu, Y. Zhang, Y. Guo and S. Yang (2019). A similarity-based cooperative co-evolutionary algorithm for dynamic interval multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 24(1):

142–156. https://doi.org/10.1109/TEVC.2019.2912204

[25] Maleki, N., Y. Zeinali and S.T.A. Niaki (2021). A k-NN method for lung cancer prognosis with the use of a genetic algorithm for feature selection. *Expert Systems with Applications*, 164: 113981. https://doi.org/10.1016/j.eswa.2020.113981

[26] Amini, F. and G. Hu (2021). A two-layer feature selection method using genetic algorithm and elastic net. *Expert Systems with Applications*, 166: 114072. https://doi.org/10.1016/j.eswa.2021.114072

[27] Rathee, S. and S. Ratnoo (2020). Feature selection using multi-objective CHC genetic algorithm. *Procedia Computer Science*, 167: 1656–1664. https://doi.org/10.1016/j.procs.2020.03.376

[28] Sayed, S., M. Nassef, A. Badr and I. Farag (2019). A nested genetic algorithm for feature selection in high-dimensional cancer microarray datasets. *Expert Systems with Applications*, 121: 233–243. https://doi.org/10.1016/j.eswa.2018.12.022

[29] Yan, C., J. Liang, M. Zhao, X. Zhang, T. Zhang and H. Li (2019). A novel hybrid feature selection strategy in quantitative analysis of laser-induced breakdown spectroscopy. *Analytica Chimica Acta*, 1080: 35–42.

[30] Xue, Y., T. Tang, W. Pang and A.X. Liu (2020). Self-adaptive parameter and strategy based particle swarm optimization for large-scale feature selection problems with multiple classifiers. *Applied Soft Computing*, 88: 106031. https://doi.org/10.1016/j.asoc.2020.106031

[31] Zhang, C., W. Wang and Y. Pan (2020). Enhancing electronic nose performance by feature selection using an improved grey wolf optimization based algorithm. *Sensors*, 20(15): 4065. https://doi.org/10.3390/s20154065

[32] El-Hasnony, I.M., S.I. Barakat, M. Elhoseny and R.R. Mostafa (2020). Improved feature selection model for big data analytics. *IEEE Access*, 8: 66989–67004. https://doi.org/10.1109/ACCESS.2020.2986232

[33] Abdel-Basset, M., K.M. Sallam, R. Mohamed, I. Elgendi, K. Munasinghe and O.M. Elkomy (2021). An improved binary grey-wolf optimizer with simulated annealing for feature selection. *IEEE Access*, 9: 139792–139822. https://doi.org/10.1109/ACCESS.2021.3117853

[34] Purushothaman, R., S.P. Rajagopalan and G. Dhandapani (2020). Hybridizing Gray Wolf Optimization (GWO) with Grasshopper Optimization Algorithm (GOA) for text feature selection and clustering. *Applied Soft Computing*, 96: 106651. https://doi.org/10.1016/j.asoc.2020.106651

[35] Pan, H., S. Chen and H. Xiong (2023). A high-dimensional feature selection method based on modified Gray Wolf Optimization. *Applied Soft Computing*, 135: 110031. https://doi.org/10.1016/j.asoc.2023.110031

[36] Hu, J. and S. Szymczak (2023). A review on longitudinal data analysis with random forest. *Briefings in Bioinformatics*, 24(2): bbad002. https://doi.org/10.1093/bib/bbad002

[37] Speiser, J.L., M.E. Miller, J. Tooze and E. Ip (2019). A comparison of random forest variable selection methods for classification prediction modeling. *Expert Systems with Applications*, 134: 93–101. https://doi.org/10.1016/j.eswa.2019.05.028