ABSTRACT

31% of the deaths worldwide each year occur due to Cardiovascular Diseases (CVDs). There are various medical instruments available in the market for predicting heart diseases but there are two major problems in them:

1. They are very expensive.
2. They are not efficiently able to calculate the chance of heart disease.

Machine learning could be a better choice to achieve high accuracy for predicting heart diseases. The main objective of this project is to build a model using machine learning that can predict whether a person is suffering from Heart Disease or not based on a combination of risk factors (parameters) describing the disease. For better accuracy 4 algorithms were analyzed namely:

- Support Vector Machine (SVM)
- Decision Tree (DT)
- K-Nearest Neighbor Algorithm (KNN)
- Random Forest Classifier

The algorithms were judged based on their accuracy and it was observed that the K-Nearest Neighbor Algorithm (KNN) was the most accurate out of the four with 87.0% efficiency. Hence, it was selected for implementation of the main application. The main application is a user-interactive web application that accepts the various parameters from the user as input and computes the result.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CVDs: Cardiovascular Diseases

ANN: Artificial Neural Networks

ML: Machine Learning

SRS: System Requirement Specification

SVM: Support Vector Machine

DT: Decision Tree

KNN: K-Nearest Neighbor Algorithm

RFC: Random Forest Classifier

HTML: Hyper Text Markup Language

CSS: Cascading Style Sheets

JS: JavaScript

CHAPTER 1

INTRODUCTION

1.1 Background and Basics

Heart is one of the most vital organs for the proper functioning of our body. According to a survey by WHO, 31% of the worldwide deaths every year occurs due to Cardiovascular Diseases (CVDs). Also, more than 75% of these deaths occur in low and middle income countries including India. The main challenge is to accurately predict the existence of CVDs inside human body. The older techniques have not been very successful in efficiently predicting the heart diseases. Many medical instruments are available in the market for the prediction of heart diseases but there are some drawbacks of these instruments like they are very costly, they are not efficient enough for predicting heart diseases[10].

Age, Sex, Blood Pressure, Cholesterol, Blood Sugar, Diabetes, etc. and some lifestyle factors like obesity, eating unhealthy food, less physical activity, smoking, consumption of alcohol, etc. are some of the major risk factors that leads to heart diseases. Most of the lifestyle risk factors are controllable. In the last few decades, medical science has used the technological advancements very well to improve the quality of healthcare. These advancements in technology have paved ways for accurate diagnosis and prediction of diseases. Machine learning could be a very good choice to achieve high accuracy for predicting heart diseases as it is able to analyse large amounts of data and identifying patterns & trends. Moreover, machine learning provides much faster and reliable results. There are other soft computing approaches as well such as Artificial Neural Networks (ANN), Deep Learning, Fuzzy Logic, Data Mining, Genetic Algorithm, etc. that can put into effect for predicting heart diseases.

1.2 Objective

The main objective of this project is to build a web application for the prediction of heart disease using Machine Learning. After analysis and comparison of various ML algorithms, the one having highest accuracy will be implemented for prediction purpose.

CHAPTER 2

LITERATURE REVIEW

This chapter provides a literature review mainly based on the publications available in IEEE explore and Science Direct. Although we have tried our best to include all the papers but it might be possible that some important studies have been skipped. Table I provides a brief description of all the papers and Table II provides the technological detail adopted in the papers reviewed. Total 53 papers have been reviewed.

TABLE 1: Brief Description Of The Papers

S.No.	Year	Ref.No.	DESCRIPTION
1	2019	[44]	This work was done to increase the accuracy of weak classifiers by implementing the ensemble techniques like bagging, boosting on the medical dataset which increased the results of these classifiers by 7% as compared to previous results.
2		[43]	They compared the different ML algorithms like Decision Trees, Logistic regression, Random Forest, Neural Networks and Support Vector Machine on the data taken from the dataset of Framingham Heart Study. Different algorithms gave different results when applied on different models that were used in the study.
3		[41]	This paper uses machine learning to predict heart disease. The effective accuracy achieved was 88.7%.
4		[40]	Two SVM models were used to predict heart failures. The experimental results show that the proposed model improves the accuracy as compared to conventional SVM models by 3.3%. The accuracy varied from 57.85% to 91.83%.
5		[27]	Decision Tree and Naïve Bayes is used to process the datasets in python and Naïve Bayes came out to be more accurate.
6		[13]	It proposes a HDPS (Heart Disease Prediction System) using machine learning to predict heart disease and label it into 5 classes: very high, high, medium, low, no.
7		[5]	They proposed the model for prediction of heart diseases on the basis of various parameters like age, sex etc. by using ML.
8		[45]	This paper uses machine learning for the prediction of cardiac arrest. A systematic review was done on cardiac arrest prediction using the Pubmed, ScienceDirect, Google Scholar and SpringerLink databases.
9		[39]	This paper uses a cloud-based 4- tier architecture to improve the accuracy of predicting heart diseases. They used 5 most popular supervised learning approach based on machine learning. ANN was found to give the best results.

10	2018	[38]	This approach proposed a hybrid machine learning algorithm which used Random Forest classifier and simple k-means algorithm and then J48 tree classifier and Naïve Bayes classifiers were used to compare the final results.
11		[37]	This paper uses machine learning for the prediction of cardiac disorders. Support Vector Machine (SVM) algorithm is used for this purpose. The accuracy achieved is 98.30%.
12		[36]	Different data mining algorithms like Decision tree, Naïve Bayes and KNN were used by taking different parameters in consideration.
13		[6]	This paper proposes a modified calculation for classification with decision trees so that the outcome is more precise. The accuracy achieved from the proposed model is 79.8%.
14		[35]	Two algorithms named Tracking-Ancestors algorithm and Find-Specified-Ancestor algorithm were developed in this study for the management of deep learning models.
15		[18]	The medical data is increasing at a tremendous rate. Cloud is being used to handle such a huge data. This paper proposes a Cloud and IoT based system that is capable of predicting various diseases. For prediction datasets from UCI repository were used.
16	2017	[53]	J48 algorithm, Random Forest algorithm and Logistic model tree algorithm were tested for the prediction of heart disease. The Cleveland dataset was used for this purpose having 76 attributes. The highest accuracy of 56.76% was achieved by J48 algorithm.
17		[46]	This paper was the first study which was conducted on taking account the all parameters and aspects of management of heart failures based on machine learning techniques.
18		[34]	The study of different data mining classification schemes by employing ensemble machine learning techniques like bagging, boosting and stacking to the attributes of Cleveland dataset was performed.
19		[10]	This paper proposes machine learning and deep leaning algorithms for accurately predicting heart diseases.
20		[22]	A hybrid model of OFBAT-RBFL was used for the prediction of heart disease. This experiment used UCI datasets and an accuracy of 78% was achieved.
21		[42]	Methods such as decision trees, Naive Bayes, Support Vector Machine and Apriori algorithm are used for prediction of heart disease. Three datasets were used for analysis: Heart Disease Database, Z-Alizadeh Sani Dataset and South African Heart Disease. Highest accuracy achieved was 89.93%.
22		[47]	Different machine learning algorithms were analysed for prediction of heart disease. Different algorithm used included Naive Bayes, Classification Tree, KNN, Logistic Regression, SVM and ANN. Out of these, Logistic Regression was found to give the best results.
23		[32]	Two supervised classification problems named as Heart disease prediction and Springleaf Marketing Response were addressed by using ML techniques.
24		[31]	They proposed the model to predict the adherence of heart failure in patients by using the 11 classification algorithms and combining the with feature selection and resampling

			techniques
25	2016	[30]	A comparative study with previous work is done by using REMI algorithm using STULONG and UCI databases to predict the risk of coronary artery atherosclerosis using ML.
26		[52]	Various techniques including ANN, K-nearest neighbor classification, decision trees, Naive Bayes classifiers and support vector machine (SVM) have been used for predicting heart disease. Results show that performance of ANN and SVM was best.
27		[33]	Various data mining algorithms like Aprior, Naïve Bayes, J48, K-NN were used to diagnose heart diseases at early stage by using the Cleveland Dataset
28		[28]	A model was proposed to improve the accuracy and efficiency of already proposed models by reducing the no. of attributes of the dataset taken from UCI machine learning algorithm and introducing the new approach called average k-nearest neighbour algorithm.
29		[29]	A deep learning approach was used for analyzing the risk of four specific type of CVDs (Cardio Vascular Diseases) by studying the outpatient records within Taichung area and by employing Softmax and Autoencoder for feature extraction and classification.
30		[4]	In this paper, weights of each attribute in the dataset were set. Fuzzy-AHP was put in place to compute these weights. The average prediction accuracy achieved was 91.10%.
31	2015	[16]	In this paper a fuzzy logic and decision tree model was proposed for predicting the CHD (Coronary Heart Disease). Dataset used was prepared by Korean National Health and Nutrition Examination Survey VI. The accuracy of results obtained was 69.51%.
32		[50]	This paper proposes various data mining techniques like Naive Bayes and Modified K-Means for prediction of heart disease. The Cleveland dataset was used with 13 attributes for predicting the heart disease.
33		[49]	Several data mining classification techniques were put into action for prediction and analysis of heart disease. The classifiers included J48 Decision Tree, K Nearest Neighbours (KNN), Naive Bayes (NB), and SMO. Results show J48 Decision Tree gave highest accuracy.
34		[17]	Fuzzy K-NN approach can be used to remove the uncertainty present in the data so that heart diseases can be more accurately predicted. Further, more attributes were added to improve accuracy.
35		[51]	This paper uses artificial neural network for diagnosis of heart disease. Backpropagation was used and the network consisted of 20 neurons in the hidden layer. The UCI repository dataset was used for training purpose. The accuracy achieved was 88%.
36	2014	[7]	This paper uses neural network and global algorithm for prediction of heart disease. A method is described for calculating the number of hidden nodes in the network. Genetic algorithm is used for initialising the neural network. It used 12 attributes in the dataset.
37		[23]	In this paper, different ANN algorithms have been applied on the dataset with 13 attributes. All of which, MLP turn out to be more accurate and efficient.
38		[25]	Predictive models were prepared using ANN & LR. The prediction was done using a

			Chinese dataset consisting of 2092 rows. The individuals were aged between 30-80 years.
39	2013	[12]	The Probabilistic Neural Network (PNN) is used to predict heart disease. The data set containing 13 attributes is used for training purpose. The accuracy achieved was 92.10%.
40		[48]	Several data mining techniques were analysed in order to predict heart disease which included decision trees, SVM and logistic regression. The UCI heart disease dataset was used for this purpose.
41		[2]	A model was put forward for the accurate prediction of heart disease based on various risk factors like obesity, age, diabetes etc. by using the techniques of genetic algorithm and neural networks in MATLAB which gives the accuracy of 89%.
42	2012	[24]	A HDPS (Heart Disease Prediction System) was proposed using multilayer perceptron neural network. 13 attributes were present in the dataset used for training and validation. For enhancing the accuracy two more attributes (obesity and smoking) were added.
43		[21]	This paper proposed a hybrid PSO-based fuzzy expert system for the diagnosis of CAD (Coronary Artery Disease). The Cleveland and Hungarian dataset were used for this purpose. The output of decision tree was converted to fuzzy rule base. An accuracy of 93.27% was achieved.
44		[3]	In this paper genetic algorithm and multilayer perceptron neural network is used for prediction of heart diseases. Genetic algorithm is used to determine the weights of neural network. The accuracy achieved using this approach is 94.17%.
45		[1]	This paper proposes a data mining technique (associative classification algorithm) that uses genetic approach for the prediction of heart diseases. Accuracy achieved was 88.9%.
46	2011	[14]	A system for prediction of heart disease was made by using several parameters and applying ANN techniques on them. The accuracy came out to be 80%.
47	2009	[8]	This paper aims to predict coronary artery disease using artificial neural network. The dataset is prepared from 330 patients. The accuracy was found to be around 91%.
48	2008	[11]	A system is proposed for prediction of Cardiovascular Disease which combines a set of four different classifiers. SVM and neural network are adopted as a base classifier. Other two classifiers include Decision tree and Bayesian networks. The model was trained using data containing 66 samples. The highest accuracy achieved was 94%.
49	2007	[15]	This paper proposes a cutting edge approach named CANFIS in order to predict heart diseases. This model is a blend of fuzzy logic and neural network. Further genetic algorithm is used for detecting the presence of heart disease.
50		[26]	Prediction of Coronary Heart Disease was performed using Data mining techniques along with 10-fold cross-validation method.
51	2000	[20]	Dataset from angiography patient records was used having 14 attributes. The neural network was trained using forward feedback propagation on 332 records. The results indicated that the network was able to identify patients who do not needed coronary angiography.

52	1997	[19]	The results of multilayer perceptron neural network were compared with the conventional logistic regression and Bayesian model. The database included 80,606 patients and was used for the purpose of training and validation. The result of neural network was found to be better.
53	1990	[9]	They trained an artificial neural network by back propagation to diagnose the presence of acute myocardial infarction in patients.

TABLE 2: Technology Used

S.No.	Year	Technology Used	References
1.	2019	Machine Learning	[5], [13], [27], [40], [41], [43], [44]
2.	2018	Machine Learning	[37], [38], [39], [45]
		Data Mining	[6], [36]
		Deep Learning	[35]
		Fuzzy-ANN	[18]
3.	2017	Machine Learning	[10], [34], [46], [53]
		Fuzzy	[22]
		Data Mining	[42]
4.	2016	Machine Learning	[30], [31], [32], [47]
		Data Mining	[28], [33], [52]
		Deep Learning	[29]
		Fuzzy-ANN	[4]
5.	2015	Fuzzy	[16], [17]
		Data Mining	[16], [49], [50]
		ANN	[51]
6.	2014	Genetic Algorithm	[7]
		ANN	[7], [23]
7.	2013	ANN	[12], [25]
		Data Mining	[48]
		Genetic Algorithm	[2]
8.	2012	ANN	[24]
		Fuzzy	[21]
		Genetic Algorithm	[3], [1]
9.	2011	ANN	[14]
10.	2009	ANN	[8]
11.	2008	Machine Learning	[11]
12.	2007	Genetic Algorithm	[15]
		Data Mining	[26]
13.	2000	ANN	[20]
14.	1997	ANN	[19]
15.	1990	ANN	[9]

CHAPTER 3

PROBLEM DEFINITION & REQUIREMENT ANALYSIS

3.1 Project Undertaken

3.1.1 Problem Definition

Cardiovascular disease is considered as one of the main causes of death around the globe. It is very difficult to be predicted by the medical practitioners as it is a complex task which requires expertise and a lot of experience for prediction.

An automated system in medical diagnosis would increase medical efficiency and would also help in reducing costs. We will design a system that can efficiently discover the rules to predict heart disease in patients based on the given parameters about their health. The goal is to find the hidden patterns by employing machine learning algorithms, which are quick and reliable enough to predict the presence of heart disease in users and patients.

3.1.2 Scope Statement

Comparative study and analysis of various Machine Learning algorithms will be done. The most efficient algorithm of them will be found and used for implementation. A User-interactive web application will be developed for prediction of heart disease.

3.2 Requirement Analysis

3.2.1 System Requirement Specification (SRS)

System Requirement specification (SRS) is considered as the base for the effort estimations and project scheduling.

3.2.1.1 System Overview

It is a web based application based on machine learning which has been trained by UCI heart disease dataset. The user inputs specific medical details in order to get the prediction of heart disease. The algorithm will then calculate the possibility of heart disease. The result is displayed on the webpage itself. Thus, reducing the cost and time needed to predict the disease.

3.2.1.2 Functional Requirements

Format of data plays an important role in this application. When the user enters the medical details, it should be in correct format and also within the specified range, else ERROR dialog box will be prompted.

The following four algorithms will be analyzed:

- Support Vector Machine (SVM)
- Decision Tree (DT)
- *K*-Nearest Neighbor Algorithm (KNN)
- Random Forest Classifier

The working of these algorithms has been explained in the sections ahead.

The algorithms have been trained using the UCI (University of California, Irvine) Cleveland data set.

75% of the entries in the data set have been used for training and the remaining 25% for testing the accuracy of the algorithm. Furthermore, some steps have been taken for optimizing the algorithms thereby improving the accuracy. These steps include cleaning the dataset as well as data preprocessing.

The algorithms were judged based on their accuracy and it was observed that the *K*-Nearest Neighbor Algorithm (KNN) was the most accurate out of the four with 87.0% efficiency. Hence, it was selected for implementation of the main application.

The main application is a web application which accepts the various parameters from the user as input and computes the result.

Inputs: Data Set, User Data

Outputs: Predicted Result

3.2.1.3 Non-Functional Requirements

Performance Requirements

- To be precise there are no certain guidelines and criteria defined for the web application related to their performance.
- The system needs to be reliable.
- If unable to process the request then appropriate error message will be displayed.
- Web pages are loaded within few seconds.

Safety Requirements

This product uses the data of users for prediction purposes only, as is evident as there is no ”Name” input field. Thus, the privacy of the user is preserved. But the program uses open source libraries and any developer having doubts about safety can directly check the source code.

Security Requirements

- Data will not be saved.
- Data will be safe and secure.

3.2.1.4 Deployment Environment

Hardware Requirements

- Hard Disk : 500GB and Above
- RAM : 4GB and Above
- Processor : i3 and Above

Software Requirements

- Operating System : Windows 7 , 8, 10 (64 bit)
- Software : Python 3.7
- Tools : Anaconda (Jupyter Notebook IDE)

3.2.2 Project Process Modelling

The incremental model is a software development model where the product is designed, implemented and tested incrementally until the product is finished. It involves both development and maintenance. This process goes on until it satisfies all of its requirements of the user/client. This model combines the elements of the waterfall model with the iterative philosophy of prototyping. The product is broken down into a number of components, each of which is designed and built separately. Each component is delivered to the client once it is completed. This helps to avoid a large initial capital outlay and subsequent long waiting period.

Characteristics of Incremental Model

- System is decomposed into a number of small development projects.
- Partial systems are implemented to yield the final system.
- The requirements having highest priority are tackled first.
- The requirement of a fraction is frozen once the incremented portion has been developed.

Advantages

- After every iteration, regression testing should be conducted. During this testing, faulty elements of the software can be quickly identified because few changes are made within any single iteration.
- It is generally easier to test and debug than other methods of software development because relatively smaller changes are made during each iteration. This allows for more targeted and rigorous testing of each element within the overall product.
- Customer has the option to respond to features and review the product for any needed or useful changes.
- Initial product delivery is faster and also lowers the initial delivery cost.

CHAPTER 4

SYSTEM DESIGN

4.1 IDEA MATRIX

An IDEA matrix is a concept that evaluate various effects that the idea has. This tells us almost everything about the project. The letters in the word idea are the initials of the components analyzed regarding the project.It is represented in a tabular form. Along with the component name, its description is written beside it in another column.

Increase	• Accuracy of Prediction of Heart Disease • Ease of Operation for prediction
Improve	• Improved Heart Disease Prediction • Quick Measurable Results
Ignore	• The parameters that are not related to heart disease

Fig. 4.1.1

Drive	• To decrease the fatalities due to heart diseases.
Deliver	• Prediction System • Most Efficient Algorithm
Decrease	• Decreased Death Rate • Prediction Time • Ambiguity

Fig. 4.1.2

Experiment	• All the algorithms will be tested to analyze their efficiency.
Evaluate	• Probability of heart disease • Efficiency of the algorithms being tested
Eliminate	• Risk of Sudden Cardiac arrests

Fig. 4.1.3

Accelerate	• The process of prediction of heart disease.
Avoid	• Absurd results • Sudden Deaths.
Actualize	Implementation of System using the data set

Fig. 4.1.4

4.2 Architecture Diagram

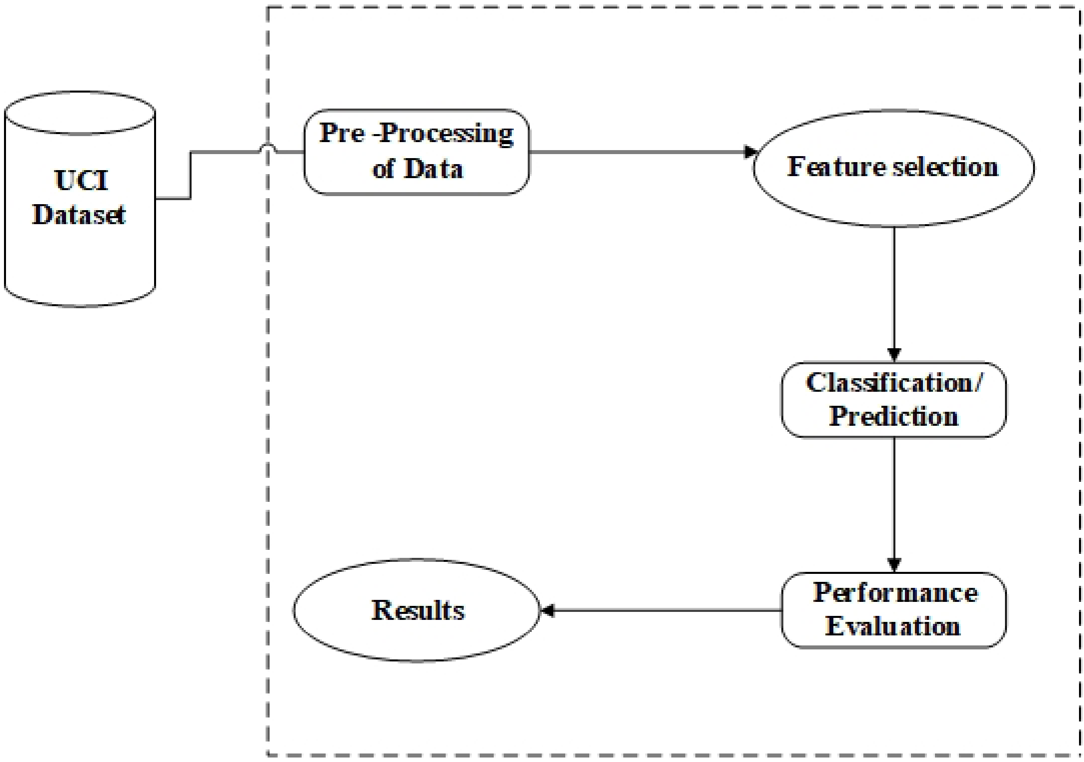


Fig. 4.2

4.3 Use Case Diagram

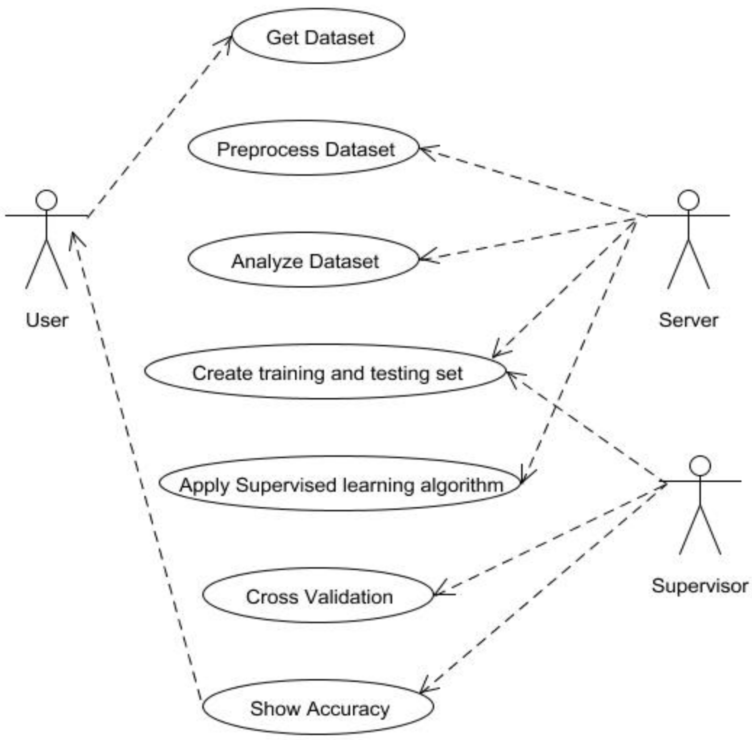


Fig. 4.3

4.4 Activity Diagram

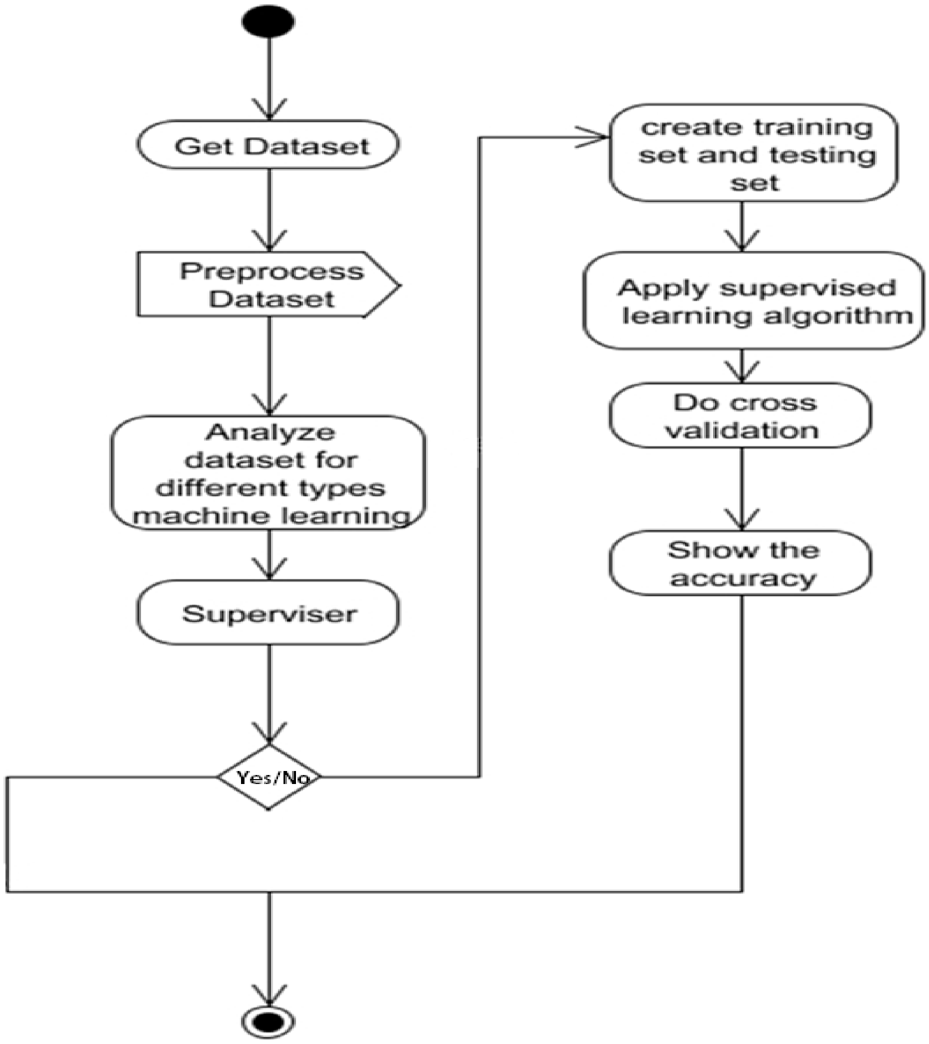


Fig. 4.4

4.5 Collaboration Diagram

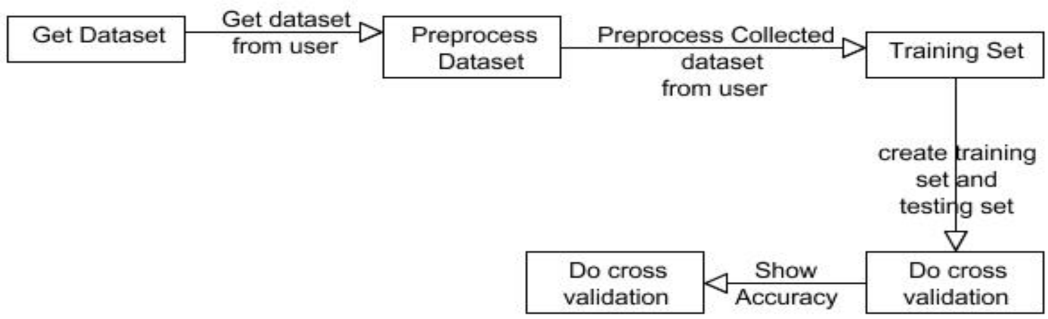


Fig. 4.5

4.6 Sequence Diagram

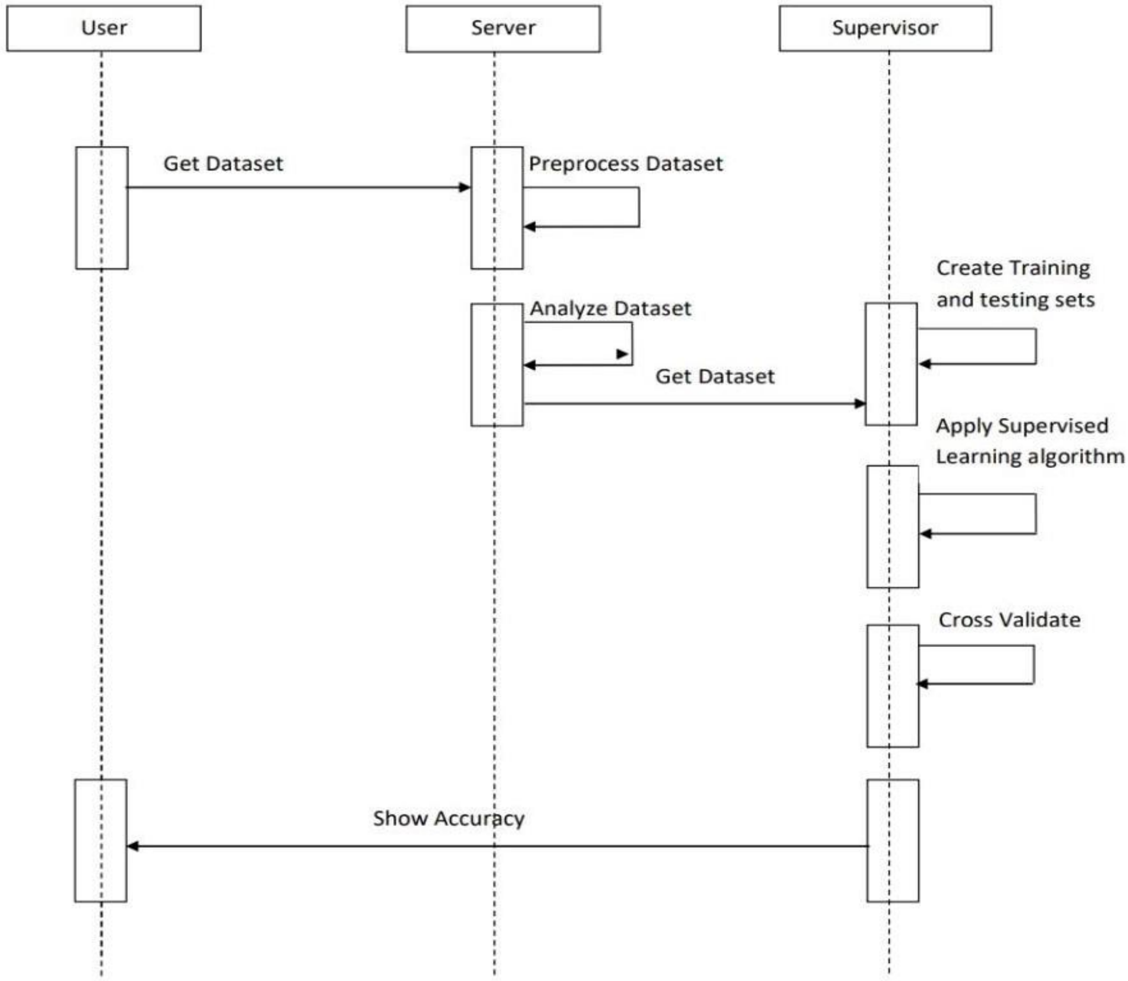


Fig. 4.6

4.7 Interface Diagram

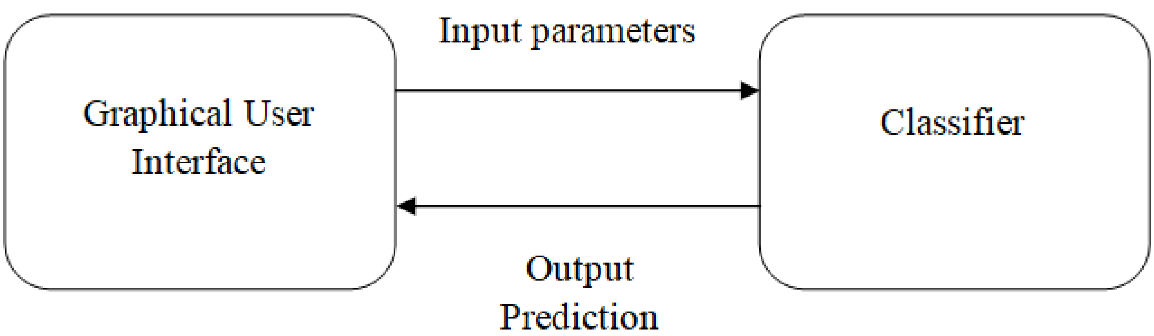


Fig. 4.7

4.8 State Machine Diagram

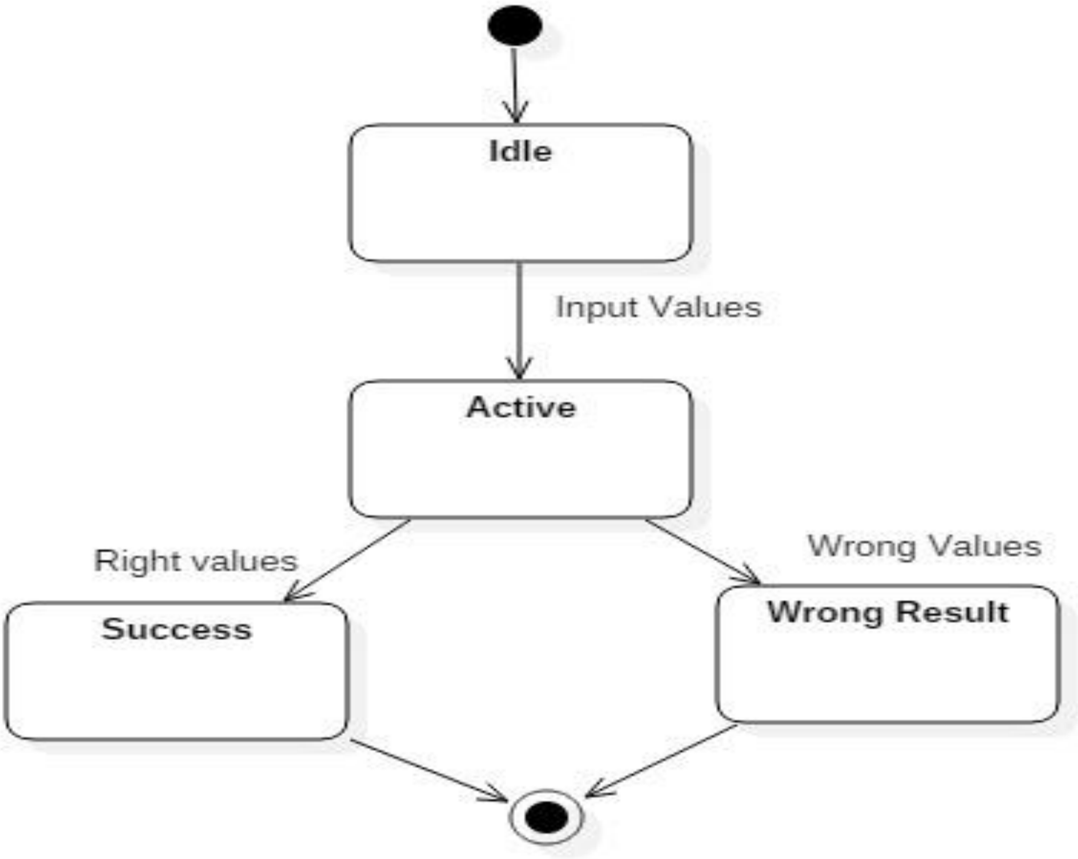


Fig. 4.8

CHAPTER 5

ANALYSIS AND IMPLEMENTATION

This chapter describes the working of the algorithms involved in this project and covers the role of various subsystems/modules/classes along with implementation details listing of the code for the major functionalities. Also, comparison and results of various machine learning algorithm has explained based on their accuracy.

5.1 Dataset Parameters Description

Field	Description	Range and Values
Age	Age of the patient	0-100 (in years)
Sex	Gender of the patient	0-1 (1:Male 0:Female)
Chest Pain	Type of chest pain	1-4 (1: Typical Angina, 2: Atypical Angina, 3: Non-anginal, 4: Asymptotic)
Resting Blood Pressure	Blood pressure during rest	in mm Hg
Cholesterol	Serum Cholesterol	mg / dl
Fasting Blood Sugar	Blood sugar content before food intake if >120 mg/dl	0-1 (0: False, 1: True)
ECG	Resting Electrocardiographic results	0-1 (0: Normal, 1: Having ST-T wave)
Max Heart Rate	Maximum heart beat rate.	Beats/min
Exercise Induced Angina	Has pain been induced by exercise	0-1 (0: No, 1: Yes)
Old Peak	ST depression induced by exercise relative to rest	0-4
Slope of Peak Exercise	Slope of the peak exercise ST segment	1-3 (1: Up sloping, 2: Flat, 3: Down sloping)
Ca	Number of vessels coloured by fluoroscopy	0-3
Thal	Displays the thalassemia	3- normal 6-Fixed Defect 7- Reversible Defect
Num	Diagnostics of Heart Disease	0-1 (0: <50% Narrowing 1: >50% Narrowing)

Table 3: Dataset Parameters Description

5.2 Operational Details

Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised learning methodology which is used to analyse data used for classification and regression analysis. The training data is fed into an SVM training algorithm which then builds a model that classifies new examples to one category or the other, that makes SVM a non-probabilistic binary linear classifier. An SVM model is an illustration of the examples as points in space, mapped in such a way that the examples of the separate categories are separated by a significant gap that is as broad as possible. New examples are then mapped into that same space and predicted to reside to a category based on which side of the gap they belong. The points are separated based on hyper planes that separate them.

Supervised learning is not possible if the data set is unlabelled, and an unsupervised learning approach is required, which tries to find natural clustering of the data to groups, and then map new data to these formed groups.

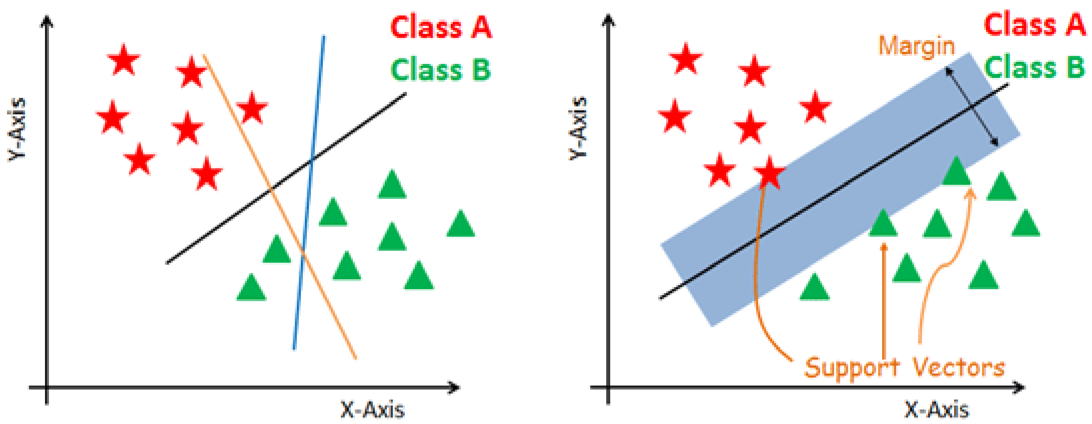


Fig.5.2.1 Working of Support Vector Machine

Decision Tree

A decision tree is a flowchart-like structure in which each internal node represents a "test" on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represent classification rules.

For decision analysis, a decision tree along with a closely related influence diagram is used as an analytical and visual decision support tool, where the expected values of competing alternatives are calculated.

There are three types of nodes in a Decision Tree:

- Decision nodes (which are represented by squares)
- Chance nodes (which are represented by circles)
- End nodes (which are represented by triangles)

Decision trees are generally used in research and management operations. If the decisions are to be taken online without any recall under the situation of insufficient data, a decision tree should be accompanied by a probability model as an online selection model algorithm. Decision trees can also be used as a descriptive means for the calculation of conditional probabilities.

Decision trees, utility functions, influence diagrams and other analysis tools for making decision are helpful in economics, business and public health.

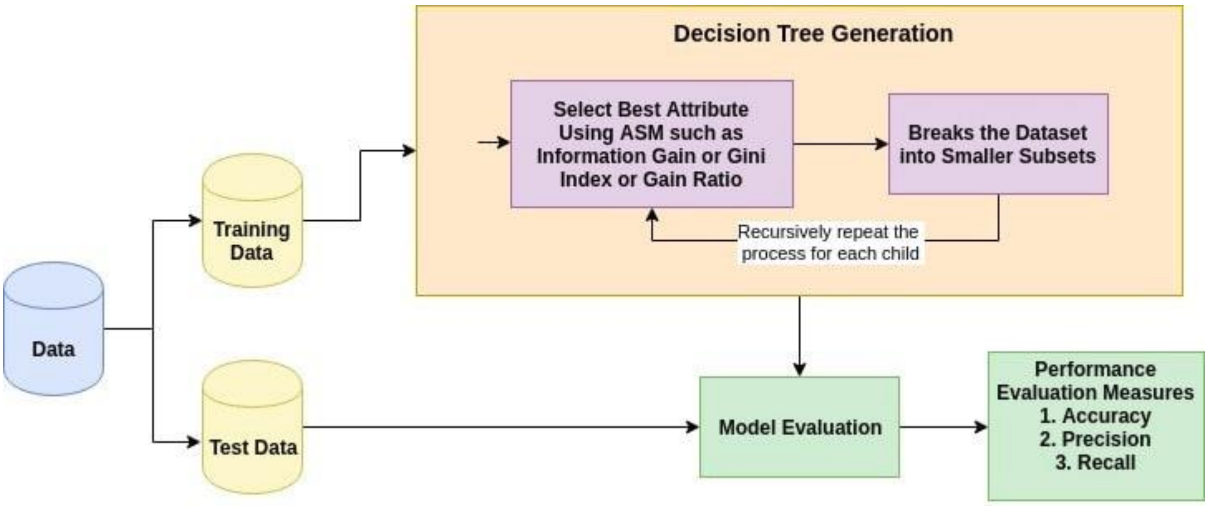


Fig. 5.2.2 Working of Decision Tree Classification

***K*-Nearest Neighbor Algorithm (KNN)**

The KNN classification algorithm is used to decide the new instance should belong to which class. When K = 1, we have the nearest neighbour algorithm. KNN classification is incremental. KNN classification does not have a training phase, all instances are stored. Training uses indexing to find neighbours quickly.

During testing, KNN classification algorithm has to find K-nearest neighbours of a new instance. This is time consuming if we do exhaustive comparison. K-nearest neighbours use the local neighborhood to obtain a prediction.

Algorithm: Let m be the number of training data samples. Let p be an unknown point.

Store the training samples in an array of data points array. This means that each element of this array represents a tuple (x, y).

For i = 0 to m:

Calculate Euclidean distance d(arr[i],p).

Make set S of K smallest distances obtained. Each of these distances corresponds to an already classified data point.

Return the majority label among S.

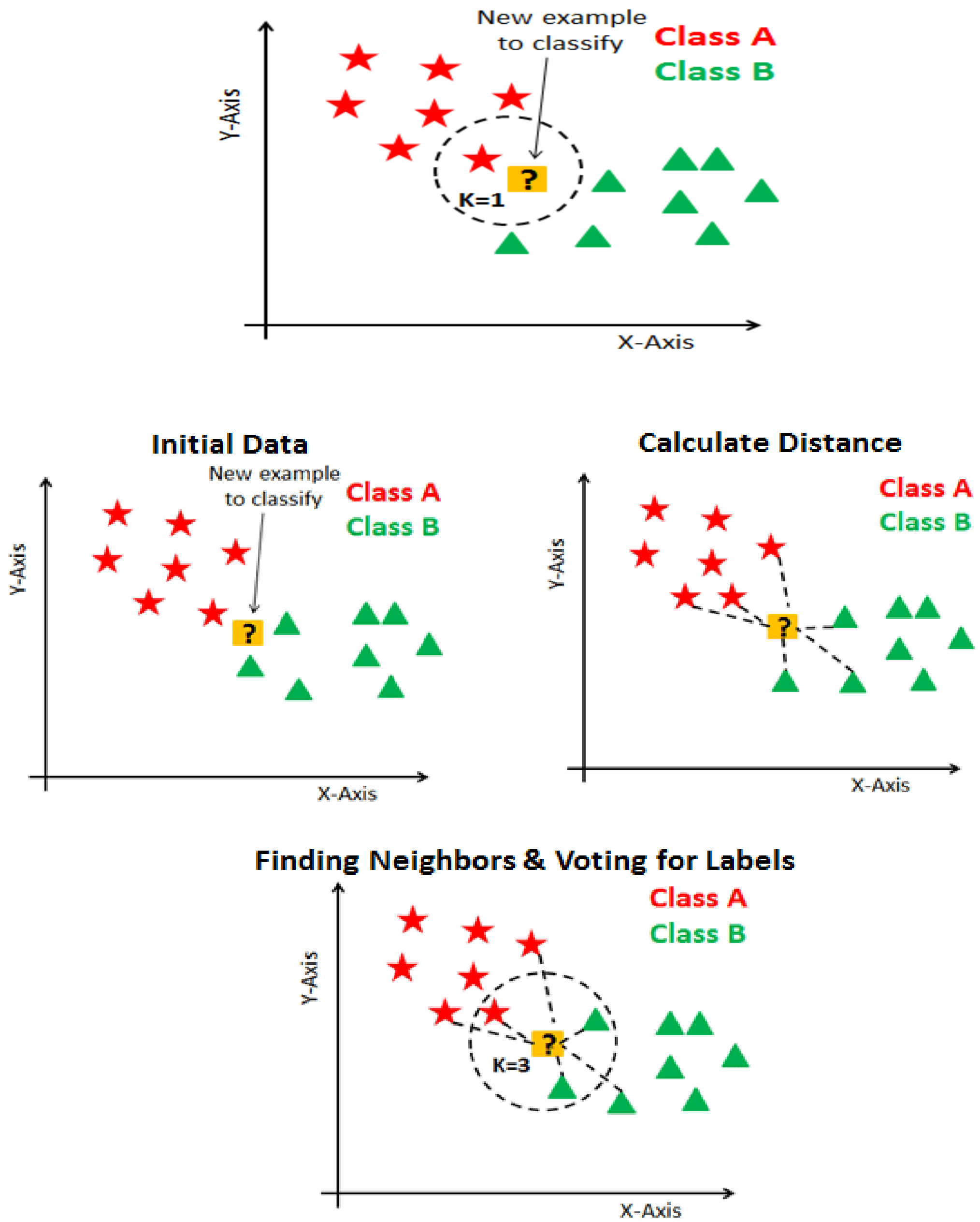


Fig. 5.2.3 Working of *K*-Nearest Neighbor Algorithm

Random Forest Classifier

Random forest classifier is also a supervised learning algorithm which can be utilized both for the purpose of classification and regression. It is also considered as the most simple, flexible and easy to use algorithm in machine learning. A forest consists of trees. A forest is said to be more robust if it has more trees. Random forests forms decision trees based on randomly selected data samples, gets prediction from each tree and selects the best solution by means of voting. It also provides a pretty good indicator of the feature importance.

Random forest classifier can be applied in various fields which includes recommendation engines, feature selection and image classification. It is also applied in classification problems such as loyal loan applicants, identify fraudulent activity and for prediction of diseases. More or less it follows the Boruta algorithm, in which important features are selected in a dataset.

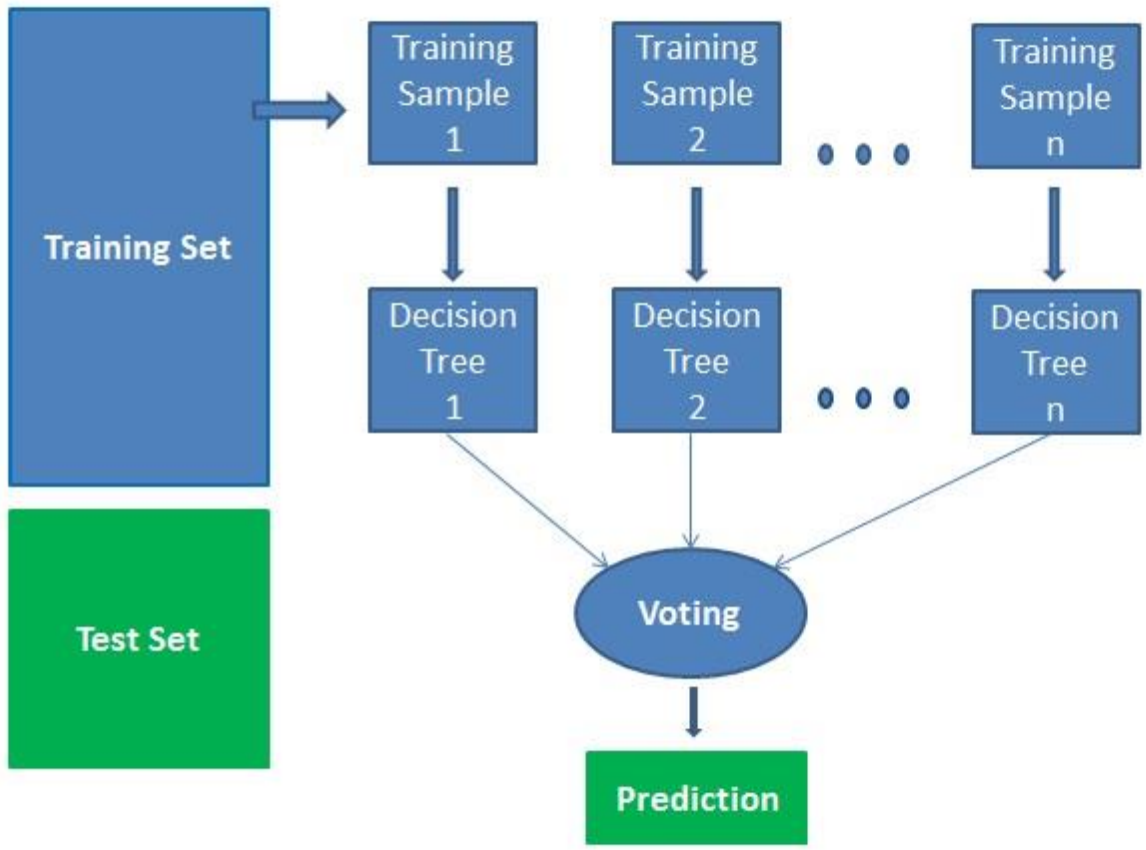


Fig. 5.2.4 Working of Random Forest Classifier

5.3 Processing the Dataset

The dataset we obtained was not completely accurate and error free. Hence we first carried out the following operations on it:

- **Pre-Processing of Data Set**
- **Feature Selection And Scaling**
- **Factorization**

Pre-Processing of Data Set

After collection of various records pre-processing of dataset was carried out. The dataset consists of a total of 303 patient records (instances), where 6 records are with some missing values. So, those 6 records were removed from the dataset and the remaining 297 patient records are used.

Feature Selection and Scaling

The dataset had 13 attributes out of which two features which included age and sex are used to identify the personal patient information. 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.

Since the range of values of raw data varies widely, in some machine learning algorithms, objective functions will not work properly without feature scaling. For example, the majority of classifiers calculate the distance between two points by the Euclidean distance. If one of the features has a broad range of values, the distance will be governed by this particular feature. Therefore, the range of all features should be scaled so that each feature contributes approximately proportionately to the final distance.

So we scaled the various fields in order to get them closer in terms of values. Eg. Sex had just two values i.e. 0,1 and cholesterol had high values like 100. So, in order to get them closer to each other, they were scaled to a value between -1, and 1.

Factorization

In this section, we assigned a meaning to the values so that the algorithm doesn’t confuse between them. For example, assigning meaning to 0 and 1 in the age section so that the algorithm doesn’t consider 1 as greater than 0 in that section.

5.4 Understanding the Dataset

Before the actual analysis of algorithm it is important to fully understand the characteristics of the dataset that is to be used.

Correlation Matrix

When the dataset contains a number of attributes, then correlation matrix is used to simplify up the things. It shows the correlation coefficients between each pair of attributes of the dataset. It helps to analyze the dataset and see the patterns between attributes.

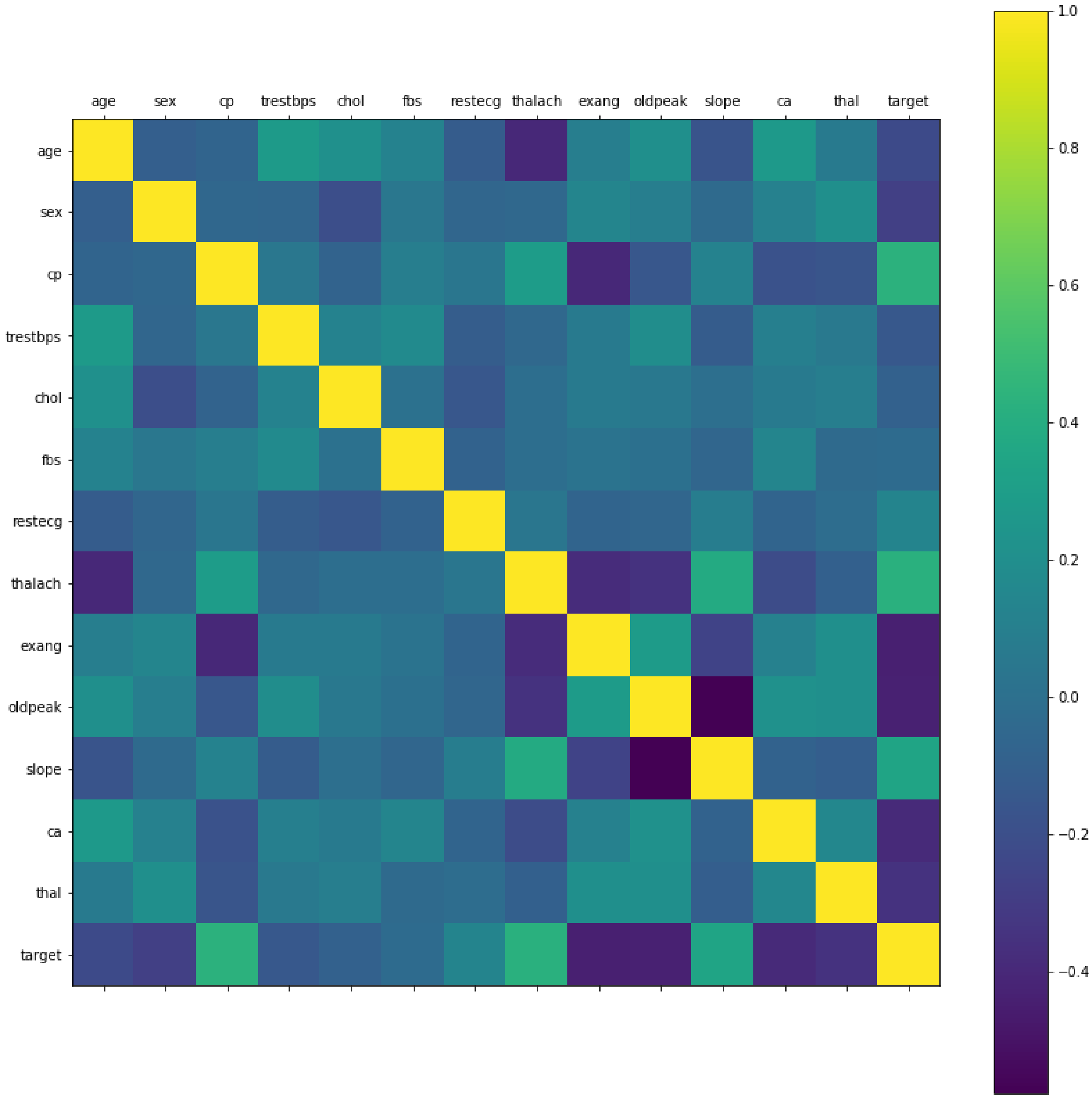


Fig. 5.4.1 Correlation Matrix

Histogram of Features/Attributes

The histogram below shows the distribution of each feature along the different ranges.

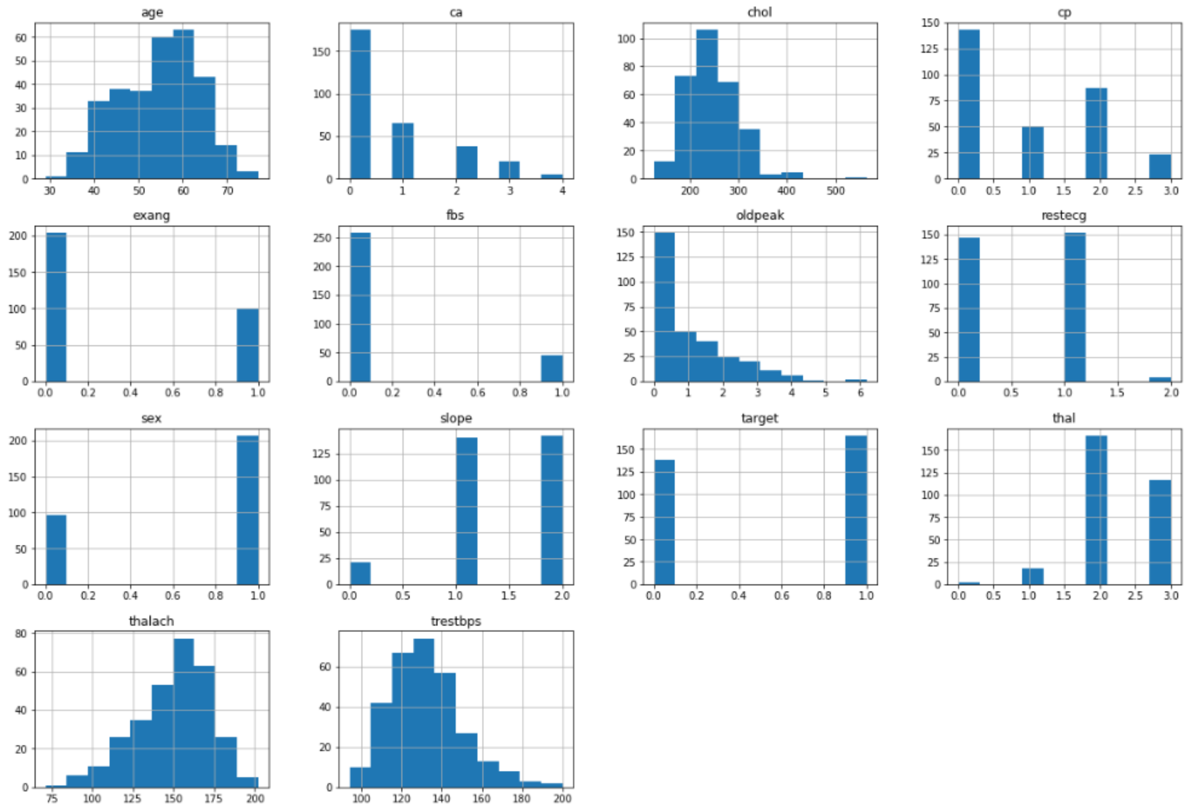


Fig. 5.4.2

Bar Graph of Target Class Count

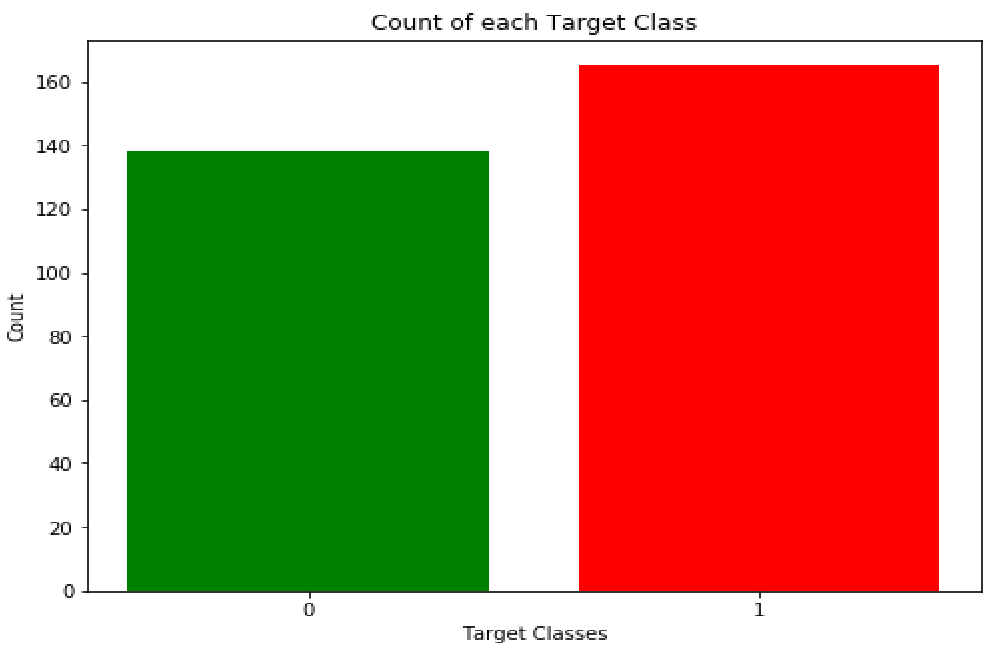


Fig. 5.4.3

There are two target classes in the dataset (0 & 1). From the above bar graph (Fig. 5.4.3), it is evident that both the classes share almost equal proportions of instances.

5.5 Comparative Analysis of ML Algorithms

After all the pre-requisites were fulfilled the following four algorithms were compared.

- Support Vector Machine
- Decision Trees Classifier
- K- Nearest Neighbour
- Random Forest Classifier

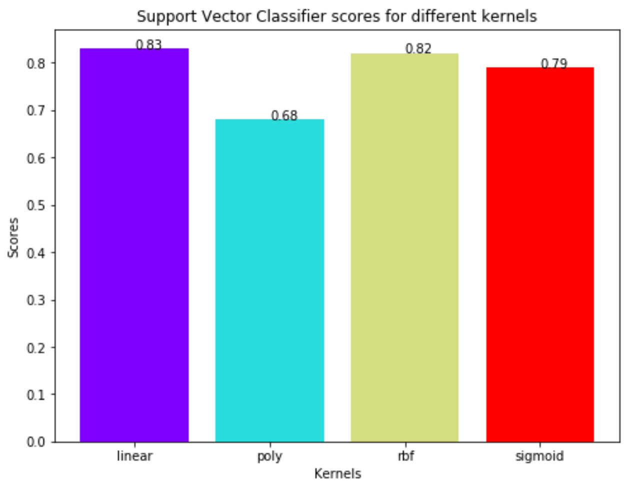
Support Vector Machine

The accuracy score of SVM was found to be **83.0%**.

```
In [17]: svc_scores = []
kernels = ['linear', 'poly', 'rbf', 'sigmoid']
for i in range(len(kernels)):
    svc_classifier = SVC(kernel = kernels[i])
    svc_classifier.fit(X_train, y_train)
    svc_scores.append(svc_classifier.score(X_test, y_test))
```

```
In [18]: colors = rainbow(np.linspace(0, 1, len(kernels)))
plt.bar(kernels, svc_scores, color = colors)
for i in range(len(kernels)):
    plt.text(i, svc_scores[i], svc_scores[i])
plt.xlabel('Kernels')
plt.ylabel('Scores')
plt.title('Support Vector Classifier scores for different kernels')
```

```
Out[18]: Text(0.5, 1.0, 'Support Vector Classifier scores for different kernels')
```




```
In [19]: print("The score for Support Vector Classifier is {}% with {} kernel.".format(svc_scores[0]*100, 'linear'))
The score for Support Vector Classifier is 83.0% with linear kernel.
```

Fig. 5.5.1

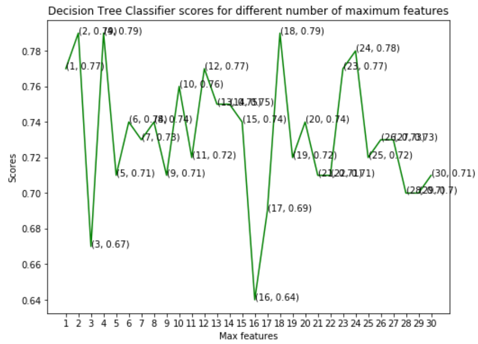
Decision Tree Classifier

The accuracy score of Decision Tree was found to be **79.0%**.

```
In [20]: dt_scores = []
         for i in range(1, len(X.columns) + 1):
             dt_classifier = DecisionTreeClassifier(max_features = i, random_state = 0)
             dt_classifier.fit(X_train, y_train)
             dt_scores.append(dt_classifier.score(X_test, y_test))
```

```
In [21]: plt.plot([i for i in range(1, len(X.columns) + 1)], dt_scores, color = 'green')
         for i in range(1, len(X.columns) + 1):
             plt.text(i, dt_scores[i-1], (i, dt_scores[i-1]))
         plt.xticks([i for i in range(1, len(X.columns) + 1)])
         plt.xlabel('Max features')
         plt.ylabel('Scores')
         plt.title('Decision Tree Classifier scores for different number of maximum features')

Out[21]: Text(0.5, 1.0, 'Decision Tree Classifier scores for different number of maximum features')
```




```
In [22]: print("The score for Decision Tree Classifier is {}% with {} maximum features.".format(dt_scores[17]*100, [2,4,18]))

         The score for Decision Tree Classifier is 79.0% with [2, 4, 18] maximum features.
```

Fig. 5.5.2

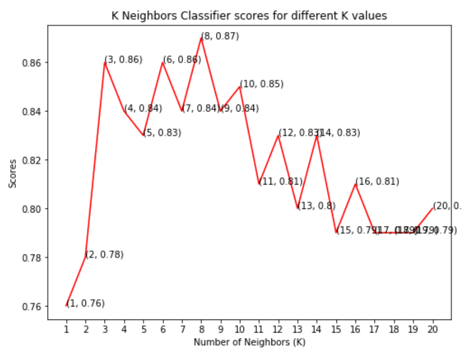
K-Nearest Neighbour

The accuracy score of KNN was found to be **87.0%**.

```
In [14]: knn_scores = []
         for k in range(1,21):
             knn_classifier = KNeighborsClassifier(n_neighbors = k)
             knn_classifier.fit(X_train, y_train)
             knn_scores.append(knn_classifier.score(X_test, y_test))
```

```
In [15]: plt.plot([k for k in range(1, 21)], knn_scores, color = 'red')
         for i in range(1,21):
             plt.text(i, knn_scores[i-1], (i, knn_scores[i-1]))
         plt.xticks([i for i in range(1, 21)])
         plt.xlabel('Number of Neighbors (K)')
         plt.ylabel('Scores')
         plt.title('K Neighbors Classifier scores for different K values')

Out[15]: Text(0.5, 1.0, 'K Neighbors Classifier scores for different K values')
```




```
In [16]: print("The score for K Neighbors Classifier is {}% with {} nieghbors.".format(knn_scores[7]*100, 8))

         The score for K Neighbors Classifier is 87.0% with 8 nieghbors.
```

Fig. 5.5.3

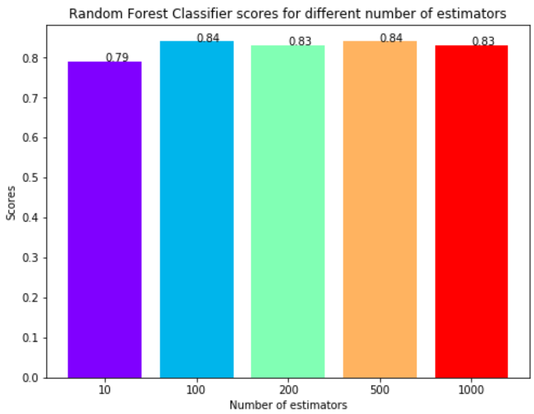
Random Forest Classifier

The accuracy of Random Forest Classifier was found to be **84.0%**.

```
In [23]: rf_scores = []
         estimators = [10, 100, 200, 500, 1000]
         for i in estimators:
             rf_classifier = RandomForestClassifier(n_estimators = i, random_state = 0)
             rf_classifier.fit(X_train, y_train)
             rf_scores.append(rf_classifier.score(X_test, y_test))
```

```
In [24]: colors = rainbow(np.linspace(0, 1, len(estimators)))
         plt.bar([i for i in range(len(estimators))], rf_scores, color = colors, width = 0.8)
         for i in range(len(estimators)):
             plt.text(i, rf_scores[i], rf_scores[i])
         plt.xticks(ticks = [i for i in range(len(estimators))], labels = [str(estimator) for estimator in estimators])
         plt.xlabel('Number of estimators')
         plt.ylabel('Scores')
         plt.title('Random Forest Classifier scores for different number of estimators')

Out[24]: Text(0.5, 1.0, 'Random Forest Classifier scores for different number of estimators')
```




```
In [25]: print("The score for Random Forest Classifier is {}% with {} estimators.".format(rf_scores[1]*100, [100, 500]))

         The score for Random Forest Classifier is 84.0% with [100, 500] estimators.
```

Fig. 5.5.4

Overall Performance

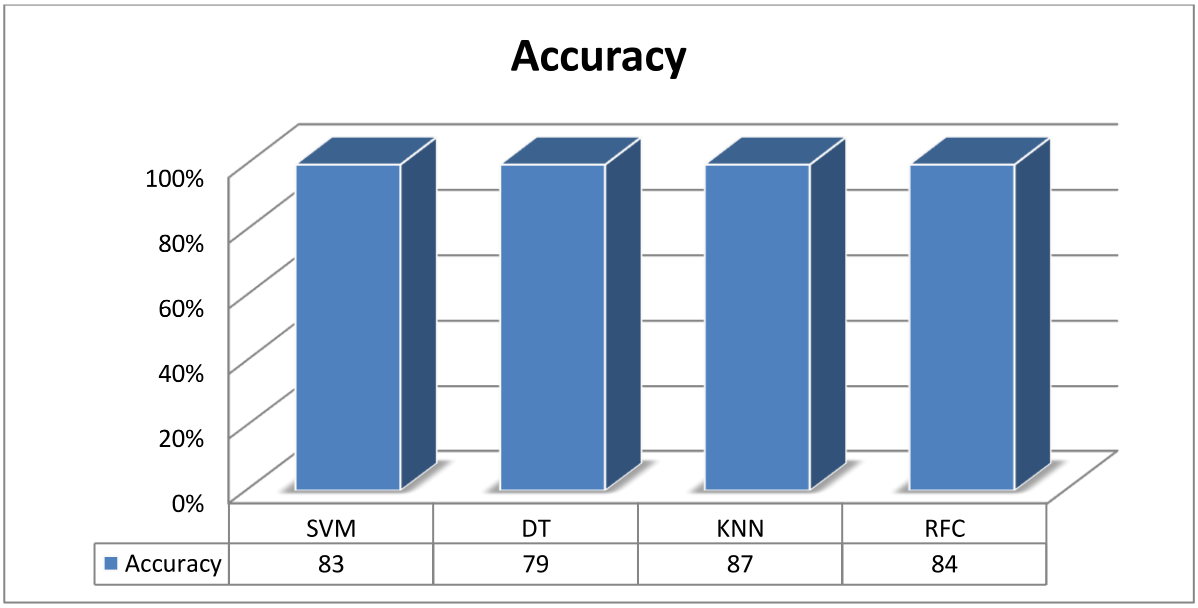


Fig. 5.5.5

5.6 Implementation Strategy

Based on the analysis, K-Nearest Neighbour (KNN) was found to be most accurate and reliable. Therefore, KNN was used for the final implementation of the project. Python 3 was used for modelling and classification. The dataset was split into training and testing data in the ratio of 3:1 i.e., 75% of the dataset was used for training purpose & the remaining 25% was used for testing and validation. Front-end is based on HTML5, CSS and JS. Python's micro web-framework Flask is also used for database connection.

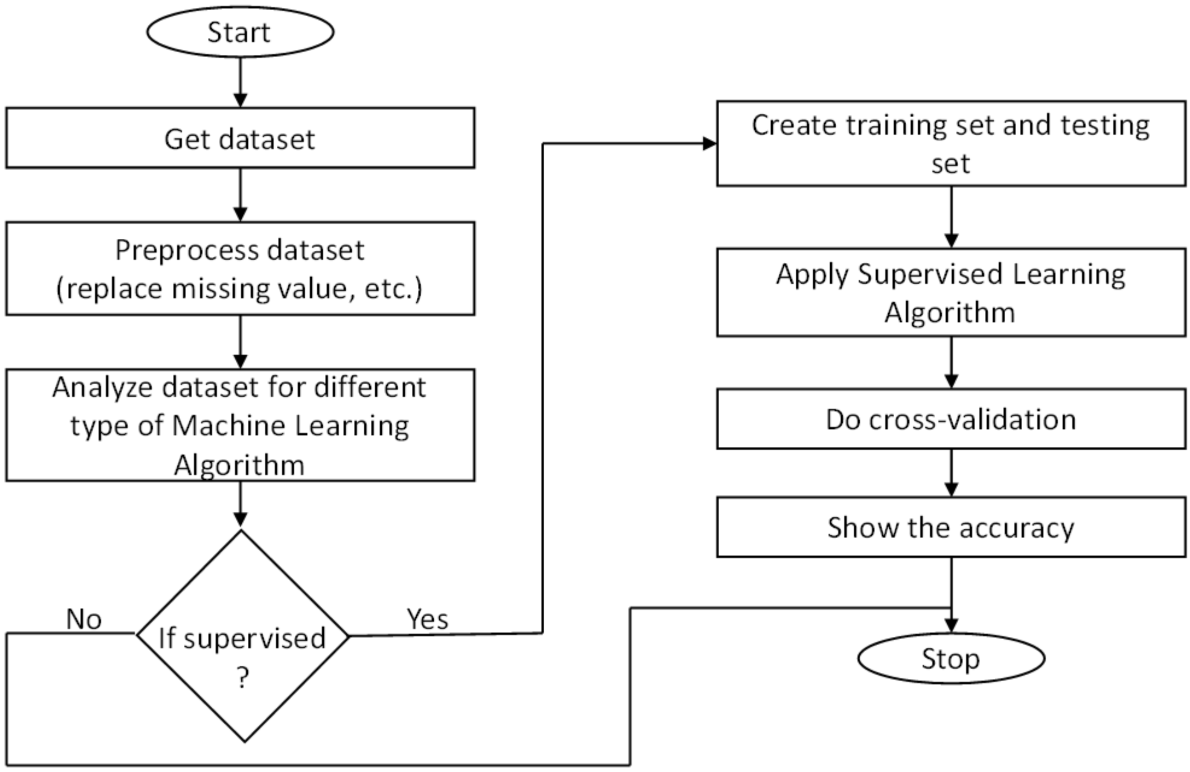


Fig. 5.6

CHAPTER 6

TESTING

This chapter covers the testing approach used in this project.

This Chapter covers the following testing approaches -

1. Unit Testing

2. Integration Testing

3. Acceptance Testing

4. Program Testing

5. Security Testing

6.1 Unit Testing

Unit testing primarily focuses on verification on the fundamental element of the program model.

By using the in depth design description most significant control paths are tested to find the errors within the limits of the module. In this system each sub module has been tested individually. Their input field validations are tested.

Test to be conducted	Input	Expected Output
Web app	13 Attributes entered by user	Should support HTML 5, CSS 3
GUI Test	On click event on predict button	All fields on the page

Table 4: Unit Testing

6.2 Integration Testing

Integration testing is a well-structured approach for building the structure of the program and at the same time carrying out tests to spot errors within the program. Individual modules are very prone to interface errors that is why we should not assume that they will work perfectly when put together. Therefore, the major problem which arises is “putting them together”.

Once all the individual unit have been tested there is need to test how they were put together to ensure no data is lost across the interface. The purpose is to make sure that each module does not have an unfavorable impact on another. After unit testing each and every sub module is tested with integrating each other.

6.3 Acceptance Unit

Acceptance testing is a testing methodology that is used to ensure that the project software has met the requirement specifications or not. The sole purpose of this testing methodology is to examine the system’s agreement with the requirements of business and then cross check if it is has met the set benchmarks for delivery to the client.

Acceptance test falls under black box system test. Basically every acceptance test represents an accepted result from the system. However, the correctness of the acceptance test has to be verified by the customers. Reviewing test scores are used to determine the priority of the failed tests. User acceptance testing of the system is crucial element for the success of any system. This is done by keeping the following points in mind -

- Input screen design.
- Output screen design.

Test ID	Test Category	Test Description
TC – 1	Network Connectivity	Check the connectivity of the network
TC – 2	Database Connectivity	Proper connection of front end and back end

Table 5: Acceptance Testing

6.4 Program Testing

The main purpose of program testing is to eradicate logical and syntax errors in the program. If a program statement violates the rules of the language in which it is written, it is called a syntax error. These errors pop up as error messages which are generated by the computer system. On the other hand, a logical error includes incorrect data fields, invalid combination and out-off-range. The problem is that compiler is not going to deduct logical errors, so the programmer must examine the output as well.

6.5 Security Testing

Security testing tries to authenticate the protection mechanisms of the software. To be more specific it helps in protecting it from illegal penetration. The security of the system must be tested for invulnerability from frontal attack as well as from rear attack.

CHAPTER 7

RESULTS

This chapter includes the snapshots of the final running project.

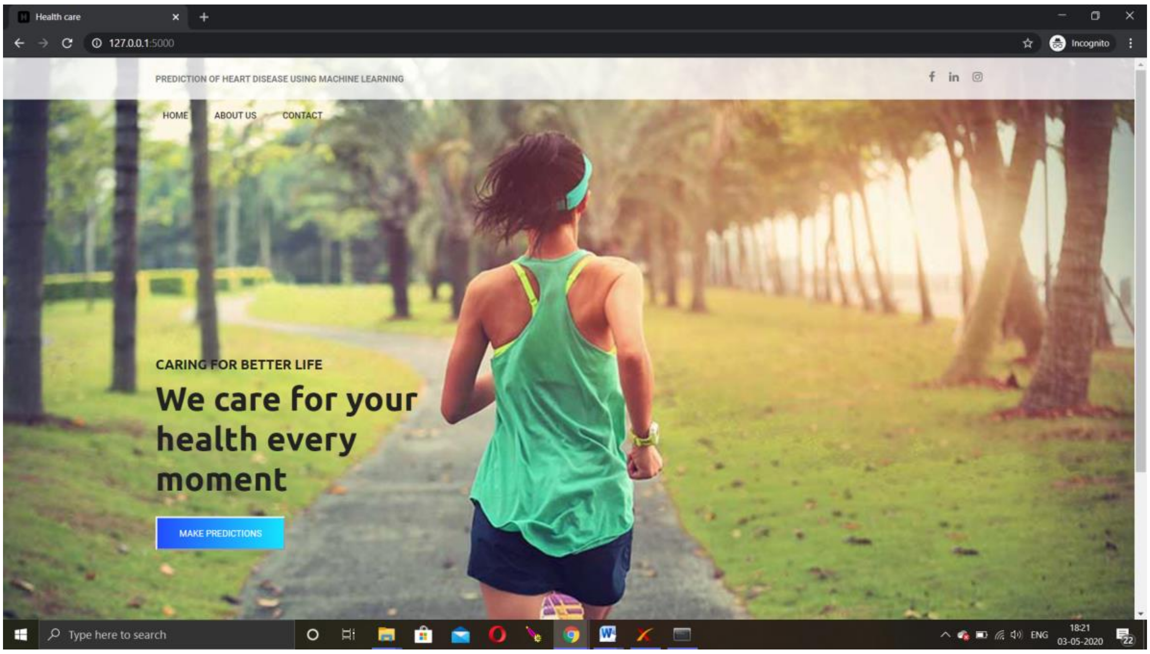


Fig. 7.1 Home Screen

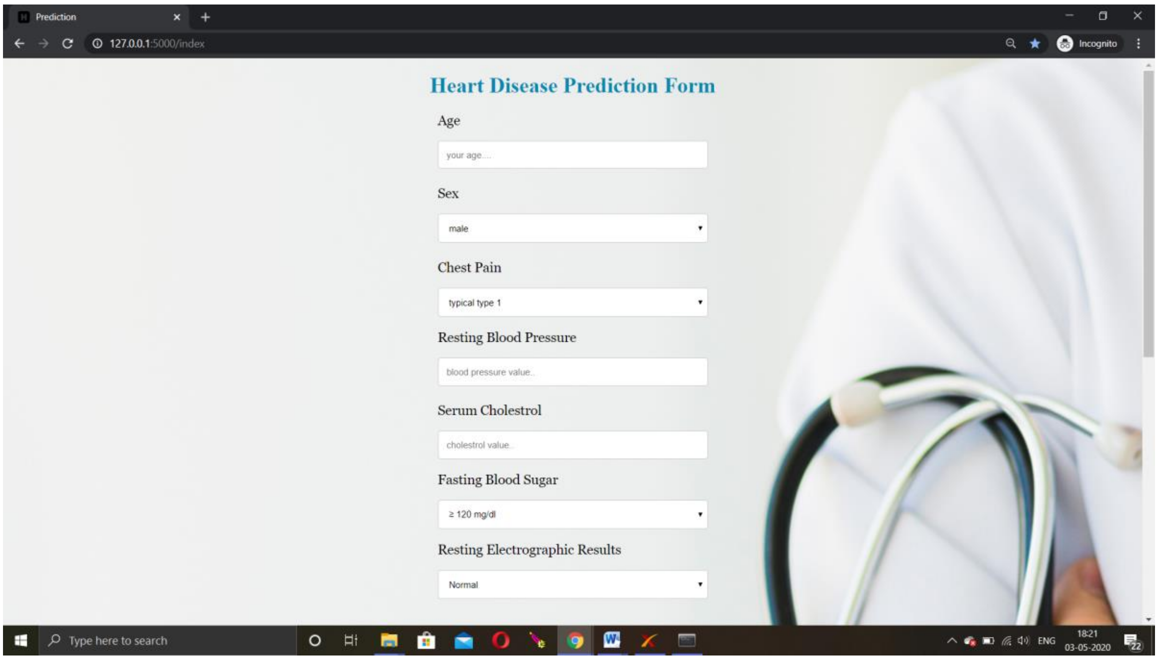


Fig. 7.2 Input Fields (i)

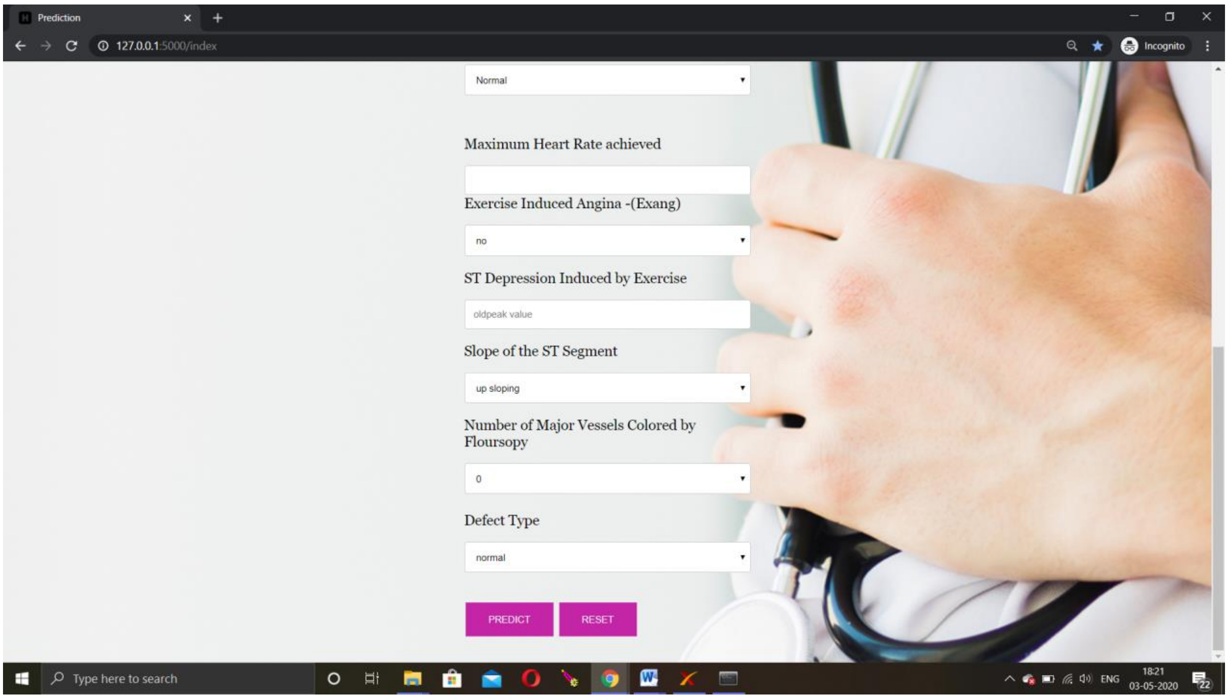


Fig. 7.3 Input Fields (ii)

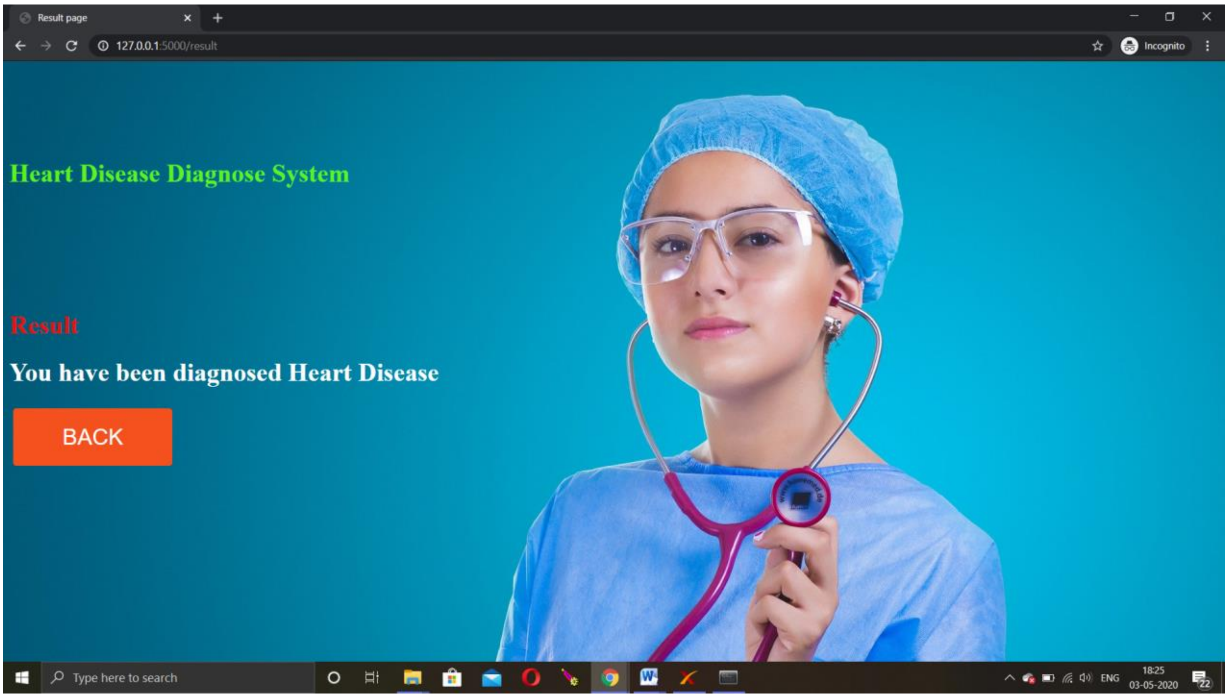


Fig. 7.4 Predicted Output (i)

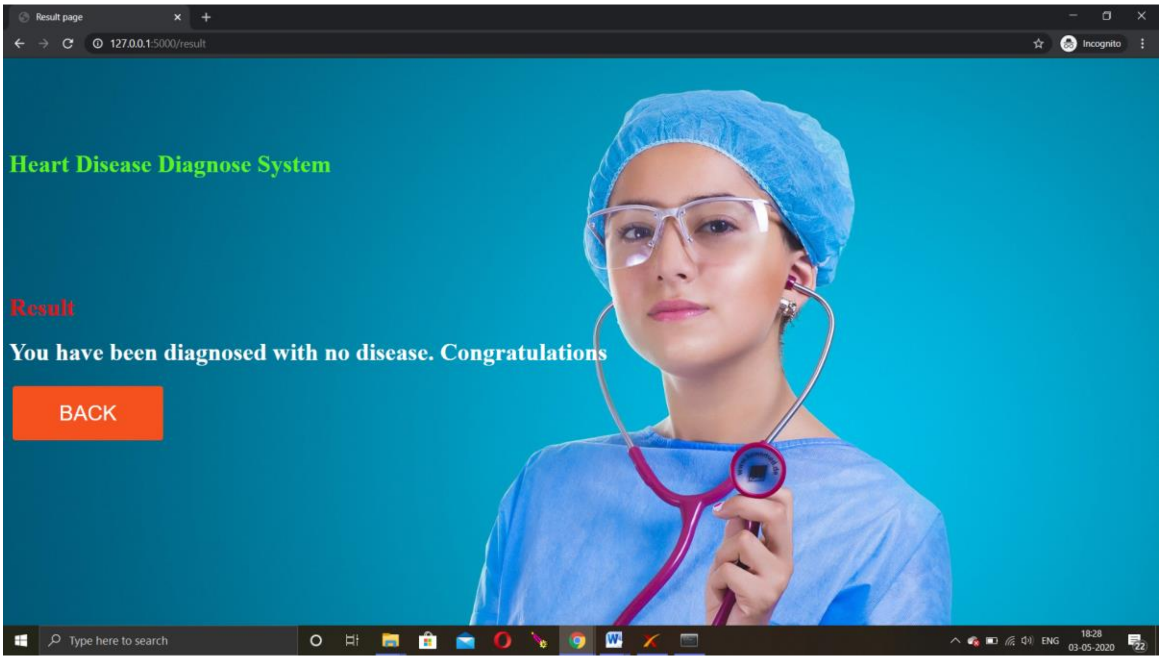


Fig. 7.5 Predicted Output (ii)

CHAPTER 8

CONCLUSION AND FUTURE SCOPE

8.1 Conclusion

At first, the four algorithms were implemented. Datasets were trained for all the algorithms individually. After this, all of them were tested. The most efficient algorithm was to be selected based on various criteria. We found out that KNN algorithm was the most efficient out of the four algorithms with an accuracy of 87.0%. Decision tree, Support Vector Machine and Random Forest Classifier had accuracy of 79.0%, 83.0% and 84.0% respectively. Thus, KNN algorithm was further implemented using a better user interface in form of a web application. For this HTML5, CSS, JS and Flask (Python's micro web-framework) were used. This would help the end users get a preliminary prediction about the condition of their heart. Since heart diseases are a major killer in India and throughout the world, application of a promising technology like machine learning to the initial prediction of heart diseases will have a profound impact on the society.

This will tell the user if they are at a risk and if they need to visit the doctor. This will help reduce the death rate due to heart attacks.

Hence by using the above approach successful analysis of heart diseases of the individual was performed and the result was obtained which predicted the risk of heart disease based on the parameters provided by the user.

8.2 Future Scope

By using the machine learning concept, newly trained dataset can be used for an even more accurate prediction system. Accounts can be created for each user and then by referring the past choice history of user's heart condition can be monitored to tell if there is any improvement or if the condition has deteriorated.

CHAPTER 9

REFERENCES

[1] M.Akhil jabbar, Dr.Priti Chandrab, Dr.B.L Deekshatuluc.: "Heart Disease Prediction System using Associative Classification and Genetic Algorithm", ICECIT, (2012)

[2] Syed Umar Amin, Kavita Agarwal, Dr. Rizwan Beg.: "Genetic Neural Network Based Data Mining in Prediction of Heart Disease Using Risk Factors", ICT, (2013)

[3] Bhuvaneswari Amma N.G., .: "Cardiovascular Disease Prediction System using Genetic Algorithm and Neural Network", IEEE, 2012

[4] Oluwarotimi Williams Samuel, Grace Mojisola Asogbona, Arun Kumar Sangaiahc, Peng Fanga, and Guanglin Lia.: "An Integrated Decision Support System Based on ANN and Fuzzy_AHP for Heart Failure Risk Prediction", SIAT, CAS, (2016)

[5] Aditi Gavhane ; Gouthami Kokkula ; Isha Pandya ; Prof. Kailas Devadkar.:"Prediction of Heart Disease using Machine Learning", ICECA(2019)

[6] K. Mathan, Priyan Malarvizhi Kumar, Parthasarathy Panchatcharam, Gunasekaran Manogaran, R. Varadharajan.: "A novel Gini index decision tree data mining method with neural network classifiers for prediction of heart disease", Kluwer Academic Publishers Norwell, vol. 22, (2018)

[7] Nilakshi P. Waghulde, Nilima P. Patil.: "Genetic Neural Approach for Heart Disease Prediction", ISSN, vol. 4, no. 3, sep-2014

[8] Ismail Babaoglu, Omer Kaan Baykan, Nazif Aygul, Kurtulus Ozdemir, Mehmet Bayrak.: "Assessment of exercise stress testing with artificial neural networkin determining coronary artery disease and predicting lesion localization", Science Direct, 2009

[9] William G. Bat.: "Use of an Artificial Neural Network for Data Analysis in Clinical Decision-Making: The Diagnosis of Acute Coronary Occlusio", MITP, vol. 2, issue. 4, 1990

[10] Himanshu Sharma, M A Rizvi. : “Prediction of Heart Disease using Machine Learning Algorithms: A Survey ”, International Journal on Recent and Innovation Trends in Computing and Communication, Volume: 5 Issue: 8 , 2017.

[11] Jae-Hong Eom, Sung-Chun Kim, Byoung-Tak Zhang.: "AptaCDSS-E: A classifier ensemble-based clinical decision support system for cardiovascular disease level prediction, ScienceDirect", 2008

[12] Indira S. Fal Dessai.: "Intelligent Heart Disease Prediction System Using Probabilistic Neural Network", IJACTE, vol. 2, issue. 3, 2013

[13] H.Jayasree, D. S. S. K. R. T. Naren, K. Sai Sankeerth, T. Kumidini.: "HEART DISEASE PREDICTION SYSTEM", www.researchgate.net/publication/334612815, July 2019

[14] AH Chen, SY Huang, PS Hong, CH Cheng, EJ Lin.: "HDPS: Heart Disease Prediction System", IEEE, 2011

[15] Latha Parthiban and R.Subramanian.: "Intelligent Heart Disease Prediction System using CANFIS and Genetic Algorithm", International Journal of Medical and Health Sciences, Vol. 1, No. 5, 2007

[16] Jaekwon Kim, Jongsik Lee, Youngho Lee.: "Data-Mining-Based Coronary Heart Disease Risk Prediction Model Using Fuzzy Logic and Decision Tree", Health Inform Research, July 2015

[17] V. Krishnaiah, G. Narsimha, and N. Subhash Chandra.: "Heart Disease Prediction System Using Data Mining Technique by Fuzzy K-NN Approach", Springer International Publishing, vol. 1, 2015

[18] Priyan Malarvizhi Kumar, S.Lokesh, R.Varatharajan, C.Gokulnath, P.Parthasarathy.: "Cloud and IoT based Disease Prediction and Diagnosis System for Healthcare using Fuzzy Neural Classifier", Future Generation Computer Systems, vol. 86, 2018

[19] Richard P. Lippmann and David M. Shahian.: "Coronary Artery Bypass Risk Prediction Using Neural Networks", The Society of Thoracic Surgeons, 1997

[20] Bert A. Mobley, Eliot Schechter, William E. Moore, Patrick A. McKee, June E. Eichner.: "Predictions of coronary artery stenosis by artificial neural network", www.elsevier.com, 1999

[21] S. Muthukaruppan, M.J. Er.: "A hybrid particle swarm optimization based fuzzy expert system for the diagnosis of coronary artery disease", www.elsevier.com/locate/eswa, 2012

[22] G. Thippa Reddy and Neelu Khare.: "An Efficient System for Heart Disease Prediction using Hybrid OFBAT with Rule-Based Fuzzy Logic Model", World Scientific Publishing Company, Vol. 26, No. 4, 2017

[23] Jayshril S. Sonawane, D.R. Patil.: "Prediction of Heart Disease Using Multilayer Perceptron Neural Network", ICICES, 2014

[24] Miss. Chaitrali S. Dangare, Dr. Mrs. Sulabha S. Apte.: "A DATA MINING APPROACH FOR PREDICTION OF HEART DISEASE USING NEURAL NETWORKS", IJCET, vol. 3, no. 3, 2012

[25] Zi-Hui Tang, Juanmei Liu, Fangfang Zeng, Zhongtao Li, Xiaoling Yu, Linuo Zho.: "Comparison of Prediction Model for Cardiovascular Autonomic Dysfunction Using Artificial Neural Network and Logistic Regression Analysis", PLOS ONE, vol. 8, no. 8, 2013

[26] Yanwei Xing, Jie Wang, Zjijong, Yonghong Gao.: "Combination Data Mining Methods with New Medical Data to Predicting Outcome of Coronary Heart Disease", IEEE, 2007

[27] Santhana Krishnan. J, Geetha. S.: "Prediction of Heart Disease Using Machine Learning Algorithms", IEEE, 2019

[28] C. Kalaiselvi.: "Diagnosing of heart diseases using average k-nearest neighbor algorithm of data mining", IEEE, 2016

[29] Han C. W. Hsiao, Sean H. F. Chen, Jeffrey J. P.: "Deep Learning for Risk Analysis of Specific Cardiovascular Diseases Using Environmental Data and Outpatient Records", IEEE, 2016

[30] Soodeh Nikan, Femida Gwadry-Sridhar, Michael Bauer.: "Machine Learning Application to Predict the Risk of Coronary Artery Atherosclerosis", IEEE, 2016

[31] Georgia Spiridon Karanasiou, Evanthia Eleftherios Tripoliti, Theofilos Grigorios Papadopoulos, Fanis Georgios Kalatzis, Yorgos Goletsis, Katerina Kyriakos Naka, Aris Bechlioulis, Abdelhamid Errachid, Dimitrious Ioannis Fotiadis.: "Predicting adherence of patients with HF through machine learning techniques", IET, vol. 3, issue 3, 2016

[32] Gaurav Dangi, Tanupriya Choudhary, Praveen Kumar.: "A smart approach to diagnose Heart disease through machine learning and Springleaf Marketing Response", IEEE, 2016

[33] Jagdeep Singh, Amit Kamra, Harbhahg Singh.: "Prediction of heart diseases using associative classification", IEEE, 2016

[34] Seyedamin Pouriyeh, Sara Vahid, Giovanna Sannino, Giuseppe De Pietro, Hamid Arabnia, Juan Gutierrez.: "A comprehensive investigation and comparison of Machine Learning Techniques in the domain of heart disease", IEEE, 2017

[35] Yang Peilli, Yin Xuezhen, Ye Jian, Yang Lingfeng, Zhao Hui, Liang Jimin.: "Deep learning model management for coronary heart disease early warning research", IEEE, 2018

[36] G. Shanmugasundaram, V. Malar Selvam, R. Saravanan, S.Balaji.: "An Investigation of Heart Disease Prediction Techniques", IEEE, 2018

[37] Sneha Borkar, M. N. Annadate.: "Supervised Machine Learning Algorithm for Detection of Cardiac Disorders", IEEE, 2018

[38] Sanchayita Dhar, Krishna Roy, Tanuree Dey, Pritha, Datta, Ankur Biswas.: "A Hybrid Machine Learning Approach for Prediction of Heart Diseases", IEEE, 2018

[39] Md. Razu Ahmed, S M Hasan Mahmud, Md. Altab Hossin, Hosney Jahan, Shek Rashed Haider Noori.: "A Cloud Based Four-Tier Architecture for Early Detection of Heart Disease with Machine Learning Algorithms", IEEE, 2018

[40] Laiqat Ali, Awais Niamat, Javed Ali Khan, Noorbakhsm amiri Golilarz, Xiong Xingzhong, Adeeb Noor, Redhwan Nour, Syed Ahmad Chan Bukhari.: "An Optimized Stacked Support Vector Machines Based Expert System for the Effective Prediction of Heart Failure", IEEE, vol. 7, 2019

[41] Senthilkumar Mohan, Chandrasegar Thirumalai, Gautam Srivastava.: "Effective Heart Disease Prediction Using Hybrid Machine Learning Techniques", IEEE, vol. 7, 2019

[42] Frantisek Babic, Jaroslav Olejar, Zuzana Vantova, Jan Paralic.: "Predictive and descriptive analysis for heart disease diagnosis", IEEE, 2017

[43] Juan Jose Beunza, Enrique Puertas, Ester Garcia Ovejero, Gema Villalba, Emilia Condes, Gergana Koleva, Cristian Hurtado, Manuel F. Landecho.: "Comparison of machine learning algorithms for clinical event prediction (risk of coronary heart disease)", Journal Of Biomedical Informatics, vol. 97, 2019

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

[45] Samaneh Layeghian Javan, Mohammad Mehdi Sepehri, Hassan Aghajani.: "Toward analyzing and synthesizing previous research in early prediction of cardiac arrest using machine learning based on a multi-layered integrative framework", Journal Of Biomedical Informatics, vol. 88, page 70-89, 2018

[46] Evanthia Tripoliti, Theofilos G. Papadopoulos, Georgia, S. Karansiou, Katerina K. Naka.: "Heart Failure: Diagnosis, Severity Estimation And Prediction Of Adverse Events Through Machine Learning Techniques", Computational and Structural Biotechnology Journal, 2016

[47] Ashok kumar Dwivedi, “Evaluate the performance of different machine learning techniques for prediction of heart disease using ten-fold cross-validation”, Springer, 17 September 2016.

[48] Mythili T, Dev Mukherji, Nikita Padalia, and Abhiram Naidu.: "A Heart Disease Prediction Model using SVM-Decision Trees-Logistic Regression (SDL)", International Journal of Computer Applications , vol. 68, no. 16, 2013

[49] Bashar Brahmi, Mirsaeid Hosseini Shirvani, “Prediction and Diagnosis of Heart Disease by Data Mining Techniques”, Journals of Multidisciplinary Engineering Science and Technology, vol.2, 2 February 2015, pp.164168.

[50] Sairabi H.Mujawar, P.R.Devale, “Prediction of Heart Disease using Modified K-means and by using Naïve Bayes”, International Journal of Innovative research in Computer and Communication Engineering, vol.3, October 2015, pp.10265-10273.

[51] Noura Ajam, “Heart Disease Diagnoses using Artificial Neural Network”, The International Insitute of Science, Technology and Education, vol.5, No.4, 2015, pp.7-11.

[52] S.Prabhavathi, D.M.Chitra, “Analysis and Prediction of Various Heart Diseases using DNFS Techniques”, International Journal of Innovations in Scientific and Engineering Research, vol.2, 1, January 2016, pp.1-7.

[53] Jayami Patel, Prof. Tejal Upadhay, Dr. Samir Patel, “Heart disease Prediction using Machine Learning and Data mining Technique”, March 2017.