PERIODICO di MINERALOGIA Volume 91, No. 4, 2022
ISSN: 0369-8963

Data Mining Approach for Diagnosing Heart Diseases through Deep Neural Network

A. ANN ROMALT[1*], MATHUSOOTHANA S. KUMAR[2]

[1]Department of Computer Science and Engineering, Stella Mary's College of Engineering, Nagercoil, Tamilnadu, India.

[2]Department of Information Technology, Noorul Islam Center for Higher Education, Kumaraoil, Nagercoil, Tamilnadu, India.

[*]Corresponding Author: A. ANN ROMALT

Abstract

Predicting and detecting cardiac illness using machine learning and data mining approaches is extremely clinically useful, and further progress will be quite difficult. Heart disease early-stage prediction can be improved by using digital patient data to provide analytical support for clinical decision-making in countries throughout the world where cardiovascular knowledge is lacking and the number of erroneous diagnoses is high. Many supervised machine-learning algorithms were utilised in the study to find classifiers with the highest accuracy in predicting heart disease. They were also discriminated based on their performance and accuracy. The goal of the study is to use AI (Artificial Intelligence) to diagnose cardiac disease in both normal and pathological settings. Various AI technologies are expected to be used, with DNN (Deep Neural Network) outperforming the others. This is predicted by the analysis an updated spider monkey optimization (USMO) technique has been proposed for the DNN as a means of determining optimal weights. The investigation's findings reveal a precision of 96.77 percent in the Cleveland database and a precision of 100 percent in the Hungarian database.

Keyword: *Machine learning and data mining, Updated Spider Monkey Optimization (USMO), Deep Neural Network, Diagnosing Heart Diseases, Cleveland and Hungarian database.*

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

INTRODUCTION

People in the modern society are far too preoccupied with their daily activities and careers to take care of their own needs. Depression, anxiety, stress, and other health concerns result as a result of these individuals' workaholic lifestyles. A person becoming ill from one of these major causes can lead to additional serious health problems, such as cancer, heart disease, and tuberculosis, all of which can be fatal. CVD (cardiovascular disease) and heart disease have the highest overall death rates [1]. Worldwide, according to World Health Organization (WHO) data, about 30% of deaths are caused by disorders of the heart. They estimate that CVD is to blame for almost 15.2 million deaths, according to their figures. The majority of deaths have been attributed to cardiovascular disease for about 15 years now. Between 2005 and 2015, India spent $237 billion on heart-related disorders, according to one estimate [2]. According to the Indian Heart Association (IHA), CVD kills four Indians per day. The 30-to-50-year-old demographic is particularly noticeable. People under the age of 40 are more likely to suffer from heart failure than those above the age of 40.

Many contributing risk factors make it difficult to diagnose heart disease. Diabetes and high blood pressure, both of which run in families, are additional risk factors for heart disease. It's possible to influence a few risk variables. Lifestyle habits such as inactivity, obesity and poor eating habits can all contribute to heart disease in addition to the previously stated variables [3, 4]. Different types of heart disease are found in it such as myocarditis and coronary heart disease, as well as arrhythmias and a congenital heart condition known as angina pectoris. The exact aetiology of heart disease can be difficult to determine based on risk factors alone [5, 6]. Heart disease is a complicated condition that necessitates expert treatment to avoid damaging the heart or causing early death. It is possible to minimise the death rate by detecting diseases early on [7].

The procedure of identifying cardiac illness is, therefore, an unavoidable process. While making remarkable strides, researchers are better able to decipher the pathophysiology of complicated cardiovascular failure. It is intended to monitor and break down the amount of information and data that is unpredictable. As a result, reliable findings on cardiovascular failure and useful evaluation options can be generated [8, 9]. This job, on the other hand, is frequently made for the benefit of a doctor's comprehension and familiarity. You will have to spend a lot of

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

money on it [10, 11]. Cardiovascular disease prediction is therefore a significant and difficult health issue. Predicting cardiac illness has recently been framed by numerous well-researched frameworks in healthcare, including machine learning and swarm optimization. Multi-kernel learning with adaptive neuro-fuzzy inference system (MANFIS) and support vector machines has been employed in the healthcare monitoring system to detect cardiac disease [12-17]. Another algorithm used to detect cardiac illness in the healthcare monitoring system is Genetic Algorithm with Particle Swarm Optimization, BAT method, and gravitational cuckoo search algorithm [18-20]. However, the majority of tactics fail to improve the accuracy of heart disease prediction because to a lack of appropriate prediction methodology and data-recognized approaches.

To improve weak algorithms' accuracy used a technique called ensemble classification, which combined numerous classifiers. While the study's primary goal was to improve the accuracy of weak classification algorithms, it also focused on how to incorporate these methods with medical datasets to demonstrate how useful they may be in identifying disease before it has fully developed. Ensemble approaches like as bagging and boosting were found to be efficient in improving the accuracy of weak classifiers' predictions, which showed satisfactory results in identifying the risk of heart disease [21]. Using feature fusion and ensemble deep learning, suggested a smart healthcare system for the early detection of cardiac disease [22]. Technique of feature fusion fuses features taken from electronic medical records and sensor data initially for the production of useful healthcare data. Another advantage of this technique is that features that are useless or duplicated are discarded in favour of important information that decreases the computing burden on the system while simultaneously improving performance.

Improved neural network performance by 10% by using a genetic algorithm to improve neural network starting weights is the method proposed in this paper. It was found that the Z-Alizadeh Sani dataset has 80% sensitivity, 93.85% accuracy, and 92% specificity using this methodology [23]. A hybrid decision support system of combining genetic algorithm (GA) and recursive feature reduction was used to choose acceptable features from the provided dataset using a hybridised approach. The pre-processing of the data had also utilised Synthetic Minority Oversampling Technique (SMOTE) as well as normal scalar approaches [24]. The algorithm had

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

produced precise results using a forest random classifier. This system's accuracy in predicting heart disease was higher than that of other similar systems, at 86.6%.

Using supervised learning approaches as the random forest algorithm, K-nearest neighbour, Nave Bayes and decision tree, created a model for heart disease prediction using supervised learning. In total, there are 76 attributes in this dataset, and there are 303 instances of each. Only 14 of the 76 qualities were put to the test, making it easier to compare the performance of different methods. K-nearest neighbour has the highest accuracy rating, according to the findings [25]. A genetic method and a support vector machine-based prediction model for the identification of heart disease [26]. When identifying heart disease using all of its features, SVM achieves an accuracy of 83.70 percent. As a result, when using SVM as a classifier to classify heart disease, the accuracy is 88.34%.

PROPOSED METHODOLOGY

The study uses databases from the Heart Disease Data Set, including Cleveland and Hungarian versions, as well as the UCI Machine Learning Repository, which has 480 rows of data to help detect heart disease. When evaluating the effectiveness of various training and testing methods, researchers alter the ratio of training to testing. Diagnosing cardiac disease requires the use of a variety of approaches, including DNN, artificial neural network-Levenberg-Marquardt (LM), scaled conjugate gradient (SCG), and robust back propagation (ANN) (RP). The suggested DNN's overall design is shown in Figure 1. The study uses 13 different inputs to arrive at a single output, which is the diagnosis of heart disease. SMO and upgraded SMO approaches are used to identify heart illnesses as part of the research, which includes optimization strategies for selecting appropriate weights.

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

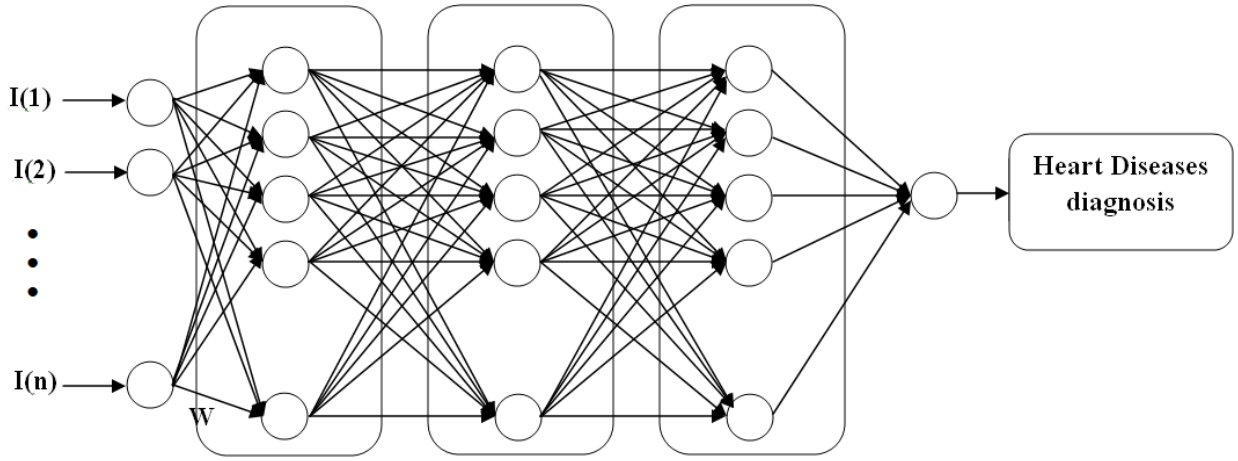


Figure 1. Overall architecture of the proposed DNN

DNN TECHNIQUE

"Deep" denotes the number of neural network layers. While DNNs have only one hidden layer, shallow neural networks include multiple hidden layers. Distributed representation can be used during machine learning based on deep learning. This can be used to say that observations are generated as a result of the interactions between numerous elements. This interaction's process is divided into many layers, signifying the multi-layered abstraction of acquired values. Layer-by-layer superimposition of simple characteristics (such as two pixels) creates numerous complex features (namely straight line).

Deep learning is generally divided into two stages: training and putting the trained model to use (i.e. testing). When dealing with nonlinear and non-convex issues, DNN is a preferred choice. Figure 2 shows an example of a DNN model structure. The DNN's inner neural network layer is divided into three parts: the input layer, the hidden layer, and the output layer. Layers 1 and 2 are shown in Figure 2, with layer 3 being the output layer. Layers 2 and 3 are shown in Figure 2, with layer 4 being the input layer. The Neuron, or Node, is the basic building block of a neural network. All of the nodes in the middle layers can be interconnected, which means that every i^{th} layer node can be linked to any $I + 1^{th}$ layer node Forward propagation performs a sequence of linear operations and activation operations in order to obtain the output result by using the weight coefficient matrix w, bias vector b, and input value vector, starting with the

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

input layer, and continuing to the output layer. The forward propagation calculation can be broken down into two steps:

$$z_i^{(l)} = w_i^{(l)} y^{(l-1)} + b_i^{(l)} \tag{8}$$

$$y_i^{(l)} = f\left(z_i^{(l)}\right) \tag{9}$$

here l denotes\l^{th} layer, i denotes ith node of layer, activation function denoted by $f(z)$.

Back propagation first needs loss function for measuring variation among output computed using output of sample of training and actual training, and training sample output can be computed through forward propagation method. Here, it used L_2 as loss function.

$$L_2 = \frac{1}{N}\sum_k \left(\hat{Z}(k) - Z(k)\right)^2 \tag{10}$$

here prediction denoted by $\hat{Z}(k)$ whereas supervision denoted by $Z(k)$ that can be the transmitted symbols during experiment. For every sample, this can be expected below mentioned formula to be minimized,

$$J(\theta) = \frac{1}{2}\left\| y_i^l - a \right\|_2^2 = \frac{1}{2}\left\| f\left(w_i^{(l)} y^{(l-1)} + b_i^{(l)} \right) - a \right\|_2^2 \tag{11}$$

Neurons of every neural network layer can generate predictions. Hence, gradient descent approach of conventional regression problem could not directly use for minimizing loss function, and prediction error necessary for considering layer by layer as well as optimizing layer by layer. Back propagation Algorithm can be utilized for optimizing prediction for multi-layer neural network. Initially, every layer prediction error can be established as vector $\delta^{(l)}$ which are shown below

$$\delta^{(l)} = \begin{cases} y^{(l)} - a & l = L \\ \left(w^{(l+1)}\delta^{(l+1)}\right)^T \Theta\, f'(z^{(l)}) & l = 2,3,\ldots L-1 \end{cases} \tag{12}$$

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

Function of activation should remain as non-linear each layer output is upper layer linear function. While nonlinear function can be established to be an activation function, neural network output will be no longer inputs linear combination which approximates the function of arbitrary. Activation functions can be sigmoid function and Relu function:

$$\sigma_{\mathrm{Re}LU}(x) = \max(0, x) \tag{13}$$

$$\sigma_{Signoid}(x) = \frac{1}{1+e^{-x}} \tag{14}$$

Relu can be generally utilized as activation function for DNNs, which is the fastest one. Sigmoid can be utilized for last layer in mapping input continuous real value for output between 0 and 1.

Algorithm DNN Training Algorithm

The input of this research holds 13 input variables namely age, sex, chest pain type (cp), maximum heart rate attained (thalach), trestbps: resting blood pressure (in mm Hg on admission to the hospital),fbs: (fasting blood sugar > 120 mg/dl) (1 = true; 0 = false), exang: exercise induced angina (1 = yes; 0 = no), serum cholestoral in mg/dl (chol), resting electrocardiographic results (restecg), depression of ST induced through exercise relative to rest (oldpeak), slope of peak exercise ST segment (slope), number of primary vessels (0-3) colored through flourosopy (ca), thal: 3 = normal; 6 = fixed defect; 7 = reversible defect. Total number of layers L, number of neurons in every layer of output as well as hidden, iteration step length MAX, loss function, maximum iteration number, activation function, and threshold to stop iteration ε; m Samples and labels $\{(s_1,r_1),(s_2,r_2),\ldots,(s_m,r_m)\}$;

The output is a linear relationship coefficient matrix w and bias vector b for each hidden and output layer; heart disease detection is included as an output. Result:

1: Initial w and b of every hidden layer as well as output layer to be random value;

2: for epoch = 1 to MAX do

3: for i = 1 to m do

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

4: Initial DNN input layer, $y^1 = x$;

5: for l = 2 to L do

6: do forward propagation, calculate y^i;

7: end for

8: calculate the output layer $\delta^{(L)}$ through loss function;

9: for l = 2 to L do

10: do back propagation, calculate $\delta^{(l)}$;

11: end for

12: end for

13: for l = 2 to L do

14: update w and b of the l layer.

15: if all w and b changes are less than stop iteration threshold ε

then

16: jump out of iteration loop;

17: end if

18: end for

19: end for

DNN Model Training

This receives input from existing or external nodes (usually on the front layer) and computes output using the activation function, providing delinearization in the process. Bias

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

(Bias) is a special type of input that has a high correlation with weight and so is apparent for all inputs that travel through that particular node.

L_2 loss can be used to describe variation in input and label when building a model. To activate the Relu function in the middle layers, one would use it. To receive input during intervals of [0, 1], one would use a sigmoid function in the output layer.

Over-fitting can be eliminated by using a dropout in training. This involves randomly discarding a small number of neurons in the hidden layer while training, and then using a neural network to remove the hidden layer in order to fit the training data. A few hidden layer neurons are deleted when parameters are adjusted before a data training batch restores the DNN model to its fully original connected state.

The two elements of the simplest technique are training and verification, which are separated by a training set and a test set for evaluating the generalisation capabilities of the model using the LOOCV technique (Leave-one-out cross-validation). For a test set, look at the MSE size in relation to different models' various parameters, including model selection via training models for training sets.

SPIDER MONKEY OPTIMIZATION (SMO) ALGORITHM

SMO (Spider Monkey Optimization) is a technique for achieving broad-based improvements. Foraging behaviour is impacted by spider monkeys' FFSS (Fission-Fusion social structure). Work division and self-organization are fundamental SMO swarm intelligence ideas. Foraging behaviour in FFSS-based animals (such as spider monkeys) can be broken down into four phases. At first, the gang goes food foraging to find out how far apart the different foods are from one another. In addition, group members improve their positions in relation to the distance between food sources based on the data collected from the distance measurements. Furthermore, the local group leader will upgrade the best position to the next level as a further step. The entire group will begin searching for food in all directions if the position has not been updated a set amount of times. Final upgrade of best position by worldwide powerhouse. This divides the group into smaller size subgroups if there is stagnation. The four processes listed above can be performed at random until the desired result is achieved.

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

Improvements to the Spider Monkey Optimization Methodology

SMO is a meta-heuristic strategy influenced by the intelligent foraging behaviour of spider monkeys. Fission-fusion-based social structure affects SMs' foraging behaviours. The qualities of the method inspire the social organisation of the group, and the female leader determines whether the group will split or combine. Global leadership can be used to the entire group, whilst local leadership can be applied to the individuals in smaller groups only. Famine can be predicted using the SMO approach with no change to the solution. A small group has less monkeys while using the SMO approach because it is built on the premise of swarm intelligence. For this reason until now, fusion only produces one group with fewer monkeys, and is considered fusion time. The potential solution SM (Spider Monkey) represents in the SMO approach. Phases of the SMO include: Local Leader, Global Leader, Local Leader Learning, Global Leader Learning and Local Leader Decision. This research also incorporates a Cauchy distribution function for increasing the likelihood of escaping from the local optimum in addition to the Local Leader Learning Phase. As previously mentioned, the following SMO phases exist:

Initialization

Here, SMO uniformly produces distributed initial swarm of N spider monkeys and SM_i denotes i^{th} SM (spider monkey) in swarm. Every SM_i can be initialized:

$$SM_{ij} = SM_{\min j} + U(0,1) * (SM_{\max j} - SM_{\min j}) \quad (15)$$

Here lower and upper bounds of search space in j^{th} dimension represented by SM_{minj} as well as SM_{maxj} whereas U (0, 1) can be distributed uniformly random number about range (0, 1).

Local Leader Phase (LLP)

At this stage, it's crucial that you use the SMO approach. When the spider monkeys reach this stage, they are all up to date. Local leader and group member experiences influence position modification in the spider monkey. The fitness value of each SM in their novel place can be calculated. If their fitness level can be improved over their previous one, then this app will be updated. Otherwise, it won't. The following equation is used to update one's position:

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

$$SMnew_{ij} = SM_{ij} + U(0,1) * (LL_{kj} - SM_{ij}) + U(-1,1) * (SM_{rj} - SM_{ij}) \quad (16)$$

Here jth dimension of i^{th} SM denoted by SM_{ij}, j^{th} dimension of k^{th} group's local leader denoted by LL_{kj} whereas SM_{rj} denotes randomly chosen SM j^{th} dimension within group of k^{th} so r ≠ i whereas U(−1, 1) can be distributed uniformly random number at range (−1, 1).

Equation brings this to life (2) SM changes its stances in order to draw attention to a local leader while also bolstering his or her confidence. The final component helps establish variations in the search process, which helps maintain the stochastic nature of the technique and avoid early stagnation. It was determined in step 4 how to perform an update complete position phase. Prefers a value between [0.1, 0.8] because it indicates how much of a disturbance the current solution will cause.

Cauchy distribution would be random variable X if Cauchy mutation operator was applied. A Cauchy density function with origin in the centre can be explained as follows:

$$f(x) = \frac{1}{\pi}\left(\frac{t}{t^2 + x^2}\right), -\infty < x < \infty \quad (17)$$

here$t > 0$ denotes scale parameter. Cauchy distribution function can be explained:

$$f_t(x) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(\frac{x}{t}\right) \quad (18)$$

By using a mutation operator of this type, it is possible to escape from local optima. After looking at the pdf (probability distribution function) of the Gaussian and Caussian distributions and doing complete simulations, the author has come to the conclusion that the variance level has a local escaping/converging ability. Cauchy mutation is more likely to occur if the chance to evade a local optimum is given to it.

Global Leader Phase (GLP)

After LLP, method steps towards GLP. The update of the solutions depends on probability of selection, which is fitness function. From objective function f_i fitness fit_i is computed as,

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

$$fitness\,function = fit_i = \begin{cases} \dfrac{1}{1+f_i}, & if \;\; f_i \geq 0 \\ 1+abs(f_i), & if \;\; f_i < 0 \end{cases} \tag{19}$$

Selection probability $prob_i$ can be determined on the basis of selection of roulette wheel. If fit_i is i^{th} SM fitness, the selected probability in GLP can be computed by any of these:

$$prob_i = \frac{fitness_i}{\sum_{i=1}^{N} fitness_i}$$

(or)

$$prob_i = 0.9 * \frac{fit_i}{\max\_ fit} + 0.1$$

For updating position, SM utilizes global leader's knowledge, the neighboring SM experience and their persistence. Position updates equation:

$$SMnew_{ij} = SM_{ij} + U(0,1) * (GL_j - SM_{ij}) + U(-1,1) * (SM_{rj} - SM_{ij}) \tag{20}$$

Global leader position in the j^{th} dimension is denoted by the acronym GLj. Each part of the equation can be classified into one of three categories: first component expresses current SM persistence; second component expresses current SM attraction to global leader; third component can maintain stochastic behaviour of algorithm. It is possible to improve the already identified search space exploitation by using the second equation component while the third equation component aids in preventing premature convergence or decreasing the likelihood of becoming stuck in local optima. Step 5 outlined the entire search procedure.

By the end of step 5, the likelihood of an update to a solution is determined by probi. As a result, when compared to less fit solutions, solutions with higher fitness have a better likelihood of updating their position. This also means that instead of using updated as well as the previous SM we can select the best fit solution using the greedy selection method.

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

Global Leader Learning Phase

A swarm optimization problem is solved here by using an algorithm. The leader of the swarm can be deemed to be the identified SM. Additionally, the company's global leadership position can be examined. When this counter for the global leader cannot be updated, it is referred to as the GLC counter (Global Limit Count). It can be increased by one, but if you don't, this value will be somewhere around 0. GLC is a global leader that is distinct from GLL, as can be proved (Global Leader Limit).

Local Leader Learning Phase

Here, the position of the local leader is updated by making greedy selections among the group's members. Unless the local leader updates their position, the counter for LLC (Local Limit Count) can be increased by one; otherwise, the counter will be reset to 0. To find out who the local leader is, it can be used by each organisation. Increment LLC until it reaches a fixed threshold, which is denoted by the letters LLL (Local Leader Limit).

Local Leader Decision Phase

Identifying global and local leaders is the first step in this process. If a local leader has not restructured in a certain area, this is referred to be the Local Leader Limit, and every member of the group uses random initialization or the global leader's experience with Equation to update their places (21). Equation (21) can be used in conjunction with a probability pr known as the rate of disturbance.

$$SMnew_{ij} = SM_{ij} + U(0,1) * (GL_j - SM_{ij}) + U(0,1) * (SM_{rj} - LL_{kj}) \quad (21)$$

If you look closely at the equation, it's clear that the solution groups can be deterred by other local leaders. It's possible to exhaust it (without updating almost LLL iterations), whereas solutions can draw attention to the global leader and cause it to shift its search directions and locations. Other solutions' placements can be disturbed by random initialization of their dimensions, which is also based on pr. LLL is a parameter used to ensure that the local leader has never fallen back to the local minimum. Generally, this can be calculated as D N, where D

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

denotes the dimension and N denotes the total number of SM. If LLC exceeds LLL, LLC will be set to zero. For better search space exploration, SM initialises as previously described.

Global Leader Decision Phase

A similar situation occurs when the global leader fails to regroup in time for the particular verge known as GLL. Fusion teams are split up into smaller teams by the Global leader or swarm. During this stage, a parameter called GLL checks to see if there is any possibility of premature convergence within the range N/2 to 2 N. As long as GLL is smaller than GLC, GLC will be set to zero (the default). You can tell how many groups there are by looking at the maximum number of groups. Groups are divided or fuses are used to create a parent or single group if the other groups' numbers are less than the predefined maximum.

Following algorithm 1 gives the complete SMO working mechanism for solving problem of optimization.

Algorithm 1: Spider Monkey Optimization

Step 1. Initialize population, global leader limit, local leader limit and rate of perturbation;
Step 2. Estimate population;
Step 3. Find leaders of global and local;
Step 4. Phase of local leader updates position
for every member $SM_i \in k^{th}$ group do
for every $j \in \{1,...,D\}$ do
if U(0,1) $\geq pr$ then
$SMnew_{ij} = SM_{ij} + U(0,1) * (LL_{kj} - SM_{ij}) + U(-1,1) * (SM_{rj} - SM_{ij})$
else
$SM_{newij} = SM_{ij}$
end if
end for
end for
Step 5. Position update through phase of global leader
count = 0:

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

while count < group size do

for every member $SM_i \in$ group do

if $U(0,1) < prob_i$ then

count = count + 1

Randomly choose $j \in \{1,.....,D\}$

Randomly choose $SM_r \in$ group s.t.r≠i

$$SMnew_{ij} = SM_{ij} + U(0,1) * (GL_j - SM_{ij}) + U(-1,1) * (SM_{rj} - SM_{ij})$$

end if

end for

end while

Step 6. Learning by phase of global leader learning

Step 7. Learning by phase of local leader learning

Step 8. Using phase oflocal leader decision update Position

if Local Limit Count > Local Leader Limit then

Local Limit Count = 0

for every $j \in \{1,.....,D\}$ do

if $U(0,1) \geq pr$ then

$$SMnew_{ij} = SM_{\min j} + U(0,1) * (SM_{\max j} - SM_{\min j})$$

else

$$SMnew_{ij} = SM_{ij} + U(0,1) * (GL_j - SM_{ij}) + U(0,1) * (SM_{rj} - LL_{kj})$$

end if

end for

end if

Step 9. Decide fission or fusion byphase of global leader decision

if Global Limit Count > Global Leader Limit then

Global Limit Count = 0

if Number of groups < MG then

Divide swarms into groups

else

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

Combine entire groups to create single group

end if

update position of Local Leaders

end if

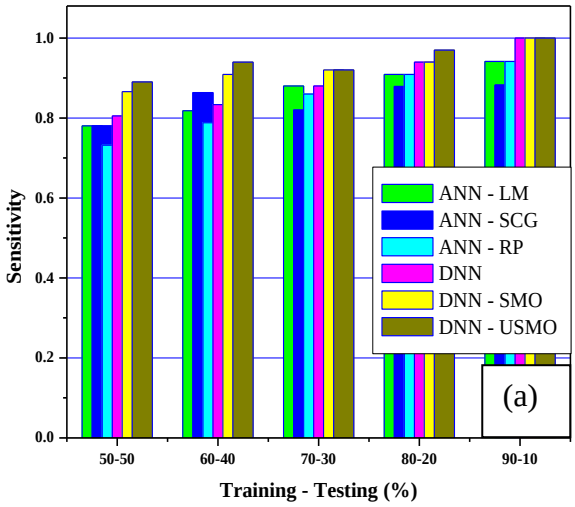
Step 10. If the condition ofterminationcan be satisfied stop which declare position of global leader to be optimal solution else go to step 4.

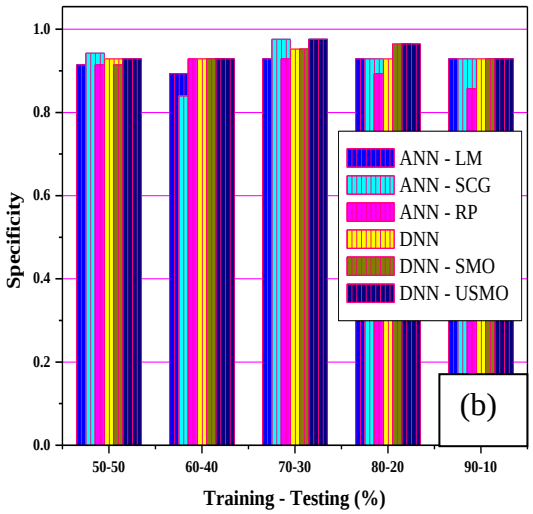
RESULTS AND DISCUSSION

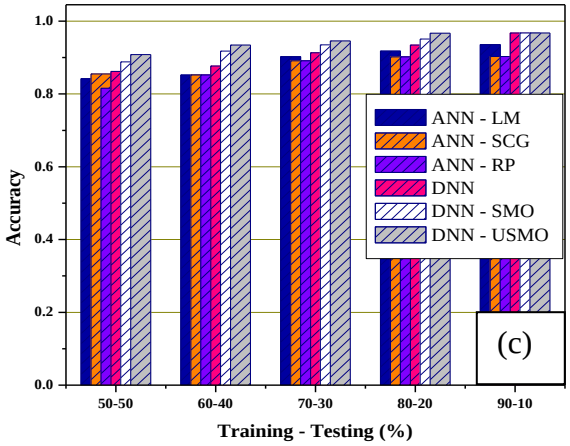
Researchers are trying to find a better way to diagnose heart disease by using DNN and standard measures like FNR, PPV, NPV, FPR, FPR, and FDR to see how well the techniques work. On the basis of this, the proposed approach by USMO associates DNN unveils accuracy levels of 96.77 percent in the Cleveland database and 100 percent in the Hungarian database, respectively. On the other hand, the accuracy of the SMO-associated DNN and traditional DNN in the Cleveland and Hungarian databases is 96.77% and 96.67%, respectively. An accuracy of 90.32 percent is achieved by using the comparative techniques of ANN-RP and ANN-SCG in the Cleveland and Hungarian databases. It is estimated that ANN-LM is 93.54% accurate in the Cleveland database and 93.33% accurate in the Hungarian database. From the results, it's plain to see that using the proposed approach and the Cauchy distribution improves its overall performance.

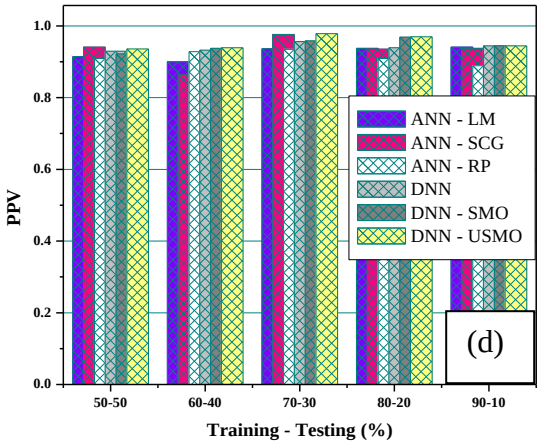
Here's a look at how the training-testing ratio changes when using the Cleveland and Hungarian databases, respectively. All performance measures in the Cleveland database show that USMO-DNN outperforms all other comparative techniques, as shown in Figure 2 (a) to (h). On all standard measures of change in training to testing ratio, USMO-DNN performs well, as shown in Table 1. Also, the superiority of USMO-DNN in Hungarian databases for performance measures can be seen in Figure 3 (a) to (h). Using all standard measures for change in the training/testing ratio, Table 2 shows how well USMO-DNN performed. In both cases, it's clear that using 90% of the database for training outperforms other approaches.

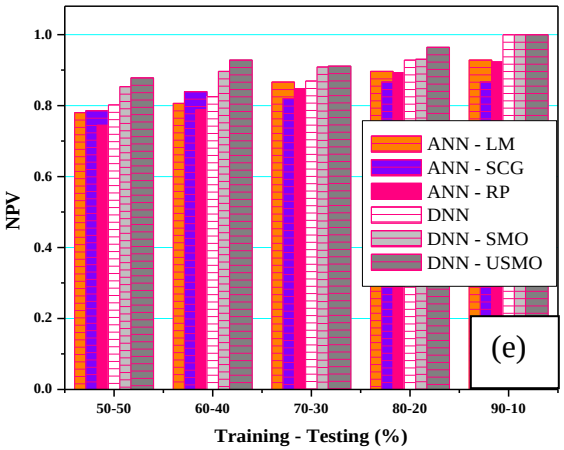
 https://doi.org/10.37896/pd91.4/91423

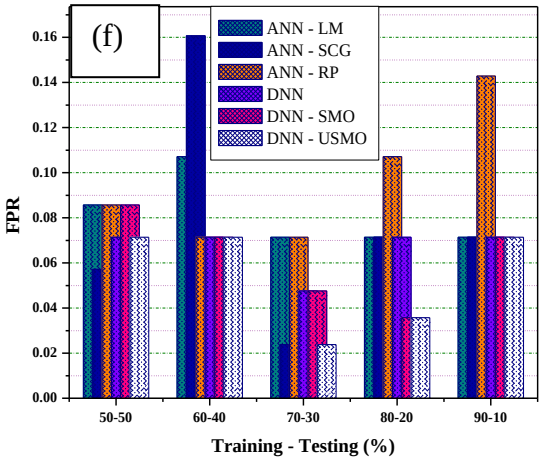












https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

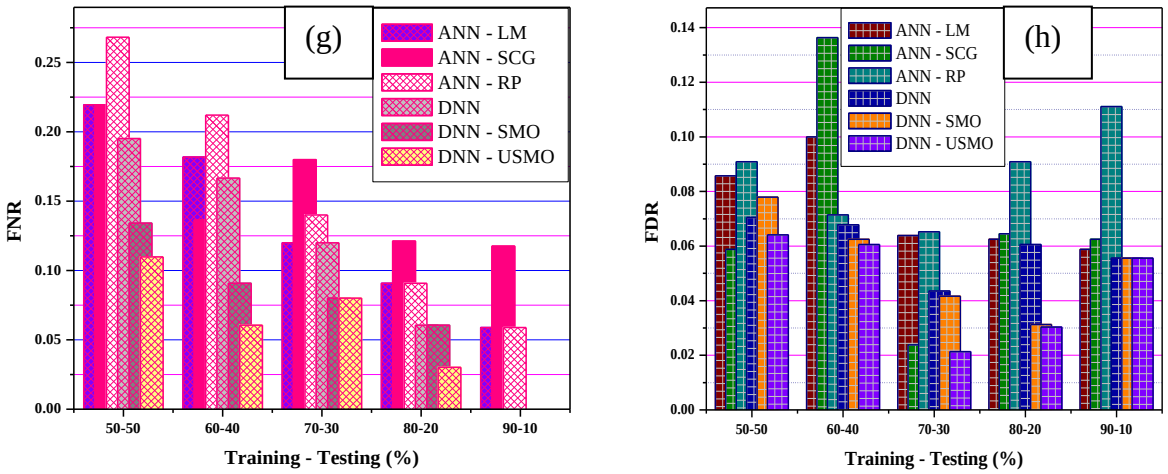


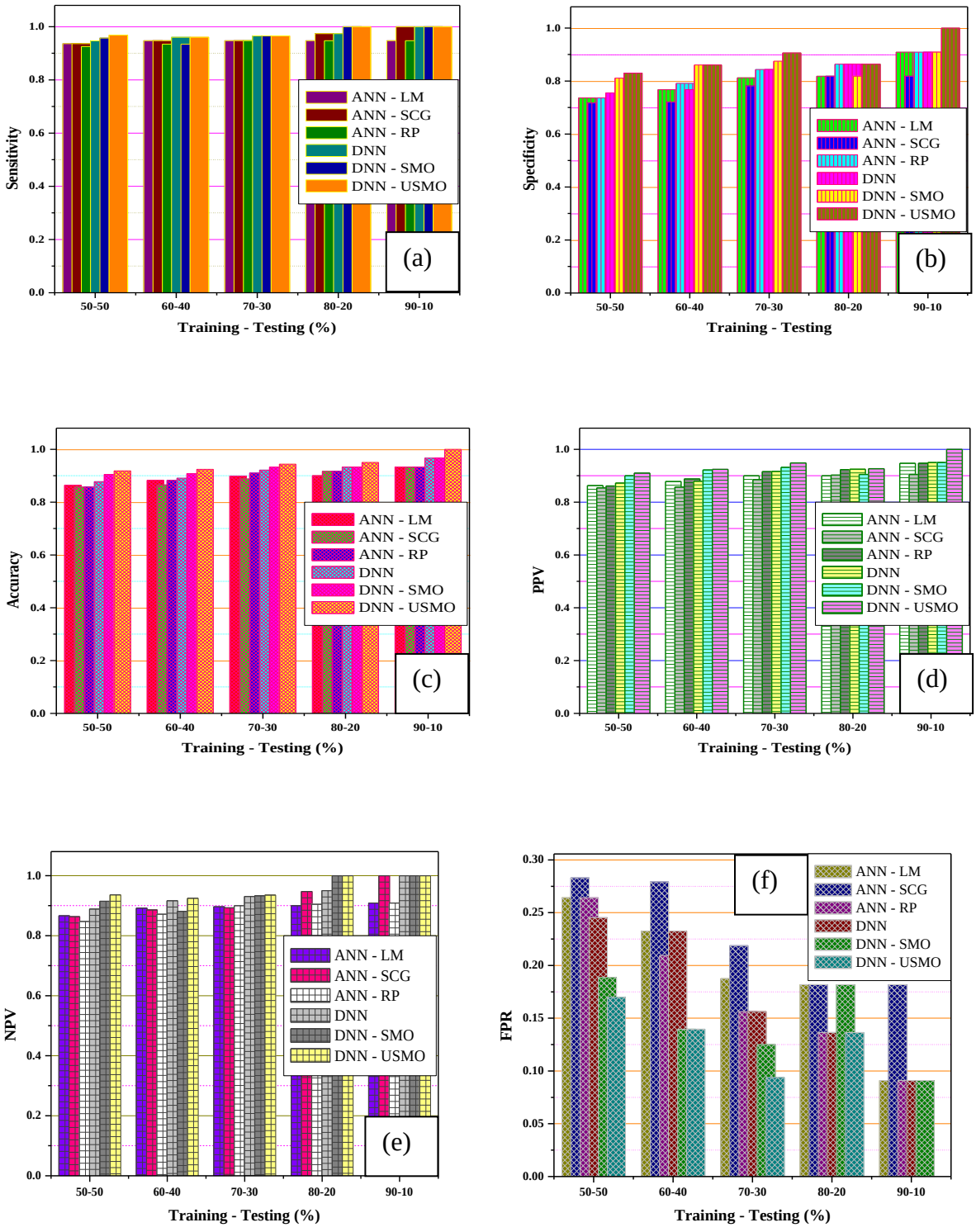
Figure2.Cleveland Database Performance Measures with Respect to Different Training –Testing.

Table 1. The USMO Performance of DNN's for the Cleveland Database

Training- Testing in Percentage					
Measures	**50-50**	**60-40**	**70-30**	**80-20**	**90-10**
Sensitivity	0.890	0.939	0.920	0.969	1
Specificity	0.928	0.928	0.976	0.964	0.928
Accuracy	0.907	0.934	0.945	0.967	0.967
PPV	0.935	0.939	0.978	0.969	0.944
NPV	0.878	0.928	0.911	0.964	1
FPR	0.071	0.071	0.023	0.035	0.071
FNR	0.109	0.060	0.080	0.030	0
FDR	0.064	0.060	0.021	0.030	0.055

https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963



https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

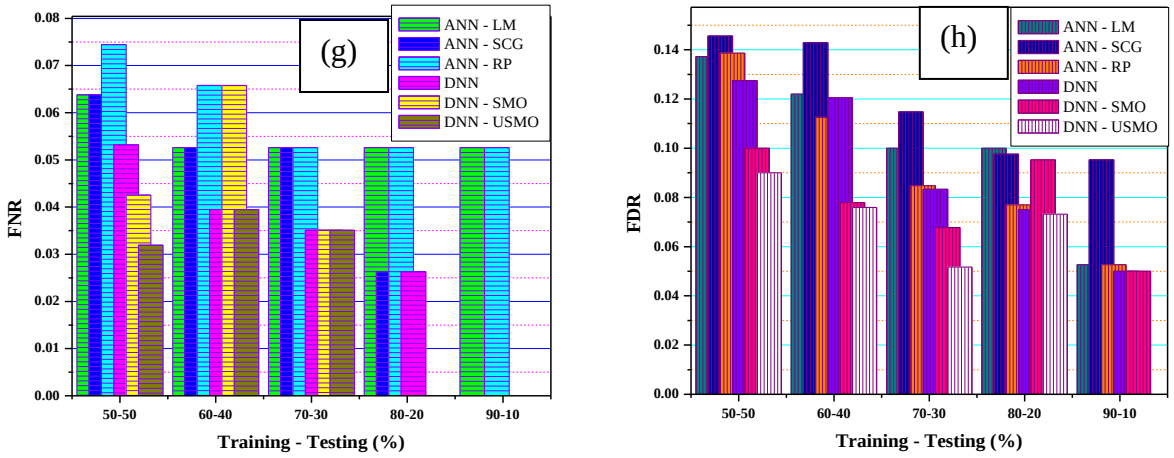


Figure 3. Measures of Hungarian Database Performance in Relation to Various Training and Testing Scenarios

Table 2. The Results of the USMO-DNN under Various Training and Testing Conditions on the Hungarian Database

Training- Testing in Percentage					
Measures	**50-50**	**60-40**	**70-30**	**80-20**	**90-10**
Sensitivity	0.968	0.960	0.964	1	1
Specificity	0.830	0.860	0.906	0.863	1
Accuracy	0.918	0.924	0.943	0.95	1
PPV	0.910	0.924	0.948	0.926	1
NPV	0.936	0.925	0.935	1	1
FPR	0.169	0.139	0.093	0.136	0
FNR	0.031	0.039	0.035	0	0
FDR	0.090	0.075	0.051	0.073	0

https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

CONCLUSION

The proposed approach performs better than comparable techniques for identifying heart diseases successfully using USMO-DNN. For the Cleveland and Hungarian databases, the proposed technique achieves a precision of 98.38 percent, which is 1.66 percent higher than SMO-DNN and traditional DNN, 4.94 percent higher than ANN-LM technique, and 6.56 percent higher than ANN associates RP and SCG. As can be seen from the results, the inclusion of updated Cauchy distribution strategy enhances DNN's performance by identifying the optimal weight parameters for DNN. The multilayers between inputs and outputs further enhance this performance. DNN reveals skilled performance in all standard measures in comparison to other employed techniques, as demonstrated by USMO's DNN partners. The goal of this research is to find the optimal DNN weight value through other optimization techniques in the future, in order to improve performance even further while identifying health care diseases.

REFERENCES

[1] Katarya, Rahul, and Sunit Kumar Meena. "Machine learning techniques for heart disease prediction: a comparative study and analysis." Health and Technology 11, no. 1 (2021): 87-97.

[2] Mendis, Shanthi, Pekka Puska, Bo Norrving, and World Health Organization. Global atlas on cardiovascular disease prevention and control. World Health Organization, 2011.

[3] Kannan, R., and V. Vasanthi. "Machine learning algorithms with ROC curve for predicting and diagnosing the heart disease." In Soft Computing and Medical Bioinformatics, pp. 63-72. Springer, Singapore, 2019.

[4] Kumar, Priyan Malarvizhi, and Usha Devi Gandhi. "A novel three-tier Internet of Things architecture with machine learning algorithm for early detection of heart diseases." Computers & Electrical Engineering 65 (2018): 222-235.

[5] Dwivedi, Ashok Kumar. "Performance evaluation of different machine learning techniques for prediction of heart disease." Neural Computing and Applications 29, no. 10 (2018): 685-693.

[6] Khan, Mohammad Ayoub. "An IoT framework for heart disease prediction based on MDCNN classifier." IEEE Access 8 (2020): 34717-34727.

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

[7] Mathan, K., Priyan Malarvizhi Kumar, Parthasarathy Panchatcharam, Gunasekaran Manogaran, and R. Varadharajan. "A novel Gini index decision tree data mining method with neural network classifiers for prediction of heart disease." Design automation for embedded systems 22, no. 3 (2018): 225-242.

[8] Pandey, Rupali, Manoj Kumar, JihenMajdoubi, Mohammad Rahimi-Gorji, and Vivek Kumar Srivastav. "A review study on blood in human coronary artery: Numerical approach." Computer methods and programs in biomedicine 187 (2020): 105243.

[9] Aggrawal, Ritu, and Saurabh Pal. "Sequential feature selection and machine learning algorithm-based patient's death events prediction and diagnosis in heart disease." SN Computer Science 1, no. 6 (2020): 1-16.

[10] Tuli, Shreshth, Nipam Basumatary, Sukhpal Singh Gill, Mohsen Kahani, Rajesh Chand Arya, Gurpreet Singh Wander, and RajkumarBuyya. "HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments." Future Generation Computer Systems 104 (2020): 187-200.

[11] Ayon, Safial Islam, Md Milon Islam, and Md Rahat Hossain. "Coronary artery heart disease prediction: a comparative study of computational intelligence techniques." IETE Journal of Research (2020): 1-20.

[12] Ahmed, Hager, Eman MG Younis, Abdeltawab Hendawi, and Abdelmgeid A. Ali. "Heart disease identification from patients' social posts, machine learning solution on Spark." Future Generation Computer Systems 111 (2020): 714-722.

[13] Nandy, Sudarshan, Mainak Adhikari, Venki Balasubramanian, Varun G. Menon, Xingwang Li, and Muhammad Zakarya. "An intelligent heart disease prediction system based on swarm-artificial neural network." Neural Computing and Applications (2021): 1-15.

[14] Nilashi, Mehrbakhsh, Othman bin Ibrahim, Hossein Ahmadi, and Leila Shahmoradi. "An analytical method for diseases prediction using machine learning techniques." Computers & Chemical Engineering 106 (2017): 212-223.

[15] Venkatesan, C., P. Karthigaikumar, and S. Satheeskumaran. "Mobile cloud computing for ECG telemonitoring and real-time coronary heart disease risk detection." Biomedical Signal Processing and Control 44 (2018): 138-145.

[16] Manogaran, Gunasekaran, R. Varatharajan, and M. K. Priyan. "Hybrid recommendation system for heart disease diagnosis based on multiple kernel learning with adaptive neuro-fuzzy inference system." Multimedia tools and applications 77, no. 4 (2018): 4379-4399.

 https://doi.org/10.37896/pd91.4/91423

ISSN: 0369-8963

[17] Abdar, Moloud, Wojciech Książek, U. Rajendra Acharya, Ru-San Tan, Vladimir Makarenkov, and Paweł Pławiak. "A new machine learning technique for an accurate diagnosis of coronary artery disease." Computer methods and programs in biomedicine 179 (2019): 104992.

[18] Reddy, G. Thippa, M. Praveen Kumar Reddy, Kuruva Lakshmanna, Dharmendra Singh Rajput, Rajesh Kaluri, and Gautam Srivastava. "Hybrid genetic algorithm and a fuzzy logic classifier for heart disease diagnosis." Evolutionary Intelligence 13, no. 2 (2020): 185-196.

[19] Reddy, G. Thippa, and Neelu Khare. "An efficient system for heart disease prediction using hybrid OFBAT with rule-based fuzzy logic model." Journal of Circuits, Systems and Computers 26, no. 04 (2017): 1750061.

[20] Jayaraman, Vijayashree, and H. Parveen Sultana. "Artificial gravitational cuckoo search algorithm along with particle bee optimized associative memory neural network for feature selection in heart disease classification." Journal of Ambient Intelligence and Humanized Computing (2019): 1-10.

[21] Latha, C. Beulah Christalin, and S. Carolin Jeeva. "Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques." Informatics in Medicine Unlocked 16 (2019): 100203.

[22] Ali, Farman, Shaker El-Sappagh, SM Riazul Islam, DaehanKwak, Amjad Ali, Muhammad Imran, and Kyung-Sup Kwak. "A smart healthcare monitoring system for heart disease prediction based on ensemble deep learning and feature fusion." Information Fusion 63 (2020): 208-222.

[23] Arabasadi, Zeinab, Roohallah Alizadehsani, Mohamad Roshanzamir, Hossein Moosaei, and Ali Asghar Yarifard. "Computer aided decision making for heart disease detection using hybrid neural network-Genetic algorithm." Computer methods and programs in biomedicine 141 (2017): 19-26.

[24] Rani, Pooja, Rajneesh Kumar, Nada MO Sid Ahmed, and Anurag Jain. "A decision support system for heart disease prediction based upon machine learning." Journal of Reliable Intelligent Environments (2021): 1-13.

[25] Shah, Devansh, Samir Patel, and Santosh Kumar Bharti. "Heart disease prediction using machine learning techniques." SN Computer Science 1, no. 6 (2020): 1-6.

[26] Gokulnath, Chandra Babu, and S. P. Shantharajah. "An optimized feature selection based on genetic approach and support vector machine for heart disease", Cluster Computing 22, no. 6 (2019): 14777-14787.

 https://doi.org/10.37896/pd91.4/91423