Thesis/Project No.: CSER-20-03

Heart Disease Prediction Using Machine Learning Techniques

By

Shahriar Parvej

Roll: 1507109

&

Md. Sharzul Mostafa

Roll: 1507107

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

February, 2020

Heart Disease Prediction Using Machine Learning Techniques

By

Shahriar Parvej

Roll: 1507109

&

Md. Sharzul Mostafa

Roll: 1507107

A thesis submitted in partial fulfilment of the requirements for the degree of "Bachelor of Science in Computer Science and Engineering"

Supervisor:

Dr. Pintu Chandra Shill
Professor
Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna 9203, Bangladesh

Signature

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

February, 2020

Acknowledgement

First and above all, we praise Allah, the almighty for providing us this opportunity and granting us the capability to proceed successfully. This thesis appears in its current from due to the assistants and guidance of several people. We would, therefore, like to offer our sincere thanks to all of them.

First and foremost we would like to express our special appreciation and thanks to our respected supervisor Dr. Pintu Chandra Shill, Professor, Department of Computer Science and Engineering, Khulna University of Engineering & Technology. His constant supervision, constructive criticism, valuable advice, scholarly guidance and encouragement at all stages of this research have made it possible to complete this research work.

We also acknowledge with sincere thanks to all the all-out cooperation and services rendered by the faculty members, staffs, and students of Computer Science and Engineering Department, Khulna University of Engineering & Technology. Finally, we would like to show our gratitude to our parents, without the support of whom nothing would be possible.

-Authors

Abstract

Heart disease is considered as one of the most crucial human diseases within the world and affects human life very badly. In coronary heart ailment, the heart is unable to push the required quantity of blood to the other additives of the body. Accurate and on time diagnosis of coronary heart disorder is necessary for coronary heart failure prevention and treatment. The analysis of heart ailment through traditional medical records has been considered as not reliable in many aspects. To classify the healthful people and people with coronary heart disease, non-invasive primarily based methods such as machine learning are very efficient and reliable. In the proposed study, we developed a system based on machine learning techniques for heart disease prediction by using heart disease dataset. We used seven popular machine learning algorithms, principal component analysis (PCA) as feature selection algorithm, and seven classifiers performance evaluation metrics such as classification accuracy, precision, recall and f1 score. Our proposed system can easily identify and classify the people with heart disease from healthy people. We have discussed all of the classifiers, feature selection algorithms, pre-processing methods, and classifiers performance evaluation metrics used in our work. The overall performance of the proposed prediction system has been validated for heart disease on full features and on a reduced set of features. The features reduction has an impact on classifiers performance in terms of accuracy and other performance measures including precision, recall and f1 score of the classifiers. On which best performance was obtained by the rbf kernel of support vector classifier with 86.3% classification accuracy and an f1 score of 88.4%. Linear kernel of support vector classifier, random forest, k-nearest neighbor, decision tree, artificial neural network, naïve bayes gave an accuracy of 82.5%, 85.5%, 85.04%, 78.6%, 83.3%, and 82.5% and an f1 score of 84.9%, 87.2%, 86.6%, 80.9%, 85.9%, and 84.9% respectively. The proposed decision support system based on machine learning will assist the doctors to diagnosis heart patients efficiently.

Table of Contents

Table of Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Heart Disease is considered as one of the most complex and life threatening human diseases in this modern world. This disease makes the heart unable to circulate the required amount of blood throughout the body parts to make it fulfil the normal functions of the human body. For this reason heart failure is often occurred in most of the cases. The rate of patients having heart disease is very high. The most common symptoms of heart disease are shortness of breathing, swollen feet, fatigue with related sign, and weakness in physical body. As an example, we can mention the elevated jugular venous pressure that is caused by the functional cardiac or non-cardiac abnormalities [1]. The processes in early stages to identify heart disease is very much complicated, also the resulting complexity is a major issue to be handled as it affects the standard of human life. For the developing countries, it is very difficult to handle this type of diseases as the diagnosis and treatment processes are very complex in nature. It is due to the lack of available apparatus needed for diagnose and the shortage of experienced physician, also lack of other resources that affects proper prediction and care delivery to the heart patients along with the treatment process [2]. So it is necessary to provide with an accurate and proper diagnosis of the heart disease risk in the patients to reduce their associated risks of severe heart disease issues and also improve the security of heart.

1.1 Background

The major challenge that the healthcare industry faces now-a-days is the superiority of facility. Diagnosing a disease correctly and providing with effective treatment to the patients will define the quality of service. Poor diagnosis of the disease can causes disastrous consequences that is not accepted.

Records or data of medical history is very large, but these records are from many dissimilar foundations. The interpretations and reports that are done by the physicians are essential components of these data. The data in real world might be noisy, incomplete and

inconsistent, so data preprocessing will be required in directive to fill the omitted values in the database [3].

Even if heart diseases is found as the important source of death in world in ancient years, these have been also announced as the most avoidable and manageable diseases. The proper timed judgement of this disease leads to the whole and accurate management. So an exact and methodical tool is a serious want for recognizing high-risk patients with the data for timely analysis of heart infections and heart related problems [4].

1.2 Statement of the Problem

Heart disease is a dangerous and challenging disease in today's world as many people die every day for the unconsciousness and lack of knowledge about this disease, also for being a disease which can result in silent heart attacks without any prior warning. It also results in other different types of diseases.

We need to implement a program that can predict the presence of heart disease with some given features of test data while we will train the system with several training data using a particular dataset.

1.3 Motivation

Safety of life is one of the major concerns in this century. Heart Disease is the leading cause of the death in this century as well as the major cause of disability that increases life threatening risk factors in human body. In fact, one in every four death is related to heart disease. If we are able to predict the presence within time, several heart disease can be handled effectively. Taking it as a future concern, we have implemented our work on heart disease prediction.

1.4 Objectives

The fundamental purposes of our study are given below:

- ✓ A standard dataset to be used for learning and prediction process.
- ✓ Pre-processing and dimensionality reduction for better performance.
- ✓ Use different machine learning techniques to classify presence/absence of the disease.
- ✓ Determination and prediction with higher performance than others.

1.5 Methodology

As a requisite of heart disease prediction we have gone through Machine Learning application on the dataset to predict the presence/absence of the disease based on training data. We preprocess on the dataset if there is any missing value in that dataset. Dataset is also splitted into training set and testing set in this step. Then we applied dimension reduction algorithm to reduce dataset dimension for obtaining better performance.

After going through all this processes on dataset, now we need to predict for the heart disease for the test dataset. In this case, we used several classifier technique to classify it into different classes.

1.6 Contributions

Major contributions in this thesis work are as follows:

1. We have gathered dataset from UCI Machine Learning Repository.
2. Data is pre-processed and converted into standardized form.
3. We have studied and compared some machine learning classification methods.
4. We have calculated confusion matrix, accuracy, precision, recall, f1 score as performance measure.

CHAPTER 2

Literature Review

2.1 Introduction

In recent decades, heart disease is the major cause of death globally with almost 31% [5] of all deaths worldwide. So study on this topic to predict the presence of heart disease has started from the very beginning. Many famous researches proposed different kind of methods and established lots of efficient mechanism to predict the presence of heart disease.

2.2 Related Works

In this thesis work we have worked so far on our collected dataset. There are several related works on this fields that are stated below.

There is ample related work within the fields directly associated with this paper. ANN has been delivered to produce the very best accuracy prediction inside the scientific field [6]. Back propagation multilayer perception (MLP) of ANN is used to predict heart disease. The acquired effects are as compared with the consequences of existing models inside the same domain and located to be improved [7]. The records of coronary heart ailment patients gathered from the UCI laboratory is used to find out patterns with NN, DT, Support Vector machines SVM, and Naïve Bayes. The outcomes are compared for overall performance and accuracy with those algorithms. The proposed hybrid approach returns outcomes of 86.8% for F-measure, competing with the other existing methods [8]. A large amount of information generated with the aid of the medical industry has not been used correctly previously. The new tactics presented here decrease the value and enhance the prediction of coronary heart ailment in a smooth and effective way. The various one of a kind research techniques taken into consideration on this paintings for prediction and category of heart disorder the usage of ML and deep learning (DL) techniques are highly correct in establishing the efficiency of these methods [9], [10].

Dwivedi et al. [11] implemented machine learning strategies consisting of Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbors (K-NN) with Artificial Neural Network (ANN) to predict coronary heart sickness. The narration of this implementation was assessed by using the eight measures of classification. By the use of these techniques 85% accuracy is achieved to predict heart disease that's the highest. This prediction became presented the use of LR together with sensitivity and specificity of 89% and 81 % in some respects. According to this experiments, LR outperformed than other classification algorithms.

Takci et al. [12] said in his observe that their purpose became to expect the heart attack by using applying high-quality classification methods along with the feature selection algorithms. In his study, the approach of reliefF choice and linear kernel-based totally guide vector device pair achieved better results with accuracy of 84.81% than others function selection algorithms and classification algorithms.

Shouman et al. [13] proposed Decision Tree with clustering algorithms (K-Means) for detecting heart illness. The type accuracy inside the prognosis of heart disease executed is 83.9% by using the inlier approach with two clusters.

Saini et al. [14] recommended K-Nearest Neighbors (K-NN) classifier to categorise coronary heart disease the usage of a wavelet remodel of electrocardiogram (ECG) and the wavelet transform indicators of the sixth level with the aid of the ECG. The outcomes of type efficiency have been 87.5% which was advanced and in comparison by using wavelet converted signals.

Yan et al. [15] proposed a decision support system for the dedication of heart contamination executing a Multilayer Perceptron (MLP) formation of input, hidden and output layers. The experiment done to reach 63.6%-82.9% of type accuracy which is considered as a first-rate decision support machine.

2.3 Scope of the Research

Naïve Bayes, Decision Tree, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbor (K-NN), Artificial Neural Network (ANN) classifiers are used to predict heart disease. These methods are used to predict the presence of heart disease from various attributes. But they all have a different accuracy in their model with the same data-set. In some model, the heart disease will occur but there actually don't have the disease. Similarly, in some model, the heart disease will not occur but there actually have heart disease. It is a problem if there exists only one model.

We have used these models to find accurate result from them. And we differentiated between them to find which model is the best.

CHAPTER 3

Theoretical Consideration

3.1 Principal Component Analysis

Principal Component Analysis (PCA) is one of the most common technique for feature reduction. It is used alone sometimes, and also as a starting point for other methods of reduction of measurements. PCA is a projection-based approach that converts data into a sequence of orthogonal axes by projecting it [16].

Obviously, it is expensive for accuracy to reduce the number of variables in the dataset, but the trick in reducing dimensionality is to compensate for simplicity with some accuracy. Since smaller data sets make data processing for machine learning algorithm without extraneous variables much easier and more quickly to explore and visualize.

To summarize, the PCA concept is easy – decreasing the number of variables in a dataset, while retaining the highest possible amount of details. The procedure of the technique is following:

1. **Standardization:** The main aims of the step is to standardize the range of continuous initial variables so that each of them is equally involved in the analysis.

   More precisely, the reason why standardization is important before PCA is because the latter is very sensitive to the variances in the initial variables. In other words, if the distribution of initial variables is very wide, those variables with large ranges will overpower those with limited ranges, that will give the biased result.

   This can be done mathematically by subtracting the mean and dividing by standard deviation for each value for every variables

$$z = \frac{value - mean}{standard\ deviation} \tag{1}$$

   All variables will be converted into the same scale until standardization is reached.

2. **Covariance Matrix computation:** The purpose of this stage is to understand the difference between the variables in the input data set or in other words whether there is a relation between them. Since variables are sometimes highly correlated to produce redundant information. Thus, we calculate the covariance matrix to describe such correlations.

   The covariance matrix is a p×p symmetric matrix (where p is the number of dimensions) which has the covariances associated with all possible pairs of the initial variables as inputs. For example, for a 3-dimensional data set of 3 variables x, y, and z, the covariance matrix is a 3×3 matrix of the following:

$$\begin{bmatrix} \text{cov}(x,x) & \text{cov}(x,y) & \text{cov}(x,z) \\ \text{cov}(y,x) & \text{cov}(y,y) & \text{cov}(y,z) \\ \text{cov}(z,x) & \text{cov}(z,y) & \text{cov}(z,z) \end{bmatrix}$$

Covariance matrix for 3-dimensional data

   Now, that we know the covariance matrix is only a table that summarizes correlations amongst all possible pairs of variables, let's proceed to the next step.

3. **Compute the eigenvectors and eigenvalues of the covariance matrix to identify the principal components:** For determining the principle component, we calculated the eigenvalues and eigenvectors from the covariance matrix.
4. **Feature vector:** In this step, we chose to preserve or discard all of those components of lesser importance, and create a matrix of vectors that we call feature-vectors with the others. The vector is essentially a matrix with the eigenvectors of the elements that we want to preserve as columns
5. **Recast the data along principal component axes:** In this last step it is necessary to reorient data from the original axis to the one of the principal parameters Components using the eigenvector generated by the feature vectors of the covariance matrix. The transformation of the original data can be achieved by multiplying the transpose of the original data set by the transpose of the feature vector.

$$FinalDataSet = FeatureVector^T \times StandarizedOriginalDataset^T \qquad (2)$$

   After the steps have been completed we have done the principal component analysis with a reduced dimensional data set.

3.2 Classification using Support Vector Machine

SVM offers very high accuracy in comparison to other classifiers which includes logistic regression, and selection trees. It is thought for its kernel trick to deal with nonlinear input spaces. It is used in numerous applications together with face detection, intrusion detection, class of emails, information articles and web pages, type of genes, and handwriting recognition.

Support Vector Machines is considered to be a classification approach, but it may be employed in both forms of classification and regression problems. It can easily take care of multiple non-stop and categorical variables. SVM constructs a hyperplane in multidimensional space to separate distinctive classes. SVM generates surest hyperplane in an iterative manner that is used to reduce an error. The core concept of SVM is to find a most marginal hyperplane (MMH) that great divides the dataset into classes [17].

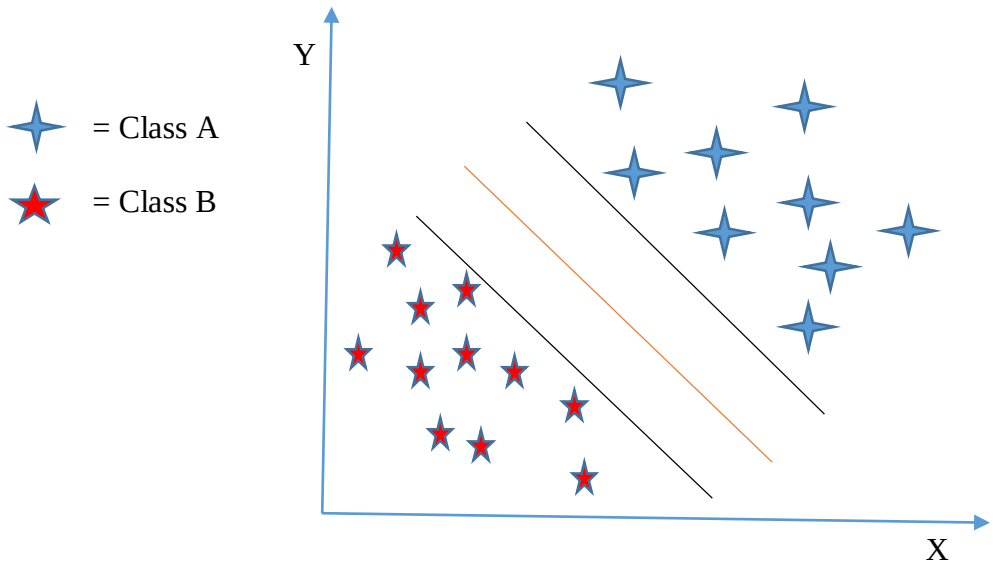


Fig 3.1: Classified data points using SVM

In the above stated way, support vector classifier is used to classify the resulting output prediction of our test records based on training information.

3.3 Classification using Decision Tree

Decision trees are very commonly used predictive models that predict the value of a target based on several input variables. The tree is built by splitting the source set consisting of the root node of the tree into subsets consisting of the successor children. The splitting is based on various splitting rules such as: Gini impurity, Information gain, etc. [18]. This process is repeated on each derived subset recursively until a given depth or until leaf nodes are found.

For a dataset containing *J* classes, Gini impurity is calculated as follows:

$$I_G(p) = 1 - \sum_{i=1}^{J} P_i^2 \tag{3}$$

Where $i \in \{1, 2, ..., J\}$ and $\boldsymbol{P_i}$ is the fraction of items labeled with class $\boldsymbol{i}$ in the dataset.

Information gain is based on entropy. Entropy is defined as follows:

$$H(T) = I_E\left(p_1, p_2, ..., p_j\right) = -\sum_{i=1}^{J} p_i \log_2 p_i \tag{4}$$

Here $p_1 + p_2 + ... + p_j = 1$. Information gain can be represented as follows:

$$IG(T|a) = H(T) - H(T|a) = -\sum_{i=1}^{J} p_i \log_2 p_i - \sum_{i=1}^{J} -Pr(i|a) \log_2 Pr(i|a) \tag{5}$$

Information gain is thusly used to decide which feature to use as a basis for splitting. The best feature is the one that provides the most information gain.

Decision trees tend to overfit to training data. It is evident from the algorithm that, if no limit is set on the depth, the tree will find route for all the items in the dataset.

3.4 Classification using Random Forest

Random forest are ensemble learning methods used extensively for various tasks such as: classification, regression, etc. Random forests are generated by constructing numerous decision trees each working with a random subsample of the entire dataset. The final output is generated by taking the mode of all the classes of the individual trees in case of classification [19].

Decision trees are very commonly used predictive models that predict the value of a target based on several input variables. The tree is built by splitting the source set consisting of the root node of the tree into subsets consisting of the successor children. The splitting is based on various splitting rules such as: Gini impurity, Information gain, etc. [18]. This process is repeated on each derived subset recursively until a given depth or until leaf nodes are found.

For a dataset containing J classes, Gini impurity is calculated as follows:

$$I_G(p) = 1 - \sum_{i=1}^{J} P_i^2 \tag{6}$$

Where $i \in \{1, 2, ..., J\}$ and $\boldsymbol{P_i}$ is the fraction of items labeled with class $\boldsymbol{i}$ in the dataset.

Information gain is based on entropy. Entropy is defined as follows:

$$H(T) = I_E(p_1, p_2, ..., p_j) = -\sum_{i=1}^{J} p_i \log_2 p_i \tag{7}$$

Here $p_1 + p_2 + ... + p_j = 1$. Information gain can be represented as follows:

$$IG(T|a) = H(T) - H(T|a) = -\sum_{i=1}^{J} p_i \log_2 p_i - \sum_{i=1}^{J} -Pr(i|a) \log_2 Pr(i|a) \tag{8}$$

Information gain is thusly used to decide which feature to use as a basis for splitting. The best feature is the one that provides the most information gain.

Decision trees tend to overfit to training data. It is evident from the algorithm that, if no limit is set on the depth, the tree will find route for all the items in the dataset.

Random forest overcomes this by providing randomly sampled dataset to each decision tree. Random forests can further battle the overfitting problem by limiting the maximum number of trees used or the depth of the trees [19].

3.5 Classification using Naïve Bayes

A Naïve Bayes classifier is a probabilistic system learning model that's used for classification task. The crux of the classifier is based totally at the Bayes theorem.

Bayes Theorem:

$$P(A\,|B) = \frac{P(B\,|A)\,P(A)}{P(B)} \tag{9}$$

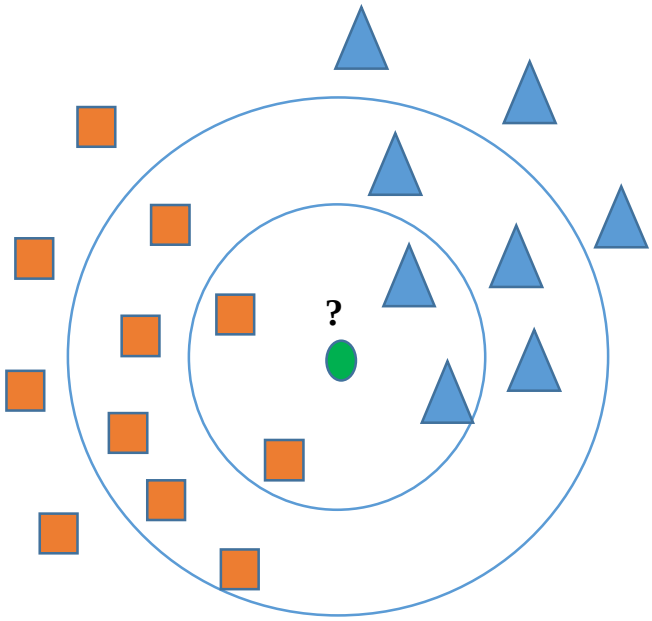
Using Bayes theorem, we can calculate the possibility of 'A' happening, given that 'B' has occurred. Here, 'B' is the evidence and 'A' is the hypothesis. The assumption made right here is that the predictors/functions are independent. That is presence of one particular characteristic does not affect the other. Hence it is known as naïve.

Naïve Bayes algorithms are in the main used in sentiment analysis, spam filtering, recommendation systems etc. They are speedy and easy to implement however their biggest downside is that the requirement of predictors to be independent. In maximum of the real existence cases, the predictors are dependent, this hinders the performance of the classifier.

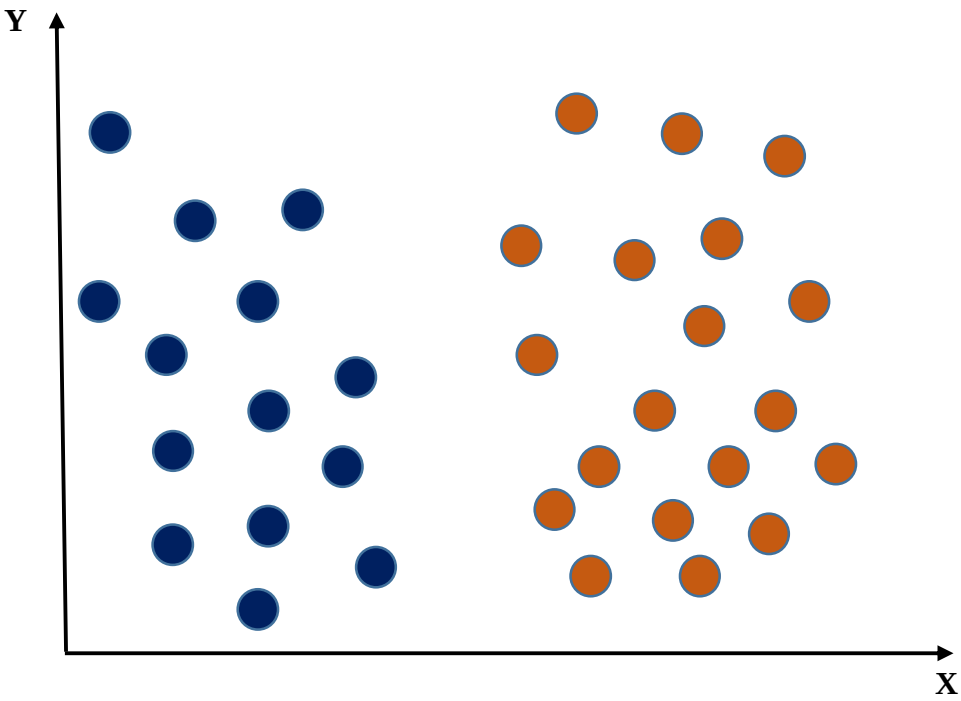
Naïve Bayes algorithms are mostly used in sentiment analysis, spam filtering, recommendation systems etc. They are fast and easy to implement but their biggest disadvantage is that the requirement of predictors to be independent. In most of the real life cases, the predictors are dependent, this hinders the performance of the classifier [20].

3.6 Classification using K-NN

In our adopted method, we have used k - nearest neighbor algorithm to predict the disease. The k - nearest neighbor algorithm (K-NN) is a classification technique which classify the objects base on training features space. It is the simplest classification technique because the computations are simple. The classification of objects based on votes of its neighbors which represented by k. In K-NN, object is classified to a particular class which has majority of votes [21]. Figure 3.2(a) shows the K-NN classification diagram.



(a): K-NN classification diagram



(b): Feature Space

Fig 3.2: K-NN classification diagram and feature space

Multiple features for each class are brought to the training function, so that the program can store the training data into its database. The training data are stored in a feature space. A feature space is a dimensional vector of numerical features that represents a certain object. These objects have their respective coordinates or positions in the feature space according to their features. Their positions scatter around the feature space, but objects which belong in the same class are usually grouped near to each other because they own similar features.

Figure 3.2(b) shows a picture of a feature space with two classes: class A and class B. In this figure, objects of class A are marked with red dots and objects of class B are marked with blue dots. Each class has their own unique features which distinguish them from other classes. These features are what determine each object's position in the feature space.

3.7 Classification using ANN

Artificial Neural Networks (ANN) are multi-layer fully-connected neural nets that appear to be the figure below. They include an input layer, a couple of hidden layers, and an output layer. Every node in one layer is hooked up to every different node inside the next layer. We make the network deeper by growing the quantity of hidden layers [22].

A single processing node of ANN can be shown as following:

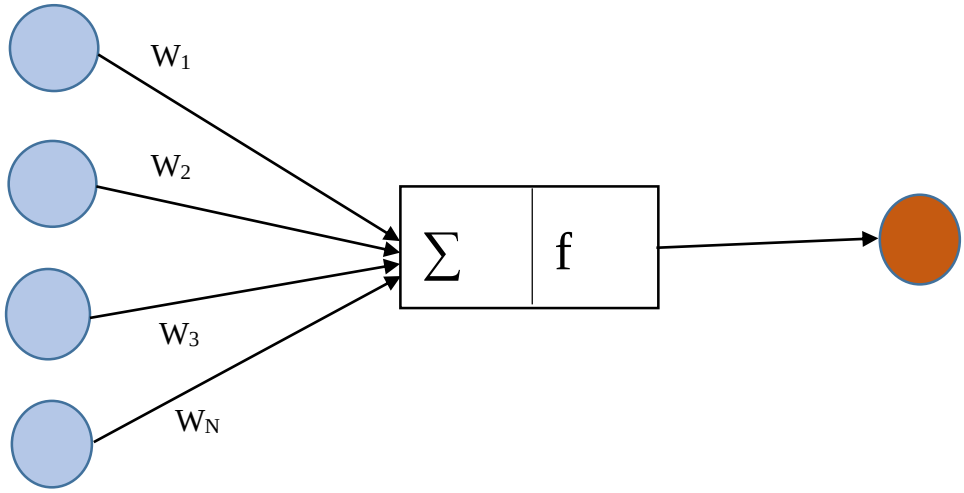


Fig 3.3: A node of ANN for classifying data

Sigmoid Activation

So as to delineate anticipated values to probabilities, we utilize the sigmoid predictions. The characteristic maps any proper incentive into another incentive somewhere inside the range of 0 and 1. In gadget learning, we make use of sigmoid to outline to probabilities.

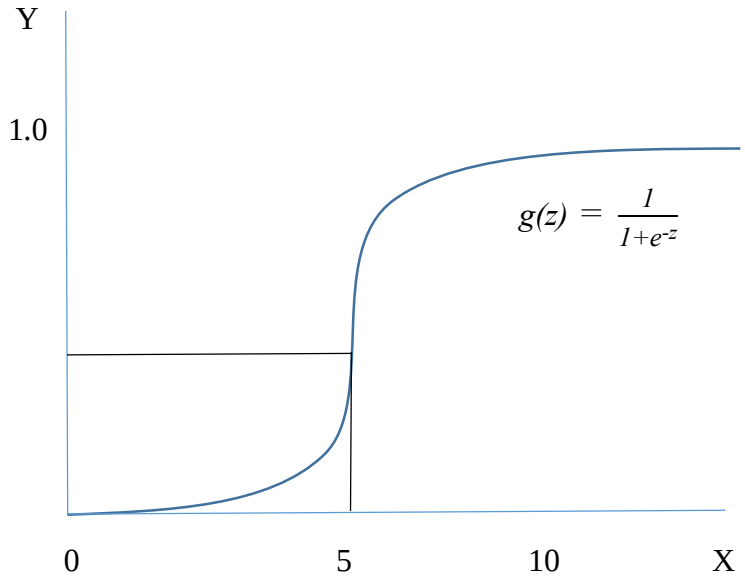


Fig 3.4: Sigmoid curve

By invoking the sigmoid function, we get the likelihood that some data x has an area with class 1 or some have an area with class 0. Take all probabilities ≥ 0.5 = class 1 and all probabilities < 0.5 = class 0. This limit must be characterised by depending upon the issue we have been working. By doing this we get the respective prediction of Heart Disease.

3.8 Performance Evaluation Measures

To evaluate the performance of the classifiers, we completed some experiments that give the measurements of performances regarding accuracy, precision, recall and f1-score. A confusion matrix contains information approximately actual and predicted classifications done by a classification system. Considering the following Confusion Matrix:

Table 3.1: Confusion matrix

	Class 1 Predicted	Class 2 Predicted
Class 1 Actual	TP	FN
Class 2 Actual	FP	TN

Here, Class 1 denotes positive class and Class 2 denotes negative class. And matrix contains information about the count of true positive (TP), true negative (TN), false positive (FP) and false negative (FN). True positive denotes both the observation and the predicted classes to be positive where true negative denotes both the observation and the predicted classes to be negative. False positive denotes that the observation is negative, but is predicted to be positive where false negative denotes that the observation is positive, but is predicted to be negative.

Classification Rate/Accuracy: Accuracy is the percentage of test tuples that are correctly classified by the classifier. It is calculated by the equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{10}$$

However, there are problems with accuracy. It considers equal costs for both kinds of errors. An accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem.

Precision: Precision is the measure obtained by dividing total number of correctly classified positive examples by the total number of positive examples that are predicted by the system. It is actually a measure of exactness of the system. High Precision indicates that an example which is labelled as positive is indeed positive. Precision is calculated by the relation:

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

Recall: Recall is defined as the ratio of the total number of positive examples that are correctly classified and the total number of actual positive examples. It denotes the correctness of the system. High recall indicates that the class is more correctly recognized. Recall is calculated by the relation:

$$Recall = \frac{TP}{TP + FN} \tag{12}$$

Condition will arise with high recall, low precision and low recall, high precision. In this case high recall, low precision means that most of the positive instances are correctly recognized with a low false negative but there are a lot of false positives. On the other hand, low recall, high precision shows that though we miss a lot of positive examples having high false negative measure but those we predicted as positive are indeed positive and have a low false positive measure.

F-measure: We considered precision and recall to measure the performance of a classifier. The F-measure in this case helps to have a measurement that represents both of them. We calculate an F-measure that uses harmonic mean instead of arithmetic mean as it punishes the extreme values more. The F-measure always remains nearer to the smaller value of precision or recall.

$$F\text{-}measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{13}$$

CHAPTER 4

Methodology

4.1 Proposed Method

There are various techniques for prediction of heart disease. Each method has both the advantages and disadvantages in the prediction process. The proposed technique uses Support Vector Classifier, K-Nearest Neighbours (K-NN) Algorithm, Naïve Bayes, Artificial Neural Networks (ANN) for prediction of heart disease. For preprocessing of data we used imputation with mean strategy to fill the missing values. In our proposed work we used feature reduction technique to reduce number of attributes and keep only attributes which contribute more towards the diagnosis of heart disease. There are several existing dataset. In our work, we have worked on dataset collected from UCI machine learning repository.

4.2 Workflow Diagram

We can represent the entire work according to the workflow diagram of Fig 4.1.

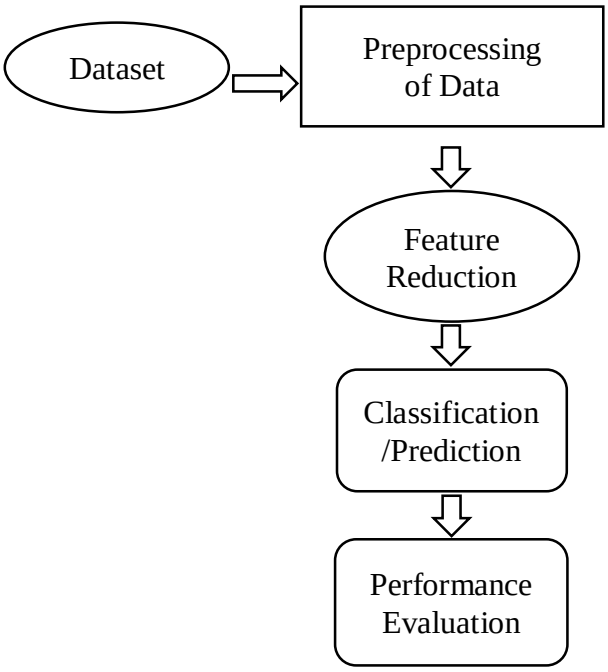


Fig 4.1: Workflow diagram of heart disease prediction system

The workflow diagram represents the entire work with a graphical view. According to the representation the system is based on a particular dataset for Heart Disease to work on.

In pre-processing stage we used a method to find out the missing values and filled them up using a renowned strategy. It then brings a completeness in the dataset to be worked on the next stage of the implementation process.

In case of the step of feature selection, we have applied dimensionality reduction process using PCA, taking a number of major components to work on further to get better accuracy and performance in the prediction of heart disease.

After preprocessing and reducing the dimensionality of the dataset, we now classified the absence/presence of heart disease from the dataset. There are several classifier techniques that can be used in this process.

Final strategy is to get the classification result in prediction stage on test data and evaluate the performance, also try to improve the system if there is any lacking or scope to work on future.

4.3 Design of the System

In this portion of our thesis we are going to discuss how we prepared or designed the whole system. In terms of how we executed the system it will be discussed later in the report.

4.4 Dataset

We collected the dataset from UCI machine learning repository that has been used in our work [23]. The dataset that we used in our thesis has in total 14 attributes. First 13 of those attributes are the features that we will be using later on in order to predict the final column ‘diagnosis’ which will tell us if the patient is going to be affected by heart disease or not.

Dataset features are listed in Table 4.1

Table 4.1: Definitions of the features

Features	Definitions
f1 : age	Age of the patient in years
f2 : sex	Gender of the patient
f3 : cp	Chest pain type
f4 : trestbps	Resting blood pressure
f5 : chol	serum cholestoral in mg/dl
f6 : fbs	fasting blood sugar > 120 mg/dl
f7 : restecg	resting electrocardiographic results
f8 : thalach	maximum heart rate achieved
f9 : exang	Angina induced for exercise
f10 : oldpeak	ST depression induced by exercise relative to rest
f11 : slope	the slope of the peak exercise ST segment
f12 : ca	number of major vessels (0-3) colored by fluoroscopy
f13 : thal	Thalassemia 3 = normal; 6 = fixed defect; 7 = reversible defect
f14 : target	Presence(1) or absence(0) of the disease

For better performance it is needed to work with a dataset where the target classes are of approximately equal in size. We checked for the unique target class count in our dataset.

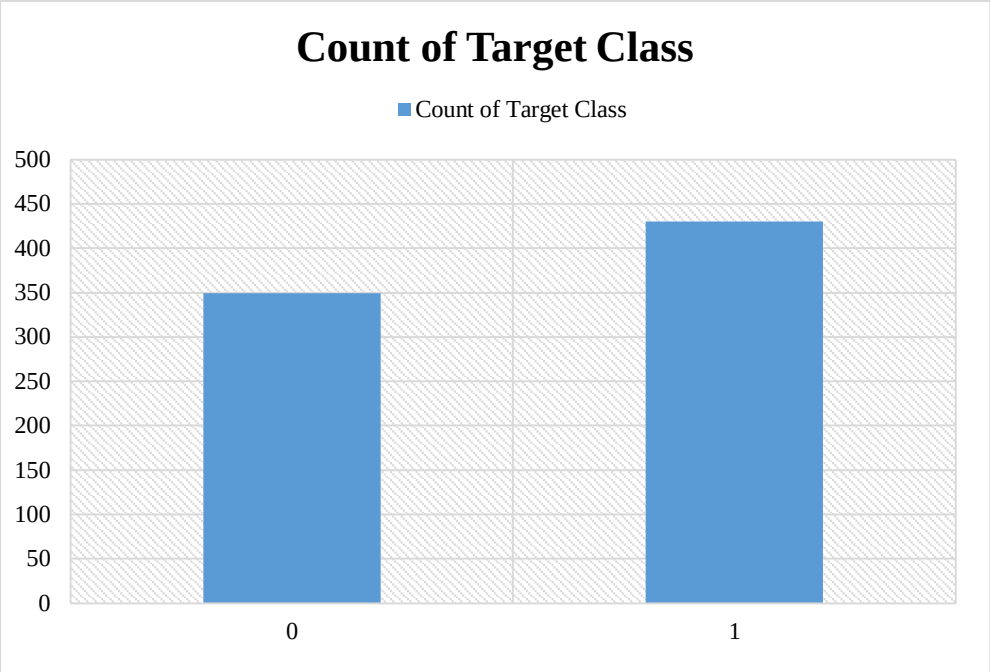


Fig 4.2: Target class count

The two classes are not exactly equal in count but the ratio is good enough to continue without dropping/increasing any data.

4.5 Preprocessing

Data preprocessing is a data processing technique that involves remodelling data into a clear format. Real world data is often incomplete, inconsistent, noisy and lacking in certain behaviours or trends and is likely to contain many errors. We standardized the data in the first phase to make all the data fall between 0 and 1. Then we applied imputation technique to fill the missing values for better performance. For this purpose we used the standard scaler from the sklearn library to standardize the data. And then we used imputer from the sklearn library to fill the missing data. This works by calculating the mean of non -missing attribute values and filling the missing values with this mean value. By this technique we preprocessed our dataset to be used for more efficient classification with better performance.

4.6 Load and Split the Dataset

We created an array called dataset. Then we read the csv file also known as the dataset file and stored the dataset on that array. There are 779 instances in our dataset. We used seventy percent (70%) of our dataset for training our model and the rest thirty percent (30%) are for testing the dataset to measure the performance of the classifier.

4.7 Feature Selection

Feature selection is the process of reducing the number of input variables when developing a predictive model. It is desirable to reduce the number of input variables to both reduce the computational cost of modelling and, in some cases, to improve the performance of the model. We adopted PCA to reduce number of attributes and keep only attributes which contribute more towards the diagnosis of heart disease.

4.8 Modelling and Predicting with Machine Learning

The main goal of the entire work is to predict heart disease occurrence with the highest accuracy. In order to achieve this we will test seven classification algorithms including SVM rbf and linear kernel, K-NN, ANN, Random Forest, Naïve Bayes, and Decision Tree. This section includes all results obtained from the study and introduces the best performer according to the accuracy metric. We have chosen these algorithms for solving supervised learning problems throughout classification methods. We trained the models with the training dataset that we generated from splitting the original dataset. Then we predicted the target class of testing instances from testing set by their features and finally measured the performance with the help of target class of the testing dataset.

4.9 Finding the Result

At the end we are going to create a summery table where we are going to show the different accuracy percentage of different algorithms. We will also show the other performance measure scores including precision, recall and f-measure. We will also plot the resulting values to a chart to make it easier for visualization. Finally we made a chart for comparing the accuracy scores that we got before applying feature reduction and after our approach. More details about the results will be discussed in Chapter 5 (Simulation Results).

CHAPTER 5

Simulation Results

In our previous chapters we have discussed about different algorithms, previous works in this field and the dataset we used for our experiments. All those were the foundation for this chapter. In this chapter, we discussed about results that we found after implementing the algorithms and analysed them.

5.1 Tuning Parameters of Implemented Algorithms

5.1.1: Support Vector Classifier

Support Vector Classifier uses kernel technique for classification process. Different kernel works better for different classifications. In our experiment we have used linear kernel and rbf kernel as the tuning parameters.

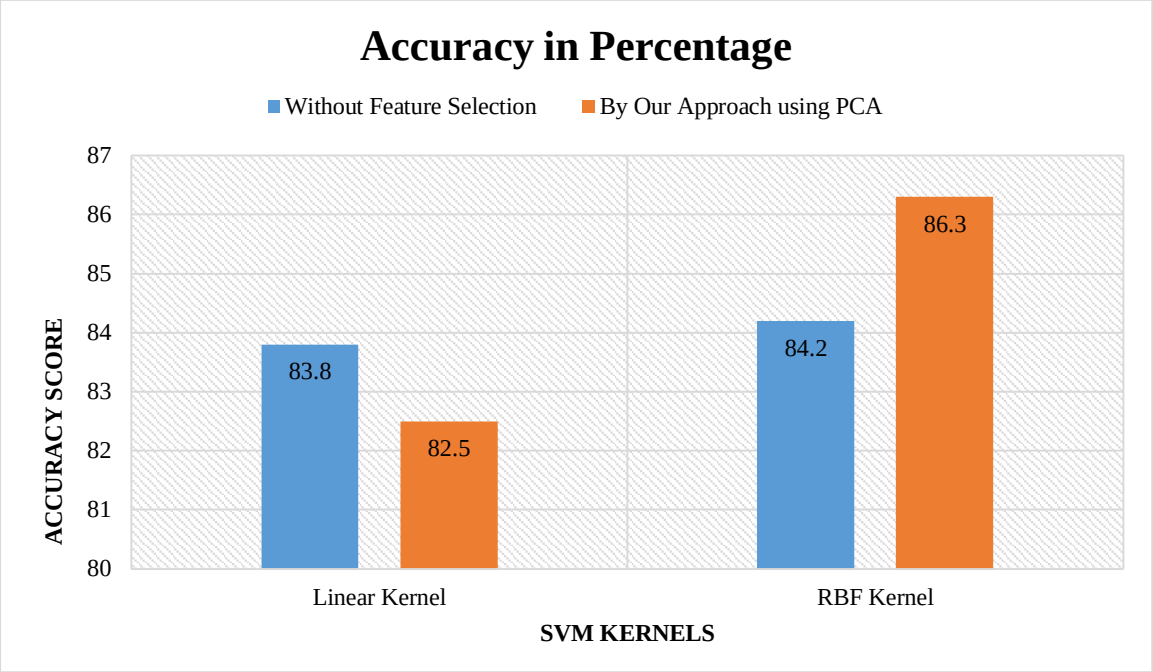


Fig 5.1: Accuracy measure for support vector classifier with different kernels

Fig 5.1 shows the accuracy obtained by the kernels without feature selection and by our approach using PCA as feature selection method and preprocessing the data to fill the missing values. Then we obtain the best accuracy of 86.3% by the rbf kernel of Support Vector Classifier.

5.1.2: Random Forest Classifier

The Random Forest classifier is an ensemble method of the decision trees to act like a forest. Each decision tree predicts the output class and a majority voting technique is applied to predict the final output class. In the chart of Fig 5.1.2, n is the number of estimators and accuracy is plotted in percentage.

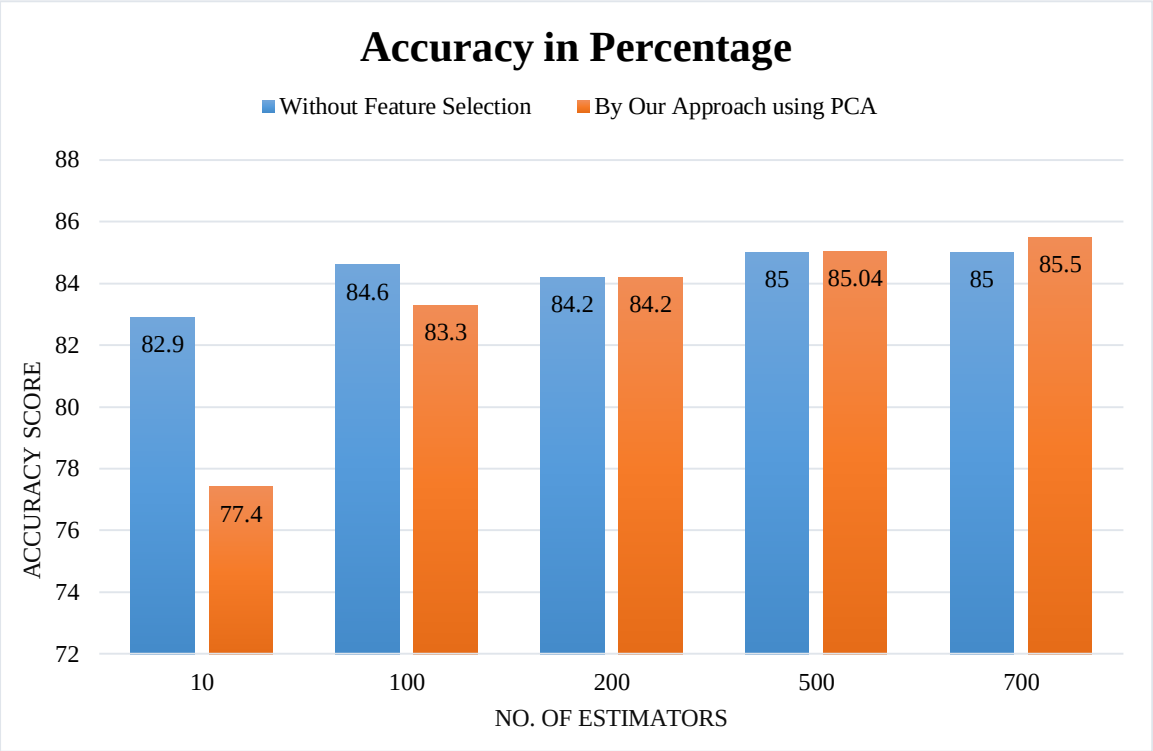


Fig 5.2: Accuracy measure for random forest for different number of estimators

Fig 5.2 shows that for 700 estimators we get the highest accuracy of 85.5% for random forest classifier by our approach using PCA as feature selection method and preprocessing the data to fill the missing values.

5.1.3: K-NN Classifier

In K-NN classifier, the number of neighbors, n in used to predict the class for the testing data. So the parameter n works as the tuning parameter of K-NN. The chart in Fig 5.1.3 shows the accuracy for different number of neighbors for the prediction system.

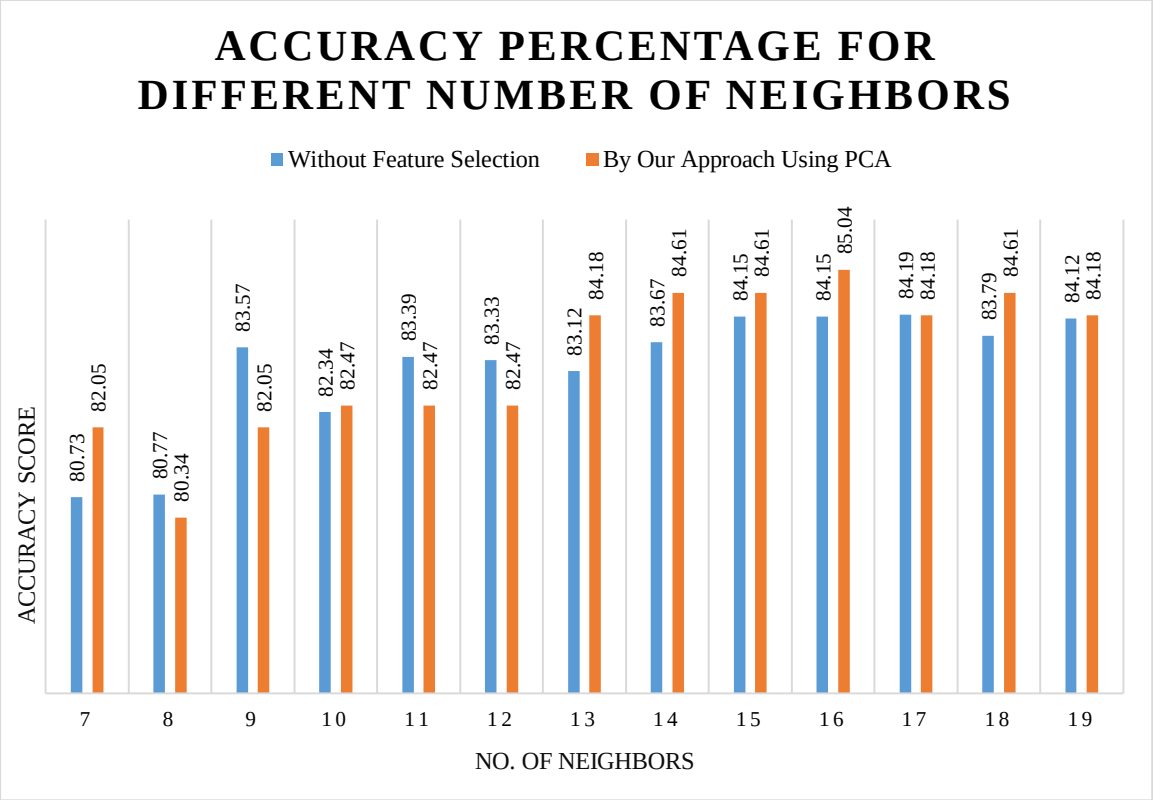


Fig 5.3: Accuracy measure for K-NN classifier for different number of neighbors

According to Fig 5.3, the K-NN classifier gives the highest accuracy of 85.04% for n=16 by our approach using PCA as feature selection method and preprocessing the data to fill the missing values.

5.1.4: ANN Classifier

In ANN classifier, the output depends on the number of hidden layers and total number of neurons in it. In our model the input dimension is 11. So there are 11 input neurons. There are two hidden layers having 10 and 6 neurons in each layers respectively.

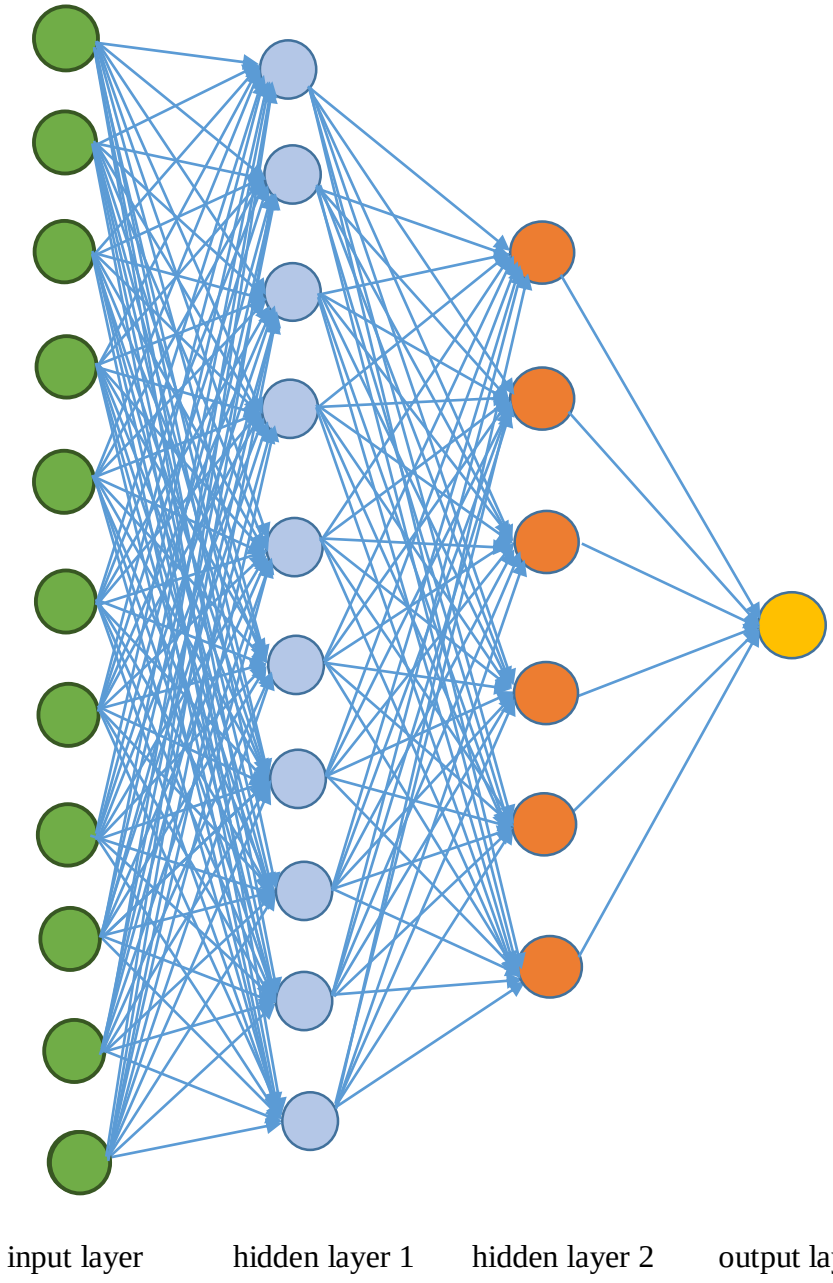


Fig 5.4: Adopted ANN architecture

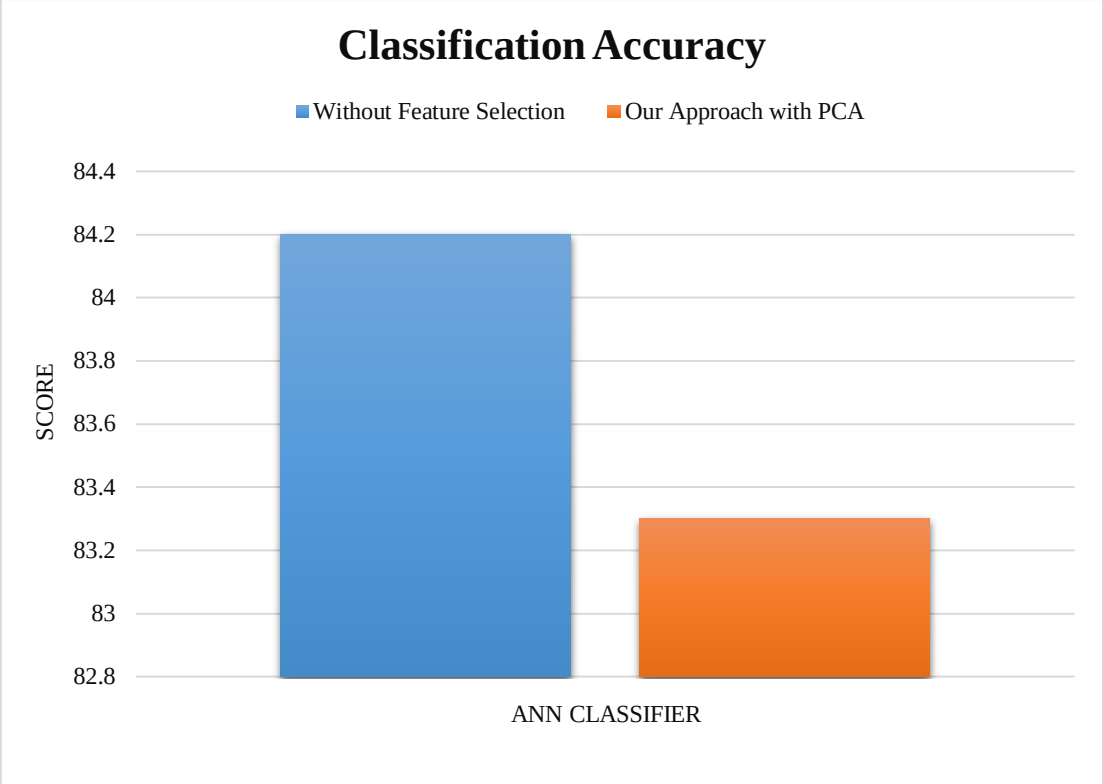


Fig 5.5: Accuracy measure for ANN classifier

Fig 5.4 shows the ANN architecture used in our work. The network gave an accuracy of 83.3% on our heart disease dataset with our approach. For the same network without the feature selection, the accuracy was 84.2%.

5.2 Performance Measure of Models

In our experiment we have used the whole dataset with reduced features and applied Support Vector Machine, Decision Tree, Random Forest, K-Nearest Neighbor, Artificial Neural Network, and Naïve Bayes classifiers as classification algorithms. Table 5.2.1 contains accuracy, precision, recall and F1 score of the different algorithms that we applied on our dataset which gives an idea of performance measure of the models.

In this case, the full features of the dataset were checked on linear and rbf kernel of SVM, Random Forest, K-NN, Decision Tree, ANN, and Naïve Bayes classifiers. Of the dataset 70% was used for training the classifiers and 30% was tested. Before proceeding to classification, we preprocessed the dataset and applied feature selection technique on it. Table 5.1 describes the results of these seven classifiers.

Table 5.1: Performance measure

Classifier	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
SVM (Linear)	82.5	83.9	85.8	84.9
SVM (RBF)	86.3	85.9	91.04	88.4
Random Forest	85.5	87.9	86.6	87.2
K-NN	85.04	88.98	84.3	86.6
Decision Tree	78.6	82.8	79.1	80.9
ANN	83.3	85.2	85.8	85.5
Naïve Bayes	82.5	86.5	85.8	84.9

From Table 5.1, the RBF kernel of SVM classifier is showing good performance that has 86.3% classification accuracy, 85.9% precision, 91.04% recall and an f1 score of 88.4%.

The classification accuracy for other classifiers are also observed: for SVM linear kernel accuracy is 82.5%, for Random Forest accuracy is 85.5%, for K-NN accuracy is 85.04%, for Decision Tree accuracy is 78.6%, for ANN accuracy is 83.3%, for Naïve Bayes accuracy is 82.5%. So in terms of accuracy the second best result was obtained by the Random Forest classifier.

Considering the f1 score of the classifier that implies both precision and recall, the best performance is obtained by rbf kernel of SVM classifier that is 88.4%. F1 score of SVM linear kernel is 84.9%, Random Forest is 87.2%, K-NN is 86.6%, Decision Tree is 80.9%, ANN is 85.5%, Naïve Bayes is 84.9%.

Fig 5.5 shows the graphical representation of the Table 5.1

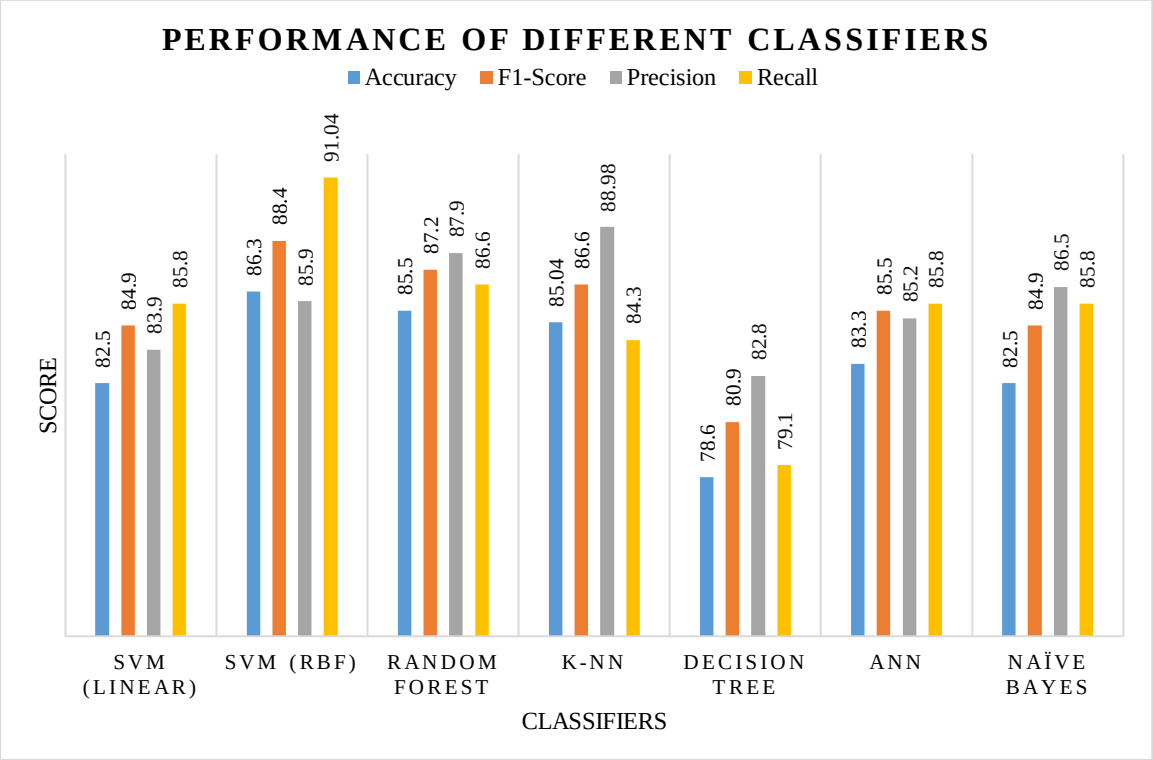


Fig 5.6: Performance of different classifiers

As shown in Figure 5.5, the performance of SVM rbf kernel outperformed to the other six classifiers in terms of accuracy and f1 score that includes both precision and recall. The predictive accuracy of SVM rbf kernel was 86.3%. The second important classifier was Random Forest classifier which has the classification accuracy of 85.5%. The worst performance was observed for Decision Tree out of all the classifiers in terms of accuracy and f1 score which were 78.6.% and 80.9%, respectively.

So we get the observation of performance measure following our approach after data preprocessing and feature selection technique. For feature selection we adopted PCA. Compared to the approach without feature selection we get the higher accuracy. Figure 5.6 shows the accuracy score comparison between the two approaches.

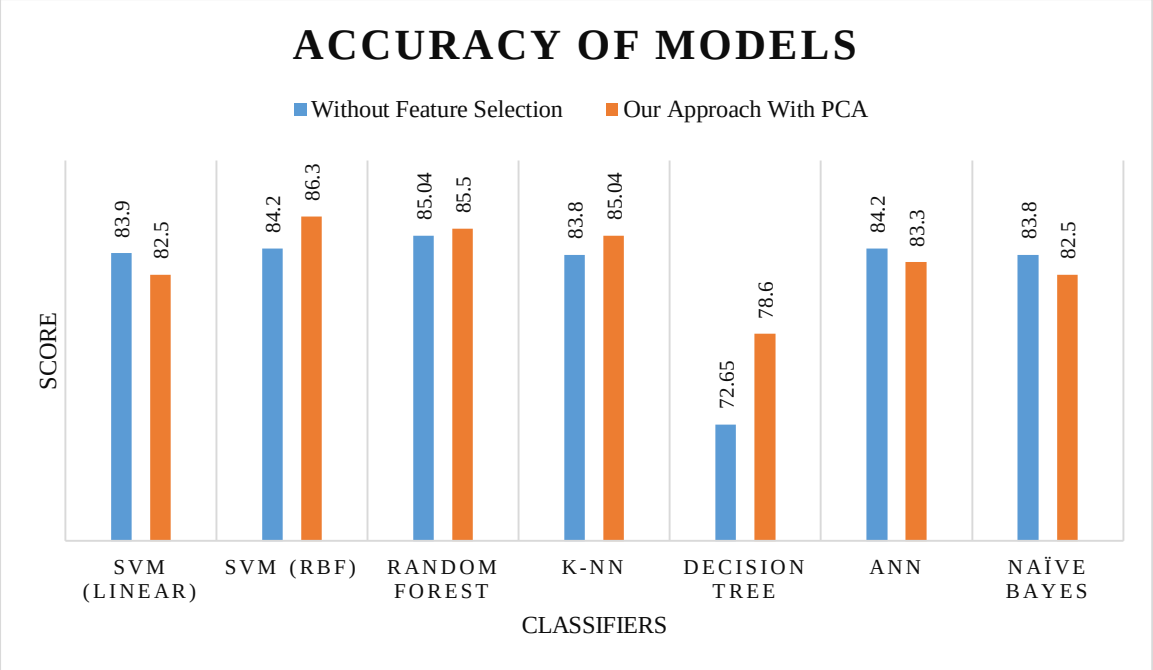


Fig 5.7: Accuracy score comparison

From the chart of Fig 5.6 we observe that without feature selection we get the best accuracy by the Random Forest Classifier that is 85.04%. Going through our approach we have been able to improve the accuracy of SVM rbf kernel, Random Forest, Decision Tree, and K-NN classifiers. Accuracy increase of SVM rbf kernel is the highest compared to the other classifiers. In this case we get the highest accuracy by rbf kernel of SVM classifier that is 86.3%.

CHAPTER 6

6.1 Discussion

Through the entire work of the prediction system on heart dataset, using Random Forest classifier, the resulting performance from the test dataset gave an accuracy of 85.04% before applying feature selection method. After applying PCA as feature selection method the accuracy increased to 85.5% for the Random Forest classifier. We got increased accuracy also for the RBF kernel of SVM, Decision Tree and K-NN classifiers. The accuracy score of RBF kernel of Support Vector Classifier gains the best accuracy of 86.3% from 84.2% after our approach which provides better results in the diagnosis of heart disease and better accuracy as compared to other existing work on this field. On the other hand accuracy increase for the K-NN was 85.04% from 83.8% and for Decision Tree is 78.6% from 72.65% each which is also better than previous approaches for classifying the presence of heart disease for this dataset.

6.2 Conclusion

The thesis work is successfully completed with the dataset using several machine learning approaches. In this case we presented an efficient approach for prediction of heart disease with increasing accuracy using Random Forest, K-Nearest Neighbor, Decision Tree and RBF kernel of Support Vector Classifier. With the help of imputing technique we preprocessed the data to fill the missing values to achieve better performance. We adopted Principal Component Analysis as feature selection technique for heart disease classification. Feature selection technique improves the classification accuracy. Our proposed approach achieved the best accuracy of 86.3% for the data set using RBF kernel of Support Vector Classifier as the classification method. We will be working on further improvement on this work for a much better performance and higher accuracy in future.

References

[1] https://towardsdatascience.com/heart-disease-prediction-73468d630cfc, accessed on: Feb. 23, 2020.

[2] Haq, A. U., Li, J. P., Memon, M. H., Nazir, S., & Sun, R. (2018). “A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms.” Mobile Information Systems, vol. 23, pp. 1–21, 2018.

[3] A. Gupta, R. Kumar, H. Singh Arora and B. Raman, "MIFH: A Machine Intelligence Framework for Heart Disease Diagnosis," in IEEE Access, vol. 8, pp. 14659-14674, 2020.

[4] Yanwei, X.; Wang, J.; Zhao, Z.; Gao, Y., “Combination data mining models with new medical data to predict outcome of coronary heart disease”. Proceedings International Conference on Convergence Information Technology 2007, pp. 868 – 872.

[5] https://www.who.int/health-topics/cardiovascular-diseases/#tab=tab_1, accessed on: Feb. 23, 2020.

[6] L. Baccour, ‘‘Amended fused TOPSIS-VIKOR for classification (ATOVIC) applied to some UCI data sets,’’ Expert Syst. Appl., vol. 99, pp. 115–125, Jun. 2018.

[7] R. Das, I. Turkoglu, and A. Sengur, ‘‘Effective diagnosis of heart disease through neural networks ensembles,’’ Expert Syst. Appl., vol. 36, no. 4, pp. 7675–7680, May 2009.

[8] C.-A. Cheng and H.-W. Chiu, ‘‘An artificial neural network model for the evaluation of carotid artery stenting prognosis using a national-wide database,’’ in Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC), Jul. 2017, pp. 2566–2569.

[9] D. K. Ravish, K. J. Shanthi, N. R. Shenoy, and S. Nisargh, ‘‘Heart function monitoring, prediction and prevention of heart attacks: Using artificial neural networks,’’ in Proc. Int. Conf. Contemp. Comput. Inform. (IC3I), Nov. 2014, pp. 1–6.

[10] W. Zhang and J. Han, ‘‘Towards heart sound classification without segmentation using convolutional neural network,’’ in Proc. Comput. Cardiol. (CinC), Sep. 2017, pp. 1–4.

[11] Dwivedi, A. K,“Performance evaluation of different machine learning techniques for prediction of heart disease.” in Neural Computing and Applications, vol. 29, pp. 685–693, Feb. 2016.

[12] H. Takci, “Improvement of heart attack prediction by the feature selection methods,” Turkish Journal Of Electrical Engineering & Computer Sciences, vol. 26, pp. 1–10, Nov. 2018.

[13] M. Shouman, T. Turner, R. Stocker, “Integrating Decision Tree and KMeans Clustering with Different Intial Centroid Selection Methods in Diagnosis of Heart Disease Patients,”Procedings of the International Conference on Data Mining, 2012, pp. 1-5.

[14] R. Saini, N. Bindal, and P. Bansal, “Classification of heart diseases from ECG signals using wavelet transform and kNN classifier,” International Conference on Computing, Communication & Automation, 2015, pp. 1–6.

[15] H. Yan, Y. Jiang, J. Zheng, C. Peng, and Q. Li, "A multilayer perceptron-based medical decision support system for heart disease diagnosis," Expert Systems with Applications, vol. 30, pp. 272, Feb. 2006.

[16] https://elitedatascience.com/dimensionality-reduction-algorithms, accessed on: Feb. 23, 2020.

[17] https://www.sciencedirect.com/topics/earth-and-planetary-sciences/support-vector-machine, accessed on: Feb. 23, 2020.

[18] http://mines.humanoriented.com/classes/2010/fall/csci568/portfolio_exports/lguo/decisionTree.html, accessed on: Feb. 23, 2020.

[19] https://towardsdatascience.com/understanding-random-forest-58381e0602d2, accessed on: Feb. 23, 2020.

[20] https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained, accessed on: Feb. 23, 2020.

[21] https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering, accessed on: Feb. 23, 2020.

[22] https://www.tutorialspoint.com/artificial_neural_network/index.htm, accessed on: Feb. 23, 2020.

[23] http://archive.ics.uci.edu/ml/datasets/Heart+Disease, accessed on: Feb. 23, 2020.