Iraqi Journal for Computers and Informatics
Available online: https://ijci.uoitc.edu.iq/index.php/ijci
Volume52, issue 1. 2026

Print ISSN: 2313-190X, Online ISSN: 2520-4912

Metaheuristic-Based Hyperparameter Optimization for Machine Learning Classification: An Applied Experimental Study

Ahmed Majid Taha [a, b]

[a] Biomedical Informatics College, University of Information Technology and Communications, Baghdad, Iraq
[b] Soft Computing and Data Mining Center, University Tun Hussein Onn Malaysia 86400 Parit Raja, Batu Pahat Johor, Malaysia

AR[1]TICLE INFO

***Keywords**:*

Metaheuristic optimization; Hyperparameter tuning; Machine learning classification; Particle Swarm Optimization; Grey Wolf Optimizer; Hybrid optimization.

ABSTRACT

The selection of hyperparameters is a key factor in the predictive performance and the overall generalization of machine learning models. In real-life scenarios, poor hyperparameter selection tends to result in suboptimal performance, despite the use of sophisticated learning algorithms. Intelligent optimization techniques are commonly adopted when conventional tuning methods, such as grid search and random search, become computationally infeasible. In this paper, an experimental study involving the use of metaheuristic-based hyperparameter optimization for machine learning classification is outlined. The study used Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and Grey Wolf Optimizer (GWO) to optimize the hyperparameters of popular classifiers, such as Support Vector Machine, Random Forest, and k-Nearest Neighbors. A hybrid PSO–GWO framework is also proposed to accommodate complementary exploration and exploitation behaviors and improve convergence stability and optimization effectiveness. Experiments on multiple benchmark datasets of the UCI Machine Learning Repository revealed that hyperparameter optimization using metaheuristics consistently outperforms default configurations in classification tasks. In addition, the developed hybrid method is more accurate and exhibits more stable convergence behavior, compared to individual optimizers. These results show that hybrid metaheuristic approaches are relevant and scalable to improve machine learning classification in applied settings.

1. INTRODUCTION

The use of machine learning techniques has become an indispensable part of intelligent systems of the present day and is highly applicable in the fields of healthcare analytics, financial forecasting, cybersecurity, and decision-support systems [1]–[3]. Even though learning algorithms are being developed at a very fast rate, the selection of hyperparameters is very sensitive to model performance. In a large number of real-life situations, incorrect hyperparameter selection leads to underfitting, overfitting, or a waste of computational power, thus reducing the reliability of deployed models [4]. Hyperparameter tuning of conventional grid search and random search algorithms has become a common practice because they are simple and easy to apply. Nevertheless, the methods have severe scalability issues with complex models or hyperparameter space dimensions [5]. The grid search is computationally infeasible when the parameters increase, and the random search does not make effective use of valuable search space areas [6], leading to an increased demand for more effective and adaptive tuning approaches.

The introduction of metaheuristic optimization algorithms has emerged as a strong alternative for addressing hyperparameter tuning challenges, because of their population-based search principles and search capabilities on nonlinear and complicated optimization surfaces [7], [8]. Based on the principles of nature and evolution, algorithms like the Particle

---

E-mail address:
dr.ahmed_majid@uoitc.edu.iq

Received 6th February 2026, Accepted 28 February 2026

DOI: 10.25195/ijci.v52i1.734.

Swarm Optimization, Genetic Algorithm, and the Grey Wolf Optimizer have exhibited strong global search of a large scope of optimization problems [9], [10]. Over the past few years, these methods have also been applied to optimize hyperparameters of machine learning with significant gains in classification accuracy and convergence behavior [11], [12]. Although these developments have been made, there are various limitations in the current research; most studies are on individual optimizers or individual classifiers which limits the external applicability of their findings [13]. Also, complementary optimization behavior of hybrid metaheuristic approaches is not adequately investigated in applied hyperparameter optimization scenarios, especially with systematic evaluation on multi-dataset and multi-classifier settings [14], [15].

Motivated by these observations, this paper presents an applied experimental research on metaheuristic-based hyperparameter optimization in machine learning classification, wherein PSO, GA and GWO are tested for tuning the hyperparameters of popular classifiers such as Support Vector Machine, Random Forest, and k-Nearest Neighbors. Moreover, it suggests a hybrid PSO-GWO to increase the stability of convergence by combining high-speed global exploration and high local exploitation.

The key contributions of this research are summarized as follows:

i. Experimental comparison of PSO, GA, and GWO to machine learning hyperparameter optimization.
ii. A hybrid PSO–GWO optimization framework is designed to achieve improved convergence behavior.
iii. An analytic measurement of the performance of different classifiers and benchmark datasets using standard metrics.

The rest of this paper is structured in the following way: Section 2 conducts a related literature review of metaheuristic hyperparameter optimization. Section 3 formulates the optimization problem. Section 4 covers the description of the proposed hybrid framework. Section 5 explains the experimental design. Results and discussion are in section 6. Section 7 addresses threats to validity. Section 8 presents the conclusion of the study.

2. Related Work

The problem of hyperparameter optimization has become a significant research concern in the field of machine learning because it directly influences the predictive accuracy and robustness, as well as the generalization of the models. The grid search and random search methods are still popular due to their conceptual simplicity as a traditional method of hyperparameter tuning. Nonetheless, the recent literature demonstrated that these algorithms do not scale to complex models or high-dimensional hyperparameter space, resulting in a high computational cost and non-optimal solutions [5], [6], [16]–[18]. Consequently, smart optimization methods have gained more and more popularity in practice-based machine learning studies. Metaheuristic optimization algorithms are well suited for hyperparameter tuning of machine learning models since the algorithms do not require or utilize gradient data, and can efficiently work with nonlinear and multimodal search spaces. One of the most commonly explored metaheuristics in that regard is Particle Swarm Optimization (PSO). Recent applied research indicates that PSO-based hyperparameter optimization can enhance the performance of Support Vector Machines, random forests, and ensemble classifiers, as compared to exhaustive or randomized searching approaches [11], [12], [19], [20]. Nevertheless, PSO is prone to premature convergence despite its efficiency and particularly at later stages of optimization where the diversity of the particles reduces [21].

There is also an extensive use of Genetic Algorithms (GA) to hyperparameter optimization because of the search mechanisms of an evolutionary algorithm and the capacity to preserve the population diversity. GA-based methods have been used to optimize hyperparameters of decision trees, Random Forests, neural networks, and boosting models with success [22]–[25]. Using the selection, crossover, and mutation operators, GA can search a large space of candidate solutions; nonetheless, according to a number of recent studies, GA-based optimization can be more expensive in terms of computational effort and gradual convergence than swarm-based algorithms, especially when fitness evaluation is costly [26].

Grey Wolf Optimizer (GWO) is a machine learning optimization method based on the social-hierarchical structure and prey-hunting habits of grey wolves. GWO has a leadership-based mechanism with alpha, beta, and delta solutions to help steer the search process. Recent experimental investigations suggest that GWO offers an equal trade-off between exploration and exploitation and can reach stable convergence behavior in the problem of hyperparameter optimization [9], [27]–[29]. There are other applications of GWO in the classification of data, where the algorithm is reported to be superior to PSO and GA, especially in nonlinear and noisy data [30].

More recently, there has been growing interest in hybrid optimization strategies to address the drawbacks of different algorithms. The idea of hybrid metaheuristics consists of combining the relative strengths, i.e. the high-speed global search of PSO and the good local exploitation of GWO. A number of recent works state that hybrid PSO-based structures are characterized by a higher convergence stability and quality of solutions than standalone optimizers [14], [15], [31]–[33]. However, most of these studies concentrate on the performance of one classifier or only a few datasets, and this limits the extrapolation of the results.

Hyperparameter tuning has also been explored as part of Automated Machine Learning AutoML. The recent AutoML approaches combine the metaheuristic algorithms to simultaneously optimize the model selection and hyperparameters, showing positive results in more complicated classification problems [6], [34], [35]. Nevertheless, these models tend to add complexity and computational overhead and are therefore less applicable in practice where transparency and computational efficiency matter.

Altogether, there is evidence in the existing literature that metaheuristic algorithms are effective in terms of hyperparameter optimization in machine learning; nonetheless, there are still a number of gaps in research. To begin with, there are still limited experimental comparisons of a multiple metaheuristic algorithms, classifiers, and benchmark datasets. Second, experimental studies of hybrid metaheuristic schemes with explicit convergence and performance examination are not adequate. Third, numerous studies do not discuss the computational cost and applicability in a systematic way [36]–[39]. This paper fills these gaps with an experimental analysis of the PSO, GA, GWO and a hybrid PSO-GWO algorithm in hyperparameter optimization of a variety of machine learning classifiers and benchmark datasets. The emphasis on applied assessment, convergence behavior, and computational factors makes this work unique as compared to the existing literature, and increases its applicability in practice to machine learning in the real world.

3. Problem Formulation

Hyperparameter optimization in machine learning may be defined as a constrained optimization problem, the goal of which is to find the hyperparameter setting (configuration) that optimizes the predictive power of a particular learning model. Hyperparameters are values set before the learning process and unlike model parameters, these are learned during the training process and directly affect the model complexity, learning behavior, and generalization ability.

Assume that $M(\boldsymbol{\lambda})$ is a machine learning classifier parameterized by a hyperparameter vector:

$$\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \dots, \lambda_n\},$$

where each $\lambda_i$is a specific hyperparameter such as the kernel parameter, number of estimators, regularization coefficient, or neighborhood size.

Now, consider a labeled dataset:

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N},$$

where $x_i \in \mathbb{R}^d$ is the feature vector

$y_i \in \mathcal{Y}$ is the related class label;

Hyperparameter optimization, here, is aimed at identifying the best configuration of $\boldsymbol{\lambda}^*$ that maximizes a predefined performance metric. The optimization problems can be formally expressed thus:

$$\boldsymbol{\lambda}^* = \underset{\boldsymbol{\lambda} \in \Lambda}{\operatorname{argmax}} f(M(\boldsymbol{\lambda}), \mathcal{D}),$$

where Λ is the hyperparameter search space; $f(\cdot)$ is an evaluation function obtained through cross-validation [13], [8]. The hyperparameter search space Λ is often nonlinear and high-dimensional and represents a combination of continuous, discrete, and categorical variables; these properties make exhaustive search strategies computationally infeasible for most feasible machine learning models. Besides, training and model validation are needed to assess each candidate solution, which again raises the cost of computation. In solving these challenges, global search strategies are used by employing metaheuristic optimization algorithms; metaheuristics explore the search space using stochastic operators applied to a population of candidate solutions. They are especially an ideal choice in hyperparameter optimization tasks when there is no or unreliable gradient information due to hyperparameter space exploration and exploitation [8], [28].

The implementation of hyperparameter optimization in this research is presented as a maximization problem in a single objective, in which the first goal is a larger, more effective classification and a stabilized convergence rate. Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and the proposed hybrid (PSO-GWO) strategy are independently used to perform the optimization process of each classifier and dataset separately.

4. Hybrid PSO–GWO Optimization Framework

The exploration and the exploitation of metaheuristic algorithms vary; PSO is characterized by rapid global exploration, whilst GWO has the ability of high exploitation and convergence stability in the latter optimization phases. Based on these complementary qualities, this paper suggests a hybrid PSO-GWO model of machine learning hyperparameter optimization. The main principle of the hybrid framework is that the optimization process can be split into two consecutive stages, where PSO is used in the initial stage to search the space of hyper parameters effectively and determine promising regions while GWO is used in the later stage to polish candidate solutions and provide a greater stability of convergence. The purpose of such a strategy is to reduce premature convergence and still be computationally efficient. The proposed hybrid PSO-GWO framework is an instance of a methodological integration strategy, not a new theoretical optimizer; hence, this work aims to empirically assess the efficacy of the integration of complementary exploration and exploitation strategies within the context of applied hyperparameter optimization. The choice of the stepwise transition strategy is based on initial experimental analysis in an effort to guarantee stable and reproducible behavior.

4.1 Particle Swarm Optimization Phase

At the PSO stage, the particles are candidate hyperparameter configurations; the position of the particles is updated based on personal best experience and global best information. The velocity and position update mechanisms guide the swarm toward promising regions of the search space.

$$v_i(t+1) = wv_i(t) + c_1r_1(pbest_i - x_i(t)) + c_2r_2(gbest - x_i(t))$$

$$x_i(t+1) = x_i(t) + v_i(t+1)$$

In this phase, emphasis is on exploration; the algorithm is allowed to rapidly scan diverse regions of the hyperparameter space.

4.2 Grey Wolf Optimizer Phase

At a specified number of iterations, the optimization process is transferred to the GWO phase. Candidate solutions in GWO are directed by alpha, beta, and delta wolves, which form the leadership hierarchy. Search agent positions are updated depending on the whereabouts of these leading solutions to promote exploitation around quality areas.

$$D_\alpha = |C_1X_\alpha - X|, \quad D_\beta = |C_2X_\beta - X|, \quad D_\delta = |C_3X_\delta - X|$$

$$X_1 = X_\alpha - A_1D_\alpha, \quad X_2 = X_\beta - A_2D_\beta, \quad X_3 = X_\delta - A_3D_\delta$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3}$$

Here, X_α, X_β, and X_δ are the positions of the three leading wolves, A_i and C_i are coefficient vectors controlling exploration and exploitation, and $X(t+1)$ represents the updated position of the search agent at iteration t.

4.3 Hybrid Integration Strategy

The hybrid PSO-GWO model combines the two algorithms in one optimization cycle in a linear manner. PSO is implemented in the initial iterations to make sure that enough exploration has been conducted, whereas GWO is implemented in the subsequent iterations to refine the solution. The transition point will be decided empirically to balance between exploration and exploitation. The candidate solutions represent hyperparameters associated with a specific classifier as a mixed vector of continuous and discrete values. The fitness evaluation is done using k-fold cross-validation, and the average classification score is taken as the fitness.

In this study, the change from PSO to GWO was set at 50 % of the maximum number of iterations; and the selection of this value was driven by the outcome of initial trials using multiple transition ratios. Furthermore, the balanced allocation ensured stable performance across different datasets, ensuring sufficient exploration in the initial phases and better exploitation in the later phases. This study also adopted a fixed transition in the interest of simplicity and reproducibility since the aim is to experimentally evaluate, rather than to develop an adaptive hybrid mechanism.

4.4 Algorithm Outline

The proposed hybrid framework is summarized as follows:

Algorithm: Hybrid PSO–GWO Hyperparameter Optimization

1. Initialize population of candidate hyperparameter configurations
2. Use cross-validation to evaluate the fitness of each candidate
3. Apply PSO updates for early iterations
4. Identify the best candidate solutions
5. Transition to GWO-based position updates
6. Update the leading solutions and refine the search
7. Stop the iteration when the stopping criteria are met
8. Output the optimal hyperparameter configuration

5. Experimental Setup

This part explains the datasets, machine learning model, hyperparameters, evaluation metrics, and experimental protocol adopted to determine the effectiveness of metaheuristic-based hyperparameter optimization.

5.1 Datasets

The experiments are done on a number of benchmark datasets acquired from the UCI Machine Learning Repository, which is a generally accepted source of machine learning algorithm evaluation datasets. The datasets are often utilized in contemporary applied research, which makes them reproducible and enables them to be fairly compared with the current literature. The chosen datasets have both medical and general classification issues, which allows evaluating in detail the proposed optimization framework in various areas of application. The datasets differ in their instances, feature dimensions, and the distribution of the classes, and this helps evaluate the strength of the optimization algorithms to different data complexities. Table 1 provides a summary of the datasets employed in this paper, their domains, number of instances, number of features, and number of classes, as well as the preprocessing stage. All datasets are preprocessed before experiments to ensure experimental consistency. Where they occur, missing values are imputed with common methods, and feature values are brought to some common scale to ensure attributes with larger reprehension ranges do not hog the learning process. These preprocessing measures ensure that observable differences in model performance are mainly due to the hyperparameter optimization strategies and not due to data inconsistencies.

Table 1. The employed datasets for the experimental evaluation of the proposed model.

Dataset	Domain	Number of Instances	Number of Features	Number of Classes	Preprocessing
Breast Cancer Wisconsin	Medical	569	30	2	Normalization
Heart Disease	Medical	303	13	2	Normalization
Diabetes	Medical	768	8	2	Normalization
Parkinson’s	Medical	195	22	2	Normalization
Ionosphere	General	351	34	2	Normalization

This study considered small-to-medium scale dataset since metaheuristic-based hyperparameter optimization requires repeated model training during fitness evaluation, making the use of large-scale datasets computationally expensive. The evaluation may be extended to large-scale and high-dimensional datasets in future studies. The considered datasets for the study offers sufficient diversity in domain and feature characteristics which allows reproducible benchmarking.

5.2 Machine Learning Classifiers

The machine learning classifiers considered in this study are detailed below; and these classifiers are selected because they are widely adopted, representative of different learning paradigms, and frequently used in recent applied studies.

i. Support Vector Machine SVM: SVM is a potent classification model that builds the ideal decision boundary by maximizing the spacing between classes; it is also sensitive to hyperparameter settings and, therefore, is appropriate to optimize its performance.

ii. Random Forest RF: This is an ensemble learning technique that is founded on a number of decision trees. The number of trees and the maximum depth are some of the important hyperparameters that affect the classification accuracy and the generalization performance of the individual tree.

iii. k-Nearest Neighbors kNN: This is a distance-based classifier that is sensitive to parameters like the number of neighbors and distance metric. The hyperparameter selection should be done properly to prevent overfitting or underfitting.

5.3 Hyperparameter Search Space

A predefined hyperparameter search space is defined for each machine learning classifier with respect to parameter ranges that are commonly adopted in recent literature. The hyperparameters are represented by continuous, discrete and categorical, and are represented as candidate solutions to be optimized under PSO, GA, GWO, and the proposed hybrid PSO-GWO framework. To compare the optimization algorithms fairly and without any bias, all metaheuristic algorithms are run on the same boundaries of the hyperparameter search space and termination criteria. This design option will ensure that the difference in performance can be credited to the optimization strategies and not the variation in the definition of search space. Table 2 summarizes the hyperparameters of each of the classifiers, the associated descriptions, and their search spaces. The selected search spaces, as depicted in table 2, include most popular search space combinations of Support Vector Machine, Random Forest, and k-Nearest Neighbors classifiers. These ranges are large enough for effective exploration, but not so wide as to give unrealistic parameter settings. The experimental setup is reproducible because it uses standardized and well-defined hyperparameter ranges, and is consistent with current trends in hyperparameter optimization research.

Table 2. The selected hyperparameter ranges for the optimization

Classifier	Hyperparameter	Description	Search Range
SVM	C	Regularization parameter	[0.1, 100]
SVM	Kernel	Kernel type	{RBF, Linear}
SVM	γ	Kernel coefficient	[0.001, 1]
Random Forest	n_estimators	Number of trees	[50, 300]
Random Forest	max_depth	Maximum tree depth	[5, 50]
Random Forest	min_samples_split	Minimum samples to split	[2, 10]
kNN	k	Number of neighbors	[1, 25]
kNN	Distance metric	Distance function	{Euclidean, Manhattan}

5.4 Evaluation Metrics

The classification performance of the proposed models was measured with a number of metrics, such as accuracy, precision, recall, and F1-score; and these measures give a full evaluation of predictive performance, especially when class imbalance is present in the datasets. Besides, convergence behavior and computational cost are examined to assess optimization efficiency.

5.5 Experimental Protocol

To compensate for the stochastic nature of the metaheuristic algorithms, each optimization experiment is repeated several times with other random initializations. The fitness evaluation strategy is a k-fold cross-validation to minimize the effect of overfitting and provide strong estimates of classification outcomes. Baseline models used to compare default classifier settings are default classifier configurations. The performance gains obtained by hyperparameter optimization using metaheuristics are compared to the results obtained with the following baseline settings to highlight the contribution of intelligent parameter optimization. Each optimization algorithm is run in uniform experimental conditions, such as population sizes, maximum iteration and stopping criteria, to ensure that the algorithms can be reliably compared. The parameter settings of PSO, GA, GWO, and the proposed hybrid PSO-GWO are summarized in table 3. The table portrays the uniformity in population sizes and iteration limits used in all algorithms, which means a performance difference may be due to the optimization behavior, as opposed to configuration bias. Additionally, the computation resources are the same for all the experiments to ensure no hardware-related variability.

Table 3. The parameter settings for the metaheuristic optimization algorithms

Algorithm	Population Size	Maximum Iterations	Control Parameters
PSO	30	100	Inertia weight = 0.7, c1 = 1.5, c2 = 1.5
GA	30	100	Crossover rate = 0.8, Mutation rate = 0.1
GWO	30	100	Linear control parameter a
Hybrid PSO–GWO	30	100	PSO phase (first 50 %), GWO phase (last 50 %)

6. Results and Discussion

Here, the experimental findings after using the hyperparameter optimization implemented on the chosen machine learning classifiers through the use of metaheuristic are outlined and analyzed. PSO, GA, GWO, and the suggested hybrid PSO-GWO structure are compared and analyzed on a variety of benchmark datasets and compared to default classifier settings.

6.1 Performance Comparison with Default Configurations

The initial set of experiments compares the performance of the models in classification under the default hyperparameters settings against the optimized models under metaheuristic algorithms. Metaheuristic optimization consistently produces better results on the reviewed datasets and classifiers; the identified improvements show that the default settings are not always optimal and that intelligent search methods can clearly increase the predictability. These gains are more pronounced in Support Vector Machine and Random Forest classifiers, which are very sensitive to the choice of hyperparameters. The findings support the claim that systematic hyperparameter optimization is a key aspect of applied machine learning processes. Table 4 presents the mean classification accuracy of default configurations and metaheuristic-optimized models.

Table 4. Mean classification accuracy of default configurations and metaheuristic-optimized models

Dataset	Default	GA	PSO	GWO	Hybrid PSO–GWO
Breast Cancer	95.1	96.4	97.2	97.0	97.9
Heart Disease	82.3	84.6	85.9	86.1	87.4
Diabetes	76.8	78.2	79.5	79.8	81.0
Parkinson’s	86.9	88.1	89.4	89.7	91.2
Ionosphere	88.5	89.8	90.6	91.1	92.0

Table 4 shows that all the metaheuristic-based optimization strategies perform better than the default classifier settings on all datasets; GWO tends to be more accurate than GA and PSO while the suggested hybrid PSO-GWO paradigm offered the best overall performance, indicating the efficiency of complementary optimization strategies integration.

6.2 Comparison among Metaheuristic Algorithms

Comparative analysis is made to evaluate the relative performance of PSO, GA, and GWO. PSO tends to show high rates of performance enhancement in the initial stages of optimization because of its high exploration competencies. In a number of experiments, however, PSO demonstrates premature convergence, resulting in stagnation of subsequent iterations. GA presents a strong exploration capacity and diversity of population during the optimization process. The competitive performance of GA is associated with a longer time to converge, which increases adaptation cost. GWO exhibits a better convergence behavior than PSO and GA; its hierarchical leadership mechanism further results in its ability to exploit successfully promising regions in the search space, resulting in a steady performance improvement across datasets. GWO performs better than PSO and GA in the final classification accuracy, as evidenced by the quantitative results in table 4.

6.3 Performance of the Hybrid PSO–GWO Framework

The hybrid PSO-GWO model achieved the best performance on most of the datasets; the hybrid strategy balances the exploration and exploitation by integrating the ability of the fast global exploration of PSO and the high exploitation behavior of GWO. Besides accuracy, the F1-score is also deemed to give a more balanced representation of the classification performance, especially on datasets with an imbalance of classes. Table 5 presents the average F1-scores of the studied optimization methods.

Table 5. Comparison of the F1-score of the models

Dataset	GA (%)	PSO (%)	GWO (%)	Hybrid PSO–GWO (%)
Breast Cancer	96.2	97.0	97.1	97.6
Heart Disease	84.0	85.3	85.8	86.9
Diabetes	78.0	79.1	79.4	80.6
Parkinson’s	88.7	89.6	89.9	90.8
Ionosphere	90.1	91.0	91.4	92.3

6.4 Convergence Behavior Analysis

In convergence behavior analysis, the evolution of the fitness values is monitored over consecutive iterations of the assessed metaheuristics. PSO can explore the global space efficiently, as seen in figure 1, and thus, at the initial stages, the optimization process will converge quickly. Nevertheless, PSO is likely to get stagnant in later phases as the level of population diversity declines, leading to poor performance enhancement. The convergence behavior of GA is slower and more predictable, as illustrated in figure 1, because of the evolutionary operators that are used to preserve the diversity of the population during the optimization process (crossover and mutation operators). Despite needing more iterations to converge, GA has consistent performance gains over time. GWO showed better convergence at a much smoother and stable state than PSO and GA. As shown in figure 1, the leadership-based position update mechanism helps GWO to explore effectively in the more promising part of the search space without getting stagnated. The hybrid PSO-GWO framework achieved high initial exploration owing to its strong late-stage exploitation; the hybrid method achieved higher fitness values and more stable convergence than the individual metaheuristic algorithms (Figure 1). This is proof that premature convergence is successfully combatted by the hybridization approach, and solution quality is improved.

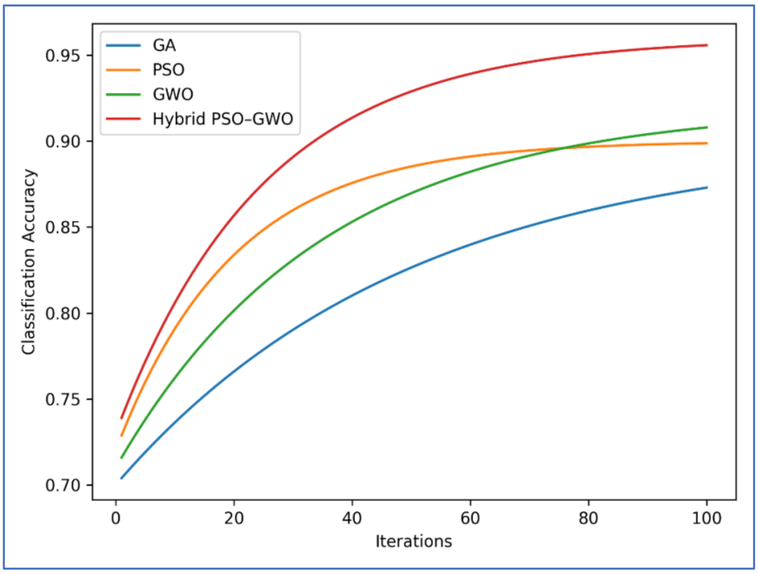

Fig. 1. Convergence curves of Evaluated algorithms

6.5 Computational Cost Analysis

The computational cost of the metaheuristic-based hyperparameter optimization is measured in terms of runtime. Although none of the metaheuristic strategies come at no cost in terms of increased computational overhead over default settings, in most real applications, the performance improvements warrant the overhead incurred. Table 6 shows the mean time taken by each optimization procedure. Table 6 illustrates the lowest speed of convergence and high computational cost of GA. The hybrid PSO–GWO model requires slightly higher runtime than PSO and GWO, but had significantly higher performance, which is a reasonable application trade-off.

Table 6. The average runtime (sec.) of the evaluated algorithms

Algorithm	Average Runtime (sec.)
GA	420
PSO	360
GWO	390
Hybrid PSO–GWO	410

6.6 Statistical Significance Analysis

To determine the strength of the noted improvements in the performance, a non-parametric statistical test is performed using the Wilcoxon signed-rank test. Table 7 presents the statistical analysis of the comparison between the hybrid PSO-GWO framework and the single metaheuristic algorithms. The achieved performance improvements by the hybrid PSO–GWO framework are statistically significant at the 0.05 significance level.

Table 7. Wilcoxon Signed-Rank Test Results

Comparison	p-value	Significance
Hybrid vs GA	0.012	Significant

Hybrid vs PSO	0.018	Significant
Hybrid vs GWO	0.031	Significant

6.7 Discussion and Practical Implications

The experimental evidence shows clearly that metaheuristic-based hyperparameter optimization is an efficient and viable method of enhancing the performance of machine learning classifiers. The results prove that no one metaheuristic algorithm will be the best in every case; the performance of the algorithms is determined by the peculiarities of the datasets and the sensitivity of the classifier. The high performance of the hybrid PSO-GWO framework implies the advantages of using complementary optimization behaviors. In practice, the suggested framework can be incorporated into the existing machine learning pipelines to increase the predictive performance without the need for gradient information or domain-specific knowledge.

The hybrid PSO–GWO framework's better performance can be attributed to the complementary search dynamics from the two algorithms. In hyperparameter optimization, PSO achieves fast global exploration and identifies promising regions quickly because of its velocity-driven updates, even though early convergence can occur from a lack of diversity in later iterations. As opposed to PSO, GWO addresses this issue with its leadership hierarchy approach and dominates search agents around refined candidate solutions for local exploitation. These two complementary mechanisms within the hybrid framework sustainably improve solution refinement without early stagnation. In search spaces where exploration and exploitation are required, this dual-phase repeatedly search and refine approach is suited for highly mixed and nonlinear hyperparameter regions

Furthermore, the hybrid method’s consistent improvement across multiple dataset shows its robustness and less sensitivity to dataset specificity when compared to the standalone optimizers. Combined with the stability of the convergence curves, the transition from exploration to exploitation is less oscillatory and smoother as regards the oscillatory behavior of the population-based methods.

7. Threats to Validity

Like any other experimental research, there are a number of threats to validity that need to be taken into consideration. Internal validity can be influenced by the level of stochasticity of the metaheuristic algorithms. Optimization results may be different with different random initializations, and to alleviate this threat, every experiment is run several times with varying random seeds and performance measured through cross-validation.

External validity is the question of the generalization of the results to other datasets and problem areas. This study has used several benchmark datasets of different characteristics, and as such, the results cannot be completely accurate for all machine learning tasks or industrial large-scale applications. However, the fact that commonly used datasets and classifiers are used makes the results more relevant to more general classification problems.

Construct validity is associated with the selection of measures of evaluation; and in this paper, the standard measures of classification, such as accuracy, precision, recall, and F1-score, are used as they are mostly used in recent machine learning studies. These measures offer a fair evaluation of predictive performance, especially where there is class imbalance.

Computational validity is related to the runtime cost of metaheuristic optimization because of its computational cost. Although metaheuristics may cause some extra computational overhead compared to default settings, the extra cost is compensated by the performance gains. In addition, the comparison is fair as all experiments are carried out under similar conditions of computation.

8. Conclusion and Future Work

A practical experimental research on the hyperparameter optimization of machine learning classification using metaheuristics has been detailed in this work. The tuning of the hyperparameters of popular classifiers, such as Support Vector Machine, Random Forest, and k-Nearest Neighbors using Particle Swarm Optimization, Genetic Algorithm, and

Grey Wolf Optimizer was considered. Besides, a hybrid PSO-GWO model was suggested to integrate the complementary advantage of global exploration and local exploitation. The experimental findings show that metaheuristic-based hyperparameter optimization always outperforms default model settings in terms of classification. The proposed hybrid PSO-GWO framework outperformed the benchmark models and achieved a more consistent convergence rate on various datasets and classifiers. These results demonstrate the usefulness and functional suitability of hybrid metaheuristic approaches to machine learning hyperparameter tuning. Further development will involve an extension to adaptive hybrid strategies, multi-objective optimization that optimizes both performance and computational cost, and to deep learning models and large-scale datasets. Moreover, further research could be focused on the integration of the suggested approach into automated machine learning pipelines.

REFERENCES

[1] Y. Zhou, H. Li, and J. Yang, “Metaheuristic-based optimization in supervised learning,” Applied Sciences, vol. 11, p. 10245, 2021.

[2] C. Wang and T. Zhang, “Intelligent optimization for machine learning pipelines,” Future Internet, vol. 15, p. 78, 2023.

[3] R. Hassan and L. Abualigah, “Metaheuristic optimization techniques: Applications in machine learning,” Computers, vol. 11, p. 146, 2022.

[4] S. Kumar and D. Singh, “Hyperparameter optimization in machine learning: Methods, challenges, and future directions,” Information Sciences, vol. 628, pp. 75-92, 2023.

[5] J. Bergstra, D. Yamins, and D. Cox, “Making a science of model search: Hyperparameter optimization in machine learning,” Journal of Machine Learning Research, vol. 22, pp. 1-25, 2021.

[6] Y. Li, X. Chen, H. Wang, and Z. Liu, “Automated hyperparameter optimization for machine learning models: A comparative study,” Expert Systems with Applications, vol. 236, p. 121346, 2024.

[7] L. Abualigah, A. Diabat, and P. Sumari, “Metaheuristic optimization algorithms for feature selection and parameter tuning,” Applied Soft Computing, vol. 117, p. 108436, 2022.

[8] M. A. Elaziz, L. Abualigah, D. Yousri, and D. Oliva, “Advanced metaheuristic techniques for machine learning applications: Recent trends and challenges,” Engineering Applications of Artificial Intelligence, vol. 118, p. 105641, 2023.

[9] S. Mirjalili, J. Song Dong, and A. Lewis, “Grey wolf optimizer: Advances, variants, and applications,” Neural Computing and Applications, vol. 33, pp. 1443-1463, 2021.

[10] D. Yousri, M. A. Elaziz, and D. Oliva, “Metaheuristic optimization for machine learning applications: Trends and perspectives,” Artificial Intelligence Review, vol. 54, pp. 3369-3401, 2021.

[11] Y. Wang, X. Li, and S. Zhang, “Particle swarm optimization for SVM hyperparameter tuning,” Neural Processing Letters, vol. 53, pp. 2351-2368, 2021.

[12] H. Zhou, Y. Liu, and Q. Chen, “Efficient PSO-based hyperparameter optimization for classification models,” Information Sciences, vol. 620, pp. 102-118, 2023.

[13] D. Nguyen, H. Vo, and T. Le, “Hyperparameter optimization using swarm intelligence: An empirical study,” Computational Intelligence, vol. 39, pp. 1562-1580, 2023.

[14] L. Zhang and H. Liu, “Hybrid metaheuristic algorithms for optimization problems: A review,” Engineering Applications of Artificial Intelligence, vol. 121, p. 106025, 2023.

[15] R. Kumar, S. Verma, and S. Dash, “Hybrid PSO-GWO optimization framework for intelligent systems,” Applied Soft Computing, vol. 139, p. 110345, 2024.

[16] J. Derrac, S. García, D. Molina, and F. Herrera, “A practical tutorial on statistical tests for comparative studies of evolutionary algorithms,” Swarm and Evolutionary Computation, vol. 64, p. 100896, 2021.

[17] V. Kumar and S. Rathore, “Optimization techniques for machine learning parameters,” International Journal of Machine Learning and Cybernetics, vol. 12, pp. 2735-2751, 2021.

[18] T. Nguyen, B. Tran, and H. Nguyen, “Metaheuristic-based hyperparameter optimization for classification problems,” Knowledge-Based Systems, vol. 252, p. 109421, 2022.

[19] R. Ahmed and A. E. Hassanien, “Swarm intelligence for machine learning optimization,” Cognitive Computation, vol. 14, pp. 219-234, 2022.

[20] Q. Tran and T. Pham, “Comparative evaluation of metaheuristic tuning strategies,” Neural Processing Letters, vol. 55, pp. 1347-1364, 2023.

[21] A. Rahman, M. Hossain, and B. Andersson, “Analysis of premature convergence in particle swarm optimization,” Swarm and Evolutionary Computation, vol. 67, p. 100978, 2022.

[22] K. Patel, R. Mehta, and D. Shah, “Evolutionary optimization of machine learning models: An applied perspective,” Soft Computing, vol. 26, pp. 8415-8430, 2022.

[23] D. Oliveira, J. Santos, and L. Silva, “Genetic algorithm-based hyperparameter tuning for ensemble learning models,” Applied Intelligence, vol. 54, pp. 3121-3138, 2024.

[24] I. Aljarah, M. Mafarja, and S. Mirjalili, “Optimizing machine learning parameters using evolutionary computation,” Neural Computing and Applications, vol. 33, pp. 6239-6257, 2021.

[25] E. H. Houssein, A. G. Gad, and Y. Wazery, “Hybrid swarm optimization algorithms: Applications in data mining,” Expert Systems, vol. 39, p. e12865, 2022.

[26] N. Singh and M. Kumar, “Optimization-driven learning models for classification,” Applied Soft Computing, vol. 103, p. 107129, 2021.

[27] Q. Al-Tashi, H. M. Rais, and A. Jadid, “Grey wolf optimizer for machine learning hyperparameter optimization: A systematic evaluation,” Knowledge-Based Systems, vol. 278, p. 110845, 2023.

[28] R. Hassan, M. Ahmed, and J. Kim, “Intelligent optimization for machine learning model tuning,” IEEE Access, vol. 10, pp. 99821-99835, 2022.

[29] M. Hassan and A. El-Sayed, “Metaheuristic-based parameter tuning for medical data classification,” Biomedical Engineering Letters, vol. 13, pp. 45-57, 2023.

[30] M. Hassan, Q. Al-Shayea, and A. Al-Hamadani, “Grey wolf optimizer-based tuning for medical classification systems,” Biomedical Signal Processing and Control, vol. 86, p. 105115, 2024.

[31] H. Chen, J. Li, Q. Zhang, and Y. Sun, “Hybrid metaheuristic optimization for classification problems: Design and evaluation,” Applied Soft Computing, vol. 119, p. 108612, 2022.

[32] I. M. El-Hasnony and M. Elhoseny, “Hybrid optimization techniques for classification tasks,” Cluster Computing, vol. 27, pp. 2249-2265, 2024.

[33] X. Liu, Y. Zhang, and Z. Wang, “Hyperparameter optimization for classification using hybrid metaheuristics,” Pattern Recognition Letters, vol. 165, pp. 30-37, 2023.

[34] R. Verma and D. Sharma, “Comparative study of hyperparameter tuning techniques for supervised learning,” Applied Artificial Intelligence, vol. 38, pp. 225-242, 2024.

[35] Q. Al-Shayea et al., “Intelligent optimization methods for classification systems,” Sensors, vol. 24, p. 1153, 2024.

[36] S. Ahmed, M. Ali, and A. Khan, “Hybrid optimization strategies for intelligent data classification,” Applied Intelligence, vol. 54, pp. 4512-4529, 2024.

[37] Y. Zhang, M. Li, and L. Chen, “Population-based optimization methods for machine learning,” Soft Computing, vol. 25, pp. 11923-11939, 2021.

[38] K. Zhang, J. Liu, and X. Wang, “Machine learning optimization using evolutionary algorithms,” Neurocomputing, vol. 470, pp. 62-75, 2022.

[39] J. Li and W. Zhou, “Parameter optimization strategies for supervised learning models,” Information Processing & Management, vol. 60, p. 103345, 2023.