Library Progress International *Print version ISSN 0970 1052*
Vol.44 No. 3, Jul-Dec 2024: P. 11285-11296 *Online version ISSN 2320 317X*

Original Article *Available online at www.bpasjournals.com*

HEART DISEASE PREDICTION USING IMROVED-PSO AND ENSEMBLE CLASSIFIER

[1]Venkata Maha Lakshmi N*,[2] Ranjeet Kumar Rout

[1] Department of Computer Science & Engineering, National Institute of Technology, Srinagar
[2] Department of Computer Science & Engineering, National Institute of Technology, Srinagar

How to cite this article: Venkata Maha Lakshmi N, Ranjeet Kumar Rout (2024) Heart Disease Prediction Using Imroved-Pso And Ensemble Classifier. *Library Progress International,* 44(3), 11285-11296.

ABSTRACT
Heart disease has recently become a leading cause of death worldwide, making the prediction of cardiovascular diseases a challenging task in clinical data analysis. With the rapid increase in the size and complexity of medical datasets, automated models leveraging data mining and machine learning techniques are proving essential for clinicians to make accurate and timely decisions. Key challenges in heart disease prediction include imbalanced datasets and the insufficient significance of certain features. The goal of this research is to propose effective feature optimization and classification methods to enhance heart disease prediction, facilitating early diagnosis for medical professionals. Initially, the min-max normalization technique is applied to rescale data from the Shahid Rajaei hospital and UCI Cleveland datasets. Then, optimal features and feature subsets are identified using the Mutual Information technique and the Improved Particle Swarm Optimization (IPSO) algorithm. Finally, the optimized feature subsets are fed into an ensemble classifier to predict the presence or absence of heart disease. Experimental results show that the proposed IPSO-based ensemble model achieved accuracy rates of 98.41% and 97.40% on the UCI Cleveland and Shahid Rajaei hospital datasets, respectively, outperforming traditional machine learning techniques.

KEYWORDS
Ensemble classifier, Heart disease prediction, Improved particle swarm optimization, Min-max normalization.

1. Introduction

In recent years, Heart disease has become a major global health concern, affecting millions of people worldwide [1-2]. Breathlessness, muscular weakness, and swelling feet are common signs of heart disease. Due to a number of risk factors, including diabetes, high blood pressure, high cholesterol, irregular pulse rate, and other contributory diseases, identifying heart disease can be difficult.[3-4]. Researchers are currently focusing on developing more accurate methods of heart disease prediction, as existing diagnostic methods often fall short in early detection due to technical limitations such as slow processing times and low accuracy [5-6]. Diagnosing and treating cardiac disease becomes especially challenging in areas where access to medical professionals and modern technology is limited.[7]. Traditionally, a clinician's diagnosis of heart disease has been made using the patient's medical history, physical examination, and symptom analysis. [8]. However, these approaches are often inaccurate, and the analysis of large datasets is computationally intensive and costly [9-10]. A machine learning classifier-based non-invasive diagnosis approach has been developed in order to overcome these challenges. [11]. To improve the effectiveness of a model, feature optimization and data balancing are essential since the model needs appropriately balanced data for testing and training. [12].
This article is organized as follows: Section 2 presents proposed methodology on heart disease prediction. Sections 3 and 4 present the mathematical derivations and experimental analysis of the proposed IPSO-based ensemble model. The conclusions of this study are summarized in Section 5.

2. Proposed Methodology

The proposed diagnostic system is primarily composed of five key phases. The various steps for this proposed

method are outlined below:
- Shahid Rajaei Hospital and UCI Cleveland dataset are used for experimental study. Min-max normalization was then applied to the data to rescale and normalize it, making the data easier to understand.
- The Mutual Information technique was used to select relevant and discriminative features for the decision-making process, and the optimal features are selected using the Improved Particle Swarm Optimization (IPSO) algorithm. This reduced the dimensionality of the data, improving model complexity and storage efficiency. In order to guarantee convergence to the global optimum and so solve the feature optimization problem, the IPSO algorithm used a fitness function that was created by using the method of a linear weighting approach.
- The optimized feature subsets were then used in an ensemble classifier to predict heart disease. The ensemble classifier included decision trees, logistic regression, naïve Bayes classifiers, Support Vector Machine (SVM) with linear and Radial Basis Function (RBF) kernels and K-Nearest Neighbor (KNN).
Based on Matthews Correlation Coefficient (MCC), specificity, accuracy and sensitivity the performance of the IPSO-based ensemble model was assessed. The Fig.1 shows the proposed method.

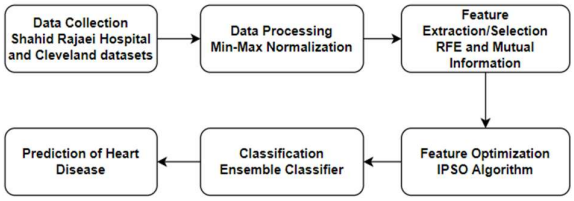

Figure 1 Flowchart of the proposed model

2.1 Data collection

In the context of heart disease prediction, this study evaluates the performance of the proposed IPSO algorithm in conjunction with an ensemble classifier using datasets from Shahid Rajaei Hospital and the UCI database. The UCI Cleveland dataset comprises 303 medical samples, with six of them containing missing data and 297 having complete data. This dataset encompasses a total of 76 features, out of which 13 key features have been identified for conducting various diagnostic tests. Similarly, the dataset from Shahid Rajaei Hospital consists of 303 samples, with a total of 59 recorded features [13]. The features are categorized into four groups: laboratory test results, cardiac electrocardiography data, demographics, physical evaluations, and ECG data.

2.2 Data pre-processing

After gathering the data, the raw data from the UCI and Shahid Rajaei Hospital datasets were rescaled and standardized using the min-max normalization technique. This method rescales the raw data to fit within predefined lower and upper bounds, typically ranging from 0 to 1 or -1 to 1. Equation (1) presents the mathematical expression for the min-max normalizing process.

$$\left(Z'_{i,n}\right) = \frac{Z_{i,n} - min(Z_i)}{max(Z_i) - min(Z_i)} (nMax * nMin) + nMin \quad (1)$$

Here, 'min' and 'max' denote the minimum and maximum values of the i^{th} attribute, 'n' denotes the total number of attributes, and 'nMin' and 'nMax' indicate the lower and upper bounds for rescaling.

2.3 Feature extraction and selection

A powerful feature selection method called Recursive Feature Elimination (RFE) groups feature vectors based on importance. [14-15]. In each iteration, it assesses the significance of each feature and removes less relevant ones, thereby enhancing computational efficiency and the training process. RFE explores different feature subsets and identifies the relative significance of each feature during the stepwise elimination process. The relevance of feature vectors is evaluated using a fixed threshold value of 80% total importance, defined using RFE analysis, in both the Shahid Rajaei Hospital dataset and the UCI dataset. Within the UCI dataset, 10 out of the 13 features contribute a combined significance of 81% to the target, leading to the selection of these 10 features. Similarly, 30 of the 59 features in the Shahid Rajaei Hospital dataset contribute to 80% of the target variable's significance. Random feature subsets are generated by considering all possible combinations of features. To improve disease classification and reduce dimensionality, IPSO uses feature subsets with stronger mutual information among the entire feature set as input.

2.4 Feature optimization

Feature optimization [16], sometimes referred to as feature engineering or feature selection, is a vital step in the developing of machine learning models. To increase the effectiveness and performance of the model, it is to identify or generating the most important and useful features (input variables) from the given data. The process of selecting a subset of the features that are most important from the initial feature set is known as feature selection. In turn, this may speed up training and enhance model generalization by reducing the dimensionality of the data. Typical approaches for choosing features include: Filtering techniques: Statistical measurements are used in these strategies to rank features according to how well they correlate with the target variable. The features that score highest are chosen.
Wrapper techniques: These techniques, which include forward selection and backward elimination, result in training the model with several feature subsets and choosing the subset that produces the best model performance. Embedded techniques: Feature selection is a component of the model training process in embedded methods. Methods that fall under this category include feature importance scores in tree-based models and L1 regularization in linear models. Feature optimization offers the following advantages: enhanced performance of the model: Removing unnecessary information can result in simpler, easier-to-understand models with higher predicted accuracy. Reduced overfitting: High-dimensional datasets with many irrelevant features can lead to overfitting. Feature optimization can mitigate this issue. Faster training and prediction: smaller feature sets result in shorter training times and faster predictions, which is important for real-time applications. It often requires experimentation and validation to determine the best set of features for a given problem.

2.5 Particle Swarm Optimization

The population-based stochastic optimization method known as Particle Swarm Optimization (PSO) was motivated by the flocking social behavior of birds. It was developed by Dr. James Kennedy and Dr. Russell Eberhart in the mid-1990s [17]. PSO is widely used in solving optimization and search problems across various domains, including engineering, finance, machine learning, and biology. PSO draws its inspiration from the collective behavior of animals, particularly birds and fish, that cooperate and communicate to find the best path or location. In a swarm, individuals adjust their positions based on their own experiences and the experiences of their peers. In order to discover the best solution, PSO keeps track of a population of viable solutions, known as particles, that travel through the search area. Particles are defined by their position and velocity throughout the search space, and each one represents a possible solution. The optimization problem at hand is defined by an objective function that needs to be minimized or maximized. Evaluating this function with the particle's current position yields the fitness of each particle.In PSO, particles adjust their velocities and positions iteratively to search for the optimal solution. This adjustment is guided by two main principles:
Cognitive Component (Personal Best): It is the aim of every particle to move toward its lowest fitness position, which is also its own best position.
Social Component (Global Best): Particles also consider the best-known position among their neighbors within the swarm and aim to move in that direction.
PSO has several important parameters, including the swarm size (number of particles), the maximum number of iterations, and parameters that control the influence of personal and social components on particle movement, such as cognitive and social learning rates. PSO continues its iterations until certain termination criteria are met. A maximum number of iterations, reaching a desirable level of fitness, or showing little improvement over a period of iterations are examples of common termination criteria. Maintaining diversity within the swarm is crucial to avoid premature convergence to suboptimal solutions. Strategies like constriction coefficients or adaptive techniques can be used to balance exploration and exploitation.
PSO is applied to various optimization problems, such as function optimization, feature selection, parameter tuning of machine learning models, neural network training, and even dynamic optimization problems. It is known for its simplicity, effectiveness, and ability to find good solutions in complex and high-dimensional optimization problems. PSO's parallel nature and ability to handle noisy or nonlinear objective functions make it a valuable tool in the field of optimization.

2.6.2 Formulation of PSO approach

Two populations comprise over the classical PSO algorithm: one represents the individual best position of each particle, and the other the global best position. The location vector and the velocity vector are the two fundamental characteristics of every particle. In a dynamic search environment, particles undergo random movements with velocity adjustments based on their own historical experiences and the experiences of their fellow particles. Equations (2) and (3) are used to update these particle locations and velocities.

$$v_{id}(t+1) = \omega \times v_{id}(t) + c_1 \times r_1 \times \ [p_{id}(t) - x_{id}(t)] + c_2 \times r_2 \times \ \left[p_{gd}(t) - x_{id}(t)\right] \quad (2)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (3)$$

Here, c1 and c2 stand for the acceleration coefficients, and the symbol ω denotes the inertia weight.

Furthermore, random numbers dispersed throughout the interval (0, 1) are indicated by r1 and r2. Figure 2. displays the flowchart of the conventional PSO algorithm.

In the conventional PSO algorithm, Particle position and velocity are defined as $X_i = [x_{i1}, x_{i2}, .x_{ij} ...x_{iD}]$ and $V_i = [v_{i1}, v_{i2}, .v_{ij}...v_{iD}]$, where $x_{ij} \in [x_{min}, x_{max}]$ and $v_{ij} \in [v_{min}, v_{max}]$ within a D-dimensional vector space. $P_g = (P_{g1}, P_{g2}, .P_{gj}...P_{jD})$ is the particle's global best position and $P_i = (P_{i1}, P_{i2}, .P_{ij} ...P_{iD})$ is the particle's current best position

2.5.1 Inertia weight

The inertia weight, represented by the symbol "ω," was first proposed by Shi and Eberhart in 1998, to govern the balance

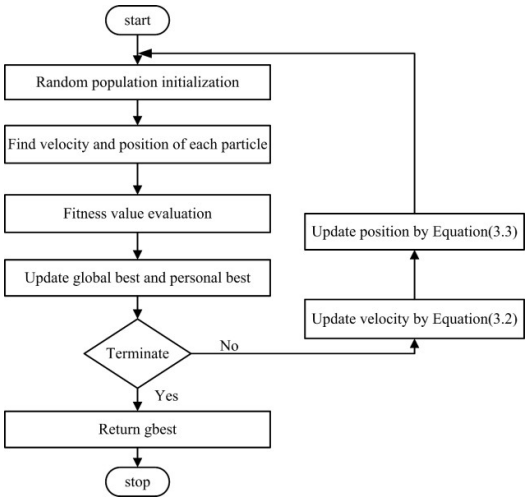

Figure. 2 Flowchart of the Conventional PSO algorithm

between swarm exploration and exploitation and to eliminate the need for velocity clamping. When larger values of 'ω' are employed, the swarm maintains greater diversity, facilitating exploration. Conversely, a smaller 'ω' encourages localized exploitation. However, if 'ω' becomes excessively small, the swarm's ability to explore diminishes.

In this study, a linearly decreasing inertia weight approach is adopted. Initially set at typically 0.9, the inertia weight is gradually reduced to a lower value, often around 0.4. To implement this linear reduction of the inertia weight, the following equation is employed.

$$\omega^t = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{t_{max}} * t \quad (4)$$

where

ω_{min} beginning inertia weight,

ω_{max} final inertia weight

ω^t inertia at the t^{th} iteration

2.6 IMPROVED PARTICLE SWARM OPTIMIZATION

In this study, the improved PSO algorithm is employed to select the best subset of features, thereby enhancing classification accuracy. This goal is accomplished by developing a fitness function that takes into account both the Kappa coefficient and the accuracy of the logistic regression through the use of a linear weighting technique. This mathematical representation is illustrated in Equation (5).

$$\textit{Fitness} = (\alpha \times (1.0 - \textit{accuracy}) + (1.0 - \alpha) \times (1 - (\textit{number of selected features} / \textit{total features}))) + (\alpha \times (1 - \textit{Kappascore})) \quad (5)$$

The weight parameter α, with α set to 0.88, is essential for ensuring balance between the number of selected features and the classification error rate. A logistic regressor is used to calculate the error rate of test datasets and is represented by 1 minus the logistic regression accuracy, while the reliability of the model is indicated by 1 minus the Kappa score.

The IPSO algorithm's parameter settings are as follows:

- Number of epochs=100
- Maximum number of iterations=100
- Problem size=Equal to the total number of features
- Population size= the number of selected features

- Inertia weight (w):=0.88
- Acceleration coefficient c_1:=2
- Acceleration coefficient c_2=2

The Pseudocode of the IPSO algorithm is given below.

Pseudoncode of the improved PSO algorithm

Input: Heart disease feature set and population size

Output: Optimal feature set

Initialize particles in the population and Logistic regression classifier

Calculate the importance of every feature based on logistic regression prediction

While iteration *I*>Maximum iterations

For $I = 1$ **to n**

Calculate fitness for every particle using equation (5)

Update particle's local best position

End for

Update particle's global optimal position

For $I = 1$ **to n**

Update particle's velocity

Update particle's global best position

End for

End while

3. Classification

Regression and classification are two common machine learning applications that use ensemble classifiers. Enhancing prediction robustness, decreasing overfitting, and raising model accuracy are some of the notable advantages. The choice of the ensemble method depends on the problem at hand and the characteristics of the data. Several algorithms are used in this ensemble classifier, such as Decision Tree, Naïve Bayes, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Logistic Regression. Ultimately, the best results are determined through a straightforward averaging technique, where each algorithm contributes its prediction, and the final result is an average of these individual predictions.

3.1 Support Vector Machine

A Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks[18]. The SVM classifier typically achieves classification by identifying an optimal hyperplane and subsequently fine-tuning the margin to maximize classification accuracy. Initially, the training data is represented as $f_i \in R^m$ (where i = 1, 2, 3, ..n), and the corresponding expected outcomes are represented as $q_i \in +1, -1$ (where i = 1, 2, 3, ...n). Consequently, Equation (6) is used to represent the target function of the hyperplane t(x).

$$t(x) = (U^T \times f) + b_v \qquad (6)$$

In the SVM classifier, U represents the standard vector, while b_v stands for the bias value. To ensure proper classification of all the provided training data, it is essential to set a sufficiently large separation margin, where 2/U is set as the maximum value.

3.2 K-Nearest Neighbor

The K-Nearest Neighbor (KNN) algorithm is a machine learning classification and regression technique that is primarily used for supervised learning tasks. During initialization, set a value for K[19], this denotes how many of nearby neighbors to take into consideration for each prediction.

Determine the distance between each new data point and every other data point in the training set before predicting anything about a previously unseen data point.

Typically, the distance metric employed is Euclidean distance, which is defined mathematically in Equation (7)

$$\text{Euclidean distance} = \sqrt{\sum_{i=1}^{n} (f_i - q_i)^2} \qquad (7)$$

Where

- n typically represents the number of samples or data points in your dataset
- f_i denotes individual training samples or data points
- q_i represents testing samples or data points for which you want to make predictions or classifications.

3.3 Naïve Baye's classifier

The Naïve Bayes classifier is a popular machine learning algorithm used for classification tasks. Naïve Bayes [20] uses probability theory to make predictions. It determines which class a data point is most likely to belong to by calculating the likelihood of each class, then choosing the class with the highest probability to be predicted class. Furthermore, the Naïve Bayes classifier can be represented using a simple Bayesian network.This system is characterized by the use of Gaussian probability functions, as outlined in Equations (8) and (9).

$$Pr\ (f_1, f_2, \dots f_n \mid C)\& = \Pi_{i=1}^{n} pr\ (f_i \mid C) \tag{8}$$

$$Pr\ (f_i \mid C_i)\& = \frac{pr\ (C_i|f)\times pr\ (f)}{(C_i)} \tag{9}$$

Finally, the method of association probability is utilized to classify the test data into their respective classes, as formally defined in Equation (10)

$$C_{nb} = argmax\ pr\ (C_l)\Pi_{i=1}^{n} pr\ (f_i \mid C_z) \tag{10}$$

3.4 Logistic regression

Logistic regression is a statistical and machine learning technique used for binary classification and, with modifications, for multiclass classification and regression analysis. The main applications of logistic regression [21] are in binary classification tasks, in which the target variable has two distinct classes or outcomes, usually represented by the numbers 0 and 1. The logistic function, commonly known as the sigmoid function, is used by the logistic regression model to simulate the probability of the positive class. The logistic function S(z) is defined as

$$S(z) = 1 / (1 + e^{(-z)}) \tag{11}$$

where 'z' is a linear combination of the input features and model coefficients.

3.5 Decision tree

A decision tree is a popular machine learning algorithm used for both classification and regression tasks. A decision tree is a hierarchical structure consisting of nodes [22]. It starts with a single root node and branches into multiple decision nodes and leaf nodes. Each leaf node represents a class label (for classification) or a value (for regression), and each decision node represents a decision or test on a feature.The algorithm divides the data into two or more subgroups at each decision node by choosing a feature and a threshold. This choice is based on a criterion that is looking for to minimize the mean squared error (for regression) or maximize information gain (for classification). Gini impurity and entropy are common splitting criteria for decision trees in classification, whereas mean squared error is utilized in regression.These criteria measure the impurity or error associated with the data subsets created by a split. Decision trees are prone to overfitting, meaning they can capture noise and exhibit high variance. Pruning is a method of reducing a tree's complexity by cutting off branches that don't significantly increase prediction accuracy. This enhances the model's capacity for generalization. Equation (12) defines the entropy value.

$$Entropy\ = \Sigma_{i,j=1}^{n} pr_{ij} log_2\ pr_{ij} \tag{12}$$

4. Experimental Results

An extensive performance analysis of the suggested model, which combines IPSO with an ensemble classifier, is provided in this section. Several experiments are conducted to validate the efficacy of this paradigm. The objective of the first experiment is to look into how accuracy, kappa score, and the number of selected features are affected by swarm size (N) and iteration count. In the second experiment, three additional meta-heuristic algorithms—SSA, GWO, and SCA—will be compared to see how well IPSO performs. In the third experiment, six distinct classifiers—logistic regression, KNN, SVM (RBF), SVM (linear), naive Bayes, and decision trees—are used to benchmark the model's performance. Two well-known datasets, the UCI Cleveland and the Shahid Rajaei Hospital datasets, are used to thoroughly test the suggested ensemble-based IPSO model. The evaluation metrics used include specificity, accuracy, sensitivity, and the Matthews Correlation Coefficient (MCC). A summary of the parameter settings for the proposed PSO, SSA, GWO, and SCA algorithms is given in the table 1.

Table 1 parameter settings of the algorithms.

Table 1 Parameter settings of the algorithms

Algorithm	Parameter setting
Common settings	Population size N=10
	Maximum Iterations=100
	No.of runs=30
	LB=0
	UB=1
SSA	C1, C2 are randomly distributed
GWO	a=[2,0]
SCA	a=[2,0] decrease linearly from 2 to 0
IPSO	w=0.9, c1,c2=2 V_{max}=6

To assess and validate the performance of the proposed model, this work employs four widely-recognized performance metrics—specificity, accuracy, sensitivity, and the Matthews Correlation Coefficient (MCC). A rigorous 10-fold cross-validation procedure is utilized to evaluate the model, using 80% of the data for training and the remaining 20% for testing.The specificity, sensitivity, MCC, and accuracy is calculated by using the following Equations (13), (14), (15), and (16). $Specificity = \frac{TN}{TN+FP} \times 100$

(13)

$$Sensitivity = \frac{TP}{TP+FN} \times 100 \quad (14)$$

$$MCC = \frac{TP \times TN - \quad \times FN}{\sqrt{(TP+FP)(TP+F\quad)(TN+FP)(TN+F\quad)}} \quad (15)$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+F} \times 100 \quad (16)$$

Table 2 provides a performance comparison of IPSO with SSA, GWO, and SCA using the UCI Cleveland and Shahid Rajaei datasets.

4.1 Experiments on UCI Cleveland dataset

The UCI Cleveland Heart Disease dataset, often referred to as the "Cleveland dataset" or simply "UCI Heart Disease dataset," is a widely used dataset in the field of machine learning and medical research. The dataset typically includes patient information and medical measurements, including attributes

Table 2 Average accuracy results of IPSO compared with other metaheuristic algorithms

Algorithm	Average accuracy		Average std	
Dataset	UCI Cleveland dataset	Shahid Rajaei hospital dataset	UCI Cleveland dataset	Shahid Rajaei hospital dataset
Proposed	98.41	97.40	0.0010	0.0013
SSA	96.42	94.85	0.0016	0.0021
GWO	95.84	95.21	0.0023	0.0019
SCA	96.81	95.43	0.0017	0.0018

Like age, sex, blood pressure, cholesterol levels, and the presence of specific symptoms or conditions. The target variable is a binary classification label that indicates if heart disease is present or not.

Nevertheless, the purpose of using the ensemble classifier is to combine the best outcomes from the several classifiers, leading to an impressive overall performance of 87.91% accuracy, 80% specificity, 71.36% MCC, and 92.06% sensitivity on the UCI Cleveland dataset. Similarly, Table 4 provides an overview of the classifiers' performance in conjunction with feature selection and optimization techniques (such as PSO with accuracy as fitness, PSO with error rate as fitness, and IPSO). Remarkably, the ensemble classifier with the IPSO method produced impressive outcomes: 98.41% prediction accuracy, 95.39% MCC, 97.05% specificity, and 98.37% sensitivity.Table 3 Performance of the classifiers without feature selection and optimization algorithms on the

UCI Cleveland dataset.

Predictive model	Accuracy(%)	Specificity(%)	Sensitivity(%)	MCC(%)
Logistic regression	85.71	73.33	91.80	67.03
KNN	83.51	71.42	88.88	60.94
SVM (RBF)	86.81	80	89.39	67.80
SVM (linear)	87.91	78.57	92.06	71.36
Naïve Bayes	70.32	70.32	70.32	67.90
Decision tree	79.12	64.28	85.71	50.51
Ensemble	**87.91**	**80**	**92.06**	**71.36**

Bold values indicates that the methods of best results

The proposed ensemble classifier with IPSO exhibited a substantial improvement in heart disease prediction, demonstrating a significant 6% to 16% enhancement compared to other combinations on the UCI Cleveland dataset. For a visual representation of these results, please refer to Figure 3, which illustrates the performance of IPSO in conjunction with various algorithms on the UCI Cleveland dataset.

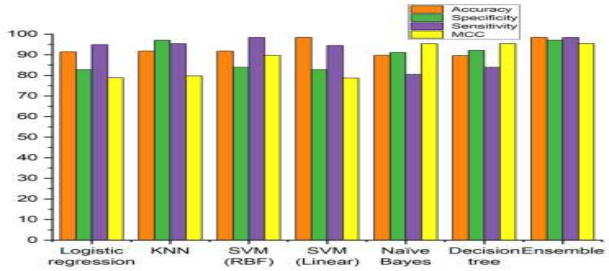

Figure. 3 Performance of the classifiers with IPSO selected features on Clevland dataset

4.2 Experiments on Shahid Rajaei hospital dataset

The performance of various classifiers is assessed using the Shahid Rajaei Hospital dataset, with a focus on metrics such as specificity, sensitivity, MCC, and accuracy.

The ensemble classifier delivered impressive performance in heart disease prediction, achieving a high accuracy of 93%, an MCC of 71%, a specificity of 85.54%, and a sensitivity of 95%. Table 5 provides a comprehensive overview of the performance of these classifiers without feature selection and optimization algorithms, on the Shahid Rajaei Hospital dataset.In order to choose features for the classifiers, PSO with accuracy (fitness), PSO with error rate (fitness), and IPSO algorithms are used. Subsequently, the performance is assessed based on metrics such as specificity, sensitivity, MCC, and accuracy. When coupled with the ensemble classifier, the IPSO method showed notable performance gains in heart disease prediction when compared to the classic PSO algorithm, PSO with accuracy (fitness), and PSO with error rate (fitness). Table 6 shows the remarkable outcomes of the ensemble-based IPSO model on the Shahid Rajaei Hospital dataset, which include a specificity of 96.08%, accuracy of 97.40%, sensitivity of 98%, and MCC of 96.29%. By utilizing the IPSO algorithm, the suggested feature optimization method significantly lowers the feature count, which lowers storage needs and optimizes the ensemble-based IPSO model. Figure 4 shows the performance of the proposed model using the Shahid Rajaei dataset.

Table 4 Performance of the classifiers with feature selection using PSO and IPSO algorithms on the UCI Cleveland dataset.

Predictive model	Accuracy(%)	Specificity(%)	Sensitivity(%)	MCC(%)
Feature selection with PSO algorithm+accuracy (fitness)				
Logistic regression	91.37	82.70	94.87	78.72
KNN	91.70	97.01	95.35	79.72
SVM (RBF)	91.72	83.85	98.35	89.62
SVM (linear)	98.38	82.72	94.43	78.63
Naïve Bayes	89.63	91.02	80.43	95.33
Decision tree	89.52	92.05	83.87	95.37
Ensemble	**98.38**	**97.01**	**98.35**	**95.36**
Feature selection with PSO algorithm + Error rate (fitness)				
Logistic regression	91.40	82.74	94.89	78.77
KNN	91.73	97.04	95.36	79.76
SVM (RBF)	91.77	83.89	98.36	89.64
SVM (linear)	98.40	82.74	94.44	78.68
Naïve Bayes	89.69	91.07	80.44	95.36
Decision tree	89.55	92.06	83.89	95.38
Ensemble	**98.40**	**97.04**	**98.36**	**95.38**
Feature selection with IPSO algorithm				
Logistic regression	91.41	82.75	94.90	78.78
KNN	91.74	97.05	95.37	79.77
SVM (RBF)	91.78	83.90	98.37	89.65
SVM (linear)	98.41	82.75	94.45	78.69
Naïve Bayes	89.70	91.08	80.45	95.37
Decision tree	89.56	92.07	83.90	95.39
Ensemble	**98.41**	**97.05**	**98.37**	**95.39**

Bold values indicates that the methods of best results

Table 5: Performance of the classifiers without feature selection on the Shahid Rajaei hospital dataset

.

Without feature selection and optimization				
Predictive model	**Accuracy (%)**	**Specificity (%)**	**Sensitivity (%)**	**MCC (%)**
Logistic regression	81	78	82	60
KNN	93	85.54	86	71
SVM (RBF)	79	68	92	56
SVM (Linear)	80.40	76	84.10	60.67
Naïve Bayes	76	63.80	95	48
Decision tree	78.70	75	78	53
Ensemble	**93**	**85.54**	**95**	**71**

Table 6: Performance of the classifiers with feature selection using PSO and IPSO algorithms on the Shahid Rajaei hospital dataset

Predictive model	Accuracy(%)	Specificity(%)	Sensitivity(%)	MCC(%)
Feature selection with PSO algorithm+accuracy (fitness)				
Logistic regression	89.36	81.30	90.82	80.94
KNN	91.95	95.23	91.35	83.14
SVM (RBF)	93.21	87.98	97.94	88.96
SVM (linear)	97.36	87.71	93.96	78.96
Naïve Bayes	89.96	96.01	80.42	96.24
Decision tree	89.77	91.96	88.84	93.30
Ensemble	**97.32**	**96.04**	**97.94**	**96.26**
Feature selection with PSO algorithm+error rate (fitness)				
Logistic regression	89.38	81.32	90.88	80.98
KNN	91.98	95.27	91.37	83.18
SVM (RBF)	93.22	87.98	97.98	88.98
SVM (linear)	97.38	87.73	93.98	78.99
Naïve Bayes	89.98	96.06	80.47	96.27
Decision tree	89.80	91.98	88.88	93.31
Ensemble	**97.38**	**96.06**	**97.98**	**96.27**
Feature selection with IPSO algorithm				
Logistic regression	89.40	81.34	90.90	81
KNN	92	95.29	91.38	83.20
SVM (RBF)	93.23	88	98	89
SVM (linear)	97.40	87.75	94	79
Naïve Bayes	90	96.08	80.49	96.29
Decision tree	89.82	92	88.90	93.33
Ensemble	**97.40**	**96.08**	**98**	**96.29**

Bold values indicates that the methods of best results

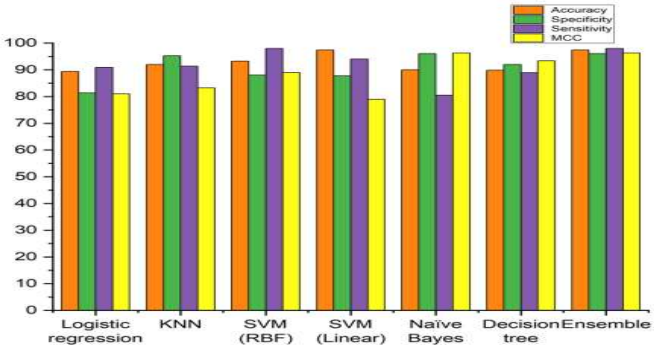

Fig..4 Performance of the classifiers with IPSO selected features on Shahid Rajaei hospital dataset.

5.3 Comparative study and discussion

Table 7 provides a summary of the comparison analysis between the existing models and the proposed ensemble-based IPSO model.

The ensemble-based IPSO model consistently performed superior in heart disease prediction on both datasets than these other models.

It's critical to emphasize that feature optimization is a key component of this research work, as it greatly reduces the computational complexity of the suggested model by employing the IPSO method to choose and optimize discriminative features from the gathered datasets. As previously stated the ensemble-based IPSO model efficiently removes redundant and unnecessary features, which increases prediction accuracy while concurrently speeding up execution.

Consequently, the proposed model is well-suited for real-time applications, enabling the early analysis of disease risks for more timely and effective treatment.Furthermore, the proposed ensemble-based IPSO model exhibited impressive efficiency, with prediction times of 2.14 seconds for the Shahid Rajaei Hospital dataset and 0.78 seconds for the Cleveland dataset. These results demonstrate its superiority over other comparative machine learning classifiers in terms of computational speed.

Table 7 Performance comparison of the proposed model with existing models

Models	Dataset	Accuracy (%)	Sensitivity (%)	Specificity (%)
Imperialist competitive approach with KNN [23]	Cleveland	94.03	96.27	90.36
	Shahid Rajaei hospital	94.43	96.57	91.06
Random search algorithm with random forest classifier [24]	Cleveland	93.33	95.12	89.79
Homogeneous ensemble technique 25]	Cleveland	93	91	92
Genetic programming with random forest [26]	Cleveland	97.52	97.29	97.70
Ensemble based IPSO model	**Cleveland**	**98.41**	**98.37**	**97.05**
	Shahid Rajaei hospital	**97.40**	**98**	**96.08**

5. CONCLUSION

An ensemble-based IPSO model for the diagnosis of heart disease is presented in this research study. Initial data preparation involves min-max normalization, enhancing data consistency across datasets. Feature selection is executed using Recursive Feature Elimination (RFE) combined with mutual information to identify the most relevant features. The suggested IPSO algorithm selects optimum feature subsets, which reduces the model's complexity and storage needs.For heart disease prediction, an ensemble classifier that incorporates decision tree, naive Bayes, KNN, SVM, and logistic regression classifiers is provided. The final predictions are determined by aggregating results from these classifiers through simple averaging. This comprehensive approach enhances the accuracy and efficiency of heart disease diagnosis.Using the Cleveland and Shahid Rajaei Hospital datasets, the performance of the proposed ensemble-based IPSO model was evaluated in terms of MCC, specificity, accuracy, and sensitivity. Our results demonstrate exceptional prediction accuracy, with scores of 98.41% on UCI Cleveland and 97.40% on Shahid Rajaei Hospital dataset. These outcomes outperform comparative models, although it is important to note that combining a large number of classifiers can be computationally expensive.

1.

2. References

1. Parsi A, Byrne D, Glavin M, Jones E (2020) Heart rate variability feature selection method for automated prediction of sudden cardiac death. Biomed Signal Process Control. https://doi.org/10.1016/j.bspc.2020.102310
2. Baghel N, Dutta M, Burget R (2020) Automatic diagnosis of multiple cardiac diseases from PCG signals using convolutional neural network. Comput Methods Programs Biomed. https://doi.org/10.1016/j.cmpb.2020.105750
3. Fitriyani N, Syafrudin M, Alfian G, Rhee J (2020) Hdpm: an effective heart disease prediction model for a clinical decision support system. IEEE Access 8:133034–133050. https://doi.org/10.1109/ACCESS.2020.3010511
4. Ali F, El-Sappagh S, Islam SMR, Kwak D, Ali A, Imran M, Kwak K (2020) A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion. Inf Fusion. https://doi.org/10.1016/j.inffus.2020.06.008
5. Gopu M, Swarnalatha P (2021) Optimal feature selection through a cluster-based DT learning (CDTL) in heart disease prediction. Evol Intell. https://doi.org/10.1007/s12065-019-00336-0
6. Khateeb N, Usman M (2017) Efficient heart disease prediction system using k-nearest neighbor classification technique, pp 21–26. https://doi.org/10.1145/3175684.3175703
7. Malav A, Kadam K, Kamat P (2017) Prediction of heart disease using k-means and artificial neural network as hybrid approach to improve accuracy. Int J Eng Technol 9:3081–3085. https://doi.org/10.21817/ijet/2017/v9i4/170904101
8. Raza K (2019) Improving the prediction accuracy of heart disease with ensemble learning and majority voting rule, pp 179–196. https:// doi.org/10.1016/B978-0-12-815370-3.00008-6

9. Zhang D, Chen Y, Chen Y, Ye S, Cai W, Jiang J, Xu Y, Zheng G, Chen M (2021) Heart disease prediction based on the embedded feature selection method and deep neural network. J Healthc Eng 2021:1– 9. https://doi.org/10.1155/2021/6260022
10. Sharma P, Saxena K (2017) Application of fuzzy logic and genetic algorithm in heart disease risk level prediction. Int J Syst Assur Eng Manag. https://doi.org/10.1007/s13198-017-0578-8
11. Cholker, A. K., & Tantray, M. A. (2020). Strain-sensing characteristics of self-consolidating concrete with micro-carbon fibre. Australian Journal of Civil Engineering, 18(1), 46-55 https://iopscience.iop.org/article/10.1088/1742-6596/1921/1/012095/met
12. Faiayaz Waris S, Koteeswaran S (2021) Heart disease early prediction using a novel machine learning method called improved k-means neighbor classifier in python. Mater Today Proc. https://doi.org/ 10.1016/j.matpr.2021.01.570
13. Islam S, Jahan N, Khatun ME (2020) Cardiovascular disease forecast using machine learning paradigms, pp 487–490. https://doi.org/ 10.1109/ICCMC48092.2020.ICCMC-00091
14. Darst B, Malecki K, Engelman C (2018) Using recursive feature elimination in random forest to account for correlated variables in high dimensional data. BMC Genet. https://doi.org/10.1186/s12863- 018-0633-8
15. Richhariya B, Tanveer M, Rashid AH (2020) Diagnosis of Alzheimer's disease using Universum support vector machine based recursive feature elimination (USVM-RFE). Biomed Signal Process Control 59:101903. https://doi.org/10.1016/j.bspc.2020.101903
16. Bansal J (2019) Particle Swarm Optimization, pp 11–23. https://doi. org/10.1007/978-3-319-91341-4_2
17. Chopard B, Tomassini M (2018) Particle Swarm Optimization, pp 97– 102. Springer, Cham. https://doi.org/10.1007/978-3-319-93073- 2_6
18. Pisner DA, Schnyer DM (2020) Chapter 6 - support vector machine. In: Mechelli A, Vieira S (eds.) Machine Learning, pp 101–121. Academic Press. https://doi.org/10.1016/B978-0- 12-815739-8.00006-7 . https://www.sciencedirect.com/science/ article/pii/B9780128157398000067
19. Zhang Y, Cao G, Wang B, Li X (2019) A novel ensemble method for k-nearest neighbor. Pattern Recognit. https://doi.org/10.1016/ j.patcog.2018.08.003
20. Salmi N, Rustam Z (2019) Naïve Bayes classifier models for predicting the colon cancer. IOP Conf Series Mater Sci Eng 546:052068. https://doi.org/10.1088/1757-899X/546/5/052068
21. Cholker, A. K., & Tantray, M. A. (2021). Electrical resistance-based health monitoring of structural smart concrete. Materials Today: Proceedings, 43, 3774-3779. https://www.sciencedirect.com/science/article/abs/pii/S2214785320390581
22. Suryakanthi T (2020) Evaluating the impact of Gini index and information gain on classification using decision tree classifier algorithm. Int J Adv Comput Sci Appl. https://doi.org/10.14569/IJACSA.2020.0110277
23. Faris H, Mafarja MM, Heidari AA, Aljarah I, Al-Zoubi AM, Mirjalili S, Fujita H (2018) An efficient binary salp swarm algorithm with crossover scheme for feature selection problems. Knowl-Based Syst 154:43–67. https://doi.org/10.1016/j.knosys.2018.05.009
24. Hafez A, Zawbaa HM, Emary E, Hassanien AE (2016) Sine cosine optimization algorithm for feature selection. In: 2016 international symposium on innovations in intelligent systems and applications (INISTA), 1–5
25. Vivekanandan T, Iyenger NCSN (2017) Optimal feature selection using a modified differential evolution algorithm and its effectiveness for prediction of heart disease. Comput Biol Med. https://doi.org/10.1016/j.compbiomed.2017.09.011
26. Kim J-K, Kang S (2017) Neural network-based coronary heart disease risk prediction using feature correlation analysis. J Healthc Eng 2017:1–13. https://doi.org/10.1155/2017/2780501