Vol. 4 No. (1) (June 2026) 225-238

Journal of Data Insights

http://journalnew.unimus.ac.id/index.php/jodi

Hyperparameter Optimization of Random Forest Using Grey Wolf Optimization for Heart Disease Classification

Ratih Khotimahtus Sa`diyah[1], Muhammad Sam'an[*2], Safuan[3], Mustafa Mat Deris[4]
[1,2,3]Department of Informatics, Universitas Muhammadiyah Semarang, Indonesia.
[4]Faculty of Business, Management and Information Technology, Universiti Muhammadiyah Malaysia, Malaysia.

DOI: https://doi.org/10.26714/jodi.v4i1.1196

Article Info

Article history:
Received June 04, 2026
Revised June 25, 2026
Accepted June 27, 2026

Keywords:
Cleveland Heart Disease Dataset; Grey Wolf Optimization; Heart Disease Prediction; Hyperparameter Optimization; Machine Learning; Random Forest.

Abstract

Cardiovascular disease remains one of the leading causes of death worldwide, making predictive models important to support early heart disease detection. Random Forest is widely used for heart disease classification, but its performance can be affected by hyperparameter selection. This study focuses on applying Grey Wolf Optimization (GWO) to selected Random Forest hyperparameters and evaluating the optimized model through a direct comparison with a baseline Random Forest model on the same testing dataset, supported by statistical verification. The dataset used is the Cleveland Heart Disease Dataset, consisting of 303 patient records, 13 predictor attributes, and one target attribute. The research stages include data preparation, preprocessing, stratified data splitting with an 80:20 ratio, hyperparameter optimization using GWO, and model evaluation. The GWO process uses the average F1-score from 5-fold cross-validation on the training set as the fitness value. Model performance is evaluated using accuracy, precision, recall, F1-score, AUC-ROC, confusion matrix analysis, and the exact McNemar test. The results show that the GWO-RF model obtains higher descriptive evaluation values than the baseline RF model, with accuracy increasing from 88.52% to 93.44%, precision from 81.82% to 90.00%, F1-score from 88.52% to 93.10%, and AUC-ROC from 95.13% to 96.86%, while recall remains at 96.43%. However, the exact McNemar test produces a p-value of 0.25, indicating that the difference is not statistically significant. Therefore, the improvement is interpreted as a descriptive performance gain rather than a statistically significant improvement.

✉ Correspondence Address:
E-mail: muhammad92sam@unimus.ac.id

e-ISSN: 2988 - 2109

1. INTRODUCTION

Heart disease remains one of the leading causes of death worldwide and continues to pose a major challenge to healthcare systems. According to the World Health Organization (WHO), cardiovascular diseases account for approximately 17.9 million deaths each year, representing 32% of all global deaths. More than 75% of cardiovascular disease-related deaths occur in low- and middle-income countries. Heart disease is influenced by various risk factors, including hypertension, diabetes mellitus, obesity, smoking, and physical inactivity. From a machine learning perspective, these heterogeneous clinical factors create a classification problem in which the model must distinguish between patients with and without heart disease based on multiple clinical attributes that may have overlapping patterns across classes. Therefore, prediction models are needed not only to support early detection but also to provide a systematic approach for analyzing clinical data [1].

Machine learning has been widely applied in heart disease prediction because it can learn patterns from clinical data and produce classification outputs based on patient attributes. Earlier studies employed conventional classification algorithms, such as Logistic Regression, Decision Tree, Naïve Bayes, K-Nearest Neighbor, and Support Vector Machine [2]. These methods are relatively simple and interpretable, but their performance can be affected by the complexity of clinical feature interactions. Ensemble learning methods, including Random Forest, Extreme Gradient Boosting (XGBoost), and Light Gradient Boosting Machine (LightGBM), have also been adopted because they combine multiple learners and can capture more complex data patterns [3], [4]. However, ensemble models usually require appropriate hyperparameter settings to control model complexity and prediction behavior. In addition, deep learning approaches, such as Convolutional Neural Network–Long Short-Term Memory (CNN–LSTM), have been applied to capture nonlinear relationships in clinical data [5], [6]. Nevertheless, deep learning models may require more complex model structures and larger computational resources than conventional machine learning models.

As classification algorithms continue to evolve, research on heart disease prediction has increasingly focused on hyperparameter optimization to improve model performance. Various optimization methods have been applied, including Grid Search, Random Search, Hyperband, Bayesian Optimization, Genetic Algorithm, Particle Swarm Optimization, Bat Algorithm, Cuckoo Search, and Grey Wolf Optimization (GWO). Previous studies have shown that selecting appropriate hyperparameter configurations influences the performance of heart disease prediction models [3], [7], [8]. Grid Search is straightforward because it evaluates predefined parameter combinations, but it can become computationally demanding when the search space is large. Random Search and probabilistic optimization methods can reduce the number of evaluated configurations, but the search result still depends on the defined search strategy and parameter space. Metaheuristic algorithms provide an alternative approach because they use population-based or nature-inspired mechanisms to explore the search space. GWO is one of the metaheuristic algorithms that uses the social hierarchy and hunting behavior of grey wolves to balance exploration and exploitation during the search process [9]. This characteristic makes GWO relevant for Random Forest hyperparameter optimization, where several parameters need to be searched simultaneously within a defined search space.

Previous studies have applied optimization-based approaches to improve heart disease classification models [3], [7], [8]. However, several methodological aspects still require further attention. Some studies focus mainly on the final optimized performance without sufficiently discussing the search space, the role of the optimized hyperparameters, or the comparison with the baseline model under the same evaluation setting. In addition, studies that specifically examine GWO for Random Forest hyperparameter optimization in heart disease prediction remain relatively limited compared with studies that apply GWO to other classification algorithms or combine it with different optimization approaches [10], [11], [12]. Therefore, the research gap in this study is not only the limited use of GWO with Random Forest, but also the need for a clearer experimental evaluation that explains the optimization design and compares the optimized model with the baseline model using the same testing dataset.

Random Forest is used in this study because it is an ensemble learning algorithm that constructs multiple decision trees and combines their predictions through majority voting. Its performance can be influenced by hyperparameter settings, such as n_estimators, max_depth, min_samples_split, and min_samples_leaf. These hyperparameters are related to the number of trees, tree depth, node splitting process, and the minimum number of samples in leaf nodes. If these parameters are not properly configured, the model may produce different levels of complexity and different prediction results. Therefore, hyperparameter optimization is applied to search for a suitable configuration for the Random Forest model.

Based on the identified gap, this study applies Grey Wolf Optimization to optimize selected Random Forest hyperparameters for heart disease classification using the Cleveland Heart Disease Dataset. The objective of this study is to evaluate the performance of the GWO-optimized Random Forest model and compare it with the baseline Random Forest model using the same testing dataset. The methodological contribution of this study lies in the application of GWO for optimizing selected Random Forest hyperparameters using a defined search space and a fitness function based on the average F1-score from 5-fold cross-validation. The experimental contribution lies in the comparative evaluation between the baseline RF and GWO-RF models using the same testing data, supported by classification metrics, confusion matrix analysis, AUC-ROC, and statistical testing. Through this evaluation, this study provides a clearer description of how GWO-based hyperparameter optimization affects Random Forest performance in heart disease classification.

2. METHOD

This study applies Grey Wolf Optimization (GWO) for Random Forest hyperparameter optimization in heart disease classification. The proposed methodology consists of several stages, including dataset collection, initial dataset analysis, data preparation, data preprocessing, Random Forest model development, hyperparameter optimization using GWO, and model evaluation. The initial dataset analysis is conducted to describe the class distribution, missing values, imbalance ratio, and descriptive statistics before preprocessing. The data preparation stage includes duplicate data checking, validation of selected clinical attribute value ranges, and transformation of the target variable into a binary classification label. The data preprocessing stage consists of splitting the dataset into training and testing sets, handling missing values, and standardizing numerical features. A Random Forest model is then developed as the baseline model, while GWO is applied to optimize selected Random Forest hyperparameters. The optimized model is evaluated on the same testing set as the baseline model to support a paired comparison between both models. The proposed research workflow is illustrated in Figure 1.

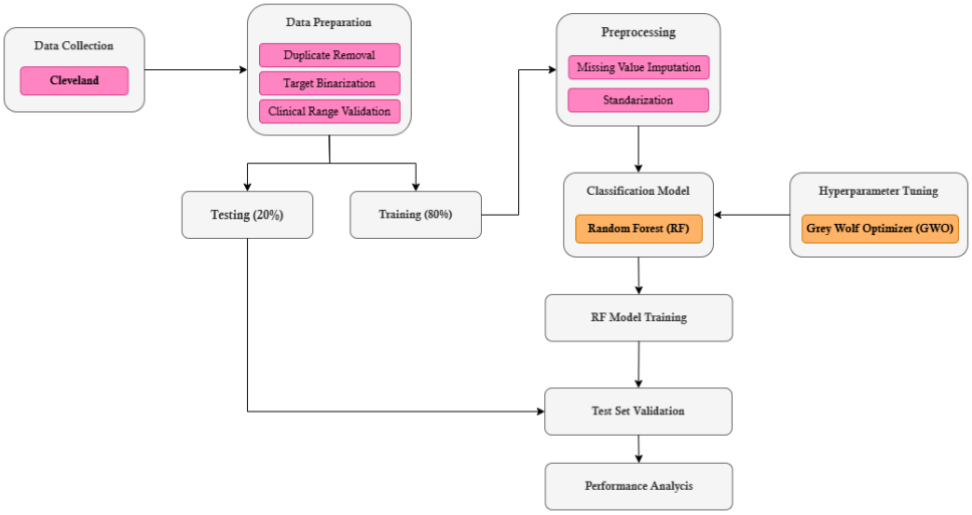

Figure 1. Overview of the Proposed Methodology for Heart Disease Prediction

2.1 Dataset

This study uses the Cleveland Heart Disease Dataset obtained from the UCI Machine Learning Repository. The dataset is one of the publicly available datasets frequently used in heart disease prediction studies. It contains various clinical attributes used to develop and evaluate machine learning models, making it suitable for heart disease classification research [13].

The dataset consists of 303 patient records, including 13 predictor attributes and 1 target attribute. The predictor attributes include demographic characteristics, physical examination results, clinical parameters, and electrocardiographic findings used for heart disease classification. A description of the attributes used in this study is presented in Table 1.

Table 1. Description of the Cleveland Heart Disease Dataset Attributes

No	Feature	Description
1	Age	Age of the patient (years)
2	Sex	Patient gender (1 = male, 0 = female)
3	CP	Chest pain category
4	Trestbps	Resting blood pressure (mmHg)
5	Chol	Serum cholesterol (mg/dL)
6	Fbs	Fasting blood sugar >120 mg/dL
7	Restecg	Resting electrocardiographic results
8	Thalach	Maximum heart rate achieved
9	Exang	Exercise-induced angina
10	Oldpeak	ST depression induced by exercise
11	Slope	Slope of the peak exercise ST segment
12	Ca	Number of major vessels colored by fluoroscopy
13	Thal	Thalassemia status
14	Target	Presence of heart disease

Before data preprocessing, an initial analysis was conducted to examine the class distribution, missing values, imbalance ratio, and descriptive statistics of the dataset. The dataset used in this study consists of 303 patient records with 13 predictor attributes and one target attribute. Before binary transformation, the target attribute consists of 164 records in class 0, 55 records in class 1, 36 records in class 2, 35 records in class 3, and 13 records in class 4. After binary transformation, class 0 is defined as no heart disease, while classes 1–4 are combined as heart disease. The class distribution after binary transformation is presented in Table 2.

Table 2. Class Distribution after Binary Transformation

Class	Number of Records	Percentage
No heart disease	164	54.13%
Heart disease	139	45.87%

Based on Table 2, the no heart disease class consists of 164 records, while the heart disease class consists of 139 records. The difference between the two classes is 25 records. The imbalance ratio of 1.18:1 shows the comparison between the majority and minority classes in the dataset.

The missing value analysis shows that missing values are found only in the ca and thal attributes. The ca attribute has 4 missing values, or 1.32% of the total data, while the thal attribute has 2 missing values, or 0.66% of the total data. The other attributes do not contain missing values. The missing value statistics are presented in Table 3. This information is used as a basis for the data preprocessing stage, particularly in the missing value imputation process.

Table 3. Missing Value Statistics

Attribute	Number of Missing Values	Percentage
ca	4	1.32%
thal	2	0.66%
Other attributes	0	0.00

Descriptive statistics are used to provide an initial overview of the numerical clinical attributes in the dataset. The attributes presented include age, trestbps, chol, thalach, and oldpeak. Based on the analysis, the patients' ages range from 29 to 77 years, with an average of 54.44 years. Resting blood pressure ranges from 94 to 200 mmHg, with an average of 131.69 mmHg. Serum cholesterol ranges from 126 to 564 mg/dL, with an average of 246.69 mg/dL. Maximum heart rate ranges from 71 to 202, with an average of 149.61. In addition, oldpeak ranges from 0 to 6.20, with an average of 1.04. The descriptive statistics of the numerical clinical attributes are presented in Table 4.

Table 4. Descriptive Statistics of Numerical Clinical Attributes

Attribute	Mean	Standard Deviation	Minimum	Maximum
age	54.44	9.04	29.00	77.00
trestbps	131.69	17.60	94.00	200.00
chol	246.69	51.78	126.00	564.00
thalach	149.61	22.88	71.00	202.00
oldpeak	1.04	1.16	0.00	6.20

2.2 Data preparation

The data preparation stage is conducted before the modeling process to ensure that the dataset is ready for the next stage. This process begins with duplicate data checking. The results show that no duplicate records are found; therefore, all data are retained for the subsequent stage. After that, the value ranges of several clinical attributes, namely trestbps, chol, thalach, and oldpeak, are examined. The results show that the values of these attributes are within the expected ranges; therefore, no value adjustment is required.

The final step in data preparation is transforming the target attribute into a binary classification label. The target value of 0 is retained as the no heart disease class, while target values 1–4 are combined into the heart disease class. This transformation is performed because this study focuses on classifying the presence or absence of heart disease in patients [13].

2.3 Data preprocessing

The data preprocessing stage is performed after the data preparation stage to prepare the dataset before the model training and testing processes. The dataset is divided into training and testing sets using the stratified train-test split method with an 80:20 ratio. This ratio is used to provide a separate testing set, so that model performance can be evaluated on data that are not used during the training process. The stratified approach is applied to preserve the class proportion in both the training and testing sets so that the class distribution remains consistent with the original dataset [10].

The 80:20 train-test split is used as the main final evaluation scheme because this study compares the baseline RF and GWO-RF models using the same testing set. This setting allows both models to be evaluated on identical testing samples and supports paired prediction comparison. Meanwhile, cross-validation is not used as the final evaluation scheme, but it is applied during the hyperparameter optimization process on the training set, as explained in Section 2.5. Since the dataset size is relatively limited, the use of a single testing set may still have limitations in estimating model

performance. Therefore, this limitation is considered in the interpretation of the results and can be addressed in future studies using broader validation strategies

After the dataset is split, missing values are handled based on the missing value statistics presented in Table 3. Missing values are found only in the ca and thal attributes. The ca attribute is imputed using the median value, while the thal attribute is imputed using the mode value. The imputation values are calculated from the training data and then applied to both the training and testing data. This procedure is performed to prevent information from the testing data from being used during the model training process, which may cause data leakage [4].

The next step is feature standardization using the StandardScaler method. Although Random Forest is generally less affected by feature scale because it is based on decision tree structures, standardization is still applied in this study as part of a consistent preprocessing procedure. The dataset contains numerical attributes with different value ranges, such as chol, trestbps, thalach, and oldpeak. Therefore, StandardScaler is used to place the numerical attributes on a comparable scale before the modeling process. In this study, standardization is not positioned as the main factor affecting the performance of the Random Forest model, but as a preprocessing step to maintain consistent data treatment for both the training and testing sets. The standardization process is performed using Eq. (1).

$$z = \frac{x-\mu}{\sigma} \quad (1)$$

where z denotes the standardized feature value, x is the original feature value, μ represents the mean of the feature calculated from the training data, and σ denotes the standard deviation of the feature calculated from the training data. The values of μ and σ obtained from the training data are then used to standardize both the training and testing data. This procedure is performed to maintain preprocessing consistency and to prevent information from the testing data from being used during the model training process [10], [11].

2.4 Random forest

Random Forest is an ensemble learning algorithm that constructs multiple decision trees using bootstrap sampling and random feature selection at each node-splitting process. The final prediction is obtained through a majority voting mechanism across all decision trees. This approach enables Random Forest to produce a more stable model, achieve good generalization performance, and reduce the risk of overfitting compared with a single decision tree [14].

In this study, Random Forest is used as the baseline model and as the model optimized using GWO. To support reproducibility, several implementation parameters are explicitly defined. The implementation parameters of Random Forest used in this study are presented in Table 5.

Table 5. Random Forest Implementation Parameters

Parameter	Values
criterion	gini
bootstrap	True
random_state	42
max_features	sqrt

For each decision tree, attribute selection is performed using the Gini Index as the splitting criterion. The attribute that produces the lowest Gini Index value is selected because it indicates a lower level of impurity after the data splitting process. The Gini Index is calculated using Eq. (2).

$$Gini = 1 - \sum_{i=1}^{C} p_i^2 \quad (2)$$

where p_i denotes the probability of samples belonging to class i, and C represents the total number of classes. A lower Gini Index value indicates a lower level of impurity, resulting in a more homogeneous data partition at a given node.

The performance of Random Forest is influenced by the hyperparameter configuration used during the model construction process. The selection of hyperparameter configurations can affect the model's generalization capability and complexity. Therefore, this study applies GWO for hyperparameter optimization to identify a hyperparameter configuration obtain a better hyperparameter configuration. The hyperparameter optimization process is described in the following subsection.

2.5 Hyperparameter optimization using grey wolf optimization

GWO is a metaheuristic optimization algorithm inspired by the social hierarchy and hunting behavior of grey wolves [9]. The algorithm consists of four hierarchical levels: alpha (α), beta (β), delta (δ), and omega (ω). During the optimization process, the alpha, beta, and delta wolves represent the three solutions with the highest fitness values and are used to update the positions of the remaining wolves in each iteration. This mechanism enables GWO to perform exploration and exploitation within the search space to identify a hyperparameter configuration that improves model performance [9].

In this study, each wolf represents one combination of Random Forest hyperparameters, namely n_estimators, max_depth, min_samples_split, and min_samples_leaf. The search ranges are determined as part of the experimental design by considering the function of each hyperparameter and the computational cost during repeated evaluation using 5-fold cross-validation. The n_estimators range of 10–500 is used to evaluate the number of trees from a smaller to a larger number, while still considering the computational cost during the optimization process. The max_depth range of 1–50 is used to evaluate trees with varying depth levels without allowing the trees to grow with unlimited depth. The min_samples_split range of 2–20 is used to control the minimum number of samples required for node splitting, while the min_samples_leaf range of 1–20 is used to control the minimum number of samples in each leaf node so that the formation of leaves with very few samples can be limited.

Meanwhile, the max_features parameter is fixed to sqrt, and class_weight is fixed to balanced for all candidate solutions. Thus, the GWO process optimizes only four Random Forest hyperparameters. The representation of wolf dimensions and the Random Forest hyperparameter search ranges is presented in Table 6.

Table 6. Representation of Wolf Dimensions for Random Forest Hyperparameters

Dimension	Hyperparameter	Search Range
x_1	n_estimators	10-500
x_2	max_depth	1-50
x_3	min_samples_split	2-20
x_4	min_samples_leaf	1-20

Each wolf is evaluated using a fitness function defined as the average F1-score obtained from 5-fold cross-validation on the training set. The F1-score is selected as the fitness function because it considers the balance between precision and recall, whereas 5-fold cross-validation is used to reduce the influence of data partitioning variability on the evaluation results during the optimization process. The optimization process is performed using the training set, while the testing set is reserved for the final model evaluation. The fitness function is defined in Eq. (3).

$$f(x) = \frac{1}{K}\sum_{i=1}^{K} F1_i \quad (3)$$

where f(x) denotes the fitness value of each wolf, K represents the number of folds used in the cross-validation process, and $F1_i$ denotes the F1-score obtained from the i-th fold. This study employs 5-fold cross-validation; therefore, the fitness value is calculated as the average F1-score across the five folds.

Based on the fitness values, the alpha, beta, and delta wolves are identified as the three solutions with the highest fitness values in the population. These three wolves are used to update the positions of all wolves in each iteration. The distance between the position of a wolf and the reference solution is calculated using Eq. (4), whereas the wolf position is updated using Eq. (5).

$$D = \left|C \cdot X_p - X\right| \tag{4}$$

$$X(t+1) = X_p - A \cdot D \tag{5}$$

The coefficients A and C are calculated using Eq. (6)-(7).

$$A = 2ar_1 - a \tag{6}$$

$$C = 2r_2 \tag{7}$$

where X_P represents the position of the alpha wolf, whereas X denotes the current position of a wolf. The parameter a decreases linearly from 2 to 0 throughout the optimization process, while r_1 and r_2 are random numbers within the interval [0,1]. In this study, the optimization process uses a population of 10 wolves with a maximum of 20 iterations. These settings are selected by considering the repeated evaluation process, because each candidate solution is evaluated using 5-fold cross-validation. Therefore, increasing the population size or the number of iterations would also increase the number of Random Forest training processes during optimization. The selected setting is evaluated through the convergence behavior presented in the Results and Discussion section. This study does not conduct a separate sensitivity analysis for different population sizes and iteration numbers; therefore, the possible effect of larger populations or more iterations is considered as a limitation and can be investigated in future studies.

During each iteration, the positions of all wolves are updated based on the positions of the alpha, beta, and delta wolves. Then, the fitness value of each solution is calculated using the average F1-score obtained from 5-fold cross-validation. After all iterations are completed, the hyperparameter combination with the highest fitness value is selected as the final solution. The hyperparameter optimization procedure for Random Forest using GWO is presented in Algorithm 1.

Algorithm 1: Hyperparameter Tuning using Grey Wolf Optimization for Random Forest

Data: Initialisation the grey wolf population X_i $(i = 1,2,....,n)$
Initialise a, A and C
Define the Random Forest hyperparameter search space:
n_estimators, *max_depth*, *min_samples_split*, *min_samples_leaf*

Calculate the fitness of each search agent using mean F1-score from 5-fold cross-validation
X_α = the best search agent
X_β = the second best search agent
X_δ = the third best search agent
while $t <$ *Maximum number of iterations* **do**
  for *each of the search agents* **do**
    Update the individual position of the current search agent using Equations (4)–(7) — (4)–(7)
    $X(t+1) = (X_1 + X_2 + X_3)/3$ — (8)
  end
  Update a, A and C — (9)
  Calculate the fitness of all search agents using mean F1-score from 5-fold cross-validation — (10)
  Update $X_\alpha, X_\beta, X_\delta$ — (11)
  $t = t + 1$ — (12)
end
return X_α

2.6 Performance Evaluation Metrics

Model performance is evaluated using several classification metrics, namely accuracy, precision, recall, F1-score, and the Area Under the Receiver Operating Characteristic Curve (AUC–ROC). Accuracy measures the proportion of correctly classified instances among all testing samples. Precision evaluates the model's ability to correctly identify patients predicted to have heart disease, whereas recall evaluates the model's ability to identify patients who actually have heart disease. The F1-score is the harmonic mean of precision and recall and is used to represent the balance between these two metrics. In addition, AUC–ROC evaluates the model's ability to distinguish between the positive and negative classes across different classification thresholds. Each evaluation metric is calculated using Eq. (8) – (12).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

$$Precision = \frac{TP}{TP+FP} \tag{9}$$

$$Recall = \frac{TP}{TP+FN} \tag{10}$$

$$F1 = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{11}$$

$$AUC = \int_0^1 TPR(FPR)d(FPR) \tag{12}$$

where TP (True Positive) denotes the number of patients with heart disease who are correctly classified, TN (True Negative) denotes the number of patients without heart disease who are correctly classified, FP (False Positive) denotes the number of patients without heart disease who are incorrectly classified as having heart disease, whereas FN (False Negative) denotes the number of patients with heart disease who are incorrectly classified as not having heart disease.

In addition to the evaluation metrics, this study also applies the exact McNemar test to determine whether there is a statistically significant difference in performance between the baseline RF and GWO-RF models. This test is used because both models are evaluated on the same testing dataset, resulting in paired prediction outputs that can be directly compared. The exact McNemar test focuses on discordant prediction pairs, namely samples that are correctly classified by one model but incorrectly classified by the other model. Since the number of discordant prediction pairs is relatively small, the p-value is calculated using the exact McNemar test based on the binomial distribution, as shown in Eq. (13).

$$p = 2 \times P(X \leq \min(b,c)), \quad X \sim Binomial(b+c, 0.5) \tag{13}$$

where b denotes the number of samples correctly classified by the baseline RF model but incorrectly classified by the GWO-RF model, while c denotes the number of samples incorrectly classified by the baseline RF model but correctly classified by the GWO-RF model.

3. RESULTS AND DISCUSSION

This section presents the evaluation results of the Random Forest model before and after hyperparameter optimization using GWO for heart disease prediction. The evaluation compares the performance of both models using several classification metrics to identify performance changes following the optimization process. The results are organized into several subsections, including the evaluation metrics, the baseline model performance, the optimization results obtained using GWO, the performance comparison between the two models, and the discussion of the research findings.

The Random Forest model is trained using the default hyperparameters as the baseline model. The baseline model serves as a reference for comparing the model performance before and after the hyperparameter optimization process. The evaluation results of the baseline model on the testing set are presented in Table 7.

Table 7. Evaluation Results of the Baseline Random Forest Model

Metric	Value
Accuracy	88.52%
Precision	81.82%
Recall	96.43%
F1-score	88.52%
AUC-ROC	95.13%

Based on Table 7, the baseline Random Forest model achieves an accuracy of 88.52% and an AUC–ROC of 95.13%. The recall of 96.43% indicates that most patients with heart disease are correctly identified by the model. Meanwhile, the precision of 81.82% indicates that some patients are predicted to have heart disease even though they actually belong to the non-heart disease class. The obtained precision and recall values result in an F1-score of 88.52%. The classification results of the baseline Random Forest model are also illustrated using the confusion matrix and the ROC curve shown in Figure 2.

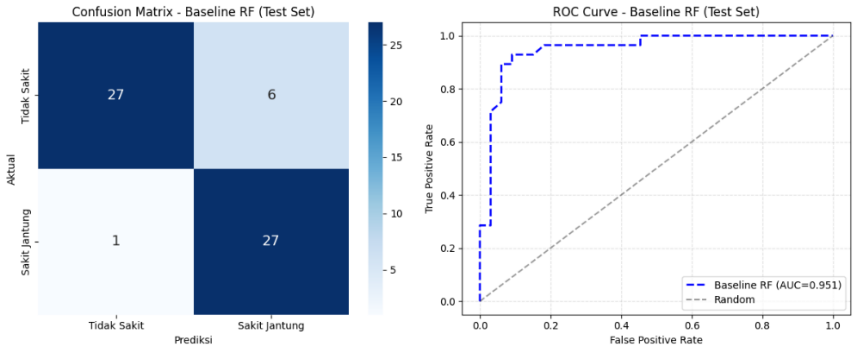

Figure 2. Confusion Matrix and ROC Curve of the Baseline Random Forest Model

Based on the confusion matrix, the model correctly classifies 27 patients without heart disease and 27 patients with heart disease. In addition, there are 6 false positive predictions, representing patients without heart disease who are predicted to have heart disease, and 1 false negative prediction, representing a patient with heart disease who is predicted not to have heart disease. The higher number of false positive predictions compared with false negative predictions is consistent with the lower precision value relative to recall. The ROC curve yields an AUC of 95.13%, indicating the model's ability to distinguish between the positive and negative classes.

The higher number of false positives in the baseline RF model may be related to the use of default hyperparameters. In this condition, the tree complexity settings are not specifically adjusted to the characteristics of the Cleveland Heart Disease Dataset. In addition, some patients without heart disease may have clinical feature values that are similar to those of patients with heart disease. This similarity may cause some samples from the negative class to be classified as the positive class. This is reflected in the lower precision value of the baseline model compared with its recall.

After the baseline Random Forest model is evaluated, hyperparameter optimization is performed using GWO. In this study, the fitness value is defined as the average F1-score obtained from 5-fold cross-validation on the training set, as described in Section 2.5. The progression of the highest fitness value throughout the optimization process is shown in Figure 3.

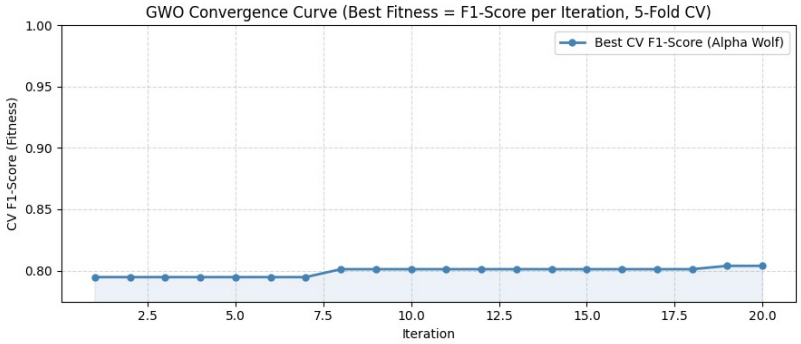

Figure 3. Convergence of Grey Wolf Optimization

Based on Figure 3, the highest fitness value generally increases throughout the optimization process, although it remains unchanged during several iterations. The highest fitness value is 0.7947 at the beginning of the optimization process, increases to 0.8012 at the eighth iteration, and reaches 0.8039 at the nineteenth iteration. After the nineteenth iteration, no further improvement in the fitness value is observed until the final iteration. These results indicate that the optimization process no longer produces an increase in the fitness value during the remaining iterations. In this study, four hyperparameters are optimized using GWO, namely n_estimators, max_depth, min_samples_split, and min_samples_leaf. Meanwhile, max_features is fixed to sqrt, and class_weight is fixed to balanced throughout the optimization process. The final Random Forest hyperparameter configuration used in this study is presented in Table 8.

Table 8. Final Random Forest Hyperparameter Configuration After GWO Optimization

Hyperparameter	Value
n_estimators	60
max_depth	50
min_samples_split	4
min_samples_leaf	17
max_features	sqrt
class_weight	balanced

Based on Table 8, the optimization process produces a hyperparameter configuration that differs from the default Random Forest hyperparameters. This configuration is used to retrain the Random Forest model, and the optimized model is then evaluated on the testing set. The evaluation results are compared with those of the baseline model, as presented in Table 9.

Table 9. Performance Comparison Between the Baseline Random Forest and GWO-RF Models

Metric	Baseline RF	GWO-RF
Accuracy	88.52%	93.44%
Precision	81.82%	90.00%
Recall	96.43%	96.43%
F1-score	88.52%	93.10%
AUC-ROC	95.13%	96.86%

Based on Table 9, the Random Forest model with the hyperparameter configuration obtained through GWO optimization obtains higher descriptive values for most evaluation metrics than the baseline model. The accuracy increases from 88.52% to 93.44%, whereas the precision increases from 81.82% to 90.00%. In addition, the F1-score increases from 88.52% to 93.10%, while the AUC–ROC increases from 95.13% to 96.86%. Meanwhile, the recall remains unchanged at 96.43%, indicating that the model's ability to identify patients with heart disease remains unchanged. The evaluation

results of the Random Forest model with the GWO-optimized hyperparameters are also presented using the confusion matrix and the ROC curve shown in Figure 4.

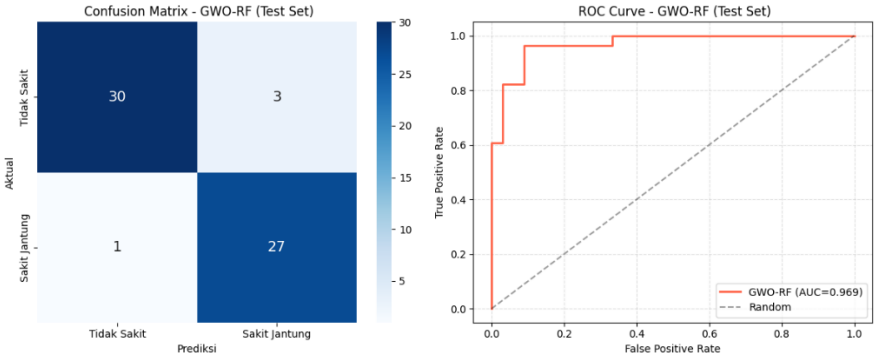

Figure 3. Confusion Matrix and ROC Curve of the GWO-RF Model

Based on the confusion matrix, the GWO-RF model correctly classifies 30 patients without heart disease and 27 patients with heart disease. In addition, there are 3 false positive predictions and 1 false negative prediction. Compared with the baseline model, the number of false positive predictions decreases from 6 to 3, whereas the number of false negative predictions remains 1. This change is accompanied by an increase in the precision value, while the recall remains 96.43%. The ROC curve yields an AUC of 96.86%, which is higher than the AUC of 95.13% obtained by the baseline model.

The decrease in the number of false positives in the GWO-RF model may be related to the hyperparameter configuration obtained through the GWO optimization process. One relevant setting is the larger min_samples_leaf value, which may help reduce decisions based on small or overly specific patterns in the data. This condition is associated with the decrease in the number of false positives from 6 in the baseline RF model to 3 in the GWO-RF model.

To verify the performance difference between the baseline RF and GWO-RF models, this study uses the exact McNemar test based on paired prediction results from the same testing dataset. The contingency table of the exact McNemar test is presented in Table 10.

Table 10. Contingency Table of the Exact McNemar Test

Baseline RF	GWO-RF Correct	GWO-RF Incorrect
Correct	54	0
Incorrect	3	4

Based on Table 10, the GWO-RF model correctly classifies 3 samples that are misclassified by the baseline RF model. In addition, there are no samples that are correctly classified by the baseline RF model but misclassified by the GWO-RF model. The exact McNemar test produces a p-value of 0.25. Since this value is greater than 0.05, the performance difference between the baseline RF and GWO-RF models is not statistically significant. Therefore, the higher evaluation values obtained by the GWO-RF model in this study are interpreted as a descriptive performance gain rather than a statistically significant improvement. This result may be influenced by the limited size of the testing dataset, which consists of 61 samples, and the small number of prediction changes between the two models, which occurs in only three samples.

A comparative analysis is conducted to compare the performance of the GWO-RF model with several previous methods, namely SVM [2], DBN-CSO [5], GWO-SVM [15], and Random Forest with hyperparameter tuning [16]. The comparison uses studies that applied the Cleveland Heart Disease Dataset so that the compared results are within the same data context. In this study, the GWO-RF model obtains an accuracy of 93.44%, precision of 90.00%, recall of 96.43%, F1-score of 93.10%, and AUC of 96.86%. The performance comparison results are presented in Table 11.

Table 11. Comparative Study of Proposed Method

Method	Dataset	Accuracy	Precision	Recall	F1-Score	AUC
Support Vector Machine (SVM)	Cleveland Heart Disease	85.00%	87.00%	82.00%	NR	91.00%
DBN-CSO	Cleveland Heart Disease	89.20%	NR	NR	NR	NR
GWO-SVM	Cleveland Heart Disease	89.83%	NR	93.00%	NR	NR
Random Forest with Hyperparameter Tuning	Cleveland Heart Disease	90.28%	85.11%	89.06%	92.00%	89.00%
Proposed GWO-RF	Cleveland Heart Disease	93.44%	90.00%	96.43%	93.10%	96.86%

Based on the testing results, hyperparameter optimization using GWO produces higher descriptive evaluation values than the baseline Random Forest model. The GWO-RF model achieves higher accuracy, precision, F1-score, and AUC-ROC values, while maintaining the same recall. This improvement is associated with a reduction in false positive predictions from 6 to 3 without increasing the number of false negatives, resulting in higher precision while preserving the model's ability to identify positive cases.

Nevertheless, the exact McNemar test indicates that the performance difference between the baseline RF and GWO-RF models is not statistically significant ($p = 0.25$). Therefore, the observed improvement should be interpreted as a descriptive improvement rather than a statistically significant improvement. These findings highlight that higher evaluation metrics do not necessarily indicate a statistically significant performance difference, particularly when the testing dataset is relatively small.

The findings of this study are consistent with previous studies showing that hyperparameter optimization and metaheuristic algorithms can improve heart disease classification performance [8], [10], [16]. However, this study has several limitations. It uses only the Cleveland Heart Disease Dataset, the testing dataset is relatively small, and the optimization process considers only four Random Forest hyperparameters. Future studies may employ more diverse datasets, broader validation strategies, a wider hyperparameter search space, and comparisons with other optimization algorithms.

4. CONCLUSION

This study applies Grey Wolf Optimization (GWO) to optimize the hyperparameters of Random Forest for heart disease classification using the Cleveland Heart Disease Dataset. The testing results show that the GWO-RF model obtains higher descriptive evaluation values than the baseline RF model in terms of accuracy, precision, F1-score, and AUC-ROC, while the recall value remains the same. However, the exact McNemar test shows that the performance difference between the two models is not statistically significant. Therefore, the performance improvement in this study is interpreted as a descriptive improvement.

The contribution of this study lies in the application of GWO as a hyperparameter optimization approach for Random Forest and in the comparative evaluation between the baseline RF and GWO-RF models on the same testing dataset. Methodologically, this study highlights the importance of comparing an optimized model with a baseline model and verifying the performance difference using a statistical test. Practically, the results of this study can serve as an initial basis for developing machine learning-based heart disease prediction models, particularly in selecting hyperparameter

configurations that are suitable for the characteristics of the dataset. However, this study has several limitations because it only uses one dataset, the testing data size is relatively limited, and the optimization process is conducted on four Random Forest hyperparameters. Future studies can use more diverse datasets, apply broader validation strategies, expand the hyperparameter search space, and compare GWO with other optimization algorithms.

REFERENCES

[1] O. World Health, "Cardiovascular diseases (CVDs)," World Health Organization. [Online]. Available: https://www-who-int.translate.goog/news-room

[2] E. A. Ogundepo and W. B. Yahya, "Performance analysis of supervised classification models on heart disease prediction," *Innov. Syst. Softw. Eng.*, vol. 19, no. 1, pp. 129–144, 2023, doi: 10.1007/s11334-022-00524-9.

[3] M. G. El-Shafiey, A. Hagag, E. S. A. El-Dahshan, and M. A. Ismail, "A hybrid GA and PSO optimized approach for heart-disease prediction based on random forest," *Multimed. Tools Appl.*, vol. 81, no. 13, pp. 18155–18179, 2022, doi: 10.1007/s11042-022-12425-x.

[4] T. O. Omotehinwa, D. O. Oyewola, and E. G. Moung, "Optimizing the light gradient-boosting machine algorithm for an efficient early detection of coronary heart disease," *Informatics Heal.*, vol. 1, no. 2, pp. 70–81, 2024, doi: 10.1016/j.infoh.2024.06.001.

[5] N. P and S. Narayan, "Cardiac disease detection using cuckoo search enabled deep belief network," *Intell. Syst. with Appl.*, vol. 16, 2022, doi: 10.1016/j.iswa.2022.200131.

[6] M. S. Al Reshan, S. Amin, M. A. Zeb, A. Sulaiman, H. Alshahrani, and A. Shaikh, "A Robust Heart Disease Prediction System Using Hybrid Deep Neural Networks," *IEEE Access*, vol. 11, pp. 121574–121591, 2023, doi: 10.1109/ACCESS.2023.3328909.

[7] A. Abdellatif, H. Abdellatef, J. Kanesan, C. O. Chow, J. H. Chuah, and H. M. Gheni, "An Effective Heart Disease Detection and Severity Level Classification Model Using Machine Learning and Hyperparameter Optimization Methods," *IEEE Access*, vol. 10, pp. 79974–79985, 2022, doi: 10.1109/ACCESS.2022.3191669.

[8] R. Torthi, A. D. K. Marapatla, S. Mande, H. K. V. Gadiraju, and C. Kanumuri, "Heart Disease Prediction Using Random Forest Based Hybrid Optimization Algorithms," *Int. J. Intell. Eng. Syst.*, vol. 17, no. 2, pp. 134–144, 2024, doi: 10.22266/ijies2024.0430.12.

[9] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[10] R. R. Badveli, N. G. Siddappa, and S. K. Kanipakapatnam, "Heart disease detection and classification using grid search with random forest," *IAES Int. J. Artif. Intell.*, vol. 15, no. 2, pp. 1300–1315, 2026, doi: 10.11591/ijai.v15.i2.pp1300-1315.

[11] G. Narasimhan and A. Victor, "Grey wolf optimized stacked ensemble machine learning based model for enhanced efficiency and reliability of predicting early heart disease," *Automatika*, vol. 65, no. 3, pp. 749–762, 2024, doi: 10.1080/00051144.2024.2317098.

[12] M. D. Teja and G. M. Rayalu, "Optimizing heart disease diagnosis with advanced machine learning models: a comparison of predictive performance," *BMC Cardiovasc. Disord.*, vol. 25, no. 1, 2025, doi: 10.1186/s12872-025-04627-6.

[13] C. Dua, D., & Graff, "UCI Machine Learning Repository." [Online]. Available: https://archive.ics.uci.edu

[14] L. Breiman, *Random forests*, vol. 45, no. 1. 2001. doi: 10.1023/A:1010933404324.

[15] Q. Al-Tashi, H. Rais, and S. Jadid, "Feature selection method based on grey wolf optimization for coronary artery disease classification," *Adv. Intell. Syst. Comput.*, vol. 843, pp. 257–266, 2019, doi: 10.1007/978-3-319-99007-1_25.

[16] M. A. Bouqentar *et al.*, "Early heart disease prediction using feature engineering and machine learning algorithms," *Heliyon*, vol. 10, no. 19, 2024, doi: 10.1016/j.heliyon.2024.e38731.