IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 2, April 2026, pp. 1300~1315
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i2.pp1300-1315

Heart disease detection and classification using grid search with random forest

Ramakrishna Reddy Badveli[1], Nijaguna Gollara Siddappa[2], Sundeep Kumar Kanipakapatnam[3]
[1]Department of Computer Science and Engineering, S.E.A. College of Engineering and Technology, Bengaluru, India
[2]Department of Information Science and Engineering, S.E.A. College of Engineering and Technology, Bengaluru, India
[3]Department of Computer Science and Allied Branches, Sri Venkateswara College of Engineering and Technology, Chittoor, India

Article Info

Article history:

Received Jun 24, 2025
Revised Jan 5, 2026
Accepted Jan 22, 2026

Keywords:

Cardiovascular disease
CMDMO
Grid search with random forest
Heart disease
Support vector machine

ABSTRACT

Cardiovascular disease (CVD) is basically stated as heart disease, is a significant impact of mortality rate in worldwide. Diagnosing heart disease is challenging because of the complexity of patient data, which establishes multiple categories of the disease and also irrelevant features, making it difficult to achieve classification accuracy. This research proposed a grid search with random forest (GS-RF) approach, which effectively identifies heart disease and significantly enhances classification accuracy by fine-tuning the random forest (RF) approach. It optimizes key hyperparameters like number of trees and greater number of features, improving model performance. The chaotic maps-based dwarf mongoose optimization (CMDMO) is used for feature selection, which efficiently selects the relevant feature and prevents the algorithm from getting trapped in local minima. The classification using grid search's effectiveness ensures that resources are spent on finding the best model rather than performing random, less efficient tuning. The proposed GS-RF model demonstrates high classification performance, achieving 99.43% accuracy on Cleveland dataset, while also attaining 99.10% accuracy on Statlog dataset, thereby confirming its robustness and effectiveness across different datasets. The proposed approach is evaluated in comparison with existing classification techniques, such as support vector machine (SVM), to demonstrate its greater effectiveness with respect to accuracy.

Corresponding Author:

Ramakrishna Reddy Badveli
Department of Computer Science and Engineering, S.E.A. College of Engineering and Technology
Bengaluru, India
Email: itramkey@gmail.com

1. INTRODUCTION

Cardiovascular disease (CVD) is also termed as heart disease, which is the most significant cause of high mortality in worldwide. Based on World Health Organization (WHO), through 2030 over 23.6 million people are projected to die from CVD, primarily due to strokes and heart failure [1], [2]. Common symptoms of heart disease include shortness of breath, physical weakness, and swollen feet. Researchers are actively exploring efficient techniques for heart disease detection, as existing machine learning (ML) methods have shown promising results in addressing diagnostic challenges and improving accuracy [3], [4]. Early detection of heart disease, combined with timely intervention, is crucial for effective management. Advancements in auxiliary diagnostic technologies, approaches to heart disease diagnosis are becoming more diverse, accurate, and personalized [5], [6]. Applying appropriate treatment based on accurate classification of heart disease is essential in reducing both the mortality and risk associated with the disease. The ML plays an increasingly

critical role in the early classification of diseases, helping to prevent and manage heart disease more effectively [7], [8]. An integration of diagnosis tools through neural networks is utilized for leveraging the linear and non-linear relationships among the variables. These approaches reduce the impact of diverse data and ambiguity on both prediction accuracy and estimation error [9], [10]. Therefore, several key factors are essential in designing a robust computerized auscultation tool to assist healthcare providers in the CVD screening process, utilizing the Cleveland and Statlog dataset for heart disease [11]. After data collection, the framework aims to preserve the signal morphology with a focus on murmurs, which are primary indicators of cardiac abnormalities. Ensuring that input features are processed from severe distortions and standardizing features are key phase in this process [12].

There are various types of heart diseases, primarily related to conditions such as blocked arteries and heart attacks. Feature selection is involved in identifying the relevant features for determining whether an individual has heart disease [13], [14]. The CVD prediction system should operate with minimal medical features that provide complementary information, thereby reducing the need for extensive feature collection [15]. The feature selection involved optimization, efficiently handles the search space and selects relevant features, ultimately improving classification performance [16]. ML techniques for classification are part of a scientific discipline that focuses on enabling machines to obtain knowledge from data and improve over time and primarily relying on principles of statistics and probability [17], [18]. Various methods have been proposed for prediction, including support vector machine (SVM), decision trees (DT), extreme gradient boosting (XGBoost), and evolutionary computing algorithms, each with its strengths and limitations [19]. So, ML with parameter tuning is utilized for predicting heart disease and handling the various heart disease then achieve better performance [20].

El-Shafiey *et al.* [21] developed an ML based technique to analyze the several levels of heart disease by using a hybrid genetic algorithm and particle swarm optimization (GAPSO) enabled random forest (RF) in the Cleveland and Statlog dataset. These methods efficiently handle different cases of heart disease by using RF, which enhanced the decision-making process and achieves better classification accuracy. The GAPSO–RF technique faces challenges because which was difficult to understand the various tree structures and the optimization process, which leads to falling into local optima, preventing efficient exploration of the search space. Nandakumar and Narayan [22] presented a deep belief network with cuckoo search algorithm (DBN-CSO) analyzing cardiac disease by using Cleveland and Statlog datasets. The combination of DBN-CSO algorithm with various levels of depth produced an accurate prediction of cardiac disease. The DBN-CSO model struggles to analyze large datasets, and the risk of underfitting affects the model's performance. Ogundepo and Yahya [23] presented predictive analysis of data for heart disease patients by using recursive feature elimination (RFE)-RF was used as a straightforward backwards feature selection for the Cleveland dataset. The prediction using an ML classification model was evaluated, and SVM was found to be the most suitable for predicting the health conditions of heart disease patients. However, the duplicate features were unlikely to be removed by RFE using RF, even though it is unnecessary. Al Reshan *et al.* [24] presented a hybrid deep neural network that involved various neural networks for extraction of relevant feature from the Cleveland and statlog dataset by using a convolutional neural network (CNN) through long short-term memory (LSTM) techniques. These models efficiently extract spatial features, identify patterns in the data, handle high-dimensional inputs, and capture temporal dependencies from time series data, which improves accuracy. The hybrid CNN-LSTM faces challenges such as requiring large data for training, increased model complexity, and difficulty in analyzing heart disease, which affects the classification performance. Abdellatif *et al.* [25] presented efficient heart disease detection as well as brutality level classification by ML and hyper parameter optimization (HPO) methods. This method relied on synthetic minority oversampling technique (SMOTE), which handled imbalanced distribution issues, HPO to find parameters for the classifier, and extra trees (ET), which were optimized by hyperband. These methods attain better performance on binary and multi-class issues. However, the performance of the model only improved for an imbalanced dataset with binary and multi-class classification.

Omotehinwa *et al.* [26] developed the light gradient boosting machine (LightGBM) algorithm for the classification of heart diseases, and Bayesian optimization through tree-structured parzen estimator (TPE) was utilized th fine-tune its hyper parameters. Multiple imputations through chained equations (MICE) were utilized individually for train and test groups for solving missing data. Borderline-SMOTE was utilized on training group for attain better performance. Suhatril *et al.* [27] developed the different ML approaches such as DT, naïve Bayes (NB), k-nearest neighbor (KNN), SVM, RF, logistic regression (LR), neural network, and gradient boosting (GB). The study utilized the different phases of research like dataset collection, preprocessing data, data modelling, estimation of different data modelling and input new data.

Singh *et al.* [28] presented the supervised ML approaches like DT, RF, SVM, and principal component analysis (PCA) for the early heart disease diagnosis. The presented approach represented significant potential in directing doctors through time and precise diagnosis. The presented ML approaches had resulted in greater intelligence as well as significant effectiveness on training and testing data.

Diagnosing diseases from several patients through the data with several categories of a specific disease so difficult to classify the heart disease. In the overall analysis from previous section, existing techniques face challenges in accurately diagnosing diseases across various patient data, especially when dealing with multiple categories of a particular disease, which makes it difficult to classify the heart disease. In this research, grid search with random forest (GS-RF) efficiently identifies heart disease and fine-tunes the RF model, significantly improving classification accuracy by selecting optimal values for parameters such as the number of trees and maximum features. The classification using grid search (GS) effectiveness ensures that resources are spent on finding the best model rather than performing random, less efficient tuning. The innovations of this research are considered as: i) the proposed GS-RF technique efficiently detects heart disease by improving GS, which performs a localized search for optimal parameters based on the RF hyperparameters; ii) The effectiveness of the GS method ensures that computational resources are focused on identifying the optimal hyperparameter settings of the RF model, resulting in clear decision boundaries and simplifying the analysis and interpretation of the model's behavior; and iii) The feature selection with chaotic maps-based dwarf mongoose optimization (CMDMO) efficiently select the relevant feature and handles high dimensionality and prevents falling into local minima, resulted in enhancing the detection performance of heart disease.

This manuscript is organized as follows: section 2 introduces the proposed GS-RF model. Section 3 discusses results and comparative analysis. The conclusion of this research paper is given in section 4.

2. METHOD

This research is targeted to developing a significant early diagnosis in GS through considering various methods. In this section, GS-RF technique efficiently handles various heart disease and reduces overfitting by averaging multiple DT, while GS systematically identifies the best fine-tuning balance the bias and variance. The data obtained from the Cleveland and Statlog datasets is pre-processed using standardization, ensuring that each feature contributes uniformly, thereby enhancing the heart disease detection across diverse data attributes. Feature selection using the CMDMO algorithm enhances the search capability through the introduction of chaos, which enhances the effectiveness of dwarf mongoose optimization (DMO) algorithm through removal of unimportant features and effectively handling high dimensionality, resulting in better classification accuracy. The RF is used for classification, which effectively handles the nonlinear relationships and complex interactions between features better than linear models. GS is then applied for fine tuning, utilizing more efficient feature optimization and improved accuracy. Figure 1 illustrates an outline of the working process of the proposed methodology for heart disease prediction.

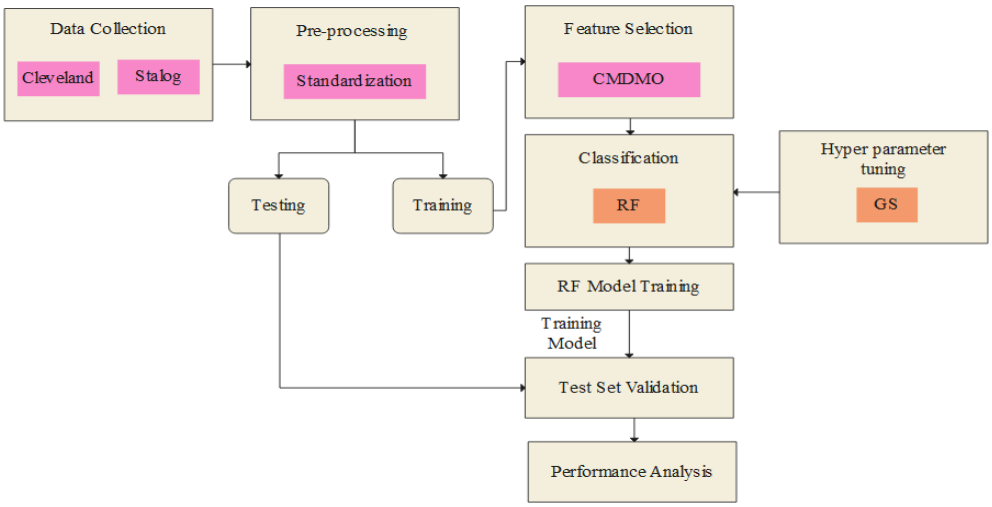


Figure 1. Outline of the working process of the proposed methodology for heart disease prediction

2.1. Data collection

The first step in designing an effective heart disease detection system is to select an appropriate dataset. The benchmark datasets like Cleveland and Statlog datasets, are utilized for heart disease detection in the network. The detailed explanation of these datasets is described as follows.

2.1.1. Cleveland dataset

The Cleveland dataset [29] for heart disease is available in the online ML and DL repository at University of California Irvine. This dataset includes an indicator variable (1 or 0) that helps to identify an occurrence of heart disease in patients. The full Cleveland database comprises 303 instances and 76 attributes. However, this research considers only 14 to 16 key attributes to assess the patient's health status.

2.1.2. Statlog dataset

Statlog dataset [30], also known as the heart disease dataset and available in the open repository at the University of California. It contains 303 instances, each representing a patient's heart disease diagnosis, along with data on various physical and biochemical attributes commonly used in medical diagnosis. Statlog dataset has 270 records, each with 13 Cleveland-like attributes. The dataset includes 76 features and classifies heart disease presence or absence.

2.1.3. Framingham dataset

This dataset was obtained as fragment of a constant study focusing on cardiovascular health performed through the involvement of inhabitants from Framingham, Massachusetts, and it is easily accessible in Kaggle [31]. It significantly performed for in classification operation to establish a likelihood of a patient's risk of introducing coronary heart disease (CHD) over a decade. It involves 4,240 patient records and 16 different features, by every feature helping as an indicator of a particular menace aspect.

2.2. Pre-processing

In this section, pre-processing using standardization, which scales the risk factors and assigns values that reflect their deviation from the mean [32]. This process rescales the risk factor values to enhance the performance of ML classification with a standard deviation σ of 1 and mean μ of 1. Its mathematical expression is expressed in (1). Where, standardization ensures that feature contributes equally, improving the reliability of prediction across diverse data attributes. After pre-processing data is fed to feature selection.

$$Standarization = \frac{X - Mean\ of\ X}{Standard\ Deviation\ of\ X} \tag{1}$$

2.3. Feature selection

After pre-processing the data, the CMDMO algorithm efficiently selects the relevant features and reduces the high dimensionality. In this research, the DMO is selected for feature selection in heart disease prediction due to it integrates global search power with chaos-induced randomness, supporting to solve local minima issue. However, the genetic algorithm (GA) suffers from premature convergence, resulted in suboptimal feature subsets, while particle swarm optimization (PSO) balance among an exploration and exploitation can weaken in high-dimensional medical data.

Unlike GA, which often suffers from premature convergence, and PSO, which may struggle in high-dimensional spaces due to loss of diversity, CMDMO combines the global search capability of DMO with the unpredictability introduced by chaotic maps. This hybrid structure improves exploration and avoids local optima more effectively. The CMDMO algorithm simulates foraging behavior in mongoose colonies, and chaotic maps dynamically adjust the solution update process, improving convergence speed and diversity. Thus, CMDMO's integration of chaotic maps adds unpredictability, enhancing a search diversity and ensuring the most relevant clinical features (e.g., cholesterol, blood pressure, and heart rate) are selected. This improves prediction accuracy and make sures model generalizes well in heart disease prediction tasks.

The chaos map efficiently explores the search space, which enhances the dingo swarm optimization (DSO) algorithm's capability to identify a solution space more effectively. The DMO algorithm simulates a foraging behavior of dwarf mongoose, which relies on compensatory behavioral adaptations. Dwarf mongoose has developed a social structure which enables every individual to endure self-sufficiently while transitioning between locations, compensating for the non-killing strategy and systematized pack hunting. Initialization of population is denoted (X), the upper and lower bound is indicated (UB and LB), the population is generated stochastically is expressed by (2).

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d-1} & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d-1} & x_{2,d} \\ \vdots & & x_{i,j} & & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,d-1} & x_{n,d} \end{bmatrix} \tag{2}$$

Where X denoted population is produced arbitrarily in (3), $x_{i,j}$ stands for the position of jth dimension in ith population, n denoted as the population size, and d indicated for dimension issue is expressed by (3).

$$x_{i,j} = VarMin + rand \times (VarMax - VarMin) \quad (3)$$

Where $rand$ denotes a random value in the range of 0 to 1, $VarMax$ and $VarMin$ represent the maximum variation and minimum variation, respectively. The best solution is identified over the iterations with the fitness function evaluating each solution after the population is initialized. A fitness of every candidate in a population is estimated using (4) and an alpha female mangoose (α) is chosen in terms of resulting probabilities. Where n indicates the count of mongooses in an alpha group. A food position candidate is expressed by (5) to (8).

$$\alpha = \frac{fit_i}{\sum_{i=1}^{n} fit_i} \quad (4)$$

$$X_{i+1} = X_i + phi \times peep \quad (5)$$

$$sm_i = \frac{fit_{i+1} - fit_i}{max\{[fit_{i+1}, fit_i]\}} \quad (6)$$

$$\varphi = \frac{\sum_{i=1}^{n} sm_i}{n} \quad (7)$$

$$X_{i+1} = \begin{cases} X_i - CF \times phi \times rand \times \left[X_i - \vec{M}\right] if\ \varphi_{i+1} > \varphi_i \\ X_i + CF \times phi \times rand \times \left[X_i - \vec{M}\right]\ elsewhere \end{cases} \quad (8)$$

Where X_{i+1} denotes a position of solution in a next iteration; X_i illustrates a position of solution in present iteration; φ demonstrates a scaling factor; sm_i denotes a difference between the fitness of the solution at the current and next iterations; fit_{i+1} denotes a fitness value of a solution at next iteration; fit_i specifies a fitness value of a solution at present iteration. CF denotes the constant factor; M means the target point. phi denoted the uniformly distributed arbitrary range is [-1,1] after each iteration sleeping hill is specified in (5). The average values of sleeping hill are identified in (6). A mongoose avoids early sleeping mounds, which allows scout search process to proceed next phase and ensure effective exploration. $CF = \left(1 - \frac{iter}{Max_{iter}}\right)^{\left(2\frac{iter}{Max_{iter}}\right)}$ denoted variable, which is decrease linearly with every iteration that control group collected movement. The $\vec{M} = \sum_{i=1}^{n} \frac{x_i \times sm_i}{X_i}$ denoted vector controls mongoose association to its recent mound of sleeping. It falls in local optimal so chaotic maps have been improved search space.

2.3.1. Chaotic maps dwarf mongoose optimization

In this phase, the enhanced chaotic map approach demonstrates non-linear changes in performance when its initial stages are even slightly altered. The chaotic maps help DMO algorithm to escape local optima easily and converge towards the global optimum. It provides diverse initial search paths and dynamically adjusts the search behavior, balancing exploration and exploitation effectively, which is shown in (9) mentions an updated version of (7).

$$X_{i+1} = \begin{cases} X_i - CF \times phi \times \rho \times \left[X_i - \vec{M}\right] & if\ \varphi_{i+1} > \varphi_i \\ X_i - CF \times phi \times \rho \times \left[X_i - \vec{M}\right] & else \end{cases} \quad (9)$$

Where ρ denoted values found from well-known chaotic maps, and the binary conversion is applied as (10). Finally, the fitness function is defined as (11).

$$X_{i,j} = \begin{cases} 1 & X_{i,j} > 0.5 \\ 0 & otherwise \end{cases} \quad (10)$$

$$Fitness = \frac{Number\ of\ wrong\ classified}{Total\ number\ of\ instance} + \frac{|X_i|}{d} \quad (11)$$

An individual feature is evaluated by fitness function, which evaluates a classification accuracy of the RF model using the selected candidate features. The chaotic maps are utilized to adjust the movement of dwarf particles in every iteration, ensuring that the exploration and exploration are balanced to maintain the dynamic feature. The introduction of chaotic helps in performing a dynamic search in the search space, avoiding the algorithm from falling into local optima. The overall process of the CMDMO algorithm is represented in Figure 2.

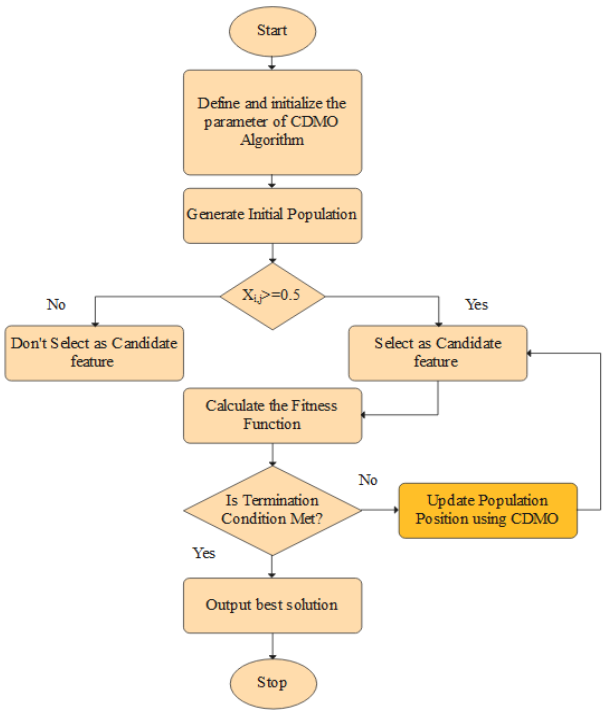


Figure 2. Flowchart of the CMDMO algorithm

2.4. Classification

In this section, the RF method is one of the most effective classification approaches and heart disease estimation is efficiently performed it involving multiple DT for classification. Each DT gives a vote that indicates the decision about the class of the object. Figure 3 shows the process of the RF technique. To classify an input vector, each tree in the forest generates a prediction and the final output is identified through most common class among these predictions. RF is well suited to handle high dimensional data and multi collinearity while also being fast and resilient to overfitting. Two key parameters in RF are the count of active variables in every arbitrary subset at each node and the count of trees in the forest. Here, the count of active variables is fixed for square root of total count of features and the count of trees is fixed at 25. A GS was performed across a wide range of parameter solutions and best set was selected to train the RF model for optimal classification accuracy.

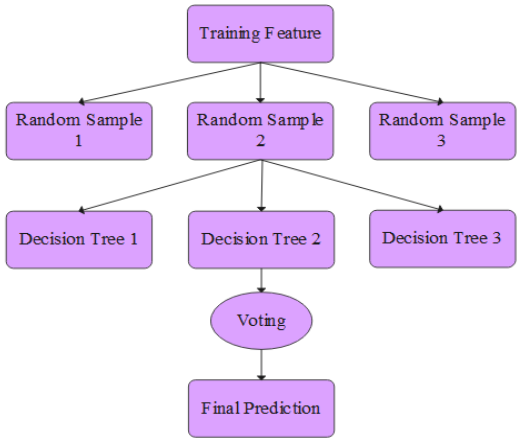


Figure 3. Working process of the RF technique

The final configuration specifies 1,000 trees, a maximum depth of 10, a voting strategy with confidence 0.5 and a split criterion that includes pruning and pre-pruning. The minimum leaf size is set to 2,

and a split parameter is set to 4 as represented in (12). Where, C indicated count of classes and $p(j)$ probability of selecting a class j data point.

$$G = \sum_{j=1}^{C} P(j)^2 \left(1 - p(j)\right) \quad (12)$$

2.4.1. Parameter tuning using grid search

In this section, GS is used for hyper parameter tuning through an iterative process to determine the optimal combination of hyperparameters. This approach systematically explores all possible combinations of predefined parameter values, ensuring the discovery of the truly optimal set for the RF model. However, the random search which miss the best configuration because of random sampling. While, Bayesian optimization is computationally expensive and complex to implement. Thus, GS offers a straightforward and exhaustive approach and guarantees a balanced trade-off among performance and interpretability, especially useful when working with well-defined parameter spaces like in RF, enhancing an approach's overall effectiveness and robustness.

This approach enhances its effectiveness, and a grid of hyper parameter values is developed for the RF method. Every loop endeavors random integration from this grid execution and used randomized GS algorithm to choose a good parameter for ten models. If the set parameter raises score values, these parameters are set for the precision of the model, while the remaining features are explored to identify new optimal parameters. The trained data is again split into $k = 5$ count of subsets called folds. The RF ensemble approach handle nonlinear relationships and effectively captures complex interactions between features, outperforming linear models and simpler classifiers. The GS optimises its parameters, ensuring the classification accuracy and tuning parameter are represented for the better reproducibility in algorithm 1.

Algorithm 1: Parameter tuning GS-RF method

```
Input:
    D ← Raw dataset (Cleveland, Statlog, or Framingham)
    CMDMO ← Chaotic maps with dwarf mongoose optimization
    RF ← Random forest classifier
    GS ← Grid search for hyperparameter tuning
Output:
    Trained GS-RF model with optimal features
Begin
1. Data preprocessing:
    - Load dataset D
    - Standardize features using Z-score normalization:
        For each feature x in D:
    x_scaled = (x − mean(x)) / std(x)
2. Feature selection using CMDMO:
    - Initialize population of candidate feature subsets
    - Apply chaotic maps to generate randomness in feature selection
    - Evaluate each subset using fitness function (accuracy of RF on subset)
    - Update population using CMDMO dynamics (exploration + exploitation)
    - Select best feature subset F_selected
3. Model training with grid search:
    - Define parameter grid:
        num_trees ∈ {100, 500, 1000}
        max_depth ∈ {5, 10, 15}
        min_leaf_size ∈ {1, 2, 4}
    - For each combination (num_trees, max_depth, min_leaf_size) in grid:
        - Train RF using F_selected on training data
        - Evaluate performance using cross-validation (e.g., K=5)
    - Select the parameter set with highest validation score
4. Train final GS-RF model:
    - Train RF with best parameters and F_selected features on full training set
5. Model evaluation:
    - Evaluate model on test set using:
        Accuracy, precision, recall, F1-score, AUC
    - Display confusion matrix and ROC Curve
Return: Trained GS-RF model with CMDMO-selected features
End
```

3. EXPERIMENTAL RESULT

In this section, the proposed GS-RF technique efficiently identifies the heart disease by using datasets, namely Cleveland and Statlog. The implementation of the proposed method is carried out using Python 3.11, Windows 10 (64-bit) operating system, memory 1 TB, GPU: 6 GB, i7 processor, and 16 GB of

RAM. All the experiments conducted for the proposed method by the environment of Python 3.11.12 through windows 10 OS, 16 GB RAM, and intel i5 processor. A training and testing portion of the obtained dataset is partitioned into a ratio of 80:20. In training procedure, the hyperparameters selected by using the GS, Adam optimizer is chosen and an initial learning rate is fixed for 0.001. After that, learning rate is decreased through 0.0001 every 100 epochs is utilized through an error of a validation set. This research utilizes the dropout mechanism in fully connected layers through a dropout ratio of 0.3 to solve the overfitting of mode.

To address class balance and prevent overfitting, stratified sampling was applied during data partitioning and cross-validation to ensure consistent class distributions. Overfitting was further mitigated using dropout (rate =0.3), learning rate decay (0.001 to 0.0001 every 100 epochs), and 5-fold cross-validation. Additionally, cross-dataset validation was conducted, where the model trained on one dataset was tested on another to evaluate robustness. Statistical analysis including t-tests, standard deviation, and permutation testing confirmed the model's stability and reliability.

The performance measures used for evaluation and the results of the feature selection and classification are explained in subsection 3.1. The performance of the proposed method is calculated using various performance metrics like accuracy, F1-score, recall, and precision. The selection of these performance metrics provides a complete evaluation of the model's classification performance. Accuracy provides an overall correctness measure, while precision concentrates on a significance of positive predictions. Recall make sures an approach significantly model all true positives, and F1-score balances precision and recall, especially under class imbalance. Combinedly, these metrics provides a robust and fair validation across different prediction aspects. are defined by (13) to (16). Where, TP, TN, FP, and FN illustrate true positive, true negative, false positive, and false negatives individually.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \tag{13}$$

$$Precision = \frac{TP}{(TP+FP)} \tag{14}$$

$$F1-score = 2 \times \frac{(Precision \times Recall)}{Precision+Recall} \tag{15}$$

$$Recall = \frac{TP}{TP+FN} \tag{16}$$

3.1. Performance analysis

In this subsection, a proposed method involving parameter tuning with classification processes is calculated using different performance indices named accuracy, F1-score, precision, and recall for Cleveland and Statlog datasets. The performance estimation of feature selection results with Cleveland and Framingham datasets are presented in Table 1. The CMDMO algorithm is compared with existing method such as weighted adaptive optimization (WAO), PSO, African buffalo optimization (ABO), and DSO algorithm.

Table 1. Evaluation of feature selection on the Cleveland and Framingham dataset

Dataset	Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
Cleveland	WAO	90.75	90.12	90.45	90.28	91.03
	PSO	91.96	91.35	91.65	91.49	92.24
	ABO	89.68	88.97	89.54	89.25	90.01
	DMSO	92.32	91.73	92.1	91.91	92.67
	CMDMO	99.43	98.79	99.40	99.09	99.50
Framingham	WAO	87.64	90.12	85.28	88.82	81.38
	PSO	91.42	93.12	87.52	90.13	83.13
	ABO	94.13	95.27	90.13	92.12	85.27
	DMSO	96.31	98.43	93.12	95.31	89.76
	CMDMO	99.12	99.24	95.78	97.32	91.32

The CMDMO achieve better accuracy of 99.43% on Cleveland and 99.10% on Statlog datasets. Figure 4 shows the graphical representation of the feature selection on the Statlog dataset. Table 2 illustrates performance evaluation of classification results on Cleveland and Statlog dataset. The performance analysis of classification using the GS-RF technique is compared with existing techniques such as SVM, DT, RF, and XGBoost model. Figure 5 graphically represents the classification on Statlog dataset. The GS-RF achieve higher accuracy on the dataset.

Table 2. Performance analysis of classification results on Cleveland and Framingham dataset

Dataset	Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
Cleveland	SVM	89.74	88.63	89.2	88.91	90.12
	DT	92.87	91.95	92.54	92.24	93.1
	RF	88.92	88.55	89.03	88.78	89.58
	XGBoost	94.23	93.88	94.11	93.99	94.64
	GS-RF	99.43	98.79	99.40	99.09	99.50
Framingham	SVM	90.32	93.13	86.43	90.22	80.12
	DT	91.37	94.12	88.12	94.12	83.12
	RF	93.37	95.23	90.13	95.12	85.32
	XGBoost	96.37	97.42	93.45	96.33	87.65
	GS-RF	99.12	99.24	95.78	97.32	91.32

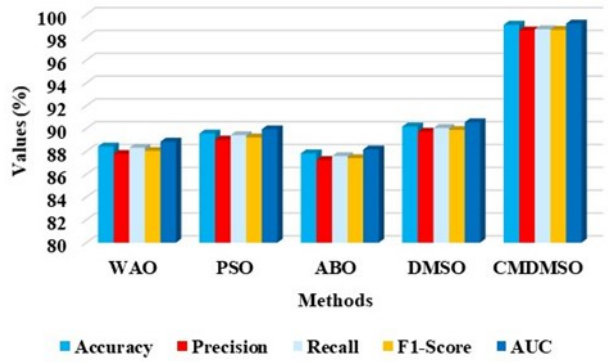


Figure 4. Feature selection on Statlog dataset

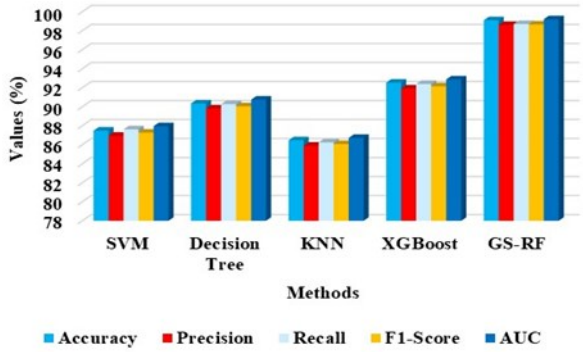


Figure 5. GS-RF on Statlog dataset

Table 3 demonstrates performance estimation of simple feature selection methods with proposed method using different datasets. The methods like Lasso and mutual information are considered with the proposed CMDMO to validate effectiveness. As compared to these simple feature selection approaches, the proposed CMDMO approach attains superior results in overall datasets.

Table 3. Performance estimation of simple feature selection methods with proposed method

Methods	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
Lasso	Cleveland	93.27	92.45	93.01	92.72	94.15
	Statlog	92.83	91.74	92.22	91.97	93.68
	Framingham	94.39	93.28	90.56	91.89	85.73
Mutual information	Cleveland	95.91	95.33	95.84	95.58	96.27
	Statlog	95.42	94.85	95.36	95.1	95.98
	Framingham	96.74	95.92	93.12	94.5	88.45
CMDMO	Cleveland	99.43	98.79	99.40	99.09	99.50
	Statlog	99.10	98.60	98.70	98.65	99.20
	Framingham	99.12	99.24	95.78	97.32	91.32

Table 4 demonstrates the performance estimation of model trained on Cleveland and tested on Statlog dataset. Table 5 demonstrates the performance estimation of model trained on Statlog and tested on Cleveland dataset. Table 6 demonstrates the performance estimation of model trained on Framingham and tested on Cleveland Statlog dataset. Table 7 describes the standard deviation of the proposed method with existing methods based on three different datasets.

Table 4. Performance estimation of model trained on Cleveland and tested on Statlog dataset

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
SVM	89.74	88.63	89.2	88.91	90.12
DT	92.87	91.95	92.54	92.24	93.1
KNN	88.92	88.55	89.03	88.78	89.58
XGBoost	94.23	93.88	94.11	93.99	94.64
GS-RF	99.43	98.79	99.40	99.09	99.50

Table 5. Performance estimation of model trained on Statlog and tested on Cleveland dataset

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
SVM	80.57	79.39	78.27	78.81	83.36
DT	82.93	81.72	80.94	81.26	85.24
KNN	79.61	78.29	77.63	77.91	82.14
XGBoost	84.79	83.51	84.04	83.73	87.36
CMDO-GS-RF	91.78	91.13	90.91	91.06	94.14

Table 6. Performance estimation of model trained on Framingham and tested on Cleveland and Statlog datasets

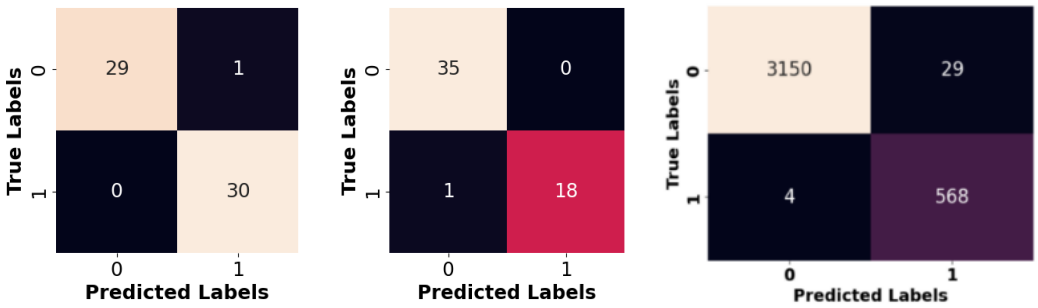
Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
SVM	78.94	77.83	76.42	77.08	81.43
DT	80.64	79.37	78.94	79.12	83.27
KNN	77.54	76.33	75.41	75.86	80.74
XGBoost	83.76	82.17	82.93	82.48	86.74
CMDO-GS-RF	92.63	92.01	91.73	91.84	93.82

Table 7. Standard deviation of the proposed method with existing methods

Methods	Dataset	Accuracy	Precision	Recall	F1-score	AUC
SVM	Cleveland	89.74 ± 0.43	88.63 ± 0.52	89.20 ± 0.37	88.91 ± 0.46	90.12 ± 0.48
	Statlog	87.49 ± 0.58	86.95 ± 0.61	87.62 ± 0.45	87.28 ± 0.50	87.95 ± 0.55
	Framingham	90.32 ± 0.62	93.13 ± 0.73	86.43 ± 0.59	90.22 ± 0.66	80.12 ± 0.60
DT	Cleveland	92.87 ± 0.39	91.95 ± 0.47	92.54 ± 0.41	92.24 ± 0.42	93.10 ± 0.40
	Statlog	90.34 ± 0.52	89.82 ± 0.58	90.27 ± 0.49	90.04 ± 0.51	90.75 ± 0.50
	Framingham	91.37 ± 0.65	94.12 ± 0.69	88.12 ± 0.64	94.12 ± 0.70	83.12 ± 0.58
KNN	Cleveland	88.92 ± 0.47	88.55 ± 0.44	89.03 ± 0.39	88.79 ± 0.42	89.58 ± 0.46
	Statlog	86.47 ± 0.51	85.90 ± 0.57	86.25 ± 0.49	86.07 ± 0.50	86.72 ± 0.52
	Framingham	93.37 ± 0.69	95.23 ± 0.73	90.13 ± 0.63	95.12 ± 0.70	85.32 ± 0.57
XGBoost	Cleveland	94.23 ± 0.35	93.88 ± 0.41	94.11 ± 0.38	93.99 ± 0.39	94.64 ± 0.37
	Statlog	92.54 ± 0.46	91.93 ± 0.48	92.39 ± 0.43	92.16 ± 0.45	92.87 ± 0.42
	Framingham	96.37 ± 0.52	97.42 ± 0.49	93.45 ± 0.44	96.33 ± 0.47	87.65 ± 0.53
CMDO-GS-RF	Cleveland	99.43 ± 0.18	98.79 ± 0.22	99.40 ± 0.20	99.09 ± 0.21	99.50 ± 0.19
	Statlog	99.10 ± 0.21	98.60 ± 0.25	98.70 ± 0.23	98.65 ± 0.24	99.20 ± 0.22
	Framingham	99.12 ± 0.23	99.24 ± 0.21	95.78 ± 0.29	97.32 ± 0.27	91.32 ± 0.31

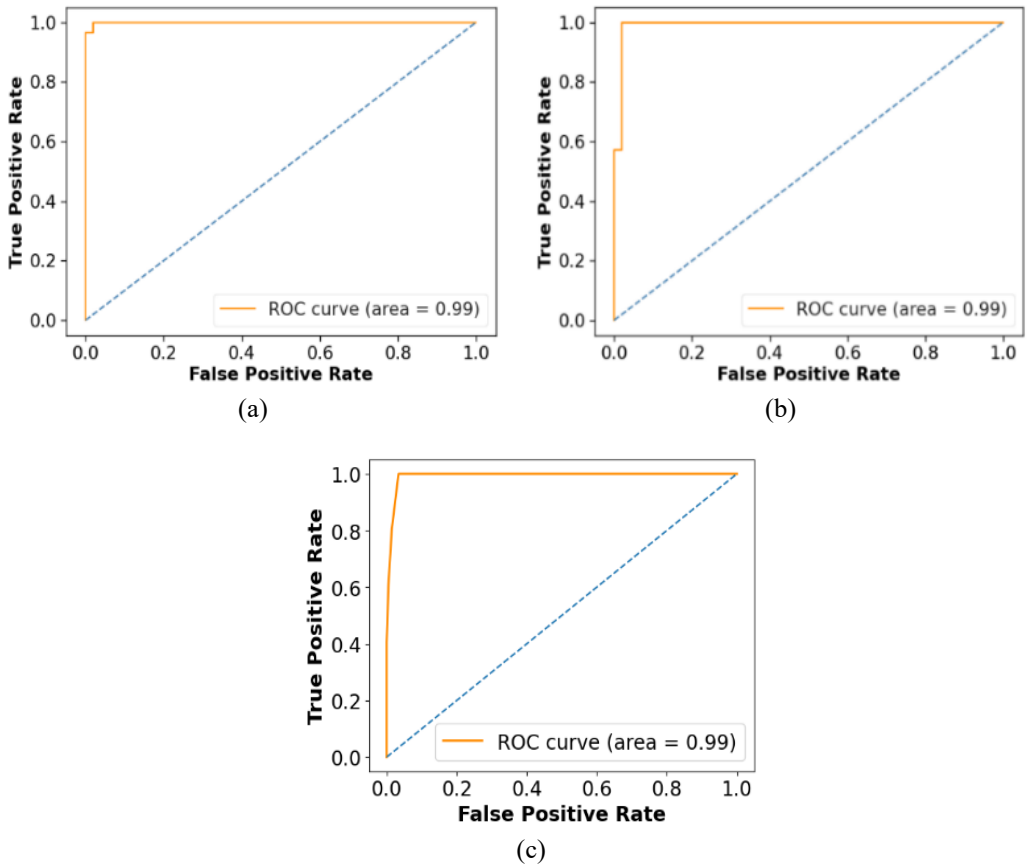
Figure 6 presents the confusion matrix analysis of the proposed method across three different datasets. Figure 6(a) demonstrates the confusion matrix for Cleveland dataset, Figure 6(b) represents the confusion matrix for Statlog dataset, and Figure 6(c) represents the Framingham dataset. The confusion matrix analyses classification and measures the accuracy of the proposed method in distinguishing between 0 and 1 instance. The confusion matrix analysis demonstrates that most misclassifications occurred between class 0 (no heart disease) and class 1 (presence of heart disease). These errors are primarily because of overlapping clinical symptoms such as similar cholesterol levels, blood pressure, and chest pain types, which can blur the distinction among healthy and at-risk patients. Moreover, small sample sizes and potential noise in the Cleveland and Statlog datasets contribute to these inaccuracies. Such misclassifications have important clinical implications such as false positives which resulted in unnecessary testing and patient anxiety, while false negatives could result in missed early diagnosis. This focusses the requirement for implementing large clinical features, expert validation, and real-world data variability to improve diagnostic reliability.

Figure 7 presents the receiver operating characteristic (ROC) curve analysis of the proposed method across three different datasets. Figure 7(a) demonstrates the ROC curve for Cleveland dataset, Figure 7(b) represents the ROC curve for Statlog dataset, and Figure 7(c) represents the ROC curve for Framingham dataset. The ROC curve visually represents performance by comparing the false positive rate and true positive rate. The proposed method significantly outperforms other existing methods. Table 8 demonstrates the cross-validation results of the proposed GS-RF method based on three different datasets. This table demonstrates that K =5 consistently attains greater results overall all datasets, denoting it provides the best balance among bias and variance. Compared to K =2, 3, 4, 7, 8, K =5 provides stable and reliable performance. Its importance lies in improving the model's generalization and make sure consistent accuracy across diverse traffic patterns, designing it the most impactful configuration for robust forecasting.



(a) (b) (c)

Figure 6. Confusion matrix for (a) Cleveland dataset, (b) Statlog dataset, and (c) Framingham dataset



(a) (b)

(c)

Figure 7. ROC curve analysis for (a) Cleveland dataset, (b) Statlog dataset, and (c) Framingham dataset

Table 8. Cross validation of proposed GS-RF approach

Dataset	K-fold values	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
Cleveland	K =2	91.45	90.45	91.47	91.37	90.43
	K =3	93.23	91.24	93.18	92.33	94.20
	K =3	95.27	94.24	94.22	94.13	95.31
	K =4	96.73	96.37	96.72	96.38	97.64
	K =5	99.43	98.79	99.40	99.09	99.50
Statlog	K =2	91.56	90.38	92.67	87.56	90.27
	K =3	93.12	94.27	94.76	91.45	92.73
	K =3	94.67	95.39	95.35	93.56	95.37
	K =4	96.37	96.27	96.36	96.38	96.37
	K =5	99.10	98.60	98.70	98.65	99.20
Framingham	K =2	87.44	88.78	85.24	85.38	82.87
	K =3	90.12	91.21	87.65	91.32	84.31
	K =3	93.32	93.32	90.13	93.31	85.32
	K =4	97.38	96.34	93.45	95.23	89.23
	K =5	99.12	99.24	95.78	97.32	91.32

Table 9 specifies the performance analysis of the proposed method with different hyperparameter tuning. The different hyperparameter tunings such as random search, Bayesian optimization, and GA are estimated and compared with GS approach. While GS is a widely used hyperparameter tuning method, its integration in this work is not merely standalone but complements the CMDMO-based feature selection. GS is particularly effective in our context due to the structured and well-bounded parameter space of the RF model. Although Bayesian optimization achieved similar accuracy, GS demonstrated more stable results, lower memory usage, and reduced computational overhead across repeated runs. This makes GS a preferable and reproducible choice for clinical settings, where model transparency and consistency are essential.

Table 9. Performance analysis of the proposed method with different hyperparameter tuning

Methods	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
Random search	Cleveland	93.74	93.1	92.88	92.99	94.26
	Statlog	92.59	92.01	91.74	91.88	92.94
	Framingham	94.88	94.31	91.42	92.59	85.73
Bayesian optimization	Cleveland	95.41	95.17	94.93	95.05	96.2
	Statlog	94.85	94.22	94.1	94.16	95.56
	Framingham	96.32	95.78	92.74	94.14	87.93
Genetic algorithms	Cleveland	96.13	95.96	96.12	96.04	97.11
	Statlog	95.84	95.47	95.55	95.51	96.65
	Framingham	97.19	96.44	94.23	95.31	89.41
Grid search	Cleveland	99.43	98.79	99.40	99.09	99.50
	Statlog	99.10	98.60	98.70	98.65	99.20
	Framingham	99.12	99.24	95.78	97.32	91.32

Table 10 demonstrates the computational complexity and statistical analysis of the proposed method based on multiple datasets. In addition to high performance metrics, the proposed CMDMO-GS-RF model is justified through statistical significance testing (t-test, p-test, and ANOVA) showing p-values consistently <0.05, indicating reliable improvements. Moreover, the computational benchmarks (Table 4) demonstrate superior efficiency through minimized training time, inference speed, and memory usage compared to baseline models. These results validate the method's robustness, scalability, and practical applicability beyond accuracy alone. In statistical analysis, a hypothesis is a formal assumption made about a population parameter to be tested through sample data. process of hypothesis testing involves evaluating these assumptions using a test statistic and a significance level (α=0.05). If the test yields a p-value less than α, the null hypothesis is rejected in favor of the alternative, indicating statistically significant results. In this study, hypothesis testing validates whether the performance differences between the proposed GS-RF model and existing methods are statistically significant.

The computational complexity involves training time, inference speed, memory usage, T-test, p-test, and ANOVA. As compared to other classification approaches, the proposed GS-RF approach attains the better computational efficiency. These results show that there is an important variation among least one model group in terms of performance. Table 11 demonstrates the Ablation study of the proposed method using different datasets. While individual GS and RF provide solid baselines, integrating them (GS-RF) crucially enhances all metrics. Adding the CMDMSO feature selection further improves results, attaining better results in all performance across all datasets. This confirms that CMDMSO effectively strengthens GS-RF through refining feature relevance and enhancing predictive effectiveness.

Table 10. Computational efficiency and statistical analysis of proposed method based on three different datasets

Dataset	Methods	Training time (s)	Inference speed (s)	Memory usage (MB)	P-test
Cleveland	SVM	12.4	0.15	120	0.041
	DT	8.7	0.10	105	0.036
	KNN	15.2	0.18	130	0.034
	XGBoost	20.5	0.12	150	0.030
	GS-RF	7.5	0.08	95	0.020
Statlog	SVM	11.9	0.14	125	0.042
	DT	8.1	0.09	110	0.037
	KNN	14.8	0.17	135	0.035
	XGBoost	19.8	0.11	155	0.029
	GS-RF	7.0	0.07	90	0.019
Framingham	SVM	13.1	0.16	128	0.040
	DT	9.0	0.11	112	0.036
	KNN	15.6	0.19	138	0.032
	XGBoost	21.3	0.13	158	0.028
	GS-RF	6.8	0.06	88	0.018

Table 11. Ablation study with different approaches

Dataset	Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC (%)
Cleveland	RF	92.12	92.12	93.32	93.13	94.21
	GS-RF	95.38	95.33	96.91	95.42	97.32
	DMO+RF	96.14	95.77	96.25	96.03	97.04
	DMO+GS-RF	97.63	97.12	97.4	97.25	98.37
	CMDO+GS-RF	99.43	98.79	99.40	99.09	99.50
Statlog	RF	92.35	92.12	92.32	95.11	95.32
	GS-RF	95.26	96.55	95.48	96.21	96.32
	DMO+RF	95.87	95.34	95.41	95.38	96.12
	DMO+GS-RF	97.35	96.94	96.66	96.8	98.01
	CMDO+GS-RF	99.10	98.60	98.70	98.65	99.20
Framingham	RF	94.13	94.12	90.89	93.12	84.12
	GS-RF	97.64	96.32	93.12	95.32	87.43
	DMO+RF	97.18	95.96	94.38	95.16	88.39
	DMO+GS-RF	98.47	97.58	95.82	96.69	90.23
	CMDO+GS-RF	99.12	99.24	95.78	97.32	91.32

3.2. Comparative analysis

A significance of proposed GS-RF is compared to existing approaches including GAPSO-RF [21], DBN-CSO [22], SVM [23], and hyperband (HB)-SMOTE-ET [25]. The comparative analysis involves two datasets: Cleveland, Statlog, and Framingham datasets. In this analysis, the values of existing methods like Borderline-SMOTE+TPE+LightGBM [26] and RF [27] are converted into percentage based on the proposed method's value. In this research, the proposed GS-RF method achieves a high accuracy of 99.43% on Cleveland and 99.10% of accuracy on Statlog dataset, respectively. Table 12 describes a comparative analysis of proposed approaches.

Table 12. Comparative study of proposed method

Methods	Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUC
GAPSO-RF [21]	Cleveland	95.60	NA	NA	NA	94.00
	Statlog	91.40	89.58	95.56	NA	92.60
DBN-CSO [22]	Cleveland	89.20	NA	NA	NA	88.00
	Statlog	91.20	NA	NA	NA	87.00
SVM [23]	Statlog	87.04	85.1	85.8	85.5	93.00
HB-SMOTE-ET [25]	Cleveland	99.2	98.7	99.33	99.33	NA
	Statlog	98.52	98.13	98.09	98.09	NA
Borderline-SMOTE+TPE+LightGBM [26]	Framingham	98.82	98.35	93.70	95.97	NA
RF [27]	Framingham	85	84	85	79	70
Proposed GS-RF technique	Cleveland	99.43	98.79	99.40	99.09	99.50
	Statlog	99.10	98.60	98.70	98.65	99.20
	Framingham	99.12	99.24	95.78	97.32	91.32

3.3. Discussion

The proposed GS-RF provides several advantages over existing methods. For instance, the GAPSO-RF [21] technique faces challenges in learning the structure of various trees and later gets stuck in

local optima, which limits the algorithm's ability to effectively explore the search space. The struggle to analyze large data by using DBN-CSO [22] and the risk of under-fitting affects the performance of the model. The duplicate features are unlikely to be removed by RFE using RF, even though it is unnecessary [23]. The performance of the model only improved for an imbalanced dataset with binary and multi-class classification [25]. GS-RF is efficiently identifying heart disease and fine-tunes the RF model, significantly improving classification accuracy by selecting optimal values for parameters such as the number of trees and maximum features. This GS-RF, which efficiently identifies heart disease and fine-tunes the RF model, significantly improving classification accuracy by selecting optimal values for parameters such as the number of trees and maximum features. Despite the strong classification performance, the relatively small size of datasets like Cleveland and Statlog presents a risk of overfitting. Although we employed regularization techniques such as dropout and learning rate decay, and validated across independent datasets, the absence of noise or mislabeled samples may not fully represent real-world conditions. In future work, we aim to validate our model on larger, more diverse datasets and collaborate with clinicians to understand the impact of real-world data variability.

4. CONCLUSION

This paper proposes the GS-RF approach, which effectively detects heart disease and enhances the performance of the RF model by fine-tuning key parameters such as the number of trees and maximum number of features, which significantly improves classification accuracy. Initially, data is obtained from the Cleveland and Statlog dataset, and pre-processing is performed using standardization to normalize the features between 0 and 1. The CMDMO is used for feature selection, which efficiently identifies relevant features while preventing the algorithm from getting trapped in local minima. The classification using GS's effectiveness ensures that resources are spent on finding the best model rather than performing random, less efficient tuning. The proposed GS-RF achieves better accuracy of 99.43% and 99.10% on Cleveland and Statlog datasets. This paper is compared with existing methods, such as the SVM technique, to evaluate its accuracy and performance. While this research demonstrates strong classification performance, we recognize that its practical diagnostic use requires real-world clinical validation. This research is constrained through the lack of collaboration with cardiology professionals, clinical trials, and patient-level interaction. In future work, we intend to involve medical experts through qualitative interviews and expert feedback to assess the interpretability, reliability, and clinical usability of our model. Moreover, advanced deep learning (DL) techniques like hierarchical models and transformers are integrated to improve multi-level classification. Additionally, this research will explore transfer learning to improve adaptability and performance on unseen, real-world datasets.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Ramakrishna Reddy Badveli	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Nijaguna Gollara Siddappa		✓				✓		✓	✓	✓	✓	✓		
Sundeep Kumar Kanipakapatnam	✓		✓	✓			✓			✓	✓		✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are openly available in Kaggle at:

- https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci, reference [29].
- https://www.kaggle.com/datasets/ritwikb3/heart-disease-statlog, reference [30].
- https://www.kaggle.com/datasets/aasheesh200/framingham-heart-study-dataset, reference [31].

REFERENCES

[1] B. Zhao, J. Peng, C. Chen, Y. Fan, K. Zhang, and Y. Zhang, "Deep learning-based segmentation and localization in CT angiography for coronary heart disease diagnosis," *IEEE Access*, vol. 13, pp. 57615–57628, 2025, doi: 10.1109/ACCESS.2025.3555991.

[2] J. P. Li, A. U. Haq, S. U. Din, J. Khan, A. Khan, and A. Saboor, "Heart disease identification method using machine learning classification in e-healthcare," *IEEE Access*, vol. 8, pp. 107562–107582, 2020, doi: 10.1109/ACCESS.2020.3001149.

[3] M. Ahmed and I. Husien, "Hybrid machine learning approach for accurate heart disease prediction," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 4, pp. 728–737, 2024, doi: 10.22266/ijies2024.0831.55.

[4] P. Anuradha and V. K. David, "Improved heart diseases risk prediction using optimized super learner ensemble model," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 5, pp. 307–317, 2024, doi: 10.22266/ijies2024.1031.25.

[5] E. E. Tripoliti, T. G. Papadopoulos, G. S. Karanasiou, K. K. Naka, and D. I. Fotiadis, "Heart failure: diagnosis, severity estimation and prediction of adverse events through machine learning techniques," *Computational and Structural Biotechnology Journal*, vol. 15, pp. 26–47, 2017, doi: 10.1016/j.csbj.2016.11.001.

[6] H. Lu *et al.*, "Research progress of machine learning and deep learning in intelligent diagnosis of the coronary atherosclerotic heart disease," *Computational and Mathematical Methods in Medicine*, vol. 2022, pp. 1–14, 2022, doi: 10.1155/2022/3016532.

[7] M. Ozcan and S. Peker, "A classification and regression tree algorithm for heart disease modeling and prediction," *Healthcare Analytics*, vol. 3, 2023, doi: 10.1016/j.health.2022.100130.

[8] S. J. Khiabani, A. Batani, and E. Khanmohammadi, "A hybrid decision support system for heart failure diagnosis using neural networks and statistical process control," *Healthcare Analytics*, vol. 2, 2022, doi: 10.1016/j.health.2022.100110.

[9] S. Mohapatra *et al.*, "A stacking classifiers model for detecting heart irregularities and predicting cardiovascular disease," *Healthcare Analytics*, vol. 3, 2023, doi: 10.1016/j.health.2022.100133.

[10] A. Mir *et al.*, "A novel approach for the effective prediction of cardiovascular disease using applied artificial intelligence techniques," *ESC Heart Failure*, vol. 11, no. 6, pp. 3742–3756, 2024, doi: 10.1002/ehf2.14942.

[11] A. Javaid *et al.*, "Medicine 2032: the future of cardiovascular disease prevention with machine learning and digital health technology," *American Journal of Preventive Cardiology*, vol. 12, 2022, doi: 10.1016/j.ajpc.2022.100379.

[12] S. N. Ali, S. B. Shuvo, M. I. S. Al-Manzo, A. Hasan, and T. Hasan, "An end-to-end deep learning framework for real-time denoising of heart sounds for cardiac disease detection in unseen noise," *IEEE Access*, vol. 11, pp. 87887–87901, 2023, doi: 10.1109/ACCESS.2023.3292551.

[13] S. K. Basha, D. Roja, S. S. Priya, L. Dalavi, S. S. Vellela, and B. V. Reddy, "Coronary heart disease prediction and classification using hybrid machine learning algorithms," in *2023 International Conference on Innovative Data Communication Technologies and Application (ICIDCA)*, Mar. 2023, pp. 1–7. doi: 10.1109/ICIDCA56705.2023.10099579.

[14] V. Chang, V. R. Bhavani, A. Q. Xu, and M. Hossain, "An artificial intelligence model for heart disease detection using machine learning algorithms," *Healthcare Analytics*, vol. 2, Nov. 2022, doi: 10.1016/j.health.2022.100016.

[15] G. N. Ahmad, H. Fatima, S. Ullah, A. S. Saidi, and Imdadullah, "Efficient medical diagnosis of human heart diseases using machine learning techniques with and without GridSearchCV," *IEEE Access*, vol. 10, pp. 80151–80173, 2022, doi: 10.1109/ACCESS.2022.3165792.

[16] K. Drożdż *et al.*, "Risk factors for cardiovascular disease in patients with metabolic-associated fatty liver disease: a machine learning approach," *Cardiovascular Diabetology*, vol. 21, no. 1, 2022, doi: 10.1186/s12933-022-01672-9.

[17] K. Kanagarathinam, D. Sankaran, and R. Manikandan, "Machine learning-based risk prediction model for cardiovascular disease using a hybrid dataset," *Data & Knowledge Engineering*, vol. 140, 2022, doi: 10.1016/j.datak.2022.102042.

[18] I. D. Mienye and N. Jere, "Optimized ensemble learning approach with explainable AI for improved heart disease prediction," *Information*, vol. 15, no. 7, 2024, doi: 10.3390/info15070394.

[19] K. Chandrashekar and A. T. Narayanreddy, "An ensemble feature optimization for an effective heart disease prediction model," *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 2, pp. 517–525, 2023, doi: 10.22266/ijies2023.0430.42.

[20] 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," *International Journal of Intelligent Engineering and Systems*, vol. 17, no. 2, pp. 134–144, Apr. 2024, doi: 10.22266/ijies2024.0430.12.

[21] 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," *Multimedia Tools and Applications*, vol. 81, no. 13, pp. 18155–18179, 2022, doi: 10.1007/s11042-022-12425-x.

[22] P. Nandakumar and S. Narayan, "Cardiac disease detection using cuckoo search enabled deep belief network," *Intelligent Systems with Applications*, vol. 16, 2022, doi: 10.1016/j.iswa.2022.200131.

[23] E. A. Ogundepo and W. B. Yahya, "Performance analysis of supervised classification models on heart disease prediction," *Innovations in Systems and Software Engineering*, vol. 19, no. 1, pp. 129–144, 2023, doi: 10.1007/s11334-022-00524-9.

[24] 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.

[25] 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.

[26] 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 and Health*, vol. 1, no. 2, pp. 70–81, 2024, doi: 10.1016/j.infoh.2024.06.001.
[27] R. J. Suhatril, R. D. Syah, M. Hermita, B. Gunawan, and W. Silfianti, "Evaluation of machine learning models for predicting cardiovascular disease based on framingham heart study data," *ILKOM Jurnal Ilmiah*, vol. 16, no. 1, pp. 68–75, 2024, doi: 10.33096/ilkom.v16i1.1952.68-75.
[28] A. Singh, H. Mahapatra, A. K. Biswal, M. Mahapatra, D. Singh, and M. Samantaray, "Heart disease detection using machine learning models," *Procedia Computer Science*, vol. 235, pp. 937–947, 2024, doi: 10.1016/j.procs.2024.04.089.
[29] Cherngs, "Heart disease Cleveland UCI," *Kaggle*. 2020. Accessed: Nov. 15, 2024. [Online]. Available: https://www.kaggle.com/datasets/cherngs/heart-disease-cleveland-uci
[30] Ritwik, "Heart disease Statlog," *Kaggle*. 2023. Accessed: Nov. 15, 2024. [Online]. Available: https://www.kaggle.com/datasets/ritwikb3/heart-disease-statlog
[31] A. Bhardwaj, "Framingham heart study dataset," *Kaggle*. 2022. Accessed: May 10, 2025. [Online]. Available: https://www.kaggle.com/datasets/aasheesh200/framingham-heart-study-dataset
[32] P. Muthulakshmi and M. Parveen, "Z-score normalized feature selection and iterative African Buffalo optimization for effective heart disease prediction," *International Journal of Intelligent Engineering and Systems*, vol. 16, no. 1, pp. 25–37, 2023, doi: 10.22266/ijies2023.0228.03.

BIOGRAPHIES OF AUTHORS

Ramakrishna Reddy Badveli is research scholar in S.E.A. College of Engineering and Technology, Bengaluru, Karnataka. He has teaching experience of 20 years. He has completed M.E. in Computer Science Engineering from Anna University, Coimbatore, Tamilnadu in the year 2011. He has published 5 research papers in reputed international and national journals. He can be contacted at email: itramkey@gmail.com.

Nijaguna Gollara Siddappa is working as professor and HOD, Department of Information Science and Engineering, S.E.A. College of Engineering and Technology, Bangalore, India. He received the B.E. degree in Information Science and Engineering from S.J.M. Institute of Technology, Chitradurga, in 2006, the M.Tech. degree in Computer Science and Engineering from BVB College of Engineering and Technology, Hubli, in 2009, and the Ph.D. degree from the Department of Computer Science and Engineering, Visvesvaraya Technological University, Belagavi, Karnataka, in 2020. He has 16 years of teaching experience. His research interests include cloud computing, information security, trust management, and machine learning. He has published 26 articles in international journals and SCI/Scopus/WSI journals. He is currently guiding three research scholars at Visvesvaraya Technological University, Belagavi. Additionally, he serves as a reviewer for several journals and conferences. He has 13 publications in national/international conferences. He has 6 book chapters/books in reputed publications. He has general chair/session chair for many national/international conferences. He is a member of IEEE, a member of FIE, a member of CSI and IAENG. He can be contacted at email: nijagunags@gmail.com or nijagunags@seaedu.ac.in.

Sundeep Kumar Kanipakapatnam is working as professor and dean in Sri Venkateswara College of Engineering and Technology, Chittoor, Andhra Pradesh. He has teaching experience of 22 years. He has completed his Ph.D. in the area of image processing from Jawaharlal Nehru Technological University, Anantapuramu, Andhra Pradesh in the year 2015. He has completed M.E. Computer Science and Engineering from Anna University, Tamilnadu. He is recognized as a research supervisor for Ph.D. in Visveswaraya Technological University, Bengaluru, Karnataka. He has awarded Ph.D. for four research scholars under his guidance. He has published 33 research papers in reputed international and national journals. He acquired six patents, published two books, and one book chapter. He can be contacted at email: sundeepkumarkk@gmail.com.