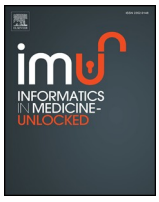
Informatics in Medicine Unlocked 16 (2019) 100203

Contents lists available at ScienceDirect

Informatics in Medicine Unlocked

journal homepage: www.elsevier.com/locate/imu



Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques

C. Beulah Christalin Latha, S. Carolin Jeeva*

Karunya Institute of Technology and Sciences, India

ARTICLE INFO

Keywords:
Heart disease
Machine learning
Ensemble classifier
Prediction model

ABSTRACT

Machine learning involves artificial intelligence, and it is used in solving many problems in data science. One common application of machine learning is the prediction of an outcome based upon existing data. The machine learns patterns from the existing dataset, and then applies them to an unknown dataset in order to predict the outcome. Classification is a powerful machine learning technique that is commonly used for prediction. Some classification algorithms predict with satisfactory accuracy, whereas others exhibit a limited accuracy. This paper investigates a method termed ensemble classification, which is used for improving the accuracy of weak algorithms by combining multiple classifiers. Experiments with this tool were performed using a heart disease dataset. A comparative analytical approach was done to determine how the ensemble technique can be applied for improving prediction accuracy in heart disease. The focus of this paper is not only on increasing the accuracy of weak classification algorithms, but also on the implementation of the algorithm with a medical dataset, to show its utility to predict disease at an early stage. The results of the study indicate that ensemble techniques, such as bagging and boosting, are effective in improving the prediction accuracy of weak classifiers, and exhibit satisfactory performance in identifying risk of heart disease. A maximum increase of 7% accuracy for weak classifiers was achieved with the help of ensemble classification. The performance of the process was further enhanced with a feature selection implementation, and the results showed significant improvement in prediction accuracy.

1. Introduction

One of the prominent diseases that affect many people during middle or old age is heart disease, and in many cases it eventually leads to fatal complications [3]. Heart diseases are more prevalent in men than in women. According to statistics from WHO, it has been estimated that 24% of deaths due to non-communicable diseases in India are caused by heart ailments [12,19]. One-third of all global deaths are due to heart diseases [10]. Half of the deaths in the United States and in other developed countries are due to heart ailments [18]. Around 17 million people die due to cardiovascular disease (CVD) every year worldwide, and the disease is highly prevalent in Asia [2,12,13]. The Cleveland Heart Disease Database (CHDD) is considered the de facto database for heart disease research [17].

Age, sex, smoking, family history, cholesterol, poor diet, high blood pressure, obesity, physical inactivity, and alcohol intake are considered to be risk factors for heart disease, and hereditary risk factors such as high blood pressure and diabetes also lead to heart disease. Some risk factors are controllable. Apart from the above factors, lifestyle habits such as eating habits, physical inactivity, and obesity are also considered to be major risk factors [5,8,15]. There are different types of heart diseases such as coronary heart disease, angina pectoris, congestive heart failure, cardiomyopathy, congenital heart disease, arrhythmias, and myocarditis. It is difficult to manually determine the odds of getting heart disease based on risk factors [1]. However, machine learning techniques are useful to predict the output from existing data. Hence, this paper applies one such machine learning technique called classification for predicting heart disease risk from the risk factors. It also tries to improve the accuracy of predicting heart disease risk using a strategy termed ensemble.

2. Literature review

Machine learning or data mining is useful for a diverse set of problems. One of the applications of this technique is in predicting a dependent variable from the values of independent variables. The healthcare field is an application area of data mining since it has vast data resources that are difficult to be handled manually. Heart disease

* Corresponding author.
E-mail address: caroljeeva@gmail.com (S.C. Jeeva).

https://doi.org/10.1016/j.imu.2019.100203
Received 26 November 2018; Received in revised form 30 June 2019; Accepted 1 July 2019
Available online 02 July 2019

has been identified as one of the largest causes of death even in developed countries [20]. One of the reasons for fatality due to heart disease is due to the fact that the risks are either not identified, or they are identified only at a later stage. However, machine learning techniques can be useful for overcoming this problem and to predict risk at an early stage. Some of the techniques used for such prediction problems are the Support Vector Machines (SVM), Neural Networks, Decision Trees, Regression and Naïve Bayes classifiers. SVM was identified as the best predictor with 92.1% accuracy, followed by neural networks with 91% accuracy, and decision trees showed a lesser accuracy of 89.6% [23]. Sex, age, smoking, hypertension, and diabetes were considered to be the risk factors for heart disease [22]. Analytical studies on data mining techniques for heart disease prediction reveal that neural networks, decision trees, Naïve Bayes and associative classification are powerful in predicting heart disease. Associative classification produces a high accuracy and strong flexibility as compared with traditional classifiers, even in handling unstructured data [7,8].

A comparative analysis of classification techniques has shown that decision tree classifiers are simple and accurate [9]. Naïve Bayes was found to be the best algorithm, followed by neural networks and decision trees [7]. Artificial neural networks are also employed for the prediction of diseases. Supervised networks have been used for diagnosis and they can be trained using the Back Propagation Algorithm. The experimental results have shown satisfactory accuracy [16].

The existing research has used ensemble methods to improve classification accuracy in prediction of heart disease [2]. A combination of genetic algorithms and neural networks based on fuzzy logic for feature extraction exhibited an increase in accuracy of up to 99.97% [6]. A genetic algorithm based trained recurrent fuzzy neural network produced an accuracy of 97.78% for diagnosing heart disease [10]. Classification accuracy of up to 93% was achieved in the prediction of heart disease risk using a rough set based classification system with a different dataset [22]. Neural networks were also used to reduce human error in the detection and measurement of blood sugar, blood pressure, and heart disease [15,18]. A new model coactive neuro-fuzzy inference system (CANFIS) combined with neural networks, fuzzy logic and genetic algorithms, was shown to produce good results for predicting heart disease. The genetic algorithm was used for tuning the parameters for CANFIS automatically, and for the selection of an optimal feature set. The model was shown to be a useful tool for assisting medical professionals in predicting heart disease [11]. In order to obtain better accuracy, an additional step of feature selection has been proposed [19].

SVM based classifiers had been shown to provide highly accurate output for classifying heartbeats. The parameters have been optimized using particle swarm optimization (PSO). The performance of the classifier was improved using PSO [1,21]. The K-means clustering algorithm was utilized to extract data from the dataset and the frequent patterns were mined using the Maximal Frequent Itemset Algorithm (MAFIA) for predicting heart disease based on different weightage assigned to different factors. The frequent patterns having a value greater than a specific threshold were found to be precise in detecting the occurrence of a myocardial infarction [18]. Though various methods were used for predicting heart disease risks with good accuracy in state-of-the-art research, some classification algorithms identify heart disease risk with poor accuracy. Most of the state-of-art research that produces high accuracy employs a hybrid method which include classification algorithms. Our study described herein focused on improving the weakness of weak classification algorithms by combining them with other classification algorithms. This assists not only to increase the efficiency of such classification algorithms, but also the prediction accuracy for heart disease. A research on using ensemble techniques such as bagging, boosting, majority voting, and stacking is done and the results are evaluated. The results are further enhanced by applying feature selection. The results are a measure to indicate how these classifiers can effectively be used in the medical field.

Table 1
Feature information of the cleveland dataset.

S.No	Attribute Name	Description	Range of Values
1	Age	Age of the person in years	29 to 79
2	Sex	Gender of the person [1: Male, 0: Female]	0, 1
3	Cp	Chest pain type [1-Typical Type 1 Angina 2- Atypical Type Angina 3-Non-angina pain 4-Asymptomatic)	1, 2, 3, 4
4	Trestbps	Resting Blood Pressure in mm Hg	94 to 200
5	Chol	Serum cholesterol in mg/dl	126 to 564
6	Fbs	Fasting Blood Sugar in mg/dl	0, 1
7	Restecg	Resting Electrocardiographic Results	0, 1, 2
8	Thalach	Maximum Heart Rate Achieved	71 to 202
9	Exang	Exercise Induced Angina	0, 1
10	OldPeak	ST depression induced by exercise relative to rest	1 to 3
11	Slope	Slope of the Peak Exercise ST segment	1, 2, 3
12	Ca	Number of major vessels colored by fluoroscopy	0 to 3
13	Thal	3 – Normal, 6 – Fixed Defect, 7 – Reversible Defect	3, 6, 7
14	Num	Class Attribute	0 or 1

3. Materials and methods

3.1. Description of the dataset

The Cleveland heart dataset from the UCI machine learning repository has been used for the experiments. The dataset consists of 14 attributes and 303 instances. There are 8 categorical attributes and 6 numeric attributes. The description of the dataset is shown in Table 1.

Patients from age 29 to 79 have been selected in this dataset. Male patients are denoted by a gender value 1 and female patients are denoted by gender value 0. Four types of chest pain can be considered as indicative of heart disease. Type 1 angina is caused by reduced blood flow to the heart muscles because of narrowed coronary arteries. Type 1 Angina is a chest pain that occurs during mental or emotional stress. Non-angina chest pain may be caused due to various reasons and may not often be due to actual heart disease. The fourth type, Asymptomatic, may not be a symptom of heart disease. The next attribute *trestbps* is the reading of the resting blood pressure. *Chol* is the cholesterol level. *Fbs* is the fasting blood sugar level; the value is assigned as 1 if the fasting blood sugar is below 120 mg/dl and 0 if it is above. *Restecg* is the resting electrocardiographic result, *thalach* is the maximum heart rate, *exang* is the exercise induced angina which is recorded as 1 if there is pain and 0 if there is no pain, *oldpeak* is the ST depression induced by exercise, *slope* is the slope of the peak exercise ST segment, *ca* is the number of major vessels colored by fluoroscopy, *thal* is the duration of the exercise test in minutes, and *num* is the class attribute. The class attribute has a value of 0 for normal and 1 for patients diagnosed with heart disease.

3.2. Classification and ensemble algorithms

Classification is a supervised learning procedure that is used for predicting the outcome from existing data. This paper proposes an approach for the diagnosis of heart disease using classification algorithms, and to improve the classification accuracy using an ensemble of classifiers. The dataset has been divided into a training set and a test set, and individual classifiers are trained using the training dataset. The efficiency of the classifiers is tested with the test dataset. The working of the individual classifiers is explained in the next section.

3.2.1. Bayes Net

The Bayesian network is a graphical prediction model based on probability theory. Bayesian networks are built from probabilistic distributions, and they utilize the laws of probability for prediction and diagnosis. Bayesian networks support both discrete and continuous variables. The network is represented as a set of variables whose conditional dependencies are described using acyclic directed graphs. In a Bayesian network, edges between the nodes represent dependent features, whereas nodes that are not connected are conditionally independent. Let X be an evidence that is dependent on n attributes $X = \{A_1, A_2, \ldots, A_n\}$. Let H be a hypothesis that the evidence belongs to a class C. The probability of the hypothesis H, given the evidence X is represented as $P(H|X)$. $P(X|H)$ is the posterior probability of X conditioned on H. The posterior probability can be calculated using the Bayes theorem as shown in equation (1).

$$P(H|X) = P(X|H)P(H)/P(X) \quad (1)$$

Where P(H) is the probability of the hypothesis being *True*. P(X) is the probability of the evidence. P(X|H) is the probability of the evidence given that hypothesis is *True* and P(H|X) is the probability of the hypothesis given that the evidence is present.

3.2.2. Naive Bayes

The Naïve Bayes classifier or simply, the Bayesian classifier, is based on the Bayes theorem. It is a special case of the Bayesian network, and it is a probability based classifier. In the Naïve Bayes network, all features are conditionally independent. The changes in one feature therefore does not affect another feature. The Naïve Bayes algorithm is suitable for classifying high dimensional datasets. The classifier algorithm uses conditional independence. Conditional independence assumes that an attribute value is independent of the values of the other attributes in a class.

Let D be a set of training data and associated class labels. Each tuple in the dataset is defined with n attributes that are represented by $X = \{A_1, A_2, \ldots, A_n\}$. Let there be m classes represented by $C_1, C_2, \ldots C_m$. For a given tuple X, the classifier predicts that X belongs to the class having the highest posterior probability, conditioned on X. The Naïve Bayes classifier predicts that the tuple X belongs to the class C_i if and only if

$$P(C_i|X) > P(C_j|X) \quad for \;\; 1 \le j \le m, \; j \ne i \quad (2)$$

Thus, $P(C_i|X)$ is maximized. The class C_i for which $P(C_i|X)$ is maximized is called the maximum posteriori hypothesis. According to Bayes' theorem,

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \quad (3)$$

If the attribute values are conditionally independent of one another,

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \quad (4)$$

Where x_k refers to the value of attribute A_k for tuple X.

If A_k is categorical, then $P(x_k|C_i)$ is the number of tuples of class C_i in D having the value x_k for A_k, divided by $|C_{i,D}|$, the number of tuples of class C_i in D. The classifier predicts the class label of X is the class C_i if and only if,

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \quad for \quad 1 \le j \le m, \; j \ne i \quad (5)$$

Bayesian classifiers are effective in the sense that they have the minimum error rate for classification.

3.2.3. Random forest

Random forest is a tree based classification algorithm. As the name indicates, the algorithm creates a forest with a large number of trees. It is an ensemble algorithm which combines multiple algorithms. It creates a set of decision trees from a random sample of the training set. It repeats the process with multiple random samples and makes a final decision based on majority voting. The Random forest algorithm is effective in handling missing values but it is prone to overfitting. Appropriate parameter tuning can be applied to avoid overfitting. The algorithm for Random forest is shown in Fig. 1.

Let D be a training set $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$
Let $h = h_1(x), h_2(x), \ldots, h_k(x)$, *an ensemble of weak classifiers*
If each h_k*is a decision tree, the parameters of the tree are defined as* $\Theta = (\theta_{k1}, \theta_{k2}, \ldots, \theta_{kp})$
Each decision tree k leads to a classifier $h_k(X) = h(X|\Theta_k)$
Final Classification f(x) = Majority of $h_k(X)$

Fig. 1. Random forest algorithm.

3.2.4. C4.5

The C4.5 algorithm is derived from the ID3 algorithm, which is a simple decision tree algorithm. The algorithm was proposed by Quinlan. It uses information gain ratio as the metric for splitting the trees. It accepts data as input and produces a decision tree as output. This algorithm creates univariate trees. Classification rules are framed in the form of decision trees. The splitting of trees is halted when the split is below a certain threshold value. It performs error based pruning and it is a good algorithm for handling numeric attributes. The algorithm for generating a decision tree from training tuples using C4.5 algorithm is shown in Fig. 2.

3.2.5. Multilayer perceptron

Artificial neurons are used in multiple layers including hidden layers in the multilayer perceptron algorithm. These algorithms are used for binary classification problems. A perceptron uses an activation function for each neuron. Multilayer perceptrons are algorithms evolved from biological neurons. They use artificial neurons that are called perceptrons. The activation function maps the weighted inputs of each neuron and reduces the number of layers to two layers. A perceptron learns by varying the weights assigned to it. The algorithm for a multilayer perceptron is shown in Fig. 3.

3.2.6. PART

PART is the acronym for Projective Adaptive Resonance Theory. PART is a rule-based classification algorithm. It is a neural network developed by Cao and Wu. It is an advanced version of the C4.5 and RIPPER algorithms. The PART algorithm is suitable for high dimensional datasets. The key feature of the PART network lies is the presence of a hidden layer of neurons, which calculate the variations between the output and input neurons, and work on reducing the similarity differences.

Let N be a node
Let D be a tuple in class C and X be the set of attributes $A = \{X_1, X_2, \ldots, X_3\}$
if $D \subseteq C$
  return $N = L(C)$, *where L is a leaf node*
if $A = \emptyset$ *then*
  return N as a leaf node labeled with the majority class in D
Split N with the best splitting criterion
for each splitting criterion j
  D_j *– set of tuples satisfying j*
if $D_j = \emptyset$ *then*
  attach a leaf labeled with the majority class in D to node N;
else
  attach the node returned by Generate decision tree(Dj, attribute list) to node N;
end for
return N

Fig. 2. C4.5 algorithm.

Initialize weights and biases in N, where N is the Network
while *condition is true {*
for *each training tuple* ***X*** *in D {*
for *each input layer unit j {*
$O_j = I_j$
for *each hidden or output layer unit j {*
$I_j = \sum_i W_{ij} O_i + \theta_j$
$O_j = \frac{1}{1 + e^{-I_j}}$
for *each unit j in the output layer*
$Err_j = O_j(1 - O_j)(T_j - O_j)$
for *each unit j in the hidden layers, from the last to the first hidden layer*
$Err_j = O_j(1 - O_j)\sum_k Err_k w_{jk}$
for *each weight* w_{ij} *in N {*
$\Delta w_{ij} = (l) Err_i O_i$
$w_{ij} = w_{ij} + \Delta w_{ij}$}
for *each bias* θ_j *in N {*
$\Delta\theta_j = (l) Err_j$
$\theta_j = \theta_j + \Delta\theta_j$
}}

Fig. 3. Multilayer perceptron algorithm.

3.3. Ensemble techniques

Ensemble is a strategy that can be used to improve the accuracy of a classifier. It is an effective meta classification technique that combines weak learners with strong learners to improve the efficacy of the weak learner. In this paper, the ensemble technique is used to improve accuracy of various algorithms for heart disease prediction. The aim of combining multiple classifiers is to obtain better performance as compared with an individual classifier. The procedure for ensemble is shown in Fig. 4.

3.3.1. Boosting

Boosting is an algorithm used for ensembling. In boosting, the original dataset is divided into various subsets. The classifier is trained with the subset to produce a series of models of moderate performance. New subsets are created based on the elements that are not correctly classified by the previous model.

Then, the ensembling process boosts their performance by combining the weak models together using a cost function. The algorithm for boosting is shown in Fig. 5.

3.3.2. Bagging

Bagging is also known as *bootstrap aggregation*. Bagging randomly selects some patterns from the training set with replacement. The newly

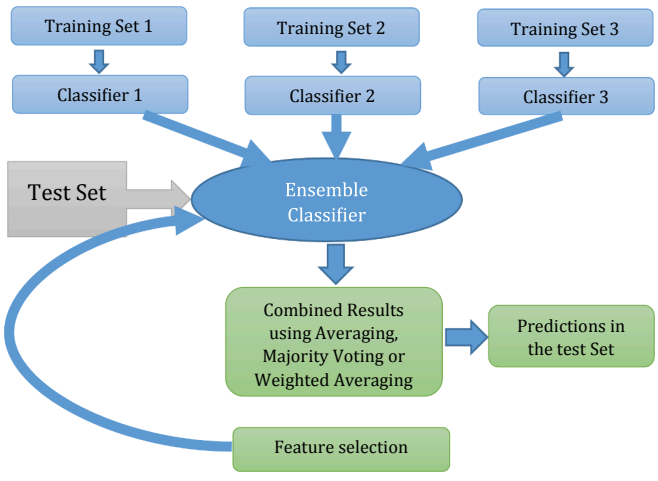


Fig. 4. The ensemble process.

Let D={d_1,d_2,d_3, … d_n} be the given dataset
E = {}, the set of ensemble classifiers
C = {c_1, c_2, c_3, …c_n}, the set of classifiers
X = the training set, X∈D
Y = the test set, Y∈D
L = n(D)
Let init = 1
S(init)=A random subset of X; S(init)⊂X
M(0) = { }
fori =1 to L do
if i>1
s(i) = Set of incorrectly classified instances of M(i-1) + S(i)
M(i) = Model trained using C(i) on S(i)
E = E∪C(i)
end if
next i
for i = 1 to L
R(i) = Y classified by E(i)
next i
Result = max(R(i): i=1,2, …, n)

Fig. 5. Algorithm for boosting.

created training set will have the same number of patterns as the original training set with a few omissions and repetitions. The new training set is known as *Bootstrap replicate*. In bagging, bootstrap samples are fetched from the data and the classifier is trained with each sample. The voting from each classifier is combined, and the classification result is selected based on majority voting or averaging. Research shows that bagging can be used to increase the performance of a weak classifier optimally. The algorithm for bagging is shown in Fig. 6.

Bagging decreases the variance of prediction, since it generates multiple sets of data from random samples of the original dataset, with replacement.

3.3.3. Stacking

Stacking is an ensemble technique in which multiple classification models are combined via a meta classifier. Multiple layers are placed one after the other, where each of the models pass their predictions to the model in the layer above, and the model in the topmost layer makes decisions based on the models below. The bottom layer models receive input features from the original dataset. The top layer model takes the

Let D={d_1,d_2,d_3, … d_n} be the given dataset
E = {}, the set of ensemble classifiers
C = {c_1, c_2, c_3, …c_n}, the set of classifiers
X = the training set, X ∈ D
Y = the test set, Y ∈ D
L = n(D)
for i =1 to L do
S(i) = {Bootstrap sample I with replacement} I⊂X
M(i) = Model trained using C(i) on S(i)
E = E∪C(i)
next i
for i = 1 to L
R(i) = Y classified by E(i)
next i
Result = max(R(i): i=1,2, …, n)

Fig. 6. Algorithm for bagging.

Let D={d_1,d_2,d_3, … d_n} be the given dataset
E = {E_1, E_2, E_3, …E_n}, the set of ensemble classifiers
C = {c_1, c_2, c_3, …c_n}, the set of classifiers
X = the training set, X ∈ D
Y = the test set, Y ∈ D
K = meta level classifier
L = n(D)
for i =1 to L do
M(i) = Model trained using E(i) on X
Next i
M=M∪K
Result = Y classified by M

Fig. 7. Algorithm for stacking.

output from the bottom layer and makes the prediction. The algorithm for stacking is shown in Fig. 7. In stacking, the original data is provided as input to several individual models. Then the meta classifier is used to estimate the input together with the output of each model and the weights are estimated. The best performing models are selected and the others are discarded. Stacking combines multiple base classifiers trained by using different learning algorithms L on a single dataset S, by means of a meta classifier.

3.3.4. Majority vote

The majority voting classifier is a meta classifier that is used to combine any classifier through majority voting. The final class label would be the class label that had been predicted by a majority of the classifiers. The final class label d_J is defined as

$$d_J = mode\{C_1, C_2, ..., C_n\}$$

Where $\{C_1, C_2, ..., C_n\}$ represents the individual classifiers that participate in the voting. The majority voting algorithm is shown in Fig. 8.

4. Experiments and results

4.1. Performance of the classifier with ensemble

A comparative analysis of various classification algorithms on the Cleveland dataset has been performed. Some algorithms show good accuracy whereas some other algorithms perform poorly. In order to improve the performance of the weak classifiers, ensemble algorithms are used. This work has used ensemble algorithms such as bagging, boosting, voting, and stacking. The Bagging algorithm performs an ensemble with the Naïve Bayes, Random Forest, Bayes Net, C4.5, multilayer perceptron and PART algorithms. For boosting, the Adaboost.M1 algorithm has been used. For the present study, ensembles are created using Naïve Bayes, Random Forest, Bayes Net, C4.5, multilayer perceptron and PART classifiers for boosting. Majority voting has also been used as one of the ensemble techniques. In stacking, the Naïve Bayes classifier is used as the meta classifier, and the results are obtained by stacking one, two, and three more classifiers, respectively.

Let $c_{i,j}$be the prediction of the i^{th} classifier on a class with j labels

$$\sum_{i=1}^{n} c_{i,j} = max_{j=1,...,m} \sum_{t=1}^{n} c_{i,j}$$

The ensemble classifier's probability for the decision to be better is

$$P_{ens} = \sum_{k=\left(\frac{n}{2}\right)+1}^{n} \binom{n}{k} p^k (1-p)^{n-k}$$

Fig. 8. Algorithm for majority voting.

The results show that weak classifiers can perform better when they are ensembled. The Weka tool is used for classification of the dataset.

At the outset, the dataset is cleaned and preprocessed for missing data and invalid data. Then, classifiers such as SVM, Naive Bayes, Bayes Net, C4.5, Multilayer Perceptron and PART are used for the classification of the dataset. C4.5, Multilayer Perceptron and PART are found to be weaker than the classifiers such as Naive Bayes, Random forest and Bayes Net. Since, ensemble is a proven strategy for boosting the classification accuracy, the weak learners are tested with the meta classification algorithms. Three types of techniques, namely, bagging, boosting and stacking are used for ensembling, and the results are analyzed. Ten-fold cross validation was utilized to evaluate the performance of the classification models. In this approach, the entire dataset is divided into ten subsets and processed ten times where, nine subsets are used as testing sets and the remaining subset is used as training. Finally, the results are obtained by averaging each ten iterations.

Fig. 9 compares the classification accuracy of individual classifiers and with bagging. When the dataset is classified using individual classifiers, the accuracy rates of Naïve Bayes, Random forest, Bayes Net, C4.5, Multilevel Perceptron and PART are found in the range of 75.58%–83.17%. The Naïve Bayes classifier exhibits the best accuracy of 83.17% whereas C 4.5, Multilevel Perceptron and PART show comparatively poor accuracy of less than 80%. It has been inferred from the results that the bagging technique can increase the classification accuracy by up to 6.92%.

Fig. 10 shows the results of the ensemble technique, namely boosting. There was an increase of 0.99% for the Naïve Bayes algorithm, 1.65% for the Bayes Net, 1% for multilayer perceptron and 5.94% for PART through boosting. The Naïve Bayes algorithm produced the highest accuracy value with boosting.

Majority voting is another ensemble strategy that combines multiple classifiers in order to improve their accuracy. In the proposed approach, for the Cleveland dataset, C4.5, multilayer perceptron and PART classifiers turned out to be weak classifiers and they showed less accuracy. Naïve Bayes and Bayes Net performed well and had better classification accuracy. It is inferred from Fig. 11 that an ensemble of weak classifiers with strong classifiers using majority voting improves the accuracy of the weak classifier to a considerable extent. Ensembling C4.5 with the strong classifiers improved the accuracy by 3.3%. Ensembling PART with the strong classifier set improved the accuracy by 7.26%. Ensembling multilayer perceptron with the strong classifier set improved the accuracy by 3.65%.

Stacking is a methodology used for ensembling in which one or more base level classifiers are stacked with a meta level classifier. In this paper, random forest and random tree classifiers are used as meta classifiers. Naïve Bayes, Bayesian Network, C4.5 and PART are chosen as the base level classifiers. It is inferred from Fig. 12 that stacking with random forest produces better accuracy than stacking with random tree as the base classifier.

While stacking with random trees, the PART algorithm alone showed an increase in accuracy by 1.98% whereas all of the other algorithms showed a decline of up to 2.96%. However, when stacked with random forest as the base classifier, the accuracy of the weak classifiers improved. The accuracy of the Bayesian network improved by 0.99%, C4.5 by 3.3%, Multilayer perceptron by 3.64% and PART by 6.93%. It is inferred that when the weak classifiers are stacked with random forest, the accuracy is higher than when they are stacked with random tree.

A comparative analysis of bagging and boosting is shown in Fig. 13. The results show that both bagging and boosting are efficient in increasing the accuracy of weak classifiers. Bagging shows a better improvement for all weak classifiers. The computation time is calculated as the average of 100 runs. The unit is in seconds. A comparison of the computation time of the classifiers with bagging and boosting techniques are shown in Table 2.

A comparison of the various ensembling strategies reveal that the

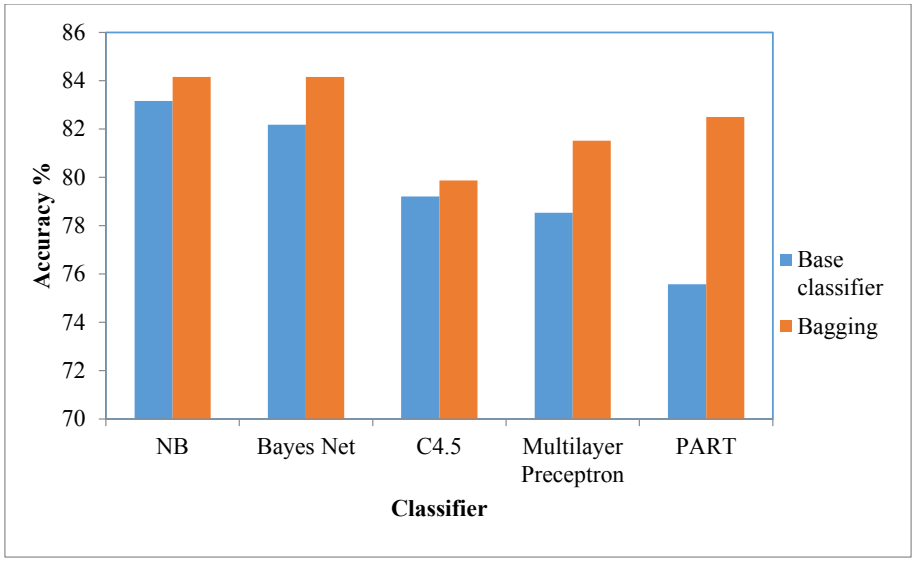


Fig. 9. Improvement in accuracy of classifiers with bagging.

accuracy of the weak classifiers can be increased by a maximum of 7.26%. The maximum increase in accuracy of a weak classifier with various ensembling techniques is shown in Fig. 14. The results show that ensemble is a good strategy for improving the accuracy of weak classifiers, and majority voting produces the highest increase in accuracy.

4.2. Performance enhancement using feature selection

The accuracy of the classifiers are further improved using feature selection [4]. Six sets of features were selected for the evaluation of the performances. The attributes 'age' and 'Sex' are considered as the personal information of the patient and the remaining 11 attributes are collected from the medical observation of the patient. The Brute force method is applied to limit the lower bound with a minimum of 3 attributes [14]. In this work, all of the possible combinations of 3 attributes from the 13 attributes were selected, and each combination was tested with the classifiers. Secondly, the experiment was repeated to select the possible combination of 4 attributes from the total 13 attributes.

The maximum number of combination from 13 attributes, without considering the empty set, is represented as $2^n - 1$. In this experiment, the combination less than 3 attributes are omitted. The total number of combination is derived as follows.

Total number of combination

$$2^n - \left(\frac{n!}{1!(n-1)!}\right) - \left(\frac{n!}{2!(n-2)!}\right) - 1 = 2^n - \left(\frac{n^2+n}{2} + 1\right)$$

Where n represents 13 attributes.

The features are named as FS1, FS2, FS3, FS4, FS5 and FS6. The description of the features are shown below:

FS1 = {sex, cp, fbs, restecg, oldpeak, ca, thal}
FS2 = {age, sex, cp, chol, fbs, exang, oldpeak, slope, ca}
FS3 = {sex, cp, fbs, thalach, exang, slope, ca, thal}
FS4 = {sex, cp, thalach, exang, oldpeak, ca}
FS5 = {age, sex, cp, chol, restecg, oldpeak, slope, ca, thal}
FS6 = {sex, cp, trestbps, fbs, restecg, thalach, exang, oldpeak, slope, ca, thal}

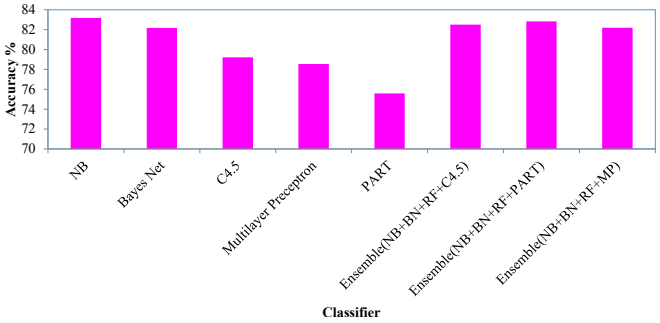


Fig. 11. Classifiers with majority voting.

The improvement in accuracy of bagging with feature selection is tabulated in Table 3.

The highest increase in accuracy of 2.31% was observed for the C4.5 classifier with bagging, with feature set 1. The accuracy of the multilayer perceptron was increased by 0.66% by feature sets FS4 and FS6. The accuracy of the random forest classifier was increased by 1.65%

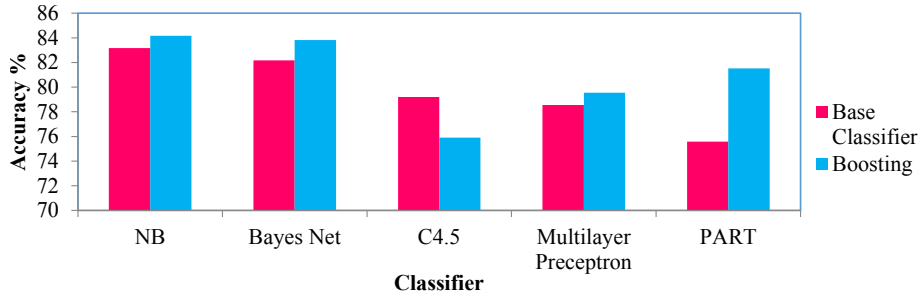


Fig. 10. Improvement in accuracy of classifiers with boosting.

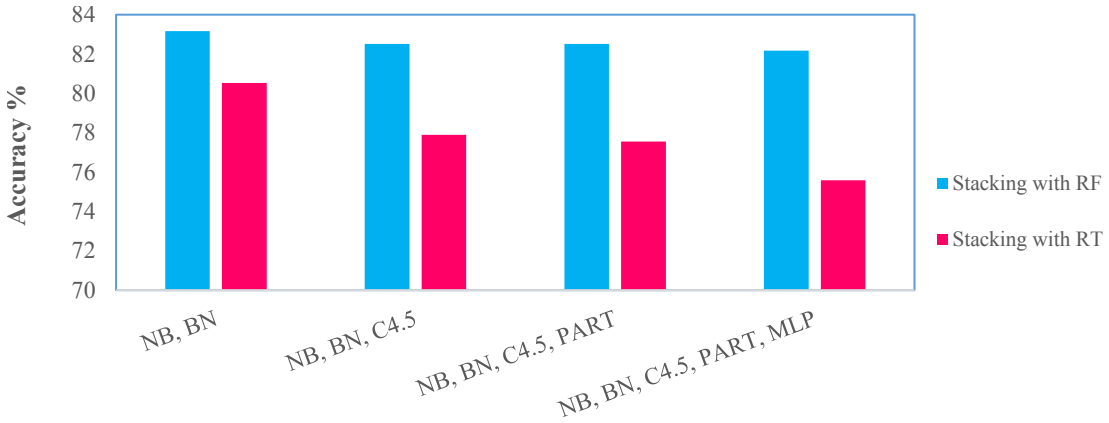


Fig. 12. Classifier with stacking.

with feature set FS6.

It has been inferred from the results that the accuracy of boosting was improved by a maximum of 3.97% with the C 4.5 classifier and feature set FS6. The maximum increase in accuracy in boosting with random forest was recorded as 3.3% with feature set FS6. For boosting with multilayer perceptron there has been an increase of 1.32% with feature set FS4. For the Naïve Bayes classifier with boosting there was an increase of 0.33% with feature set FS6. Thus, feature set FS6 has been effective in increasing the prediction rate of classifiers C 4.5, Random forest, and Naïve Bayes. The results are tabulated in Table 4.

Feature selection shows improvement in majority voting also. Majority voting of Naïve Bayes, Bayes Net, Random Forest and Multilayer Perceptron was improved by all of the feature sets; the highest increase in accuracy was 3.29%, with feature set FS2. The increase in accuracy of the majority voting of Naïve Bayes, Bayes Net, Random Forest and Multilayer Perceptron is shown in Fig. 15.

The maximum increase in accuracy was achieved by feature set FS2. The increase in accuracy of majority voting of Naïve Bayes, Bayes Net, Random Forest and C 4.5 is shown in Fig. 16.

The maximum increase in accuracy was caused by feature set FS6. The increase in accuracy of the majority voting of Naïve Bayes, Bayes Net, Random Forest and PART is shown in Fig. 17.

Feature selection shows improvement in stacking also. Feature set FS3 increases the accuracy of stacking Naïve Bayes, 201Bayesian Net, C4.5 and PART with Random Forest by 0.94%. However, significant improvements in accuracy were observed when features selection was applied to stacking with Random Tree. The results are shown in Table 5.

The highest increase of 4.63% was observed when feature set FS2 was applied to the stack of Naïve Bayes, Bayes Net, C4.5, PART and MLP with random tree. The comparison of the proposed model with the existing model is shown in Table 6.

Table 2
Comparison of Computation time of Bagging and Boosting.

Classification Algorithm	Without Ensembling (secs)	With Bagging (secs)	With Boosting (secs)
Naïve Bayes	0.04	0.03	0.21
Bayes Net	0.04	0.11	0.13
C 4.5	0.04	0.46	0.3
Multilayer Perceptron	2.8	8.06	14.36
PART	0.09	0.45	0.99

5. Conclusion

This paper analyses the accuracy of prediction of heart disease using an ensemble of classifiers. The Cleveland heart dataset from the UCI machine learning repository was utilized for training and testing purposes. The ensemble algorithms bagging, boosting, stacking and majority voting were employed for experiments. When bagging was used, the accuracy was improved by a maximum of 6.92%. When boosting

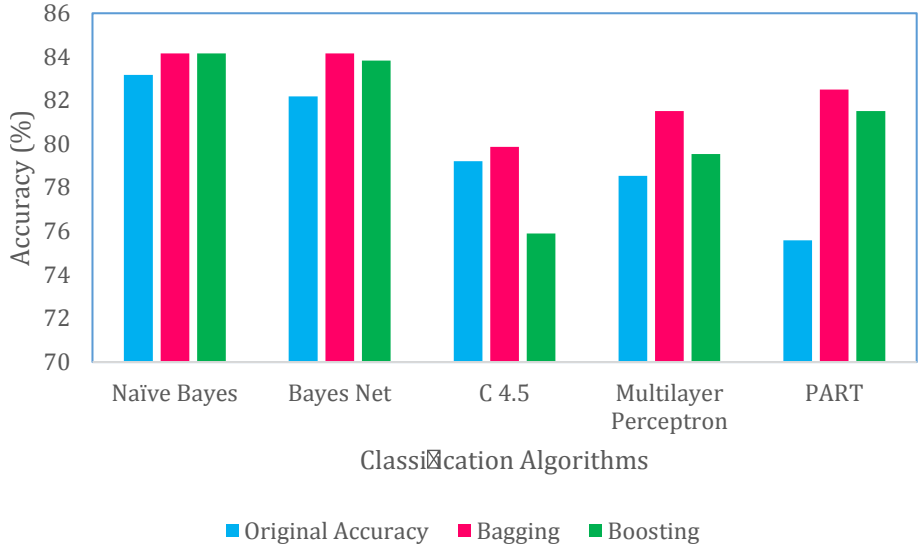


Fig. 13. Comparative analysis of bagging and boosting.

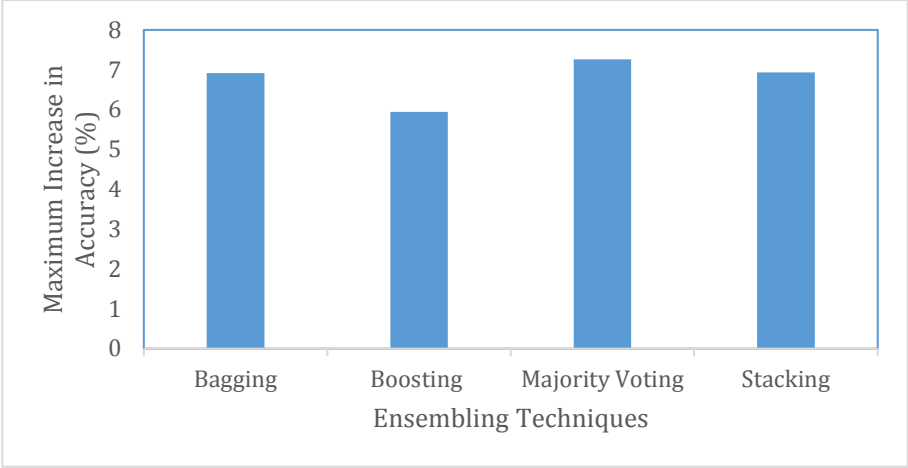


Fig. 14. Comparison of ensembling methods.

Table 3
Improvement in bagging accuracy with feature selection.

Algorithm	Bagging Accuracy	Increase in accuracy with Feature Selection	Feature Set
C4.5	79.87	82.18	FS1
Random Forest	80.53	82.18	FS6
Random Forest	80.53	81.52	FS2
Multilayer Perceptron	81.52	82.18	FS6
Multilayer Perceptron	81.52	82.18	FS4
Multilayer Perceptron	81.52	81.85	FS3
Bayes Net	84.16	84.82	FS1
Naïve Bayes	84.16	84.49	FS6

Table 4
Improvement in Boosting Accuracy with feature selection.

Algorithm	Boosting Accuracy	Increase in Accuracy with Feature Selection	Feature Set
C 4.5	75.9	79.87	FS6
C 4.5	75.9	79.21	FS3
C 4.5	75.9	78.22	FS2
C 4.5	75.9	77.23	FS5
C 4.5	75.9	76.57	FS4
Random Forest	78.88	82.18	FS6
Random Forest	78.88	80.86	FS2
Random Forest	78.88	80.86	FS3
Random Forest	78.88	79.87	FS5
Multilayer Perceptron	79.54	80.86	FS4
Multilayer Perceptron	79.54	80.53	FS5
Naïve Bayes	84.16	84.49	FS6

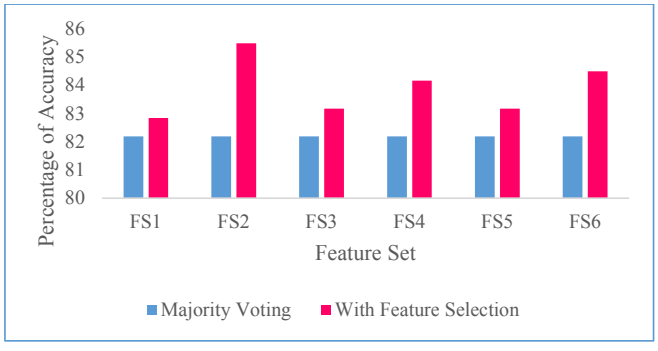


Fig. 15. Increase in Accuracy of Majority Voting of NB, BN, RF and MP using Feature Selection.

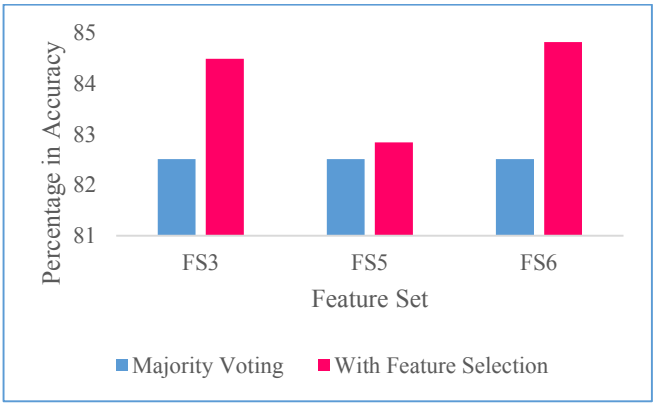


Fig. 16. Increase in Accuracy of Majority Voting of NB, BN, RF and C4.5 using Feature Selection.

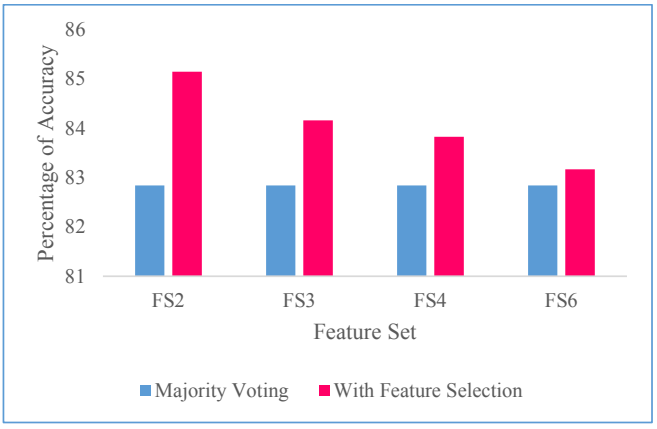


Fig. 17. Increase in Accuracy of Majority Voting of NB, BN, RF and PART using Feature Selection.

was used, the accuracy was improved by a maximum of 5.94%. When the weak classifiers are ensembled with majority voting, the accuracy was improved by a maximum of 7.26%, and stacking improved the accuracy by a maximum of 6.93%. A comparison of results showed that majority voting produces the highest improvement in accuracy. The performance was further enhanced using feature selection techniques. The feature selection techniques helped to improve the accuracy of the ensemble algorithms. The highest accuracy was obtained with majority voting with the feature set FS2.

Table 5
Accuracy with stacking and feature selection.

Algorithms Stacked with random tree	Stacking Accuracy	Accuracy with feature selection	Feature Set
Naïve Bayes, Bayes Net, C4.5	77.89	78.55	FS3
Naïve Bayes, Bayes Net, C4.5	77.89	78.55	FS4
Naïve Bayes, Bayes Net, C4.5, PART	77.56	78.22	FS3
Naïve Bayes, Bayes Net, C4.5, PART	77.56	77.89	FS1
Naïve Bayes, Bayes Net, C4.5, PART, MLP	75.58	80.21	FS2
Naïve Bayes, Bayes Net, C4.5, PART, MLP	75.58	76.24	FS4

Table 6
Comparison of the Proposed Model with the existing approaches.

Source	Approach	Accuracy
Proposed Model	*Majority vote with* NB, BN, RF and MP	85.48%
Paul et al. (2016)	*Neural Network with fuzzy*	80%
Verma et al.(2016)	*Decision Tree*	80.68%
Ei-bialy et al. (2015)	*Decision Tree*	78.54%
Nahar et al.(2013)	*Naïve Bayes*	69.11%

Conflicts of interest

None.

Ethical approval

None.

Acknowledgment

None.

References

[1] Ali Khazaee. Heart beat classification using particle swarm optimization. Intell. Syst. Appl. 2013:25–33.
[2] Fida Benish, Nazir Muhammad, Naveed Nawazish, Akram Sheeraz. Heart disease classification ensemble optimization using genetic algorithm. IEEE; 2011. p. 19–25.
[3] Centers for Disease Control and Prevention (CDC). Deaths: leading causes for 2008. Natl Vital Stat Rep June 6, 2012;60(No. 6).
[4] EI-Bialy R, Salamay MA, Karam OH, Khalifa ME. Feature analysis of coronary artery heart disease data sets. Procedia Comput. Sci. 2015;65:459–68.
[5] Lee HeonGyu, Noh Ki Yong, Ryu Keun Ho. Mining biosignal data: coronary artery disease diagnosis using linear and nonlinear features of HRV. LNAI 4819: emerging technologies in knowledge discovery and data mining. May 2007. p. 56–66.
[6] Singh Jagwant, Kaur Rajinder. Cardio vascular disease classification ensemble optimization using genetic algorithm and neural network. Indian J. Sci. Technol. 2016;9(S1).
[7] JyotiSoni Ujma Ansari, Sharma Dipesh. Predictive data mining for medical diagnosis: an overview of heart disease prediction‖. Int. J. Comput. Appl. March 2011;17(8). (0975 – 8887).
[8] Sudhakar K. Study of heart disease prediction using data mining. 2014;4(1):1157–60.
[9] Thenmozhi K, Deepika P. Heart disease prediction using classification with different decision tree techniques. Int J Eng Res Gen Sci 2014;2(6).
[10] KaanUyar Ahmet Ilhan. Diagnosis of heart disease using genetic algorithm based trained recurrent fuzzy neural networks. 9th international conference on theory and application of soft computing, computing with words and perception. Budapest, Hungary: ICSCCW; 2017. 24-25 Aug 2017.
[11] LathaParthiban, Subramanian R. Intelligent heart disease prediction system using CANFIS and genetic algorithm. Int. J. Biol. Biomed. Med. Sci. 2008;3(No. 3).
[12] Mackay J, Mensah G. Atlas of heart disease and stroke. Nonserial Publication; 2004.
[13] Vasighi Mahdi, Ali Zahraei, Bagheri Saeed, Vafaeimanesh Jamshid. Diagnosis of coronary heart disease based on Hnmr spectra of human blood plasma using genetic algorithm-based feature selection. Wiley Online Library; 2013. p. 318–22.
[14] Amin Mohammed Shafennor, et al. Identification of Significant features and data mining techniques in predicting heart disease. Telematics Inf 2019:82–93.
[15] Nahar J, Imam T, Tickle KS, Chen YPP. Computational intelligence for heart disease diagnosis: a medical knowledge driven approach. Expert Syst Appl 2013;40(1):96–104.
[16] Guru Niti, Dahiya Anil, NavinRajpal. Decision support system for heart disease diagnosis using neural network, Delhi Business Review. 2007;8(1). January-June.
[17] Detrano Robert. Cleveland heart disease database. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation; 1989.
[18] Patil SB, Kumaraswamy YS. Extraction of significant patterns from heart disease warehouses for heart attack prediction. Int. J. Comput. Sci. Netw. Secur(IJCSNS) 2009;9(2):228–35.
[19] Chauhan Shraddha, Aeri Bani T. The rising incidence of cardiovascular diseases in India: assessing its economic impact. J. Prev. Cardiol. 2015;4(4):735–40.
[20] Vanisree K, JyothiSingaraju. Decision support system for congenital heart disease diagnosis based on signs and symptoms using neural networks. Int J Comput Appl April 2011;19(6). (0975 8887).
[21] Verma L, Srivastava S, Negi PC. A hybrid data mining model to predict coronary artery disease cases using non-invasive clinical data. J Med Syst 2016;40(7):1–7.
[22] Liu Xiao, Wang Xiaoli, Su Qiang, Zhang Mo, Zhu Yanhong, Wang Qiugen, Wang Qian. A hybrid classification system for heart disease diagnosis based on the RFRS method. Comput. Math. Methods Med. 2017;2017:1–11.
[23] Xing Yanwei, Wang Jie, Yonghong Gao Zhihong Zhao. Combination data mining methods with new medical data to predicting outcome of Coronary Heart Disease. Convergence Information Technology. 2007. p. 868–72.