

EFFECTIVE HEART DISEASE PREDICTION USING DATA MINING TECHNIQUE

Submitted in partial fulfilment of the requirements for
the award of
Bachelor of Engineering degree in Computer Science and Engineering
by

Manendra Mazumder(38110294)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119**

MARCH 2022



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Manendra Mazumder(38110294)** who carried out project entitled "**A study on Effective Heart Disease Prediction using Data Mining Technique**" under my supervision from November 2021 to March 2022.

Internal Guide

Dr.N.Sridevi, M.E., (Ph.D.)

Head of the Department

Dr. S. VIGNESHWARI M.E., Ph.D.

DECLARATION

I **Manendra Mazumder(38110294)** hereby declare that the Project report entitled “**A study on Effective Heart Disease Prediction using Data Mining Technique**” done by us under the guidance of **Dr. N. Sridevi,M.E., (Ph.D.)** is submitted in partial fulfilment of the requirements for the award of **Bachelor of Engineering degree** in the **Department of Computer Science and Engineering**.

DATE:

PLACE: CHENNAI

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to **Boards of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. T. Sasikala M.E., Ph.D., Dean of the Department, School of Computing**, for providing us necessary support and details at the right time during the progressive reviews.

We convey our thanks to **Dr. S. Vigneshwari M.E., Ph.D., Head of the Department, Department of Computer Science & Engineering**, for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr.N. Sridevi M.E., (Ph.D.)**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science & Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The data mining is a process that is basically used to mine the data and give the result that are hidden to the internal database. The data mining is done in very formal that are basically used in medical field, engineering field and quite also in technical field. The data mining basically uses the machine learning algorithm which are predictable in nature. The heart disease prediction is basically a process which took some of the information from the user and then mine the data to predict the answer i.e, it has heart disease or not. Following are some data mining technique that are used for the prediction. These are Random Forest Decision Tree & Nave Bayes etc. from the algorithm procedure it is formed the Random Forest has the best accuracy and precision with 81% when composed to other algorithm for heart disease prediction.

Data Mining is the process of non-trivial extraction of implicit, previously unknown and potentially useful information from data. A pattern is interesting if it is valid for a given test data with some degree of certainty, novel, potentially useful and easily understood by humans. The huge amount of data generated for prediction of heart disease is too complex and voluminous to be processed and analysed by traditional methods. Advanced Data Mining tools overcome this problem by discovering hidden patterns and useful information from complex and voluminous data. Researchers reviewed literature on prediction of heart disease using data mining techniques and reported that Neural Network technique overcome all other techniques with higher levels of accuracy. Applying Data Mining techniques on healthcare data can help in predicting the likelihood of patients getting heart disease. This paper highlights the important role played by data mining tools in analysing huge volumes of healthcare related data in prediction and diagnosis of disease.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	i
	LIST OF FIGURES	v
	LIST OF TABLES	vii
	ABBREVRATIONS	viii
1.	INTRODUCTION	01
	1.1 TYPES OF ALGORITHM USED	04
	1.1.1 K-Nearest Neighbour classification	04
	1.1.2 Support Vector Machine algorithm	08
	1.1.3 Naïve Bayes algorithm	10
	1.1.4 Decision Tree algorithm	13
	1.1.5 Random Forest classification	13
	1.2 ORGANIZATION OF THE PROJECT	
	WORK	15
2.	LITERATURE SURVEY	16
3.	AIM AND SCOPE OF PRESENT	
	INVESTIGATION	18
	3.1 AIM	18
	3.2 SCOPE	18
	3.3 PROBLEM DEFINATION	19
	3.4 RELATED WORKS	19
	3.5 EXISTING SYSTEM	20
	3.5.1 Disadvantages of existing system	20

3.6 OVERVIEW OF PROPOSED SYSTEM	20
3.6.1 Advantages of proposed system	21
4. METHODOLOGY	22
4.1 HARDWARE REQUIREMENTS	22
4.2 SOFTWARE REQUIREMENTS	22
4.2.1 Overview of jupyter notebook	22
4.3 SYSTEM DESIGN	25
4.4 SYSTEM ARCHITECTURE	25
4.5 MODULE DESIGN	26
4.5.1 Data set collection	26
4.5.2 Pre-processing of data set	26
4.5.3 Extraction of data set	27
4.5.4 Prediction of results	27
4.6 ALGORITHM DESCRIPTION OF METHODS	27
4.7 PERFORMANCE MEASURES	31
4.8 HRFLM ALGORITHMS	32
4.8.1 Algorithm1 Decision tree-based partition	32
4.8.2 Algorithm2 Apply ml to find less error rate	33
4.8.3 Algorithm3 Feature extraction using less error classifier	33
4.9 EXPERIMENTAL ENVIRONMENT	34
4.9.1 Datasets	34
4.9.2 Experimental setup for evaluation	35
4.10 EVALUATION RESULTS	36
4.11 DISCUSSION OF HRFLM TO IMPROVE	

	THE RESULTS	36
5.	RESULTS AND DISCUSSION	38
	5.1 RESULTS AND DISCUSSIONS	38
	5.2 PERFORMANCE ANALYSIS	38
	5.3 BENCHMARKING OF THE PROPOSAL MODEL	39
6.	SUMMARY AND CONCLUSION	40
	6.1 CONCLUSION	40
	6.2 FUTURE ENHANCEMENT	40
	REFERENCES	41
	APPENDIX	43
	A. SAMPLE CODE	43
	B. SCREENSHOTS	47

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
Fig 1.1	Time unit for different algorithms	4
Fig 1.2	Different clusters of data on graph	5
Fig 1.3	Cluster of red circles using blue star	5
Fig 1.4	Clustering using different values of K.	6
Fig 1.5	Graph of error	7
Fig 1.6	Graph of validation error	7
Fig 1.7	Graph of different classes	8
Fig 1.8	Graph of segregating the classes.	9
Fig 1.9	Classification of different clusters	9
Fig 1.10	Graph of different classes	10
Fig 4.1	Browser view of Jupyter Notebook	23
Fig 4.2	System architecture of data prediction	25
Fig 4.3	Overall error rate of data set	35
Fig 4.4	Overall classification error rate of data set	37
Fig 5.1	Comparison between proposed and existing system	38
Fig 5.2	Performance comparison with various models	39
Fig B.1	Manual test accuracy of 86.89%	48

Fig B.2	Maximum accuracy of KNN algorithm	48
Fig B.3	Support vector Machine algorithm graph	49
Fig B.4	Decision tree algorithm graph	49
Fig B.5	Comparison between different algorithms	50
Fig B.6	Confusion Matrix for all the algorithm	50

LIST OF TABLES

TABLE NO	TABLE NAME	PAGE NO
Table 1.1	Classification of fruits	11
Table 1.2	Classification of fruits on different attributes	11
Table 4.1	UCI dataset attributes detailed information	30
Table 4.2	UCI dataset range datatype	31
Table 4.3	Classification rules for HRFLM	33
Table 4.4	Results generated based on HRFLM	34
Table 4.5	Result of various models with proposed models.	36

KEYWORDS

ABBREVRATION	EXPANSION
ML	-Machine Learning
SDK	-Software Development Kit
KNN	-K Nearest Neighbour
SVM	-Support Vector Machine
CAS	-Carotid Artery Stenting
HRFLM	-Hybrid Random Forest with Linear Model
IHDPS	-Intelligent Heart Disease Prediction System
MLP	-Multi Layer Perception
CNN	-Convolutional Neural Networks
CPU	-Central Processing Unit
RAM	-Random Access Memory
RF	-Random Forest
LM	-Linear Method

CHAPTER 1

INTRODUCTION

All person wants to live healthy life. But in the race of technique development we are compromising with our health. The basic and main part of human health is heart. For a good and longer life, one should have a good and healthy heart. The heart is responsible for blood cleaning and pumping of the blood to all other organ.

According to a survey, every year more than 10 million people were died due to heart diseases. The heart disease includes, all type of problems related to heart. There are some specific disease of the heart which are still not known and their cure is not possible. It is also said that the heart diseases are passed over the ancestors. If your father has any type of heart disease, there must be a probability of 60% that you should have the same. The heart disease is basically caused due to eating junk food, stress on mind, restless, depression and many other factors like obesity, lack of diet, family history, blood sugar problem, smoking & drinking and hypertension. The cure of heart disease is also very tough for the doctor (cardiologist) just because it is very sensitive organ of the human body. There are some common symptoms which indicate above heart attack/disease such as pain in chest, breathing problem, & palpitations of heart. Filing of heart (heart failure) is also a result of heart disease and breathing problem can occur when the heart becomes very weak to pump blood very fast.

Recently, the healthcare or health organization took data of the patients and started to build diagnosis report. The report led to the scientist that it should be predictive if the data are collected very much. The data mining technique is a technique that uses the data and basically it mine and performs the task and then predict the answer. The classification of the Forest, SVM etc. These algorithms have to basically use the data and then it has to predict the data whether it is affected by heart disease or not. Using medical data, one should take all the medical history of patient and basically predict the data on the mining technique. The machine learning algorithm is basically used to predict the answer. It is not 100% sure that the answer is right. The most accurate answer comes with Random Forest with accurate of 81%. The heart disease analysis is done based on data set that are collected. The data set contains all type of information about the patient. The stat log dataset from ULI

machine learning responsibility is utilized for making heart disease prediction in research work. The prediction of any type of heart disease can be done using the UCI machine. Various methods have been used for knowledge abstraction by using known methods of data mining for prediction of heart disease. In this work, numerous readings have been carried out to produce a prediction model using not only distinct techniques but also by relating two or more techniques. These amalgamated new techniques are commonly known as hybrid methods. We introduce neural networks using heart rate time series. This method uses various clinical records for prediction such as Left bundle branch block (LBBB), Right bundle branch block (RBBB), Atrial fibrillation (AFIB), Normal Sinus Rhythm (NSR), Sinus bradycardia (SBR), Atrial flutter (AFL), Premature Ventricular Contraction (PVC)), and Second degree block (BII) to find out the exact condition of the patient in relation to heart disease. The dataset with a radial basis function network (RBFN) is used for classification, where 70% of the data is used for training and the remaining 30% is used for classification. We propose the diagnosis of heart disease using the GA. This method uses effective association rules inferred with the GA for tournament selection, crossover and the mutation which results in the new proposed fitness function. For experimental validation, we use the well-known Cleveland dataset which is collected from a UCI machine learning repository. We will see later on how our results prove to be prominent when compared to some of the known supervised learning techniques. The most powerful evolutionary algorithm Particle Swarm Optimization (PSO) is introduced and some rules are generated for heart disease. The rules have been applied randomly with encoding techniques which result in improvement of the accuracy overall. Heart disease is predicted based on symptoms namely, pulse rate, sex, age, and many others. The ML algorithm with Neural Networks is introduced, whose results are more accurate and reliable as we have seen in network. Neural networks are generally regarded as the best tool for prediction of diseases like heart disease and brain disease. The proposed method which we use has 13 attributes for heart disease prediction. The results show an enhanced level of performance compared to the existing methods in works like [3]. The Carotid Artery Stenting (CAS) has also become a prevalent treatment mode in the medical field during these recent years. The CAS prompts the occurrence of major adverse cardiovascular events (MACE) of heart disease patients that are elderly. Their evaluation becomes very important. We generate results using a

Artificial Neural Network ANN, which produces good performance in the prediction of heart disease. Neural network methods are introduced, which combine not only posterior probabilities but also predicted values from multiple predecessor techniques. This model achieves an accuracy level of up to 89.01% which is a strong result compared to previous works. For all experiments, the Cleveland heart dataset is used with a Neural Network NN to improve the performance of heart disease as we have seen previously in. We have also seen recent developments in machine learning ML techniques used for Internet of Things (IoT) as well. ML algorithms on network traffic data has been shown to provide accurate identification of IoT devices connected to a network. Meidan et al. collected and labelled network traffic data from nine distinct IoT devices, PCs and smartphones. Using supervised learning, they trained a multi-stage meta classifier. In the first stage, the classifier can distinguish between traffic generated by IoT and non-IoT devices. In the second stage, each IoT device is associated with a specific IoT device class. Deep learning is a promising approach for extracting accurate information from raw sensor data from IoT devices deployed in complex environments. Because of its multilayer structure, deep learning is also appropriate for the edge computing environment. In this work, we introduce a technique we call the Hybrid Random Forest with Linear Model (HRFLM). The main objective of this research is to improve the performance accuracy of heart disease prediction. Many studies have been conducted that results in restrictions of feature selection for algorithmic use. In contrast, the HRFLM method uses all features without any restrictions of feature selection. Here we conduct experiments used to identify the features of a machine learning algorithm with a hybrid method. The experiment results show that our proposed hybrid method has stronger capability to predict heart disease compared to existing methods. The rest of the paper is organized as follows, Section II discusses heart related works, existing methods and techniques available. We also provide an overview of our results in Section III. Section IV discusses HRFLM Data pre-processing followed by feature selection, classification modeling and performance measure. Section V gives the algorithms used and the experimental setup. Section VI shows the evaluation of datasets and experimental setup. It also shows how the experiment was conducted and the results that were achieved. Section VII contains a discussion about the HRFLM method results and benchmarking of the proposed model. Finally, Section VIII ends with a conclusion of current work and some notes on future enhancement

1.1 TYPES OF ALGORITHM USED

For determination of disease and for prediction purposes, there are five important types of algorithms are used. These algorithms have all the type of prediction accuracy and have all the precision control. These are:

- K-Nearest Neighbour (KNN) classification.
- Support Vector Machine (SVM) algorithm.
- Naïve Bayes algorithm.
- Decision Tree algorithm.
- Random Forest classification.

1.1.1 K-Nearest Neighbour (KNN) classification

KNN can be used for both classification and regression predictive problems. However, it is more widely used in classification problems in the industry. To evaluate any technique we generally look at 2 important aspects:

1. Ease to interpret output
2. Calculation time

Let us take a few examples to place KNN in the scale:

	Logistic Regression	CART	Random Forest	KNN
1. Ease to interpret output	2	3	1	3
2. Calculation time	3	2	1	3
3. Predictive Power	2	2	3	2

Fig1.1 Time unit for different algorithms

KNN algorithm fares across all parameters of considerations. It is commonly used for its easy of interpretation and low calculation time.

Let's take a simple case to understand this algorithm. Following is a spread of red circles (RC) and green squares (GS):

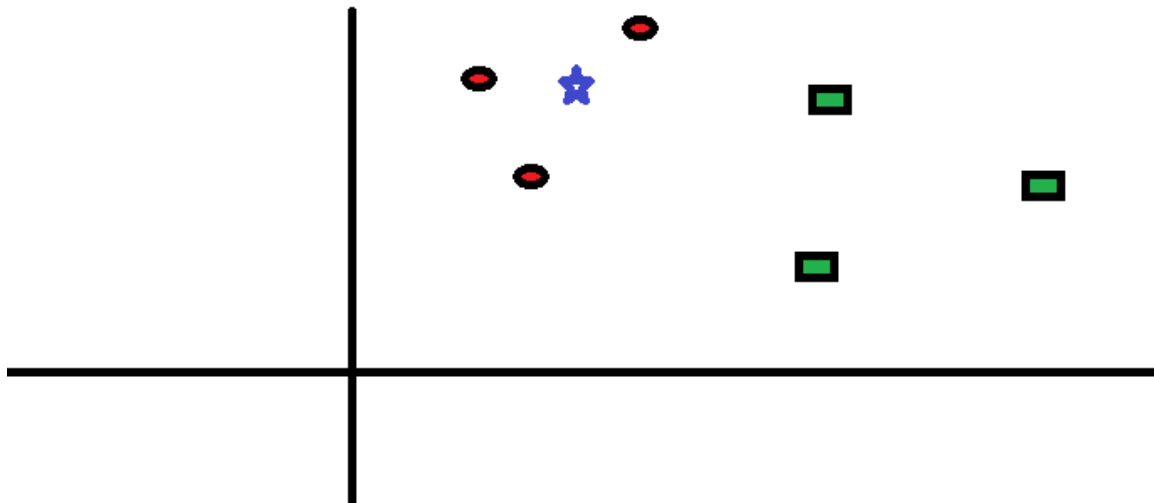


Fig1.2 Different clusters of data on graph

You intend to find out the class of the blue star (BS). BS can either be RC or GS and nothing else. The “K” in KNN algorithm is the nearest neighbours we wish to take vote from. Let's say $K = 3$. Hence, we will now make a circle with BS as centre just as big as to enclose only three datapoints on the plane. Refer to following diagram for more details:

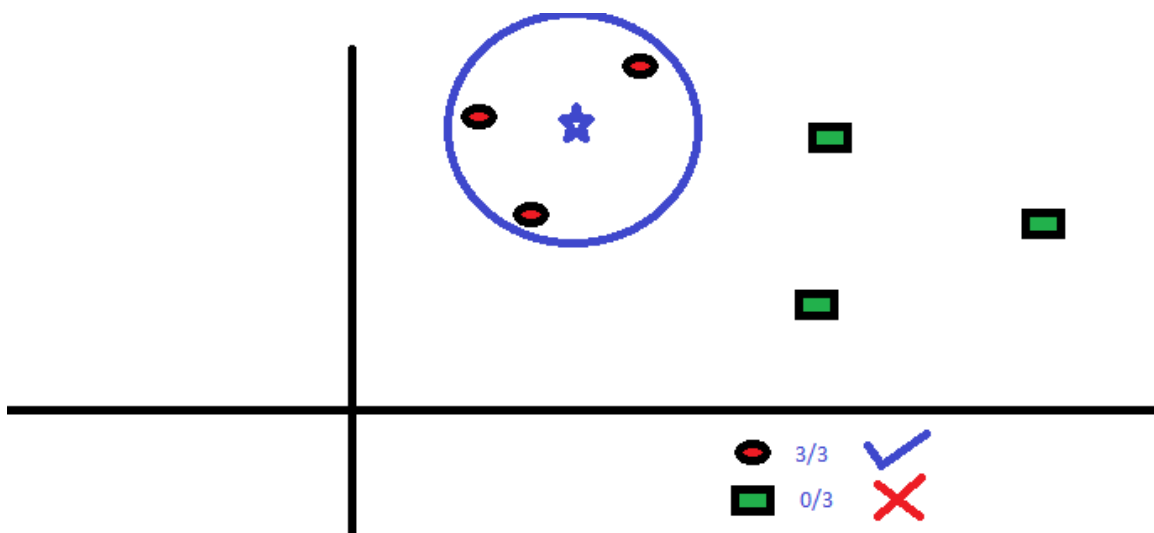


Fig1.3 Cluster of red circles using blue star

The three closest points to BS is all RC. Hence, with good confidence level we can say that the BS should belong to the class RC. Here, the choice became obvious as all three votes from the closest neighbour went to RC. The choice of the parameter K is very crucial in this algorithm. Next we will understand what are the factors to be considered to conclude the best K.

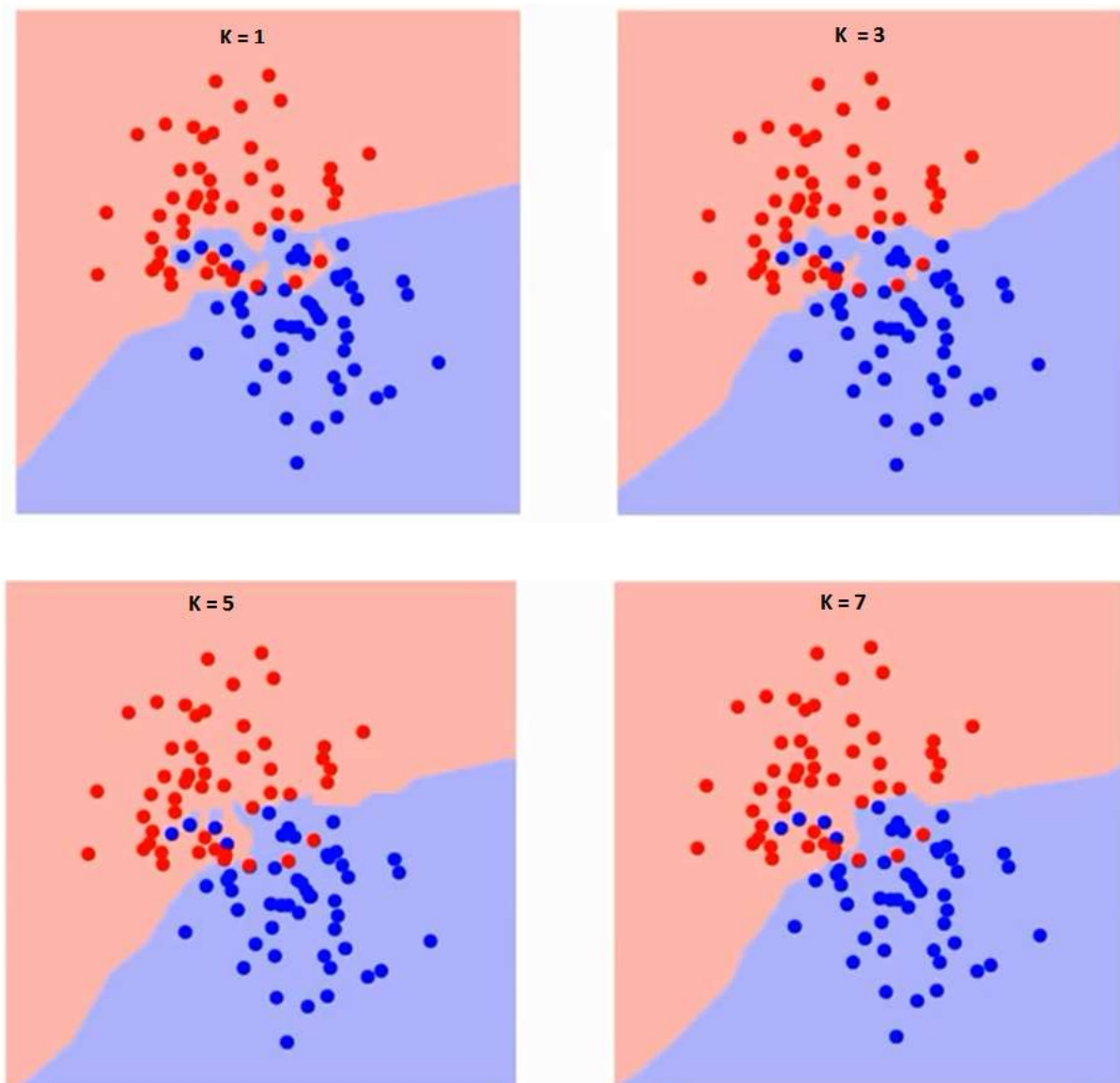


Fig1.4 Clustering using different values of K.

If you watch carefully, you can see that the boundary becomes smoother with increasing value of K. With K increasing to infinity it finally becomes all blue or all red depending on the total majority. The training error rate and the validation error rate are two parameters we need to access on different K-value. Following is the curve for the training error rate with varying value of K :

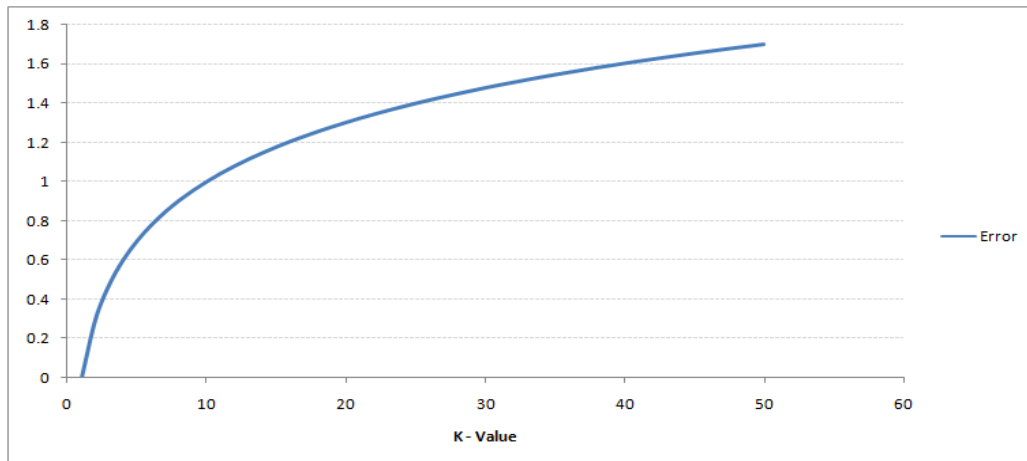


Fig1.5 Graph of error

As you can see, the error rate at K=1 is always zero for the training sample. This is because the closest point to any training data point is itself. Hence the prediction is always accurate with K=1. If validation error curve would have been similar, our choice of K would have been 1. Following is the validation error curve with varying value of K:

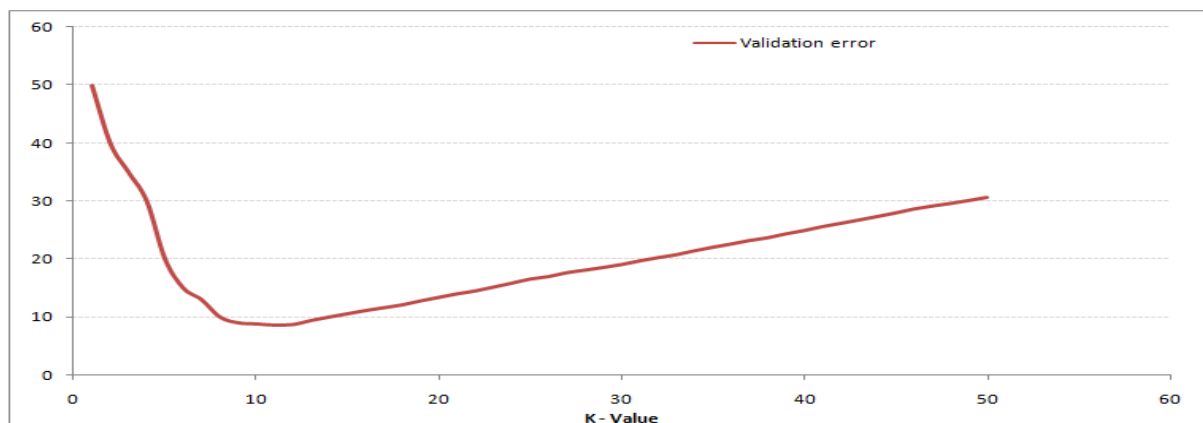


Fig1.6 Graph of validation error

This makes the story more clear. At $K=1$, we were overfitting the boundaries. Hence, error rate initially decreases and reaches a minima. After the minima point, it then increase with increasing K . To get the optimal value of K , you can segregate the training and validation from the initial dataset. Now plot the validation error curve to get the optimal value of K . This value of K should be used for all predictions.

1.1.2 Support Vector Machine (SVM) algorithm

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n -dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

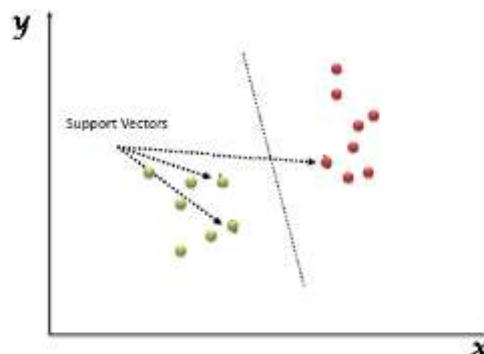


Fig1.7 Graph of different classes

There are certain points which must be followed and must be studied for understanding of support vector machine algorithm.

- Identify the right hyper-plane (Scenario-1) : Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify stars and circles.

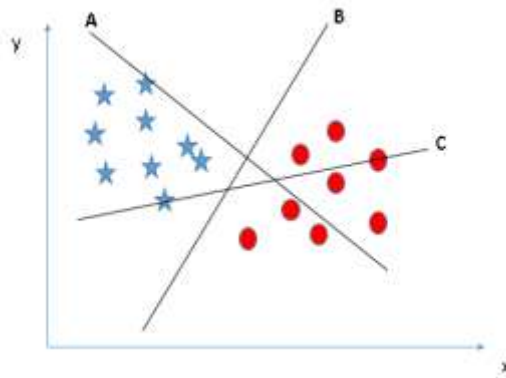


Fig1.8 Graph of segregating the classes.

- One need to remember a thumb rule to identify the right hyper-plane: “Select the hyper-plane which segregates the two classes better”. In this scenario, hyper-plane “B” has excellently performed this job.
- Identify the right hyper-plane (Scenario-2): Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, How can we identify the right hyper-plane?

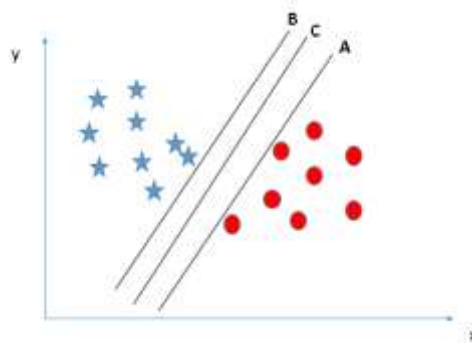


Fig1.9 Classification of different clusters

Here, maximizing the distances between nearest data point (either class) or hyper-plane will help us to decide the right hyper-plane. This distance is called as margin. Let's look at the below snapshot:

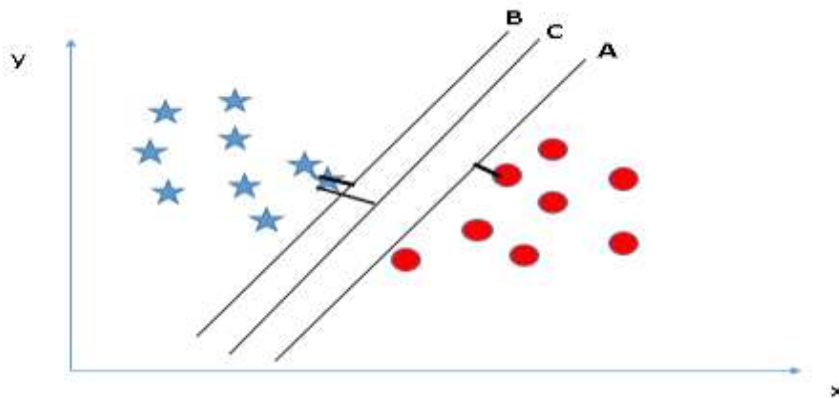


Fig1.10 Graph of different classes

Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

1.1.3 Naïve Bayes algorithm

Naive Bayes is a probabilistic machine learning algorithm that can be used in a wide variety of classification tasks. Typical applications include filtering spam, classifying documents, sentiment prediction etc. It is based on the works of Rev. Thomas Bayes (1702-61) and hence the name. Let us take an example and then study this algorithm. Say you have 1000 fruits which could be either 'banana', 'orange' or 'other'. These are the 3 possible classes of the Y variable. We have data for the following X variables, all of which are binary (1 or 0).

- Long
- Sweet
- Yellow

The first few rows of the training dataset look like this:

Table1.1 Classification of fruits

Fruit	Long (x1)	Sweet (x2)	Yellow (x3)
Orange	0	1	0
Banana	1	0	1
Other	1	1	1

For the sake of computing the probabilities, let's aggregate the training data to form a counts table like this.

Table1.2 Classification of fruits on different attributes

Type	Long	Not Long	Sweet	Not Sweet	Yellow	Not Yellow	Total
Banana	400	100	350	150	450	50	500
Orange	0	300	150	150	300	0	300
Other	100	100	150	50	50	150	200
Total	500	500	650	350	800	200	1000

So the objective of the classifier is to predict if a given fruit is a 'Banana' or 'Orange' or 'Other' when only the 3 features (long, sweet and yellow) are known. Let's say you are given a fruit that is: Long, Sweet and Yellow, can you predict what fruit it is? This is the same of predicting the Y when only the X variables in testing data are known. Let's solve it by hand using Naive Bayes. The idea is to compute the 3 probabilities, that is the probability of the fruit being a banana, orange or other. Whichever fruit type gets the highest probability wins. All the information to calculate these probabilities is present in the above tabulation.

- Compute the 'Prior' probabilities for each of the class of fruits.

That is, the proportion of each fruit class out of all the fruits from the population. You can provide the 'Priors' from prior information about the population. Otherwise, it can

be computed from the training data. For this case, let's compute from the training data. Out of 1000 records in training data, you have 500 Bananas, 300 Oranges and 200 Others. So, the respective priors are 0.5, 0.3 and 0.2.

$$P(Y=\text{Banana}) = 500 / 1000 = 0.50$$

$$P(Y=\text{Orange}) = 300 / 1000 = 0.30$$

$$P(Y=\text{Other}) = 200 / 1000 = 0.20$$

- Compute the probability of evidence that goes in the denominator.

This is nothing but the product of P of Xs for all X. This is an optional step because the denominator is the same for all the classes and so will not affect the probabilities.

$$P(x_1=\text{Long}) = 500 / 1000 = 0.50$$

$$P(x_2=\text{Sweet}) = 650 / 1000 = 0.65$$

$$P(x_3=\text{Yellow}) = 800 / 1000 = 0.80$$

- Compute the probability of likelihood of evidences that goes in the numerator.

It is the product of conditional probabilities of the 3 features. If you refer back to the formula, it says $P(X_1 | Y=k)$. Here X_1 is 'Long' and k is 'Banana'. That means the probability the fruit is 'Long' given that it is a Banana. In the above table, you have 500 Bananas. Out of that 400 is long. So, $P(\text{Long} | \text{Banana}) = 400/500 = 0.8$.

Probability of Likelihood for Banana

$$P(x_1=\text{Long} | Y=\text{Banana}) = 400 / 500 = 0.80$$

$$P(x_2=\text{Sweet} | Y=\text{Banana}) = 350 / 500 = 0.70$$

$$P(x_3=\text{Yellow} \mid Y=\text{Banana}) = 450 / 500 = 0.90$$

So, the overall probability of Likelihood of evidence for Banana = $0.8 * 0.7 * 0.9 = 0.504$

1.1.4 Decision Tree algorithm

Decision Tree algorithm belongs to the family of supervised learning algorithm. Unlike other supervised learning algorithms, decision tree algorithm can be used for solving regression and classification problems too. The general motive of using Decision Tree is to create a training model which can use to predict class or value of target variables by learning decision rules inferred from prior data (training data). The understanding level of Decision Trees algorithm is so easy compared with other classification algorithms. The decision tree algorithm tries to solve the problem, by using tree representation. Each internal node of the tree corresponds to an attribute, and each leaf node corresponds to a class label. In decision trees, for predicting a class label for a record we start from the root of the tree. We compare the values of the root attribute with record's attribute. Based on comparison, we follow the branch corresponding to that value and jump to the next node. We continue comparing our record's attribute values with other internal nodes of the tree until we reach a leaf node with predicted class value. As we know how the modelled decision tree can be used to predict the target class or the value. Now let's understanding how we can create the decision tree model.

Random forest algorithm is a supervised classification algorithm. As the name suggest, this algorithm creates the forest with a number of trees. In general, the more trees in the forest the more robust the forest looks like. In the same way in the random forest classifier, the higher the number of trees in the forest gives the high accuracy results. If you know the decision tree algorithm. You might be thinking are we creating more number of decision trees and how can we create more number of decision trees. As all the calculation of nodes selection will be same for the same dataset. Yes. You are true. To model more number of decision trees to create the forest you are not going to use the same apache of constructing the decision with information gain or gain index approach.

If you are not aware of the concepts of decision tree classifier, Please spend some time on the below articles, As you need to know how the decision tree classifier works before you learning the working nature of the random forest algorithm.

The random algorithm used in wide varieties applications. In this article, we are going address few of them. Below are some the application where random forest algorithm is widely used.

1. Banking
2. Medicine
3. Stock Market
4. E-commerce

1.Banking:

In the banking sector, random forest algorithm widely used in two main application. These are for finding the loyal customer and finding the fraud customers.

The loyal customer means not the customer who pays well, but also the customer who can take the huge amount as loan and pays the loan interest properly to the bank. As the growth of the bank purely depends on the loyal customers. The bank customers data highly analysed to find the pattern for the loyal customer based the customer details.

In the same way, there is need to identify the customer who are not profitable for the bank, like taking the loan and paying the loan interest properly or find the outlier customers. If the bank can identify theses kind of customer before giving the loan the customer. Bank will get a chance to not approve the loan to these kinds of customers. In this case, also random forest algorithm is used to identify the customers who are not profitable for the bank.

2.Medicine

In medicine field, random forest algorithm is used identify the correct combination of the components to validate the medicine. Random forest algorithm also helpful for identifying the disease by analysing the patient's medical records.

3. Stock Market

In the stock market, random forest algorithm used to identify the stock behaviour as well as the expected loss or profit by purchasing the particular stock.

4. E-commerce

In e-commerce, the random forest used only in the small segment of the recommendation engine for identifying the likely hood of customer liking the recommend products based on the similar kinds of customers.

Running random forest algorithm on very large dataset requires high-end GPU systems. If you are not having any GPU system. You can always run the machine learning models in cloud hosted desktop. You can use cloud desktop online platform to run high-end machine learning models from sitting any corner of the world.

1.2 ORGANIZATION OF THE PROJECT WORK

This section describes about various chapters included in this report.

Chapter 1 deals with an introduction to the project where the existing system is been discussed. It also gives an overview of algorithms' accuracy, precision control, data clustering and the data clustering techniques.

In Chapter 2, a detailed description of the literature survey of the papers which are referred during the project was summarized.

Chapter 3 and 4 gives a brief explanation on the aim and scope of the project and the methodology of the project. Here proposed system has been compared with the existing system. The issues in the existing system and the advantages of the proposed system are also discussed in this chapter.

In chapter 5, the Results and Discussion along with the screen layouts of each module has been depicted. It also includes the performance analysis of the system.

Chapter 6 deals with the summary and conclusion of the project. It also includes the future scope of the project.

CHAPTER 2

LITERATURE SURVEY

Tony C.T. Kuo and Arbee L.P. Chen in March 2000, proposed “Prediction of Heart Disease” using hybrid machine learning technique. The paper predicted the result by using data and the data set by regression method. The method is then implemented on the basic data set and then predicted using classification technique. The data are basically summarised and then it is gone to HRM process and accuracy control was 78%.

Sellappan Palaniappan and Rafiah Awang in August 2008, proposed “Intelligent Heart Disease Prediction” using data mining technique. The paper describes the healthcare industry collects huge amounts of healthcare data which, unfortunately, are not “mined” to discover hidden information for effective decision making. Discovery of hidden patterns and relationships often goes unexploited. Advanced data mining techniques can help remedy this situation. This research has developed a prototype Intelligent Heart Disease Prediction System (IHDPs) using data mining techniques, namely, Decision Trees, Naïve Bayes and Neural Network. Results show that each technique has its unique strength in realizing the objectives of the defined mining goals. IHDPs can answer complex “what if” queries which traditional decision support systems cannot. Using medical profiles such as age, sex, blood pressure and blood sugar it can predict the likelihood of patients getting a heart disease.

Nidhi Bhatla and Kiran Jyoti in October 2012 proposed “An Analysis of Heart Disease Prediction” using different data mining techniques. The paper describes that Heart disease is a term that assigns to a large number of medical conditions related to heart. These medical conditions describe the abnormal health conditions that directly influence the heart and all its parts. Heart disease is a major health problem in today’s time. This paper aims at analyzing the various data mining techniques

introduced in recent years for heart disease prediction. The observations reveal that Neural networks with 15 attributes has outperformed over all other data mining techniques. Another conclusion from the analysis is that decision tree has also shown good accuracy with the help of genetic algorithm and feature subset selection.

Amin ul Haq, Jiya Nazir and Ruinan Shah in September 2018, proposed “A Hybrid Intelligent System Framework for the Prediction of Heart Disease” using machine learning algorithm. This paper describes heart disease is one of the most critical human diseases in the world and affects human life very badly. In heart disease, the heart is unable to push the required amount of blood to other parts of the body. Accurate and on time diagnosis of heart disease is important for heart failure prevention and treatment. The diagnosis of heart disease through traditional medical history has been considered as not reliable in many aspects. To classify the healthy people and people with heart disease, non-invasive-based methods such as machine learning are reliable and efficient. In the proposed study, we developed a machine-learning-based diagnosis system for heart disease prediction by using heart disease dataset. We used seven popular machine learning algorithms, three feature selection algorithms, the cross-validation method, and seven classifiers performance evaluation metrics such as classification accuracy, specificity, sensitivity, Matthews’ correlation coefficient, and execution time. The proposed system can easily identify and classify people with heart disease from healthy people. Additionally, receiver optimistic curves and area under the curves for each classifier was computed.

Thus from the literature survey, the issues in the existing system has been studied which helps in improvising the proposed system. The aim and scope of the project is been discussed in the next chapter.

CHAPTER 3

AIM AND SCOPE OF PRESENT INVESTIGATION

3.1 AIM

The main aim of this project work is to predict the heart disease with accuracy and precision using data mining technique.

3.2 SCOPE

The heart disease prediction has the main ethical view using data mining technique and have the predictive algorithm based on the data. In HRFLM, we use a computational approach with the three association rules of mining namely, apriorism, predictive and Tertius to find the factors of heart disease on the UCI Cleveland dataset. The available information points to the deduction that females have less of a chance for heart disease compared to males. In heart diseases, accurate diagnosis is primary. But the traditional approaches are inadequate for accurate prediction and diagnosis. HRFLM makes use of ANN with back propagation along with 13 clinical features as the input. The obtained results are comparatively analysed against traditional methods.

The risk levels become very high and several attributes are used for accuracy in the diagnosis of the disease. The nature and complexity of heart disease require an efficacious treatment plan. Data mining methods help in remedial situations in the medical field. The data mining methods are further used considering DT, NN, SVM, and KNN. Among several employed methods, the results from SVM prove to be useful in enhancing accuracy in the prediction of disease. The nonlinear method with a module for monitoring heart function is introduced to detect the arrhythmias like bradycardia, tachycardia, atrial, atrial ventricular flutters, and many others. The performance efficacy of this method can be estimated from the accuracy in the

outcome results based on ECG data. ANN training is used for the accurate diagnosis of disease and the prediction of possible abnormalities in the patient.

3.3 PROBLEM DEFINITION

All the algorithms that are used in the different papers and used in the data prediction are well defined but mathematical calculation is not so perfect in all the papers. The mathematical calculation is nothing but a scientific approach to solve any problem and it is one of the basic rules to solve the problem. The data mining technique don't use the basic attributes and also don't uses the data accuracy control. The data mining technique in some papers have not the hybrid approach. The hybrid approach of the machine learning is best way of prediction.

3.4 RELATED WORKS

There is ample related work in the fields directly related to this paper. ANN has been introduced to produce the highest accuracy prediction in the medical field. The back propagation multilayer perception (MLP) of ANN is used to predict heart disease. The obtained results are compared with the results of existing models within the same domain and found to be improved. The data of heart disease patients collected from the UCI laboratory is used to discover patterns with NN, DT, Support Vector machines SVM, and Naive Bayes. The results are compared for performance and accuracy with these algorithms. The proposed hybrid method returns results of 86.8% for F-measure, competing with the other existing methods. The classification without segmentation of Convolutional Neural Networks (CNN) is introduced. This method considers the heart cycles with various start positions from the Electrocardiogram (ECG) signals in the training phase. CNN is able to generate features with various positions in the testing phase of the patient. A large amount of data generated by the medical industry has not been used effectively previously. The new approaches presented here decrease the cost and improve the prediction of heart disease in an easy and effective way. The various research techniques considered in this work for prediction and classification of heart disease using ML

and deep learning (DL) techniques are highly accurate in establishing the efficacy of these methods.

In the UCI data set 297 instances of patient records, in total, are considered of which 252 records are used for training and the remaining for testing. The results have been located to be satisfying based on the assessment. Heart disease prediction with SVM and ANN is proposed. In this approach, two methods are used for the premise of the accuracy and time of testing. The proposed model arranges the data records into two classes in SVM as well as ANN for further analysis as shown in. The Back Propagation Neural Network (BPNN) with classification method is introduced, where the hypertension gene sequence is generated and then, thereafter the exact gene sequence. The performance of the BPNN techniques has been measured in the training phase as well as the testing phase with the various numbers of samples. The accuracy of this technique has improved in correspondence to the number of records.

3.5 EXISTING SYSTEM

HRFLM makes use of ANN with back propagation along with 13 clinical features as the input. The obtained results are comparatively analysed against traditional methods. The risk levels become very high and several attributes are used for accuracy in the diagnosis of the disease. The nature and complexity of heart disease require an efficacious treatment plan. Data mining methods help in remedial situations in the medical field. The data mining methods are further used considering DT, NN, SVM, and KNN. Among several employed methods, the results from SVM prove to be useful in enhancing accuracy in the prediction of disease. The nonlinear method with a module for monitoring heart function is introduced to detect the arrhythmias like bradycardia, tachycardia, atrial, atrial ventricular flutters, and many others. The performance efficacy of this method can be estimated from the accuracy in the outcome results based on ECG data. ANN training is used for the accurate diagnosis of disease and the prediction of possible abnormalities in the patient.

3.5.1 Disadvantages of existing system

1. The data set don't have mathematical approach for prediction.

2. The data attributes have basic data transaction which do not have accuracy.

3. The data prediction is also not good on note of different algorithms.

3.6 OVERVIEW OF THE PROPOSED SYSTEM

In this study, we have used an R studio rattle to perform heart disease classification of the Cleveland UCI repository. It provides an easy-to-use visual representation of the dataset, working environment and building the predictive analytics. ML process starts from a pre-processing data phase followed by feature selection based on DT entropy, classification of modelling performance evaluation, and the results with improved accuracy. The feature selection and modelling keep on repeating for various combinations of attributes. It shows the UCI dataset detailed information with attributes used. It shows the data type and range of values. The performance of each model generated based on 13 features and ML techniques used for each iteration and performance are recorded. Section A summarizes the data pre-processing, Section B discusses the feature selection using entropy, Section C explains the classification with ML techniques and Section D presented for the performance of the results.

3.6.1 Advantages of the proposed system

1. The system contains five predictive algorithms that are used for the measurement of the data accuracy and precision.
2. The data prediction and the data accuracy raised up to 88.93% by random forest algorithm.
3. The mathematical approach is very clear, and the data prediction is done based on the measurement of the accuracy control.
4. The data calculation is very simple by using sigma function and sigmoid function.

Thus, the aim and scope of the project has been discussed and the issues with the existing system are also studied. The advantages in the proposed work are also mentioned. The methods and the algorithms implemented in the project are discussed in the next chapter.

CHAPTER 4

METHODOLOGY

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality.

4.1 HARDWARE REQUIREMENTS

Processor	: i4 or Later versions
Hard Disk	: 500 GB
RAM	: 1 GB

4.2 SOFTWARE REQUIREMENTS

Operating System	: Windows XP/7/8
Tool used	: Jupyter Notebook.

4.2.1 Overview of Jupyter Notebook

The Jupyter *Notebook App* is a server-client application that allows editing and running notebook documents via a web browser. The *Notebook App* can be executed on a local desktop requiring no internet access (as described in this document) or can be installed on a remote server and accessed through the internet. In addition to displaying/editing/running notebook documents, the Jupyter *Notebook App* has a “Dashboard” (Notebook Dashboard), a “control panel” showing local files and allowing to open notebook documents or shutting down their kernels. A notebook kernel is a “computational engine” that executes the code contained in a Notebook document. The ipython kernel, referenced in this guide, executes python code. Kernels for many other languages exist (official case). When you open a Notebook document, the associated *kernel* is automatically launched. When the notebook is *executed* (either cell-by-cell or with menu *Cell -> Run All*), the *kernel* performs the computation and produces the results. Depending on the type of computations, the *kernel* may consume significant CPU and RAM. Note that the RAM is not released until the kernel is shutdown. The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter. Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

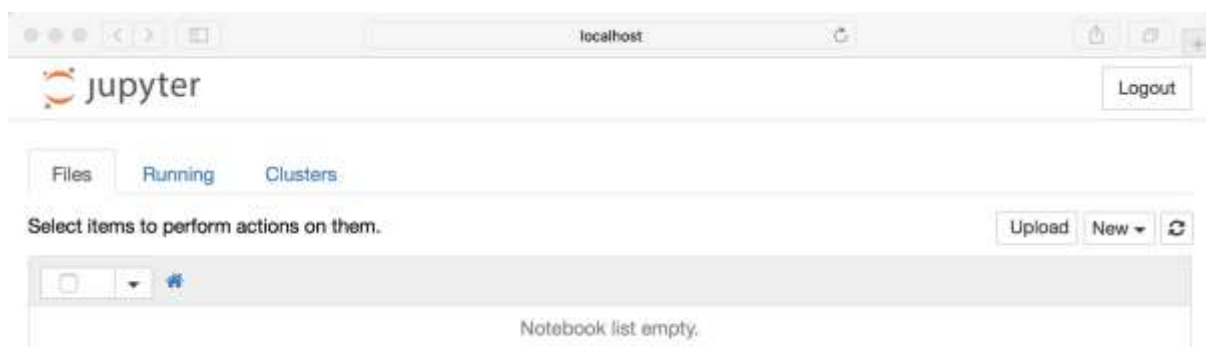


Fig4.1 Browser view of Jupyter Notebook

The Jupyter Notebook has several menus that you can use to interact with your Notebook. The menu runs along the top of the Notebook just like menus do in other applications. Here is a list of the current menus:

- File
- Edit
- View
- Insert
- Cell
- Kernel
- Widgets
- Help

Let's go over the menus one by one. This article won't go into detail for every single option in every menu, but it will focus on the items that are unique to the Notebook application.

The first menu is the File menu. In it, you can create a new Notebook or open a pre-existing one. This is also where you would go to rename a Notebook. I think the most interesting menu item is the *Save and Checkpoint* option. This allows you to create checkpoints that you can roll back to if you need to.

Next is the Edit menu. Here you can cut, copy, and paste cells. This is also where you would go if you wanted to delete, split, or merge a cell. You can reorder cells here too. Note that some of the items in this menu are greyed out. The reason for this is that they do not apply to the currently selected cell. For example, a code cell cannot have an image inserted into it, but a Markdown cell can. If you see a greyed out menu item, try changing the cell's type and see if the item becomes available to use. The *View* menu is useful for toggling the visibility of the header and toolbar. You can also toggle *Line Numbers* within cells on or off. This is also where you would go if you want to mess about with the cell's toolbar. The *Insert* menu is just for inserting cells above or below the currently selected cell.

The *Cell* menu allows you to run one cell, a group of cells, or all the cells. You can also go here to change a cell's type, although I personally find the toolbar to be

more intuitive for that. The other handy feature in this menu is the ability to clear a cell's output. If you are planning to share your Notebook with others, you will probably want to clear the output first so that the next person can run the cells themselves. The *Kernel* cell is for working with the kernel that is running in the background. Here you can restart the kernel, reconnect to it, shut it down, or even change which kernel your Notebook is using.

You probably won't be working with the Kernel all that often, but there are times when you are debugging a Notebook that you will find you need to restart the Kernel. When that happens, this is where you would go. The *Widgets* menu is for saving and clearing widget state. Widgets are basically JavaScript widgets that you can add to your cells to make dynamic content using Python (or another Kernel). Finally you have the *Help* menu, which is where you go to learn about the Notebook's keyboard shortcuts, a user interface tour, and lots of reference material.

4.3 SYSTEM DESIGN

System design is a process by which the software requirements are translated into a representation of software components, interfaces and data necessary for the implementation phase. Implementation is the final phase of the project which reveals us the real outcome of the previous steps. Here the 25 software-based tasks directly dealing with the projects target is carried out. The implementation explains about the way the project work. This includes the software support, modules and their implementation.

Design is a meaningful engineering of something that is to be built. Software Design sites at the technical kernel of software engineering. Software design is a process through which the requirements are translated into a representation of the software i.e. the blueprint for constructing software. Design provides us with representation of software that can be assesses for quality.

4.4 SYSTEM ARCHITECTURE

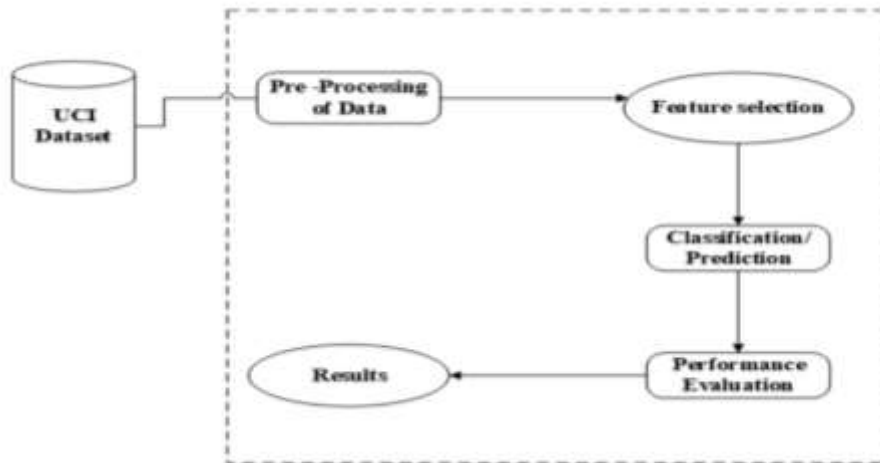


Fig4.2 System architecture of data prediction

4.5 MODULE DESIGN

The proposed system to be implemented has four modules namely

- Data set collection
- Pre-processing of data set
- Extraction of data set
- Prediction of results

4.5.1 Data set collection

Heart disease data is pre-processed after collection of various records. The dataset contains a total of 303 patient records, where 6 records are with some missing values. Those 6 records have been removed from the dataset and the remaining 297 patient records are used in pre-processing. The multiclass variable and binary classification are introduced for the attributes of the given dataset. The multi-class variable is used to check the presence or absence of heart disease. In the instance of the patient having heart disease, the value is set to 1, else the value is set to 0 indicating the absence of heart disease in the patient. The pre-processing of data is carried out by converting medical records into diagnosis values. The results of data pre-processing for 297 patient records indicate that 137 records show

the value of 1 establishing the presence of heart disease while the remaining 160 reflected the value of 0 indicating the absence of heart disease.

4.5.2 Pre-processing of data set

From among the 13 attributes of the data set, two attributes pertaining to age and sex are used to identify the personal information of the patient. The remaining 11 attributes are considered important as they contain vital clinical records. Clinical records are vital to diagnosis and learning the severity of heart disease. As previously mentioned in this experiment, several (ML) techniques are used namely, NB, GLM, LR, DL, DT, RF, GBT and SVM. The experiment was repeated with all the ML techniques using all 13 attributes. Figure 2 shows the prediction method of HRFLM.

4.5.3 Extraction of data set

The clustering of datasets is done on the basis of the variables and criteria of Decision Tree (DT) features. Then, the classifiers are applied to each clustered dataset in order to estimate its performance. The best performing models are identified from the above results based on their low rate of error. The performance is further optimized by choosing the DT cluster with a high rate of error and extraction of its corresponding classifier features. The performance of the classifier is evaluated for error optimization on this data set.

4.5.4 Prediction of results

The multiclass variable and binary classification are introduced for the attributes of the given dataset. The multi-class variable is used to check the presence or absence of heart disease. In the instance of the patient having heart disease, the value is set to 1, else the value is set to 0 indicating the absence of heart disease in the patient. The pre-processing of data is carried out by converting medical records into diagnosis values. The results of data pre-processing for 297 patient records indicate that 137 records show the value of 1 establishing the

presence of heart disease while the remaining 160 reflected the value of 0 indicating the absence of heart disease.

4.6 ALGORITHM DESCRIPTION OF METHODS

There are certain methods that are used for the prediction of the algorithms. Let us study each algorithm one by one.

1) Decision Trees

For training samples of data D, the trees are constructed based on high entropy inputs. These trees are simple and fast constructed in a top down recursive divide and conquer (DAC) approach. Tree pruning is performed to remove the irrelevant samples on D.

$$Entropy = - \sum_{j=1}^m p_{ij} \log_2 p_{ij} \quad (1)$$

2) Language Model

For given input features x_i , y_i with input vector x_i of data D the linear form of solution $f(x) = mx + b$ is solved by subsequent parameters:

$$m = \frac{(\sum_i x_i y_i) - n \bar{x} \bar{y}}{(\sum_i x_i^2) - n \bar{x}^2} \quad (2)$$

$b = \bar{y} - m \bar{x}$ where $\bar{x}, \bar{y}$ are the means.

3) Support Vector Machine

Let the training samples having dataset $Data = \{y_i, x_i\}; i = 1, 2, \dots, n$ where $x_i \in R^n$ represent the i th vector and $y_i \in R^n$ represent the target item. The linear SVM finds the optimal hyperplane of the form $f(x) = w^T x + b$ where w is a dimensional coefficient vector and b is a offset. This is done by solving the subsequent optimization problem:

$$\text{Min}_{w,b,\xi_i} \frac{1}{2} w^2 + C \sum_{i=1}^n \xi_i \quad (3)$$

4) Random Forest

This ensemble classifier builds several decision trees and incorporates them to get the best result. For tree learning, it mainly applies bootstrap aggregating or bagging. For a given data, $X = \{x_1, x_2, x_3, \dots, x_n\}$ with responses $Y = \{y_1, y_2, y_3, \dots, y_n\}$ which repeats the bagging from $b = 1$ to B . The unseen samples x_0 is made by averaging the predictions PB $b=1$ $f_b(x_0)$ from every individual trees on x_0 :

$$\hat{y} = \frac{1}{B} \sum_{b=1}^B f_b(x') \quad (4)$$

The uncertainty of prediction on these trees is made through its standard deviation,

$$\sigma = \sqrt{\frac{\sum_{b=1}^B \left(f_b(x') - \hat{y} \right)^2}{B - 1}} \quad (5)$$

6

5) Naive Bayes

This learning model applies Bayes rules through independent features. Every instance of data D is allotted to the class of highest subsequent probability. The model is trained through the Gaussian function with prior probability $P(X_f) = \text{priority} \in (0 : 1)$

$$P(X_{f1}, X_{f2}, \dots, X_{fn}|c) = \prod_{i=1}^n P(X_{fi}|c) \quad (6)$$

$$P(X_f|c_i) = \frac{P(c_i|X_f) P(X_f)}{P(c_i)} \quad c \in \{ \text{benign}, \text{malignant} \}$$

At last, the testing data is categorized based on the probability of association:

$$c_{nb} = \arg \max P(c_k) \prod_{i=1}^n P(X_{fi}|c_k), \text{ for } k = 1, 2$$

6) Neural Networks

The neuron components include inputs x_i , hidden layers and output y_i . The final result is produced through the activation function like sigmoid and a bias constant b .

$$f\left(b + \sum_{i=1}^n x_i u_i\right) \quad (7)$$

Table4.1 UCI dataset attributes detailed information

Attribute	Description	Type
Age	Patient's age in completed years	Numeric
Sex	Patient's Gender (male represented as 1 and female as 0)	Nominal
Cp	The type of Chest pain categorized into 4 values: 1. typical angina, 2. atypical angina, 3. non-anginal pain and 4. asymptomatic	Nominal
Trestbps	Level of blood pressure at resting mode (in mm/Hg at the time of admitting in the hospital)	Numeric
Chol	Serum cholesterol in mg/dl	Numeric
FBS	Blood sugar levels on fasting > 120 mg/dl; represented as 1 in case of true, and 0 in case of false	Nominal
Resting	Results of electrocardiogram while at rest are represented in 3 distinct values: Normal state is represented as Value 0, Abnormality in ST-T wave as Value 1, (which may include inversions of T-wave and/or depression or elevation of ST of > 0.05 mV) and any probability or certainty of LV hypertrophy by Estes' criteria as Value 2	Nominal
Thali	The accomplishment of the maximum rate of heart	Numeric
Exang	Angina induced by exercise. (0 depicting 'no' and 1 depicting 'yes')	Nominal
Oldpeak	Exercise-induced ST depression in comparison with the state of rest	Numeric
Slope	ST segment measured in terms of the slope during peak exercise depicted in three values: 1. unsloping, 2. flat and 3. downsloping	Nominal
Ca	Fluoroscopy coloured major vessels numbered from 0 to 3	Numeric
Thal	Status of the heart illustrated through three distinctly numbered values. Normal numbered as 3, fixed defect as 6 and reversible defect as 7.	Nominal
Num	Heart disease diagnosis represented in 5 values, with 0 indicating total absence and 1 to 4 representing the presence in different degrees.	Nominal

7) K-Nearest Neighbour

It extract the knowledge based on the samples Euclidean distance function $d(x_i, x_j)$ and the majority of k-nearest neighbours.

$$d(x_i, x_j) = \sqrt{(x_{i,1} - x_{j,1})^2 + \dots + (x_{i,m} - x_{j,m})^2} \quad (8)$$

Table4.2 UCI dataset range datatype

AGE	Numeric [29 to 77;unique=41;mean=54.4;median=56]
SEX	Numeric [0 to 1;unique=2;mean=0.68;median=1]
CP	Numeric [1 to 4;unique=4;mean=3.16;median=3]
TESTBPS	Numeric [94 to 200;unique=50;mean=131.69;median=130]
CHOL	Numeric [126 to 564;unique=152;mean=246.69;median=241]
FBS	Numeric [0 to 1;unique=2;mean=0.15;median=0]
RESTECG	Numeric [0 to 2;unique=3;mean=0.99;median=1]
THALACH	Numeric [71 to 202;unique=91;mean=149.61;median=153]
EXANG	Numeric [0 to 1;unique=2;mean=0.33;median=0.00]
OLPEAK	Numeric [0 to 6.20;unique=40;mean=1.04;median=0.80]
SLOPE	Numeric [1 to 3;unique=3;mean=1.60;median=2]
CA	Categorical [5 levels]
THAL	Categorical [4 levels]
TARGET	Numeric [0.00 to 4.00;unique=5;mean=0.94;median=0.00]

4.7 PERFORMANCE MEASURES

Several standard performance metrics such as accuracy, precision and error in classification have been considered for the computation of performance efficacy of this model. Accuracy in the current context would mean the percentage of instances correctly predicting from among all the available instances. Precision is defined as the percentage of corrective prediction in the positive class of the instances. Classification error is defined as the percentage of accuracy missing or error available in the instances. To identify the significant features of heart disease, three performance metrics are used which will help in better understanding the behavior of the various combinations of the feature-selection. ML technique focuses on the best

performing model compared to the existing models. We introduce HRFLM, which produces high accuracy and less classification error in the prediction of heart disease. The performance of every classifier is evaluated individually and all results are adequately recorded for further investigation.

4.8 HRFLM ALGORITHMS

There are certain algorithms that are used to extract the HRFLM method. Some of the algorithm are described below.

4.8.1 Algorithm1 Decision tree-based partition

Require: Input: D dataset – features with a target class

For $\forall$ features do

For Each sample do

Execute the Decision Tree algorithm

End For

Identify the feature space $f_1, f_2, \dots, f_x$ of dataset UCI. (9)

End For

Obtain the total number of leaf nodes $l_1, l_2, l_3, \dots, l_n$ with its constraints (10)

Split the dataset D into $d_1, d_2, d_3, \dots, d_n$ based on the leaf nodes constraints. (11)

Output: Partition datasets $d_1, d_2, d_3, \dots, d_n$

4.8.2 Algorithm2 Apply ML to find less error rate

Require: Input: Datasets with partition – d1, d2, d3,..., dn

For $\forall$ apply the rules do

On the dataset $R(d1, d2, d3, \dots, dn)$

End For

Classify the dataset based on the rules $C(R(d1), R(d2), \dots, R(dn))$ (12)

Output: Classified datasets with rules $C(R(d1), R(d2), \dots, R(dn))$

4.8.3 Algorithm3 Feature Extraction using less error Classifier

Require: Input: Classified datasets $C(R(d1), R(d2), \dots, R(dn))$

For $\forall$ Find out min error rate from the input do

$\text{Min}(C(R(d1), R(d2), \dots, R(dn)))$ (13)

End For

Find out max(min) error rate from the classifier.

Output: Features with classified attributes $F(d1, d2, d3, \dots, dn)$

Table4.3 Classification rules for HRFLM

Rule No	Attributes	Values	Range
1.	root	212 97 0	(0.54 0.18 0.13 0.12 0.033)
2.	CA=0	123 29 0	(0.76 0.15 0.049 0.024 0.0081)
3.	CA=1,2,3	89 67 3	(0.24 0.21 0.24 0.25 0.067)
4.	EXANG	< 0.5 97 11 0	(0.89 0.1 0 0.01 0)
5.	EXANG	>=0.5 26 17 1	(0.31 0.35 0.23 0.077 0.038)
6.	SEX	< 0.5 24 11 0	(0.54 0 0.17 0.25 0.042)
7.	SEX	>=0.5 65 46 1	(0.12 0.29 0.26 0.25 0.077)
8.	THAL	=3 11 3 0	(0.73 0.18 0.091 0 0)
9.	THAL	=3,6,7 15 8 1	(0 0.47 0.33 0.13 0.067)
10.	THAL	=3 15 2 0	(0.87 0 0.067 0.067 0)
11.	THAL	=6,7 9 4 3	(0 0 0.33 0.56 0.11)
12.	THALACH	>=139.5 33 20 1	(0.15 0.39 0.33 0.061 0.061)
13.	RESTECG	< 1 17 10 2	(0.29 0.29 0.41 0 0)
14.	RESTECG	>=1 16 8 1	(0 0.5 0.25 0.12 0.12)
15.	THALACH	< 139.5 32 18 3	(0.094 0.19 0.19 0.44 0.094)

Table4.4 Results generated based on HRFLM

Data Split	Overall error rate		Best Model	Overall classification error rate		Best Model
	RF	LM		RF	LM	
1	4	6.7	RF	14.9	16.2	DT /RF
2	12.2	22.6	RF	37.7	38.7	RF
3	11.1	16.6	RF	27.8	50	RF
4	20.9	29.2	RF	54.1	54.2	RF
5	13.3	13.3	RF/ LM	53.4	53.3	LM
6	12	18.1	RF	57.6	54.6	LM
7	0	28.5	RF	28.5	42.8	RF
8	18.2	9.1	LM	27.3	27.3	RF/ LM

4.9 EXPERIMENTATIONAL ENVIRONMENT

4.9.1 Datasets

Heart disease data was collected from the UCI machine learning repository. There are four databases (i.e. Cleveland, Hungary, Switzerland, and the VA Long Beach). The Cleveland database was selected for this research because it is a commonly used database for ML researchers with comprehensive and complete records. The dataset contains 303 records. Although the Cleveland dataset has 76 attributes, the dataset provided in the repository furnishes information for a subset of only 14 attributes. The data source of the Cleveland dataset is the Cleveland Clinic Foundation. Table 1 depicts the description and type of attributes. There are 13 attributes that feature in the prediction of heart disease, where only one attribute serves as the output or the predicted attribute to the presence of heart disease in a patient.

The Cleveland dataset contains an attribute named num to show the diagnosis of heart disease in patients on different scales, from 0 to 4. In this scenario, 0 represents the absence of heart disease and all the values from 1 to 4 represent patients with heart disease, where the scaling refers to the severity of the disease.

4.9.2 Experimental setup for evaluation

We have used an R studio rattle to perform the classification of heart disease from Cleveland UCI repository. Figure 1 depicts the evaluation of the experiment by step-by-step stages. In the first step, the UCI dataset is loaded and the data becomes ready for pre-processing. The subset of 13 attributes (Age, sex, cp, treetops, chol, FBS, restecg, thalach, exang, olpeak, slope, ca, that, target) is selected from the pre-processed data set of heart disease. The three existing models for heart disease prediction (DT, RM, LM) are used to develop the classification. The evaluation of the model is performed with the confusion matrix. Totally, four outcomes are generated by confusion matrix, namely TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative). The following measures are used for the calculation of the accuracy, sensitivity, specificity.

4.10 EVALUATION RESULTS

The prediction models are developed using 13 features and the accuracy is calculated for modeling techniques. The best classification methods are given below in Table 3. This table compares the accuracy, classification error, precision, F-measure, sensitivity and specificity. The highest accuracy is achieved by HRFLM classification method in comparison with existing methods.

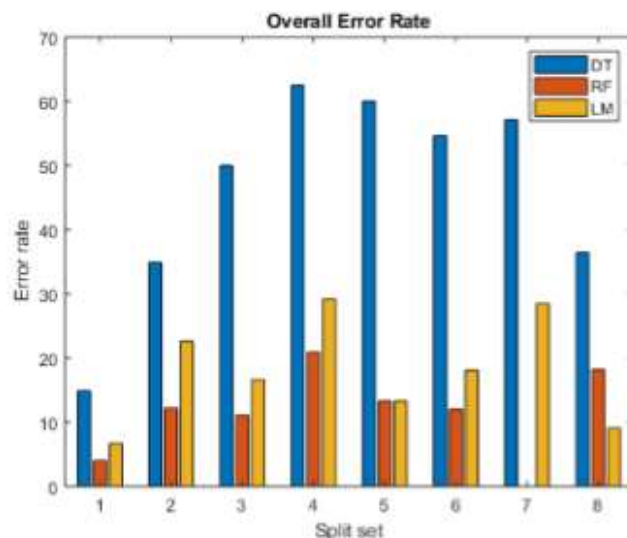


Fig4.3 Overall error rate of data set

4.11 DISCUSSION OF HRFLM TO IMPROVE THE RESULTS

The UCI dataset is further classified into 8 types of datasets based on classification rules. The classification rules are listed in Table 4.3. Each dataset is further classified and processed by R Studio Rattle. The results are generated by applying the classification rule for the dataset. The classification rules generated based on the rule after data pre-processing is done. After pre-processing, the data's three best ML techniques are chosen, and the results are generated. The various datasets with DT, RF, LM are applied to find out the best classification method. Table 4.4 shows that results of existing and proposed methods.

Table 4.5 Result of various models with proposed models.

Data Split	Overall error rate			Best Model	Overall classification error rate			Best Model
	DT	RF	LM		DT	RF	LM	
1	14.9	4	6.7	RF	14.9	14.9	16.2	DT /RF
2	34.9	12.2	22.6	RF	39.6	37.7	38.7	RF
3	50	11.1	16.6	RF	50	27.8	50	RF
4	62.5	20.9	29.2	RF	62.5	54.1	54.2	RF
5	60	13.3	13.3	RF/ LM	60	53.4	53.3	LM
6	54.6	12	18.1	RF	60.6	57.6	54.6	LM
7	57.1	0	28.5	RF	57.1	28.5	42.8	RF
8	36.4	18.2	9.1	LM	36.4	27.3	27.3	RF/ LM

The results show that RF and LM are the best. The RF error rate for dataset 4 is high (20.9%) compared to the other datasets. The LM method for the dataset is the best (9.1%) compared to DT and RF methods. We combine the RF method with LM and propose HRFLM method to improve the results. Table 6 show the results of the proposed method. Figure 3 shows the overall error rate of the dataset. Figure 4.4 shows the overall classification error rate of the dataset.

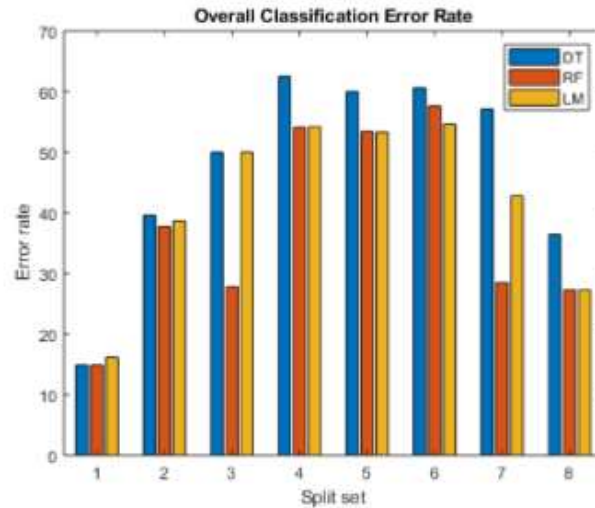


Fig4.4 Overall classification error rate of data set

Benchmarking is needed to compare the performance of the existing models compared with the proposed model. This method is used to identify whether the proposed method is the best and improves accuracy or not. The accuracy is calculated with the number of feature selection and the model generated results. HRFLM has no restriction in selecting of features to use. All the features selected in this model accomplish the best results. Table 7 shows that comparison of various models with our proposed method. Figure 5 and Figure 6 shows the performance comparison of the various model with respect to proposed method respectively.

Table 4.4 depicts the details of features selected by various models from the UCI dataset for heart disease. The proposed method is used on all 13 attributes and classified, based on the error rate. This result clearly proves that all the features selected and ML techniques used, prove effective in accurately predicting heart disease of patients compared with known existing models.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 RESULTS AND DISCUSSION

In this section, we find that the UCI dataset is further classified into 8 types of datasets based on classification rules. The classification rules are listed in Table 4. Each dataset is further classified and processed by R Studio Rattle. The results are generated by applying the classification rule for the dataset. The classification rules generated based on the rule after data pre-processing is done. After pre-processing, the data's three best ML techniques are chosen, and the results are generated. The various datasets with DT, RF, LM are applied to find out the best classification method. Table 4.5 shows that results of existing and proposed methods. The results show that RF and LM are the best. The RF error rate for dataset 4 is high (20.9%) compared to the other datasets. The LM method for the dataset is the best (9.1%) compared to DT and RF methods. We combine the RF method with LM and propose HRFLM method to improve the results.

5.2 PERFORMANCE ANALYSIS

The proposed system when compared to that of the existing system is efficient in terms of accuracy, speed and reliability of matching

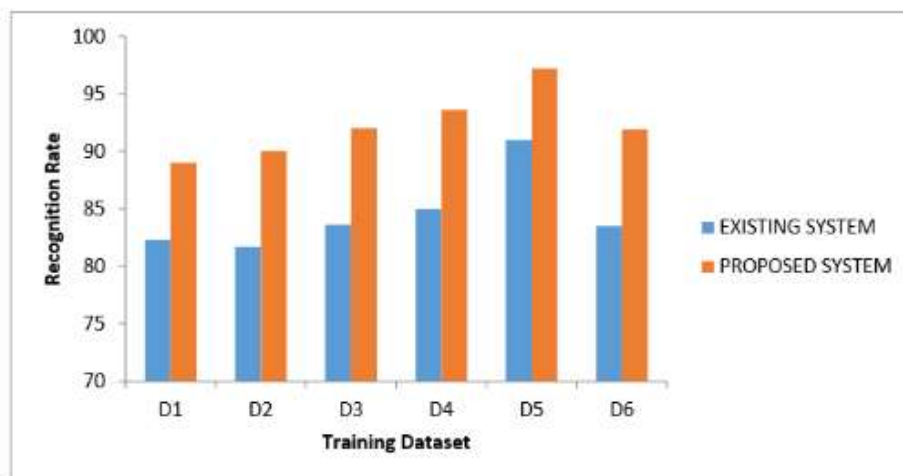


Fig:5.1 :Comparison between Proposed and Existing System

In the existing system Linear regression algorithm is used which is slow in processing and computation and is affected by variation to noisy data and fussy data whereas Random forest algorithm is not variant to noisy, unorthodox and fussy and fast in predicting results compared to other prediction matching algorithms.

5.3 BENCHMARKING OF THE PROPOSED MODEL

Benchmarking is needed to compare the performance of the existing models compared with the proposed model. This method is used to identify whether the proposed method is the best and improves accuracy or not. The accuracy is calculated with the number of feature selection and the model generated results. HRFLM has no restriction in selecting of features to use. All the features selected in this model accomplish the best results. Table 5.1 shows that comparison of various models with our proposed method. Figure 4.5 and Figure 4.6 shows the performance comparison of the various model with respect to proposed method respectively.

Table 5.1 depicts the details of features selected by various models from the UCI dataset for heart disease. The proposed method is used on all 13 attributes and classified, based on the error rate. This result clearly proves that all the features selected, and ML techniques used, prove effective in accurately predicting heart disease of patients compared with known existing models.

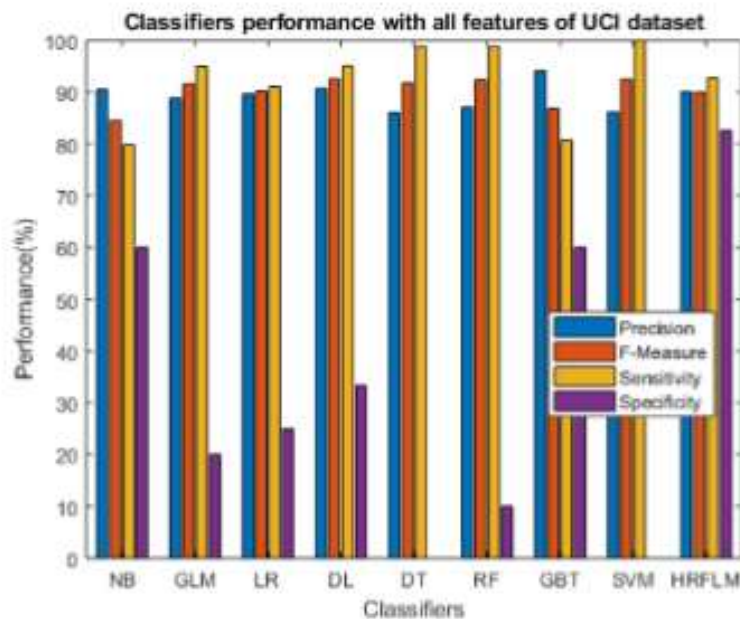


Fig5.2 Performance comparison with various models

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 CONCLUSION

Identifying the processing of raw healthcare data of heart information will help in the long term saving of human lives and early detection of abnormalities in heart conditions. Machine learning techniques were used in this work to process raw data and provide a new and novel discernment towards heart disease. Heart disease prediction is challenging and very important in the medical field. However, the mortality rate can be drastically controlled if the disease is detected at the early stages and preventative measures are adopted as soon as possible. Further extension of this study is highly desirable to direct the investigations to real-world datasets instead of just theoretical approaches and simulations. The proposed hybrid HRFLM approach is used combining the characteristics of Random Forest (RF) and Linear Method (LM). HRFLM proved to be quite accurate in the prediction of heart disease. The future course of this research can be performed with diverse mixtures of machine learning techniques to better prediction techniques. Furthermore, new feature selection methods can be developed to get a broader perception of the significant features to increase the performance of heart disease prediction.

6.2 FUTURE ENHANCEMENT

The model is supposed to be enhanced with accuracy rate and precision over different algorithm which are used to predict the algorithm. The algorithms that are used have 89.93% accuracy by random forest algorithm, but all different algorithms have precision up to 78% on an average. The data prediction is basically have all the data set present that are well defined and well placed according to prediction average.

REFERENCES

- [1] Abdullah, A.S., 2012. A Data mining Model for predicting the Coronary Heart Disease using Random Forest Classifier., (Icon3c), pp.22–25.
- [2] Alkeshuosh, A.H., Moghadam, M.Z. Mansoori, I. Al, 2017. Diagnosis of Heart Disease., pp.306–311.
- [3] Al-milli, N., 2013. Backpropagation Neural Network for Prediction of Heart Disease., 56(1), pp.131–135.
- [4] A.Devi, S.Rajamhoana, C.K.Umamaheswari and R.Kiruba Analysis of Neural Networks Based Heart Disease Prediction System, 2018 11th International Conference on Human System Interaction (HSI), Gdansk, 2018, pp. 233-239.
- [5] Anooj, P.K., 2012. Clinical decision support system: Risk level prediction of heart disease using weighted fuzzy rules. Journal of King Saud University - Computer and Information Sciences, 24(1), pp.27–40.
- [6] Baccour, L., 2018. Amende d fuse d TOPSIS-VIKOR for classification (ATOVIC) applied to some UCI data sets R. Expert Systems With Applications, 99, pp.115–125..
- [7] Cheng, C. Chiu, H., 2017. An Artificial Neural Network Model for the Evaluation of Carotid Artery Stenting Prognosis Using a National Wide Database. 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp.2566–2569.
- [8] H.A. Esfahaniand, M. Ghazanfari,"Cardiovascular disease detection using a new ensemble classifier," 2017 IEEE 4th International Conference on Knowledge-Based Engineering and Innovation (KBEI), Tehran, 2017, pp. 1011-1014.
- [9] Dammak, F., Baccour, L. Alimi, A.M., The Impact of Criterion Weights Techniques in TOPSIS Method of Multi-Criteria Decision Making in Crisp and Intuitionistic Fuzzy Domains. 2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), (9), pp.1–8. s

- [10] Das, R., Turkoglu, I. Sengur, A., 2009. Expert Systems with Applications Effective diagnosis of heart disease through neural networks ensembles. Expert Systems with Applications, 36(4), pp.7675–7680.
- [11] Durairaj, M. Revathi, V., 2015. Prediction of Heart Disease Using Back Propagation MLP Algorithm., 4(08), pp.235–239.
- [12] Gandhi, M., 2015. Predictions in Heart Disease Using Techniques of Data Mining. 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), pp.520–525.
- [13] Gavhane, A., 2018. Prediction of Heart Disease Using Machine Learning. 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), (Iceca), pp.1275–1278.
- [14] Jpdlo, V. et al., 2018. Heart diseases prediction with Data Mining and Neural Network Techniques., 6(7 2), pp.1–6.
- [15] K, S.B.N., 2016. Prediction of Heart Disease at early stage using Data Mining and Big Data Analytics: A Survey., pp.256–261.
- [16] Kelwade, J.P., 2016. Radial basis function Neural Network for Prediction of Cardiac Arrhythmias based on Heart rate time series., (Cmi), pp.454–458.
- [17] Krishnaiah, V. Chandra, N.S., 2016. Heart Disease Prediction System using Data Mining Techniques and Intelligent Fuzzy Approach: A Review., 136(2), pp.43–51.
- [18] Kumar, P.S. et al., 2016. A Computational Intelligence Method for Effective Diagnosis of Heart Disease using Genetic Algorithm., 8(2), pp.363–372.
- [19] Liberatore, M.J. Nydick, R.L., 2008. The analytic hierarchy process in medical and health care decision making: A literature review., 189, pp.194–207.
- [20] Mahboob, T., Irfan, R. Ghaffar, B., 2017. Evaluating Ensemble Prediction of Coronary Heart Disease using Receiver Operating Characteristics., pp.110–115.

APPENDIX

A. SAMPLE CODE

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
import os
print(os.listdir("../input"))
sns.countplot(x="target", data=df, palette="bwr")
plt.show()
countNoDisease = len(df[df.target == 0])
countHaveDisease = len(df[df.target == 1])
print("Percentage of Patients Haven't Heart Disease: {:.2f}%".format((countNoDisease / (len(df.target))*100)))
print("Percentage of Patients Have Heart Disease: {:.2f}%".format((countHaveDisease / (len(df.target))*100)))
sns.countplot(x='sex', data=df, palette="mako_r")
plt.xlabel("Sex (0 = female, 1= male)")
plt.show()
sns.countplot(x='sex', data=df, palette="mako_r")
plt.xlabel("Sex (0 = female, 1= male)")
plt.show()
def update(weight,bias,x_train,y_train,learningRate,iteration) :
    costList = []
    index = []

    #for each iteration, update weight and bias values
    for i in range(iteration):
```

```

cost,gradients = forwardBackward(weight,bias,x_train,y_train)
weight = weight - learningRate * gradients["Derivative Weight"]
bias = bias - learningRate * gradients["Derivative Bias"]

costList.append(cost)
index.append(i)

parameters = {"weight": weight,"bias": bias}

print("iteration:",iteration)
print("cost:",cost)

plt.plot(index,costList)
plt.xlabel("Number of Iteration")
plt.ylabel("Cost")
plt.show()
for i in range(1,20):
    knn2 = KNeighborsClassifier(n_neighbors = i) # n_neighbors means k
    knn2.fit(x_train.T, y_train.T)
    scoreList.append(knn2.score(x_test.T, y_test.T))

plt.plot(range(1,20), scoreList)
plt.xticks(np.arange(1,20,1))
plt.xlabel("K value")
plt.ylabel("Score")
plt.show()

acc = max(scoreList)*100
accuracies['KNN'] = acc
print("Maximum KNN Score is {:.2f}%".format(acc))
from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()
nb.fit(x_train.T, y_train.T)

```

```

acc = nb.score(x_test.T,y_test.T)*100
accuracies['Naive Bayes'] = acc
print("Accuracy of Naive Bayes: {:.2f}%".format(acc))

```

```

from sklearn.tree import DecisionTreeClassifier
dtc = DecisionTreeClassifier()
dtc.fit(x_train.T, y_train.T)

acc = dtc.score(x_test.T, y_test.T)*100
accuracies['Decision Tree'] = acc
print("Decision Tree Test Accuracy {:.2f}%".format(acc))

from sklearn.ensemble import RandomForestClassifier
rf = RandomForestClassifier(n_estimators = 1000, random_state = 1)
rf.fit(x_train.T, y_train.T)

acc = rf.score(x_test.T,y_test.T)*100
accuracies['Random Forest'] = acc
print("Random Forest Algorithm Accuracy Score : {:.2f}%".format(acc))

```

Predicted values

```

y_head_lr = lr.predict(x_test.T)
knn3 = KNeighborsClassifier(n_neighbors = 3)
knn3.fit(x_train.T, y_train.T)
y_head_knn = knn3.predict(x_test.T)
y_head_svm = svm.predict(x_test.T)
y_head_nb = nb.predict(x_test.T)
y_head_dtc = dtc.predict(x_test.T)
y_head_rf = rf.predict(x_test.T)

```

In [40]:

```

from sklearn.metrics import confusion_matrix

cm_lr = confusion_matrix(y_test,y_head_lr)
cm_knn = confusion_matrix(y_test,y_head_knn)
cm_svm = confusion_matrix(y_test,y_head_svm)

```

```

cm_nb = confusion_matrix(y_test,y_head_nb)
cm_dtc = confusion_matrix(y_test,y_head_dtc)
cm_rf = confusion_matrix(y_test,y_head_rf)

plt.suptitle("Confusion Matrixes",fontsize=24)
plt.subplots_adjust(wspace = 0.4, hspace= 0.4)

plt.subplot(2,3,1)
plt.title("Logistic Regression Confusion Matrix")
sns.heatmap(cm_lr,annot=True,cmap="Blues",fmt="d",cbar=False, annot_kws={"size": 24})

plt.subplot(2,3,2)
plt.title("K Nearest Neighbors Confusion Matrix")
sns.heatmap(cm_knn,annot=True,cmap="Blues",fmt="d",cbar=False, annot_kws={"size": 24})

plt.subplot(2,3,3)
plt.title("Support Vector Machine Confusion Matrix")
sns.heatmap(cm_svm,annot=True,cmap="Blues",fmt="d",cbar=False, annot_kws={"size": 24})

plt.subplot(2,3,4)
plt.title("Naive Bayes Confusion Matrix")
sns.heatmap(cm_nb,annot=True,cmap="Blues",fmt="d",cbar=False, annot_kws={"size": 24})

plt.subplot(2,3,5)
plt.title("Decision Tree Classifier Confusion Matrix")
sns.heatmap(cm_dtc,annot=True,cmap="Blues",fmt="d",cbar=False, annot_kws={"size": 24})

plt.subplot(2,3,6)

```

```
plt.title("Random Forest Confusion Matrix")
```