International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V71I6P240

Volume 71 Issue 6, 406-420, June 2023

Original Article

Prediction of Heart Disease and Diabetes (HDD) using Self-Adaptive Particle Swarm Optimization- Based Random Forest Algorithm(SAPSORF)

S. Usha[1*], S. Kanchana[1]

[1]*Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology, Kattankulathur, TamilNadu, India*

[*]*Corresponding Author : us3648@srmist.edu.in*

Received: 06 January 2023 Revised: 10 May 2023 Accepted: 27 May 2023 Published: 25 June 2023

Abstract *- Heart Disease and Diabetes (HDD) is widely recognized as the most lethal conditions afflicting humans. Preventing and treating HDDs requires accurate risk assessment at an early stage. Experts have created several machines learning-based intelligent systems to diagnose HDD automatically to address this problem. However, their classification accuracy is still below par. Furthermore, most existing machine learning models are tailored toward predicting certain diseases, such as cardiovascular disease, diabetes, lung illness, etc. For this reason, a classifier that can reliably predict the occurrence of several diseases is desirable. This paper proposes the Self-Adaptive Particle Swarm Optimization-based Random Forest Algorithm (SAPSORF) to predict cardiovascular and diabetes disease. The performance of the modified Random Forest Algorithm is enhanced via a bio-inspired algorithm, namely Self-Adaptive Particle Swarm Optimization. SAPSORF enriches sampling and dimensionality reduction phases of modified random forest. This study assesses the effectiveness of the proposed classifier on two distinct datasets: the Cardiovascular Disease Dataset and the PIMA Indian Diabetes Dataset. The evaluation results indicate that the proposed classifier surpasses existing classifiers in terms of accuracy when it comes to classification tasks.*

Keywords *- Diabetes, Heart disease, Optimization, Particle swarm, Random forest.*

1. Introduction

Healthcare management, often called healthcare administration, directs, organizes, and controls healthcare organizations, including clinics, clinic networks, hospitals, and other medical institutions. Professionals in this field are tasked with a wide range of responsibilities, including but not limited to ensuring that all departments function smoothly. Qualified individuals are hired, that information is disseminated effectively throughout the organization, desired results are achieved, and all available resources are used effectively [1]. In the healthcare industry, there are both generalists and specialists. While specialists like marketers, financiers, policy analysts, and accountants manage more specialized units inside more prominent organizations, generalists are responsible for the overall operation of their respective facilities.

Managing the healthcare system entails planning, coordinating, and directing all aspects of medical care delivery. Healthcare industry managers are responsible for their facility's operation, including finances, human resources, patient records, etc. [2]. These professionals in healthcare administration do not provide direct treatment to patients but rather assist with the operational aspects of the healthcare system. Those who work in health care administration must be flexible to handle the wide variety of daily tasks since they are responsible for managing the clinic's or department's finances and overall operations [3].

Machine learning (ML), a branch of AI techniques, is a relatively new phenomenon in the healthcare industry. Nonetheless, ML applications have expanded, and more data scientist and ML specialist jobs and educational opportunities are becoming available. Still, some physicians doubt ML 's efficacy and some worry about its use. Developing biased algorithms is a significant cause for concern since it can lead to incorrect results and the continued existence of treatment disparities [4]. The opaque nature of some ML algorithms can also make it hard to analyze results, evaluate quality, or spot mistakes. There is a

possibility of assuming causation when none focuses on the ML relationships. Misuse or misuse of ML methods by those who do not appreciate their limits is a cause for worry. Any computer system is only as good as the best human process in terms of the criteria used for the conclusion [5], which is why the criticism that ML has an inconsistent standard is not an ML-specific fault [6].

Clinical decision support is one use of ML in healthcare that aids doctors in carrying out routine activities that computers can do more efficiently than humans. Compared to humans, machines excel at endurance, tolerance, boredom, and detecting events beyond the range of human senses. For example, humans excel in inductive reasoning and intuitively understand complicated patterns like facial recognition [7]. This research work believes that to use ML algorithms' potential fully, action must be taken utilizing carefully built ML algorithms. Clinicians must be included in the design process and must be aware of the limits of the ML systems they employ. Working together, the therapist and the system should be more effective than each could be separately [8].

1.1. Problem Statement

In recent years, several automated diagnostic strategies have been developed for various diseases and conditions, includingHDD. Optimization-based ML techniques are used in HDD detection on several datasets, and this research field has become widely recognized as having a significant influence on medical research. Multiple ML models are used to classify or predict the illness diagnosis. Using ML techniques, large amounts of genetic data get evaluated rapidly. By incorporating and analyzing medical data and training algorithms, better pandemic predictions can be made. Dataset analysis yields many takeaways that shed light on the significance of the individual variables and their interconnections. In the last few years, healthcare production has accumulated a massive trove of patient data. However, researchers and doctors are not using this data well, and it is not used to diagnose diseases. Concerns about the healthcare sector's ability to provide patients with prompt, accurate diagnoses of their conditions and effective treatment have been raised. Harmful and unacceptably destructive results follow from a wrong diagnosis.

Bio-inspired algorithms offer a novel heuristic approach by modeling natural processes as restricted optimization [9], [10], [11]–[18], [19], [20]. These algorithms draw inspiration from various biological phenomena, such as the behaviour of social insects, evolutionary processes, and neural networks, to solve complex optimization problems. By emulating the inherent characteristics and mechanisms found in nature, bio-inspired algorithms provide a unique perspective for tackling optimization problems. They harness the power of evolutionary processes, swarm intelligence, or neural network dynamics to search for optimal solutions within a given problem space. One of the critical advantages of bio-inspired algorithms is their ability to explore vast solution spaces efficiently. These algorithms leverage the collective intelligence of a population, imitating the cooperative behaviour of social insects like ants or bees to navigate through the problem domain and converge on promising solutions. This collaborative search strategy often leads to finding near-optimal or even optimal solutions, particularly in large-scale or complex optimization problems where traditional methods struggle.

1.2. Objective

The primary purpose of this research is to introduce the Self-Adaptive Particle Swarm Optimization-Based Random Forest Algorithm ($SAPSORFA$). This optimization-based classifier takes biological inspiration from the natural characteristics of particles (i.e., birds) for predicting cardiovascular disease and diabetes.

2. Literature Review

In this section, the literature about the classification of cardiovascular disease and diabetes is categorized into two distinct groups. The first group encompasses algorithms specifically designed for the type of cardiovascular disease. The second group focuses on algorithms developed for the classification of diabetes. A comprehensive overview of the existing research and advancements in each domain can be obtained by classifying the associated literature into these two groups. This approach allows for focused analysis and comparison of the algorithms employed for disease classification in these two significant healthcare areas.

2.1. Heart Disease Classifiers

"Empirical Study" [21] is conducted to find the classification algorithm's performance towards its mining and prediction. The two models, association and classification, are used to perform mining on the heart disease dataset where the time limit was ten years and fell between 2006 and 2016. "Efficient Heart Disease Prediction System" [22] is proposed to effectively find the protocol to detect the level of risk in CAD patients. Protocols are given high priority depending on user requirements. Parameter plays a significant role in defining the protocols for the patients. "Multilayer Perceptron Neural Network" [23] is proposed to predict heart disease with more accuracy, which helps in providing earlier treatment. A neural network selects the appropriate features where the back-propagation is applied to train the algorithm. "Disease Risk Prediction" [24] is proposed to overcome the issue of missing values in e-health records. It involves cleaning data and taking actions to transform it to complete data from incomplete data. kNN and Naïve Bayes are applied for the classification of structured records. CNN is used for the prediction of heart disease. "Medical Choice Backing Framework ($MCBF$)"

[25] is proposed to avoid artificial intelligence-enabled devices for the prediction of CAD. To increase the accuracy of classification, $MCBF$ utilizes SVM and ANN. The performance of $MCBF$ is analyzed using the two benchmark datasets, Cleveland and statlog.

"Classification Associative Rules" [26] are proposed to provide one step more care to the patients affected with heart diseases by predicting the same earlier. Associative rule mining is used for prediction and is compared with J48, kNN, and Naïve Bayes for classification accuracy. "Disease Prediction with Deep Learning" [27] is proposed to perform prediction on input classes and transform the same to output. CNN is used for classifying images, whereas ANN is used for training the dataset. The dataset consists of sick and normal heartbeat sounds. "Heart Failure Risk Prediction Model" [28] is proposed to predict the survival rate of heart failure patients using e-healthcare information. This model attempts to create patterns using logistic regression. Data extraction is performed on e-health records by incorporating co-morbidities of patients. The probabilistic loss function is utilized to find errors in classification. "Prediction Model of Embryonic Development" [29] is proposed to detect pregnancy loss at an early stage using fetal heart rate (FHR). Initially, the importance of features is analyzed from the samples of ongoing pregnancy and early pregnancy loss. Different classification algorithms are applied to classify the FHR. Residual analysis was further done to estimate the pregnancy outcome. "Novel Ventricular Arrhythmia Prediction" [30] is proposed to predict ventricular-oriented diseases linked to cardiac attacks. To minimize the computing level in prediction, features are given more preference, and an appropriate selection strategy has been applied. Selected features are fed into classifiers where kNN achieves maximum classification accuracy.

To improve the accuracy of risk assessment for cardiovascular disease, the "Boosting Support Vector Machine ($BSVM$)" [46] was suggested. The Cleveland data set was used to evaluate the effectiveness of $BSVM$. Data cleaning was performed using the listwise approach to eliminate the six blanks. The experiment was unaffected by this technique since the sample size was too small, and the random selection process produced no biases. The boosting technique chooses essential features to minimize effort and maximize accuracy. The train/test split approach divides the data into two groups: training and testing. After that, SVM is used to both train and evaluate the dataset's records. As such, we use a linear kernel with a C-value of 0.05. Predicting cardiovascular disease with precision using a "Swarm-Artificial Neural Network" ($S-ANN$) [32] is suggested. The model is an ANN that takes as input all available medical records about heart disease. To build NNs with a fixed population size, a random weight is applied to each neuron in the network. $S-ANN$ consists of three independent stages, or phases, of data processing. To begin, an ANN is constructed using the data collected. At the start of each iteration, the $S-ANN$ creates a new three-layer feedforward ANN with randomly generated weights. In the second phase, a back-propagation method is used to change the weights of each $S-ANN$ that was formed at random. Once the third phase is complete, the swarm implements a stochastic weight modification mechanism. In the end, the population of ANNs is evaluated for its performance, and a winner is chosen. The weight and bias the victorious neuron have gained are distributed among the remaining neurons in the population according to a stochastic function.

2.2. Diabetes Classifiers

"Deep-Transfer Learning Approach" [33] is proposed for diagnosing people with diabetes with the help of heart rate signals acquired from Electrocardiogram data. It was employed with two-dimensional signals in which the model's weighting is applied to one-dimensional heart rate signals. All the indications are converted into frequency spectrum images and are used to pre-trained models like $VggNet$, $ResNet$, $DenseNet$, and $AlexNet$. Among those, the DenseNet model achieves a higher classification rate for detecting people with diabetes through heart rate signals. "Random Forest Method (RFM)" [34] is applied to discover Single Nucleotide Polymorphisms among diabetic people by providing a weight ranging from 0 to 1 for every attribute. Its performance is compared with machine learning techniques, namely, Logistic Regression and Support Vector Machines. The results indicate that RFM, along with the k-Nearest Neighbor method, performs better than the RFM method.

"Stochastic Gradient Descent Technique" [35] is proposed for predicting people with diabetes by training the different models. The Synthetic-Minority Oversampling-Technique ($SMOTE$) is applied to handle the imbalanced class of models to examine the classifier's efficiency. The model which was implemented was able to identify the diabetic and non-diabetic distribution present in the dataset. "Support Vector Machine" [36] is proposed for type 2 diabetes classification using discriminatory and non-discriminatory processes. SVM is modified to remove the irrelevant genes from the data set. The gene network has also been created to define the purpose of diabetic causes. A pathway study was made to discover the gene involved in type II diabetes. "Glucose Level Measurement Algorithm" [37] is proposed to increase the accuracy of the hematocrit (Hct) level compensation method. An adaptive Calibration curve with linear filter prediction and SVM was employed to reduce the bias in glucose concentration level. Chronoamperometry was also executed to validate their variations. Their performance was measured by considering the valid blood samples.

"Diabetic Monitoring System" [38] is demonstrated for detecting and classifying thermographic images based on temperature differences. It is a step-by-step process that uses deep learning techniques for dividing the visible spectrum images using the Mask $R-CNN$ model. Gaussian functions, filtering, and convolutions are performed for segregating the ulcerous from the necrotic zones. "Localization Model with Deep Neural Network" [47] is designed to retrieve the patches trained during the classification process. Standard Diabetic Retinopathy Database, Calibration Level 1 dataset was used and tested on different databases to measure the classification efficiency. The efficient process with two CNN models is defined to evaluate the performance under the ROC curve for diabetic screening. "Efficient Algorithm" [40] is developed for the early diagnosis of diabetics in which thermograms are used to read the temperatures. All the features related to texture and temperatures are fetched from the various regions, and analysis for extracting the elements from ipsilateral and contralateral areas is done. The SVM classifier was used to divide the standard and ulcer-affected regions. This model was evaluated and is used to identify the ulceration pre-signs at an early stage. "Weighted Paths into Convolutional Neural Network" [41] is optimized with the Back-Propagation method for effectively classifying type 2 diabetes. The output features were averaged for better efficiency, and its results show better performance and work more effectively for diabetic recognition compared with another existing algorithm. "Classification of Diabetic Retinopathy" [42] is proposed to detect fundus automatically. Preprocessing, segregation, and classification are performed using the model. Irrelevant noises are removed, and essential areas of images are segmented using a histogram-based approach. Synergic Deep Learning (SDL) model was employed for classifying the images at different levels of security.

Prediction of diabetes using a "Modified Support Vector Machine ($MSVM$)" [43] is provided. To do this first processing, $MSVM$ employs a modified version of the principal component analysis (PCA) method to extract features from the input medical record. Using $MPCA$, we may determine the eigenvectors of the covariance matrix and then project the data onto a new subspace with the exact dimensions as the original. Selecting the most important and valuable features for further classification is a job for $MSVM$. The best recoverable attributes are used as the basis for the classifications and forecasts. Finding the optimal hyperplane uses a custom approach to get the highest feasible separation margin that meets the classification criteria. The experimental categorization error can be reduced by using this classification for linear and nonlinear data analysis. Once the hyperplane of maximum separation has been found, the input vector is projected onto a higher-dimensional space. Hidden layers in deep neural networks are used to build a system for predicting diabetes called "Deep Learning for Predicting Diabetes ($DLPD$)" [44]. By utilizing dropout regularization, the issue of overfitting can be mitigated. In the case of DLPD, a binary cross-entropy loss function is employed, with its parameters optimized to accomplish the specific task at hand. Normalization layers are incorporated to preserve the model's prior knowledge while enhancing its adaptability to new data. Dropout, as a technique, randomly resets a fraction of units to zero after each training update. Adding previous gradients is accomplished by setting a constant scaling factor for the hyperparameters. The sum of gradients can be expressed through recursion as the average damping of all preceding square gradients. It is important to note that the current mean and gradient solely influence classification times without any relation to other factors.

"Revived Ant Colony Optimization-Based Adaboost Algorithm (RACOAA)" [45] is a hybrid algorithm that combines two different algorithms to improve the prediction accuracy of heart disease and diabetes. The algorithm starts by selecting the most relevant features from the dataset using Ant Colony Optimization (ACO). This nature-inspired algorithm mimics the behaviour of ants to find the shortest path between two points. ACO is used to identify the features that have the most significant impact on prediction accuracy. Once the most critical components are selected, the algorithm uses Adaptive Boosting (AdaBoost) to construct a classification model based on these features. AdaBoost is a machine learning algorithm that combines multiple weak classifiers to create a robust classifier. In this case, AdaBoost is used to train a model that can predict whether a patient has heart disease or diabetes based on the selected features. The algorithm then repeats the process of choosing the essential elements using ACO and training a model using Adaboost until the desired prediction accuracy is achieved. RACOAA has been shown to outperform other state-of-the-art algorithms regarding prediction accuracy and feature selection.

3. Self-Adaptive Particle Swarm Optimization-Based Random Forest Algorithm

This section initially discusses the modified version of random forest and then self-adaptive particle swarm optimization, which enhances the classification accuracy.

3.1. Modified Random Forest

Random attribute selection has become a typical feature of the Decision Tree (DT) training process, and it will result in the incorporation of bagging techniques with DT-based learning in RF. Despite its modest processing requirements and straightforward design, the RF method has become an effective strategy in many real-world applications. Combining more than one DT will result in the enhancement of RF towards the predictive power of the conventional DT approach.

In the classification world, DT fits the mold of the standard single classifier. To put it to use for classification, this research first constructs a DT model from the available training data and then employs it to categorize unseen sample data. Pruning involves removing branches from a DT model to make it simpler and more resistant to overfitting. First, starting at the top-level root node, child nodes are formed iteratively downwards until the lowest-level leaf node is reached; all things are done under the specified feature assessment criteria.

Iterative Dichotomiser 3 (ID3) is a technique used in DT node splitting that utilizes the gain of information obtained strategy to choose which feature to employ; then, it calculates information gain using the entropy of the data. Assuming that P is a random variable with a finite range of values, the probability distribution is calculated using Eq.(1):

$$M(P = p_s) = M_s, \quad s = 1,2, \dots., t, \tag{1}$$

The random variable P entropy is therefore described as Eq.(2):

$$L(P) = -\sum_{s=1}^{t} M_s log M_s. \tag{2}$$

Eq.(3) describes the convergence of DT's generalization error in all RF.

$$\{\_t\backslash rightarrow\backslash infty\^{}\{lin\}\}\{MH\}\^{}\backslash ast = M\_\{pq\}\left(\begin{matrix} M_\Theta(a(P,\Theta) = Q) - maxM_\Theta(a(P,\Theta) = W) < 0 \\ w \neq Q \end{matrix}\right) \tag{3}$$

Which t refers to the count of trees in the RF.

The concept behind RF classification are:

- a number of prototypes are taken from the initial training set using bootstrap sampling.
- Each sample has the same number of samples as the training set.
- For a samples, a DT models are made, and a number of classifications are discovered.
- Based on the results of a, perform selection on every record to decide how it should be classified.

3.2. Sampling

When constructing Random Forest (RF) algorithms, bagging sampling is commonly employed to create subgroups from the complete training set. Each training subset is typically approximately two-thirds the size of the original training set. The sampling process is performed randomly and repeated several times, resulting in varying duplication among the samples in each training subset. Also, it prevents local optimum solutions from being produced by DT in theRF.

There are two interpretations of "Bagging" in the development of "RF," which are:

- A potential benefit is that it increases the RF algorithm's precision during the classification process.
- To avoid anomalous data and noisy data entering the training subset, roughly 31% of the samples are excluded from the training sample after being shuffled around.

Following the two points above improves the DT performance over any dataset. The set of vectors present in RF can minimize the available connectivity, minimizing the overfitting and maximizing the precision with which the RF performs classification.

When dealing with continuous variables in RF, it is usually practised to discretize the parameters into threshold distinct intervals. However, because of the strong correlation between the algorithm's complexity after partitioning and the pace at which the dataset is reduced, it takes a long time to study and compute the nodes splitting standard, significantly impacting the algorithm's execution performance. It follows that the RF method has material that has to be improved, specifically the fuzzification of continuous variables.

3.3. Dimensionality

Compared to more common low-dimensional data, high-dimensional datasets typically include more characteristics. Classifying high-dimensional data is always challenging and assists in mining the data for perfect classification. For high-dimensional data, traditional classification algorithms suffer from issues of (i) lengthy processing times, (ii) overfitting, and (iii) poor classification accuracy. This research attempts to provide an intelligent automated methodology for extracting and optimizing RF features to fix the RF method's low classification performance and significant generalization error when used with high-dimensional data.

Selecting features aims to have a classification model created by the feature subset. F', which will attain an increased level of classification accuracy. Feature selection may be characterized as a process in which an ideal list of features $F' = \{f_{1'}, f_{2'}, f_{3'}, \dots, f_{(s-2)'}, f_{(s-1)'}, f_{s'}\}$ is obtained from the set of features $F = \{f_1, f_2, f_3, \dots, f_{(e-2)}, f_{(e-1)}, f_e\}$ and F' can contain the majority of the information from the initial stage, among which $e' < e$.

The initial ideology is to look at how information is dispersed by employing the balanced RF technique, which can be seamlessly included in RF. The $SMOTE$ algorithm is an enhanced version of the traditional random upsampling technique. Initially, it replicates the negative samples randomly, leading to multiple duplicated new datasets and making it more challenging to address the data imbalance issue.

The $SMOTE$ algorithm initially looks for the nearest negative instances surrounding each minus sample before generating a new negative pattern between an independent sample and its neighbours. The interpolation synthesis is expressed as Eq.(4):

$$M_{sw} = p_s + rand(0,1) \times (q_{sw} - p_s), \quad (4)$$

$p_s(s = 1,2,3, \dots., (t-2), (t-1), t)$ is the sum of negative samples, and t is the number of negative samples; $q_{sw}(w = 1,2, \dots, c)$is the c closest neighbour sample of p_s; M_{sw} does the sample synthesize a new sample p_s; and $rand(0,1)$ is any pseudorandom value that falls between 0 and 1.

But there are two problems with the $SMOTE$ algorithm:

- Initially, when selecting the nearest neighbour, there is an inevitable loss of vision regarding how much a significance to take.
- New finite difference algorithm for continuous variables that are based on p^2 correction is done.

Determine the intervals H_{sw} by counting the number of decision attributes a that fall between two consecutive intervals of a fixed attribute value, as shown in Eq.(5).

$$H_{sw} = b_s \times \frac{U_w}{T}. \quad (5)$$

where b_s is calculated as $\sum_{w=1}^{a} D_{sw}$ and it acts as the sample count in the interval s, U_w is calculated as $\sum_{s=1}^{a} D_{sw}$ and it is the sample count in class w, and T indicates the overall sample number in two intervals next to each other.

If a group's theoretical degree H_{sw} is less than 8, then that group should be joined or merged with any neighbouring groups until its theoretical degree level attains H (i.e., 8) or until there is only one piece of data in intervals, and then the a is calculated.

In both cases, the value of p^2 is determined using Eq.(6) or Eq.(7). In case of $a < 2$, Eq.(6) is applied.

$$p^2 = \prod \sum_{s=1}^{2} \sum_{w=1}^{a} \left(\frac{(|D_{sw} - H_{sw}| - 0.5)^2}{H_{sw}}\right), \quad w = 1; s = 1,2. \quad (6)$$

In the case of $a \geq 2$, Eq.(7) is applied.

$$p^2 = \prod \sum_{s=1}^{2} \sum_{w=1}^{a} \frac{(D_{sw} - H_{sw})^2}{H_{sw}} \quad w = (1,2,3, \dots., (a-2), (a-1), a); s = 1,2, \quad (7)$$

Where a is the total number of classes available for the dependent variable, s is the total number of intervals between the two adjacent intervals, and D_{sw} is the total number of class w data between the s intervals

Once all continuous attribute variables have been processed in this manner, the dataset has been reduced to its minimal viable form. As a result, the continuous variables are successfully discretized. Eq.(8) is applied to determine the Y value that merges intervals.

$$D = \frac{p_d^2 - p^2}{\sqrt{2r}}. \quad (8)$$

Assuming the following inequality criteria are true, the pseudo data is valid.

$$p(1 - \pi) \leq p' \leq p(1 + \pi). \quad (9)$$

With the original genuine data at p, the random number range at n, and the fictitious data at $3p'$, DT has C leaf nodes, a decision area of $B_c(c = 1,2,3, \dots.(C-2), (C-1), C)$ underneath the c -th node, and a decision value of U_c (constant) that represents the fraction of targets correctly classified inside that area. The bottom-up approach is used to obtain a new sample which will be assigned to DT discrimination, with the probability that it corresponds to the target class represented by the constant. U_c. Eq.(10) assist in discriminating function towards generating leaf nodes.

$$gu(P) = \sum_{c=1}^{C} U_c S(P, B_c). \quad (10)$$

This research uses the Wrapper approach for feature selection due to its benefits over other methods. It is common to practice normalizing data before feeding it into a model. Normalization's strength lies in its ability to remove the dimensional impact of disparate data. The higher evaluation index may mitigate the effect of the lower index test on the model when the information is not standardized. The max-min normalizing approach is the most popular choice among the researchers. The index is often placed in a range that falls between -1 and 1.

3.4. Self-Adaptive Particle Swarm Optimization

To improve its exploratory and exploitative abilities of traditionalPSO, $SAPSO$ is proposed to make adaptive changes to the particle neighbourhoods. Neighbourhood structures govern the swarm's information flow. More selection pressure acts on larger interconnected topologies, while smaller, less interconnected ones promote greater diversity. Each swarm particle is attracted to different particles based on the swarm's structure. So, they modify the paths that particles take while it flies.

With even and uneven particles, $SAPSO$ varies the swarm's searching strategy. Particles employ a different mechanism of updates similar to another swarming approach. However, there is no limit to the number of examples. The surrounding area can expand and contract on the fly. Based

on local fitness, an adaptive process modifies the quantity and size of neighbouring particles. As a result, various degrees of discovery and use are possible for these particles. All even particles usually move uniformly, which is why particles can follow paths aligning with the dependencies already in the optimization problem. In general, particles have great exploitation potential. Individual particles can address intricate issues involving inter and intra-dependent factors. However, a particle might get scrambled if it experiences attractions from several different sources simultaneously. When particles in a highly linked topology have multiple attractors, the searching region contracts to the region immediately surrounding the swarm's centre position. $SAPSO$ employs nonuniform particles with overlapping but distinct functions to address these deficiencies.

3.5. Particle Update

All particles' acceleration coefficients are calculated at once during the update process. $SAPSO$ maintains not one but two databases of potential acceleration coefficients. $\aleph$. stands for the theoretical acceleration coefficient for a non-spherical particle. At the iteration number a, the P_s for each uneven particle, chooses an accelerating coefficient from this pool that best suits the needs. Eq.(11) assist in deriving the particle's acceleration coefficient.

$$u_s(a) = Max[Min\{\aleph_{bs} + \varsigma(0,0.2), U^{MAX}\}, U^{MIN}], \quad (11)$$

where in $\aleph_{bs}$ is a coefficient of acceleration chosen at random from $\aleph$, $\varsigma(0,0.2)$ is a random integer created using a Cauchy distributed with a mean of 0, U^{MIN} and U^{MAX} are the lowest and highest values for the coefficient of acceleration, respectively. In a nutshell, the acceleration coefficients stored in the archives are somewhat adjusted for each particle that is not perfectly uniform. After calculating the particle's acceleration coefficient, Eq.(12) is used to revise its speed.

$$\begin{aligned} r_s^y(a) = {} & nr_s^y(a-1) \\ & + u_s(a) b_s^y \left(M_{Xs.y}^y(a-1) \right. \\ & \left. - p_s^y(a-1)\right), \end{aligned} \quad (12)$$

The index of P_s attractions are defined by $X_s = [X_{s,1}, \ldots., X_{s,Y}]$.

3.6. Velocity Synchronization

A new velocity for the even particle P_s is determined after each iteration by comparing the particle's $mbest$ with t_s number of samples from the z_s^{th} level of the tree. Particle neighborhood structure is defined by the t_s and z_s, which are modified during the search process.

In the updating phase, two different acceleration coefficients are applied. Each particle's acceleration coefficient is determined at each iteration. The ξ represents the coefficient of particles having potential velocity. Eq.(13) and Eq.(14) determines the acceleration coefficients for each particle P_s during iteration a.

$$U_s^1(a) = Max[min\{\xi_{bs}^1 + \varsigma(0,0.2), U^{MAX}\}, U^{MIN}], \quad (13)$$

$$U_s^2(a) = Max[min\{\xi_{bs}^2 + \varsigma(0,0.2), U^{MAX}\}, U^{MIN}], \quad (14)$$

The acceleration coefficients in the tuple ξ_{bs} are chosen at random from a set of ξ. After determining the acceleration coefficients, the particle's speed is modified using Eq.(15).

$$\begin{aligned} r_s(a) = {} & nr_s(a-1) + u_s^1(a) b_{s1} \\ & \otimes \left(m_s(a-1) - p_s(a-1)\right) \\ & + e_s \frac{u_s^2(a)}{t_s} b_{s2} \otimes \sum_{o\omega h(d(node(P_s).z_s),t_s)} \left(m_{P^{(o)}}(a-1) \right. \\ & \left. - p_s(a-1)\right) \\ & + (1-e_s) \frac{u_s^2(a)}{t_s} b_{s3} \\ & \otimes \sum_{o\omega h(d(node(P_s).z_s),t_s)} \left(m_{P^{(o)}}(a-1) \right. \\ & \left. - p_s(a-1)\right), \end{aligned} \quad (15)$$

In Eq.(15), the components of the Y-dimensional vectors b_{s1} and b_{s2} are sampled from $o(0,1)$. Element multiplication is denoted as $'\otimes'$. Real numbers e_s and b_{s3} are chosen, and it falls in the range $o(0,1)$. It is important to notice that the random integer determines the value of each scaling scheme e_s, in the case of Eq.(15). To implement linear scaling, random scalars are used rather than random scalar vectors (b_1 and b_2). Eq.(15) is an attempt to integrate the positive aspects of both methods.

After each particle's velocity is updated, it determines the location of each particle. The particle's optimum location is revised based on the fitness function evaluation. $SAPSO$ stores the total number of consecutive failures for each particle in the fail count. After $SAPSO$ successfully changes the positions of the particles, it returns them to their former $mbest$, i.e., via Eq.(3). It is suggested that $t_s = 1$ and $z_s = 2$ are used to implement frequent updates for all even particles like P_s. As a result, P_s still partially loses its mobility from its previous velocity, is attracted to P_s as well as the $mbest$ of the particles directed by the nodes that $d(node(P_s), 2)$. There is a significant likelihood that $d(node(P_s), 2)$ refers to P_s itself. Since, m_s is equal to p_s after successful updates, this means the particle can travel very slowly for a while after a successful update. To counteract this, $SAPSO$ updates the previously promised $mbest$ position (p_s) to the current $mbest$.

Algorithm 1: Self-Adaptive Particle Swarm Optimization-Based Random Forest Algorithm (SAPSORF)

Step 1: Initialize a population of NP particles randomly, where each particle represents a potential solution.
Step 2: Evaluate the fitness of each particle by constructing a random forest with T trees using the particle's selected features. The fitness can be determined by a performance metric such as accuracy, error rate, or other suitable measure.
Step 3: Set the particle's local best (pbest) as each particle's current position and fitness.
Step 4: Set the global best (gbest) as the particle with the best pbest among all the particles.
Step 5: Repeat the following steps until a termination condition is met (e.g., reaching a maximum number of iterations or satisfactory fitness level):
Step 6: For each particle:
(i). Randomly select a particle p' from the neighbourhood of the current particle.
(ii). Update the velocity and position of the particle:
(iii). Calculate the acceleration coefficients based on the previous velocity, the difference between the particle's personal best position (pbest) and its current position, and the influence of the randomly selected particle p'.
(iv). Modify the particle's velocity by combining its previous velocity, the acceleration coefficients, and additional random factors.
(v). Update the particle's position by adding the new velocity to its current position.
Step 7: Adapt the particle neighbourhood structure:
(i). Calculate the performance of each particle based on its fitness.
(ii). Update the neighbourhood structure by adjusting the neighbourhood size or connectivity based on the particles' performance and other parameters.
Step 8: Evaluate the fitness of each particle's new position using the random forest.
Step 9: Update each particle's pbest if the new position yields a better fitness.
Step 10: Update the gbest based on the best pbest among all the particles.
Step 11: Select the particle with the highest fitness as the final solution.
Step 12: Construct a random forest with T trees using the selected features of the final solution.
Step 13: Return to the constructed random forest.

4. About Dataset and Performance Metrics

4.1. Dataset

To evaluate the effectiveness of the suggested classifier, two specific datasets are utilized: the Cardiovascular Disease dataset (CDD) and the PIMA Indians Diabetes Dataset (PIDD). These datasets can be freely obtained from the Kaggle website, a popular platform for sharing and exploring datasets. The CDD encompasses a vast collection of 70,000 records, each containing valuable information relevant to cardiovascular diseases. This dataset likely includes diverse data points such as age, gender, medical history, lifestyle factors, and various health measurements. These records serve as a comprehensive resource for analyzing patterns, trends, and potential risk factors associated with cardiovascular diseases. The PIDD focuses specifically on the health outcomes of the PIMA Indian population. This dataset is comparatively smaller, with 768 records, but it provides a unique perspective on diabetes prevalence and related factors within this ethnic group. The records within the PIDD dataset may include attributes such as glucose levels, insulin measurements, age, BMI, and other clinical indicators relevant to diabetes diagnosis and management.

By utilizing these two distinct datasets, researchers and data scientists can assess the performance of the proposed classifier in different healthcare contexts. The CDD allows for a broader examination of cardiovascular disease prediction and risk assessment on a larger scale, considering a diverse population. Meanwhile, the PIDD offers a more focused analysis of diabetes prediction within the PIMA Indian population, providing insights into potential genetic or cultural factors contributing to the disease. It is worth noting that both datasets are openly accessible on Kaggle, allowing researchers, analysts, and developers to explore and utilize the data for various purposes. The detailed contents of each dataset, including specific variables and their descriptions, can be found in Table 1 for the CDD and Table 2 for the PIDD, providing further clarity and context for utilizing these datasets in evaluating the suggested classifier's performance.

4.2 Performance Metrics

4.2.1. Classification Accuracy

The ratio of accurate predictions to total predictions provides a concise performance summary.

4.2.2. F-Measure

It represents precision and recall metrics harmonic mean.

4.2.3. Matthews Correlation Coefficient

It is a measure of how well the actual values match up with the forecasted ones.

4.2.4. Fowlkes-Mallows Index

It is a tool for determining the degree of similarity between clusters and external assessment approach (i.e., classification). It is a representation of precision and recalls geometric mean.

Table 1. CDD

Feature	Feature Type	Description
Age	Objective	Patient's age.
Height	Objective	The patient's height was measured in centimeters.
Weight	Objective	The patient's weight was measured in kilograms.
Gender	Objective	Patient's gender, where a value of 1 represents male and 2 illustrates female.
Systolic Blood Pressure	Examination	The sudden surge of blood creates pressure within the arteries, quantified in millimetres of mercury (mmHg).
Diastolic Blood Pressure	Examination	Arterial blood pressure denotes the force exerted by the blood on the artery walls between heartbeats, measured in millimetres of mercury (mmHg).
Cholesterol	Examination	The cholesterol level in the patient's bloodstream is categorized into three levels: 1 indicates an average level, 2 indicates an above-normal level, and 3 shows a significantly above-normal level.
Glucose	Examination	The glucose level in the patient's blood is classified into three categories: 1 represents an average level, 2 represents an above-normal level, and 3 represents a significantly above-normal level.
Smoking	Subjective	Whether the patient is a smoker or not.
Alcohol Intake	Subjective	Whether the patient consumes alcohol or not.
Physical Activity	Subjective	Whether the patient engages in any form of physical activity or not.
Cardiovascular Disease Presence/Absence	Subjective	Whether the patient has been diagnosed with cardiovascular disease or not.

Table 2. PIDD

Feature	Feature Type	Description
Preg	Subjective	Number of pregnancies the patient has had.
Gluc	Examination	The glucose level in the patient's blood.
Diastolic Blood Pressure	Examination	Arterial blood pressure is the force the blood exerts on the arterial walls between heartbeats.
Triceps Skin Fold Thickness	Objective	Measurement of triceps skinfold thickness.
Insulin	Examination	Insulin serum level after two hours.
Body Mass Index	Examination	A medical screening tool that assesses the patient's height and weight to identify the presence of excess fat.
Diabetes Pedigree Function	Subjective	Assessment of the patient's familial risk for developing diabetes.
Age	Objective	Age of the patient.
Diabetes Presence/Absence	Subjective	Whether the patient has been diagnosed with diabetes or not.

The calculation of metrics mentioned earlier involves four variables, namely True Positive ($TrPst$), False Positive ($FlPst$), True Negative ($TrNgt$), False Negative ($FlNgt$):

- $TrPst$ (i.e., $True\ Positive$) is a variable that represents the number of medical cases that have been accurately diagnosed as a patient by a medical professional. These cases have been identified and confirmed to have a specific medical condition or disease.
- $FlPst$ (i.e., $False\ Positive$) represents the count of medical cases that have been diagnosed as patients, but not accurately or precisely. These cases may have been diagnosed with a medical condition, but the diagnosis may not have been entirely accurate or incomplete.
- $TrNgt$ (i.e., $True\ Negative$) refers to the count of medical cases that have been accurately diagnosed as healthy. Medical professionals have confirmed that these cases do not have any specific medical condition or disease.
- $FlNgt$ (i.e., False Negative) represents the count of medical cases diagnosed as healthy, but not accurately or precisely. These cases may have been diagnosed as not having any medical condition or disease, but the diagnosis may not have been entirely accurate or may have been incomplete.

These variables are typically used in medical research or clinical settings to evaluate the accuracy and precision of medical diagnoses. By tracking and analyzing the number of accurately diagnosed cases versus inaccurately diagnosed cases, medical professionals can improve their diagnostic accuracy and provide better patient care.

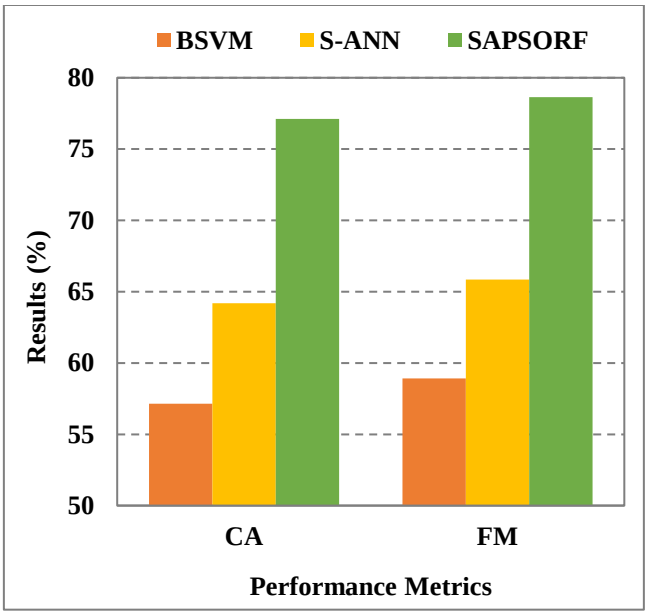

Fig. 1 *CA* and *FM* Analysis on CD Dataset

Table 3. Analysis of CA and FM results on CD dataset

Metrics → / Algorithms ↓	CA	FM
BSVM	57.134	58.920
S-ANN	64.200	65.856
SAPSORF	77.120	78.645

5. Results And Discussion

5.1. CA and FM Analysis of Classifiers with CD Dataset

Figure 1 represents the performance analysis of three classifiers against the CD dataset: BSVM, S-ANN, and SAPSORF. The analysis is based on classification accuracy (CA) and F-measure (FM) metrics.

Table 3 shows the following results: CA for BSVM is 57.134%, and FM is 58.920%. BSVM combines the boosting technique with Support Vector Machines (SVMs) to improve classification performance. However, BSVM demonstrates relatively lower accuracy and F-measure than the other classifiers in this particular analysis. CA for S-ANN is 64.200%, and the FM is 65.856%. S-ANN employs a swarm intelligence approach in artificial neural networks to optimize classification. In this case, S-ANN outperforms BSVM with higher accuracy and F-measure. CA for SAPSORF is 77.120%, and the FM is 78.645%. SAPSORF is a proposed classifier that combines the self-adaptive particle swarm optimization (SAPSO) algorithm with a random forest ensemble. The algorithm adapts the particle swarm optimization to optimize the random forest parameters. Based on the results, SAPSORF achieves significantly higher accuracy and F-measure than BSVM and S-ANN.

Figure 1 shows the performance of the three classifiers, BSVM, S-ANN, and SAPSORF, on the Cardiovascular Disease (CD) dataset. The graph provides a comparative analysis of each classifier's classification accuracy (CA) and F-measure (FM) metrics. The results of the analysis are presented as values on the graph. We can observe that SAPSORF achieves the highest CA and FM among the three classifiers, indicating its superior performance. S-ANN shows moderately higher CA and FM than BSVM, demonstrating its better classification capability. BSVM, on the other hand, exhibits relatively lower CA and FM values. The graph visualises the performance differences between the classifiers, allowing for a quick and easy comparison of their accuracy and F-measure. It shows that SAPSORF outperforms BSVM and S-ANN regarding classification accuracy and F-measure on the CD dataset.

5.2. CA and FM Analysis of Classifiers with PID Dataset

Figure 2 represents a graph based on the results obtained from Table 4, which shows the performance of three different classifiers, namely MSVM, DLPD, and SAPSORF, on the PID dataset. The graph provides a visual comparison of each classifier's classification accuracy (CA) and F-measure (FM). From Table 4, it can be observed that MSVM has a CA of 63.932% and an FM of 67.526%. DLPD, on the other hand, achieves a higher CA of 69.792% and FM of 72.960%. Finally, SAPSORF outperforms MSVM and DLPD, with a significantly higher CA of 85.807% and FM of 87.514%.

MSVM is an existing classifier that utilizes support vector machines with modifications. Although it performs reasonably well, as indicated by its CA of 63.932% and FM of 67.526%, it is not as effective as the other classifiers in this comparison. MSVM employs a margin-based approach for classification, aiming to find an optimal hyperplane that maximally separates the data points. However, in this case, it may struggle to capture the complex patterns and relationships within the PID dataset, resulting in relatively lower accuracy and F-measure values.

Table 4. Analysis of CA and FM Results on PID Dataset

Metrics → / Algorithms ↓	CA	FM
MSVM	63.932	67.526
DLPD	69.792	72.960
SAPSORF	85.807	87.514

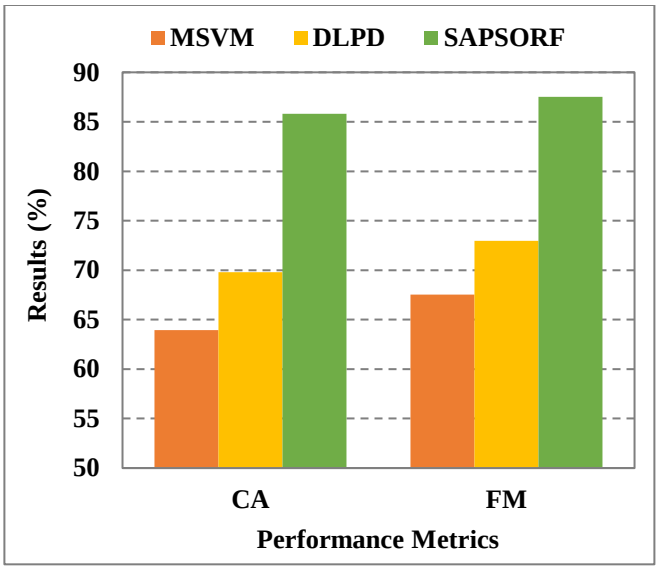

Fig. 2 CA and FM analysis on PID dataset

DLPD is another existing classifier that leverages deep learning techniques for diabetes prediction. It achieves better results than MSVM, with a CA of 69.792% and FM of 72.960%. Deep learning models, such as neural networks, are known for learning complex representations from data, allowing them to capture intricate patterns and dependencies. In the context of diabetes prediction, DLPD likely utilizes multiple hidden layers and nonlinear activation functions to learn and extract relevant features from the PID dataset. This leads to improved accuracy and F-measure compared to MSVM.

SAPSORF is a proposed classifier that outperforms MSVM and DLPD, achieving a significantly higher CA of 85.807% and FM of 87.514%. SAPSORF combines two powerful techniques: particle swarm optimization (PSO) and random forest (RF). PSO is a metaheuristic optimization algorithm that aims to find the optimal solution by simulating the behaviour of a swarm of particles in a search space. It enhances RF's performance, an ensemble learning method that constructs multiple decision trees and combines their predictions. The self-adaptive PSO in SAPSORF adapts its parameters dynamically during the optimization process, improving the quality of the random forest model. This adaptive optimization process likely enables SAPSORF to discover more informative features and make accurate predictions, resulting in the highest CA and FM among the three classifiers.

Figure 2 illustrates the comparative performance of MSVM, DLPD, and SAPSORF on the PID dataset. It demonstrates that SAPSORF, a proposed classifier combining PSO and RF, outperforms MSVM and DLPD regarding classification accuracy and F-measure. DLPD achieves better results than MSVM, showcasing the effectiveness of deep learning techniques in diabetes prediction.

5.3. FMI and MCC Analysis of Classifiers with CD Dataset

Table 5 presents the Fowlkes-Mallows Index (FMI) and Matthews Correlation Coefficient (MCC) values for three classifiers—BSVM, S-ANN, and SAPSORF—let's provide a critical explanation for Figure 3, taking into account the working mechanisms of each classifier. Figure 3 is expected to illustrate the superior performance of SAPSORF over BSVM and S-ANN in classifying the cardiovascular disease dataset based on the higher FMI and MCC values reported in Table 5.

BSVM achieved an FMI of 14.124% and an MCC of 58.924%. The lower values of FMI and MCC suggest that BSVM performed relatively poorly compared to the other two classifiers. BSVM combines multiple weaker classifiers to improve performance. However, in this case, BSVM seems to have struggled to effectively classify the cardiovascular disease dataset, resulting in lower evaluation metrics. S-ANN achieved an FMI of 28.247% and an MCC of 65.860%. These higher values indicate that S-ANN performed better than BSVM. S-ANN combines artificial neural networks with swarm intelligence algorithms to optimize classification. The higher FMI and MCC scores suggest that S-ANN successfully captured the patterns and relationships in the cardiovascular disease dataset, leading to improved classification performance compared to BSVM. SAPSORF achieved the highest values in both FMI (54.129%) and MCC (78.672%) among the three classifiers.

SAPSORF combines self-adaptive particle swarm optimization with random forest algorithms. The significantly higher FMI and MCC scores indicate that SAPSORF outperformed BSVM and S-ANN in classifying the cardiovascular disease dataset. SAPSORF's innovative approach likely allowed it to effectively capture complex patterns and optimize the classification process, resulting in superior performance.

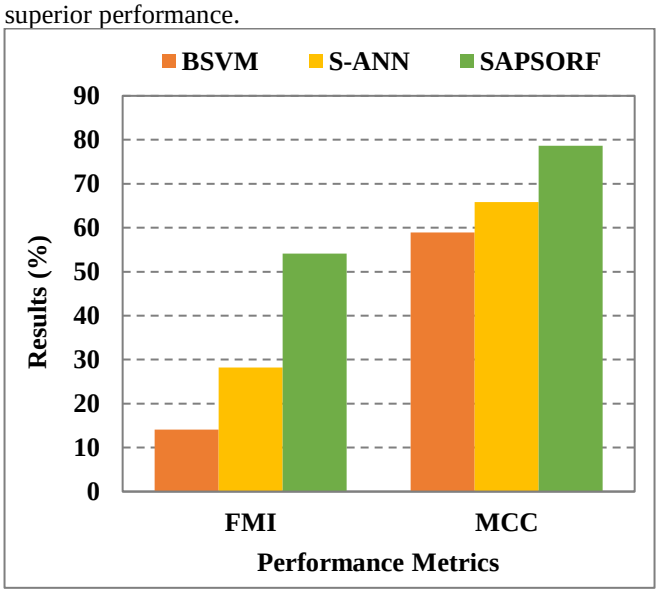

Fig. 3 FMI and MCC analysis on CD dataset

Table 5. Analysis of FMI and MCC Results on CD Dataset

Metrics → / Algorithms ↓	*FMI*	*MCC*
BSVM	14.124	58.924
S-ANN	28.247	65.860
SAPSORF	54.129	78.672

Table 6. Analysis of FMI and MCC results on PID dataset

Metrics → / Algorithms ↓	*FMI*	*MCC*
MSVM	67.657	27.588
DLPD	72.964	38.757
SAPSORF	87.515	71.077

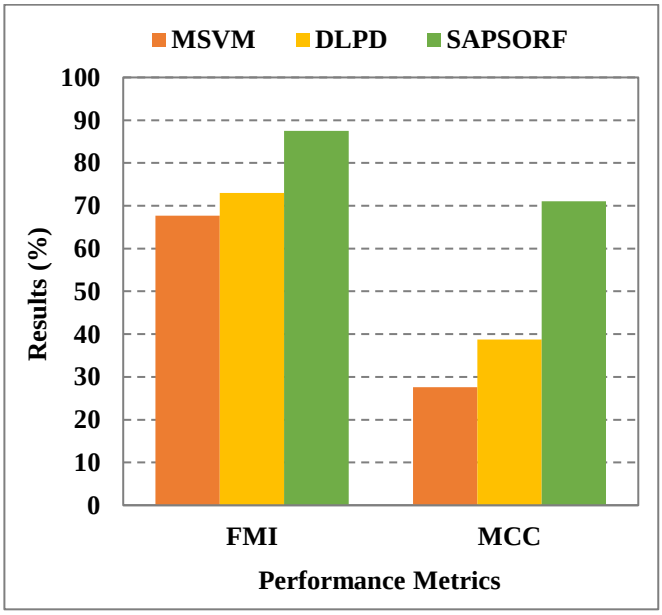

Fig. 4 FMI and MCC analysis on PID dataset

5.4. FMI and MCC Analysis of Classifiers with PID Dataset

The Fowlkes-Mallows Index (FMI) is a metric that measures the similarity between two clusters. A higher FMI score indicates better clustering performance, where similar data points are grouped effectively. In the case of the classifiers evaluated, SAPSORF achieved the highest FMI score of 87.515, indicating that it excels in clustering similar data points in the PID dataset. This suggests that SAPSORF's self-adaptive particle swarm optimization-based random forest algorithm successfully identifies and groups relevant features or patterns associated with diabetes.

DLPD follows with an FMI score of 72.964%, demonstrating relatively good clustering performance. This suggests that the deep learning approach used in DLPD, which likely involves multiple layers of interconnected neurons, effectively captures complex patterns and representations in the PID dataset. Although it performs slightly lower than SAPSORF, DLPD still competently identifies clusters of data points associated with diabetes.

On the other hand, MSVM, the existing classifier, achieved the lowest FMI score of 67.657%. While it still demonstrates some ability to cluster similar data points, it appears less effective than SAPSORF and DLPD in this regard. MSVM's modified support vector machine algorithm might face challenges in accurately separating the different classes or finding optimal hyperplanes for clustering in the PID dataset.

Matthews Correlation Coefficient (MCC) measures the correlation between the predicted and true binary classifications. A higher MCC score indicates more substantial predictive power and reliability of the classifier's predictions. SAPSORF achieved the highest MCC score of 71.077%, suggesting that it has a strong correlation between its predicted classifications and the true diabetes labels in the dataset. This implies that SAPSORF's self-adaptive particle swarm optimization-based random forest algorithm is effective in making accurate predictions for diabetes cases. DLPD follows with an MCC score of 38.757%, indicating a moderate correlation between its predictions and the correct labels. While it performs lower than SAPSORF, DLPD still demonstrates a reasonably reliable predictive capability for diabetes.

The deep learning approach used in DLPD allows it to learn and extract meaningful features from the input data, contributing to its predictive power. MSVM, on the other hand, has the lowest MCC score of 27.588%. This suggests that its modified support vector machine algorithm might face challenges in accurately predicting the diabetes labels in the dataset. MSVM's performance in the correlation between predictions and true labels is relatively weaker than SAPSORF and DLPD.

The expanded analysis highlights that SAPSORF exhibits superior performance compared to MSVM and DLPD regarding clustering similarity and predictive power. Its self-adaptive particle swarm optimization-based random forest algorithm can cluster similar data points and make accurate predictions for diabetes cases. DLPD, the existing classifier, performs well, albeit slightly lower than SAPSORF, indicating its competence in predicting diabetes based on deep learning techniques. MSVM, the current classifier with a modified support vector machine algorithm, performs relatively weaker in clustering and predictive aspects, suggesting room for improvement in its algorithm.

6. Conclusion

One of the most frequent ailments nowadays is heart disease and diabetes (HDD). Numerous data analytics methods have been utilized to aid healthcare practitioners in identifying some of the earliest indicators of HDD, which has been a long-sought goal. This research has proposed a Self-Adaptive Particle Swarm Optimization-Based Random Forest Algorithm ($SAPSORF$) to predict HDD. Better HDD predictions can be made using optimization-based sampling and dimensionality reduction. The particles' position and speed are often updated to improve classification accuracy on the CD and PID datasets. Classification Accuracy, F-Measure, Matthews Correlation Coefficient, and the Fowlkes-Mallows Index are used to evaluate $SAPSORF$'s efficacy. The results demonstrate conclusively that the suggested classifier outperforms the state-of-the-art classifier.

References

[1] Pratima Upretee, and Mehmet Emin Yüksel, "Accurate Classification of Heart Sounds for Disease Diagnosis by using Spectral Analysis and Deep Learning Methods," *Data Analytics in Biomedical Engineering and Healthcare*, pp. 215-232, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[2] Forum Desai et al., "Healthcloud: A System for Monitoring Health Status of Heart Patients Using Machine Learning and Cloud Computing," *Internet of Things (Netherlands)*, vol. 17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Saiteja Prasad Chatrati et al., "Smart Home Health Monitoring System for Predicting Type 2 Diabetes and Hypertension," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 3, pp. 862–870, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Priyanka S. Sangle, R. M. Goudar, and A.N. Bhute, "Methodologies and Techniques for Heart Disease Classification and Prediction," *2020 11th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2020*, pp. 1–6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[5] Kannadasan K, Damodar Reddy Edla, and Venkatanareshbabu Kuppili, "Type 2 Diabetes Data Classification Using Stacked Autoencoders in Deep Neural Networks," *Clinical Epidemiology and Global Health*, vol. 7, no. 4, pp. 530–535, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[6] Masayoshi Higashiguchi et al., "Prediction of the Duration Needed to Achieve Culture Negativity in Patients with Active Pulmonary Tuberculosis Using Convolutional Neural Networks and Chest Radiography," *Respiratory Investigation*, vol. 59, no. 4, pp. 421–427, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] Ashish Kumar, Rama Komaragiri, and Manjeet Kumar, "Heart Rate Monitoring and Therapeutic Devices: A Wavelet Transform Based Approach for the Modeling and Classification of Congestive Heart Failure," *ISA Transaction*, vol. 79, pp. 239–250, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[8] R. Valarmathi, and T. Sheela, "Heart Disease Prediction Using Hyper Parameter Optimization (HPO) Tuning," *Biomedical Signal Processing and Control*, vol. 70, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] R. Jaganathan, and R. Vadivel, "Intelligent Fish Swarm Inspired Protocol (IFSIP) for Dynamic Ideal Routing in Cognitive Radio Ad-Hoc Networks," *International Journal of Computing and Digital Systems*, vol. 10, no. 1, pp. 1063–1074, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] J. Ramkumar, and R. Vadivel, "Improved Wolf Prey Inspired Protocol for Routing in Cognitive Radio Ad Hoc Networks," *International Journal of Computer Networks and Applications*, vol. 7, no. 5, pp. 126–136, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11] J. Ramkumar, and R. Vadivel, "Whale Optimization Routing Protocol for Minimizing Energy Consumption in Cognitive Radio Wireless Sensor Network," *International Journal of Computer Networks and Application*, vol. 8, no. 4, pp. 455–464, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[12] J. Ramkumar, and R. Vadivel, "Multi-Adaptive Routing Protocol for Internet of Things Based Ad-Hoc Networks," *Wireless Personal Communications*, vol. 120, no. 2, pp. 887–909, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Ramkumar Jaganathan, and Vadivel Ramasamy, "Performance Modeling of Bio-Inspired Routing Protocols in Cognitive Radio Ad Hoc Network to Reduce End-to-End Delay," *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 1, pp. 221–231, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[14] P. Menakadevi, and J. Ramkumar, "Robust Optimization Based Extreme Learning Machine for Sentiment Analysis in Big Data," *International Conference on Advanced Computing Technologies and Applications (ICACTA)*, pp. 1–5, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] J. Ramkumar et al., "Energy Consumption Minimization in Cognitive Radio Mobile Ad-Hoc Networks Using Enriched Ad-Hoc on-Demand Distance Vector Protocol," *International Conference on Advanced Computing Technologies and Applications (ICACTA)*, pp. 1–6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] J. Ramkumar, and R. Vadivel, "Meticulous Elephant Herding Optimization Based Protocol for Detecting Intrusions in Cognitive Radio Ad Hoc Networks," *International Journal of Emerging Trends in Engineering Research,* vol. 8, no. 8, pp. 4548–4554, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[17] J. Ramkumar, and R. Vadivel, "Bee Inspired Secured Protocol for Routing in Cognitive Radio Ad Hoc Networks," *Indian Journal of Science and Technology,* vol. 13, no. 30, pp. 3059-3069, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] J. Ramkumar, and R. Vadivel, "Improved Frog Leap Inspired Protocol (IFLIP) – for Routing in Cognitive Radio Ad Hoc Networks (CRAHN)," *World Journal of Engineering,* vol. 15, no. 2, pp. 306–311, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[19] J. Ramkumar, R. Vadivel, and B. Narasimhan, "Constrained Cuckoo Search Optimization Based Protocol for Routing in Cloud Network," *International Journal of Computer Networks and Applications* vol. 8, no. 6, pp. 795–803, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] J. Ramkumar, and R. Vadivel, "CSIP—Cuckoo Search Inspired Protocol for Routing in Cognitive Radio Ad Hoc Networks," *Advances in Intelligent Systems and Computing,* vol. 556, pp. 145–153, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[21] Gaurav Meena, Pradeep Singh Chauhan, and Ravi Raj Choudhary, "Empirical Study on Classification of Heart Disease Dataset-Its Prediction and Mining," *International Conference on Current Trends in Computer, Electrical, Electronics and Communication, CTCEEC 2017,* pp. 1041–1043, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[22] Purushottam, K. Saxena, and R. Sharma, "Efficient Heart Disease Prediction System Using Decision Tree," *International Conference on Computing, Communication and Automation, ICCCA 2015,* pp. 72–77, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[23] J. S. Sonawane, and D. R. Patil, "Prediction of Heart Disease Using Multilayer Perceptron Neural Network," *2014 International Conference on Information Communication and Embedded Systems, ICICES 2014,* pp. 1–6, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[24] Sayali Ambekar, and Rashmi Phalnikar, "Disease Risk Prediction by using Convolutional Neural Network," *Proceedings - 2018 4th International Conference on Computing, Communication Control and Automation, ICCUBEA 2018,* pp. 1–5, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[25] S. Radhimeenakshi, "Classification and Prediction of Heart Disease Risk Using Data Mining Techniques of Support Vector Machine and Artificial Neural Network," *2016 3rd International Conference on Computing for Sustainable Global Development, INDIACom,* pp. 3107–3111, 2016. [Google Scholar] [Publisher Link]

[26] Jagdeep Singh, Amit Kamra, and Harbhag Singh, "Prediction of Heart Diseases Using Associative Classification," *2016 5th International Conference on Wireless Networks and Embedded Systems, WECON 2016,* pp. 1–7, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[27] Murat Alan, Mustafa Caner Aküner, and Alper Kepez, "Biosignal Classification and Disease Prediction with Deep Learning," *2020 Innovations in Intelligent Systems and Applications Conference, ASYU,* pp. 1–5, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[28] Vahid Taslimitehrani et al., "Developing EHR-Driven Heart Failure Risk Prediction Models Using CPXR(Log) with the Probabilistic Loss Function," *Journal of Biomedical Informatics,* vol. 60, pp. 260–269, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[29] Lijue Liu et al., "Machine Learning Algorithms to Predict Early Pregnancy Loss After in Vitro Fertilization-Embryo Transfer with Fetal Heart Rate as a Strong Predictor," *Computer Methods Programs Biomed,* vol. 196, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[30] Ashkan Parsi, "Heart Rate Variability Feature Selection Method for Automated Prediction of Sudden Cardiac Death," *Biomedical Signal Processing and Control,* vol. 65, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[31] Vaibhav Gupta, and Dr.Pallavi Murghai Goel, "Heart Disease Prediction Using ML," *SSRG International Journal of Computer Science and Engineering* , vol. 7, no. 6, pp. 17-19, 2020. [CrossRef] [Publisher Link]

[32] Sudarshan Nandy et al., "An Intelligent Heart Disease Prediction System Based on Swarm-Artificial Neural Network," *Neural Computing and Applications,* pp. 14723–14737, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[33] Ozal Yildirim et al., "Automated Detection of Diabetic Subject Using Pre-Trained 2D-CNN Models with Frequency Spectrum Images Extracted From Heart Rate Signals," *Computers in Biology and Medicine,* vol. 113, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[34] Beatriz López, "Single Nucleotide Polymorphism Relevance Learning with Random Forests for Type 2 Diabetes Risk Prediction," *Artificial Intelligence in Medicine,* vol. 85, pp. 43–49, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[35] Binh P Nguyen et al., "Predicting the Onset of Type 2 Diabetes Using Wide and Deep Learning With Electronic Health Records," *Computer Methods Programs in Biomedicine,* vol. 182, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[36] Atul Kumar et al., "SVMRFE Based Approach for Prediction of Most Discriminatory Gene Target for Type II Diabetes," *Genomics Data,* vol. 12, pp. 28–37, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[37] Jaeyeon Shin et al., "A Correction Method using a Support Vector Machine to Minimize Hematocrit Interference in Blood Glucose Measurements," *Computers in Biology and Medicine,* vol. 52, pp. 111–118, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[38] H. Maldonado et al., "Automatic Detection of Risk Zones in Diabetic Foot Soles by Processing Thermographic Images Taken in an Uncontrolled Environment," *Infrared Physics & Technology,* vol. 105, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[39] Jayakumar Sadhasivam, Senthil Jayavel, and Arpit Rathore, "Survey of Genetic Algorithm Approach in Machine Learning," *International Journal of Engineering Trends and Technology*, vol. 68, no. 2, pp. 115-133. [CrossRef] [Publisher Link]

[40] J. Saminathan et al., "Computer Aided Detection of Diabetic Foot Ulcer Using Asymmetry Analysis of Texture and Temperature Features," *Infrared Physics & Technology*, vol. 105, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[41] Yi-Peng Liu et al., "Referable Diabetic Retinopathy Identification from Eye Fundus Images with Weighted Path for Convolutional Neural Network," *Artificial Intelligence in Medicine*, vol. 99, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[42] K. Shankar et al., "Automated Detection and Classification of Fundus Diabetic Retinopathy Images Using Synergic Deep Learning Model," *Pattern Recognition Letters*, vol. 133, pp. 210–216, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[43] S. Thenappan, M. Valan Rajkumar, and P. S. Manoharan, "Predicting Diabetes Mellitus Using Modified Support Vector Machine With Cloud Security," *IETE Journal of Research*, pp. 1–11, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[44] Huaping Zhou, Raushan Myrzashova, and Rui Zheng "Diabetes Prediction Model Based on an Enhanced Deep Neural Network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[45] S.Usha, and Dr.S.Kanchana "Revived Ant Colony Optimization-Based Adaboost Algorithm for Heart Disease and Diabetes (HDD) Prediction," *Journal of Theoretical and Applied Information Technolog*, vol. 101, no. 4, pp. 1552–1567, 2023. [Google Scholar] [Publisher Link]

[46] Ebenezer Owusu, "Computer-Aided Diagnostics of Heart Disease Risk Prediction Using Boosting Support Vector Machine," *Computational Intelligence and Neuroscience*, vol. 2021, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[47] Gabriel Tozatto Zago et al., "Diabetic Retinopathy Detection using Red Lesion Localization and Convolutional Neural Networks," *Computers in Biology and Medicine*, vol. 116, 2020. [CrossRef] [Google Scholar] [Publisher Link]