International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 14, Number 3 (2021), pp. 227-240
© International Research Publication House. http://www.irphouse.com

An Experimental Study of Heart Disease Prediction Using Different Supervised Machine Learning Algorithms

Neeraj Kumar Sharma[1], Mounika Vemula[2], Vani Tadiboyina[3]

[1, 2, 3] *Department of Computer Science and Engineering, SRM University Amravati, Andhra Pradesh, India.*

Abstract

In the last few decades, heart disease is a leading cause of death across worldwide casualty, and heart diseases have emerged as a major health threat, not only in India but worldwide. Therefore, there is a need to diagnose the heart disease at the appropriate time to avoid heart attack. In this paper, we study 6 different supervised machine learning algorithms such as K-Nearest Neighbor (KNN), Decision Trees (DT), Random Forest (RF), and Logistic Regression (LR), Artificial Neural Network (ANN), and Neural Network with Pearson Coefficient (NNPS) to detect heart disease. Further, we identify the most appropriate/most dangerous features of heart disease and predict the highest risk using mechanical Learning Methods. The experimental results show that heart disease prediction accuracy obtained by DT, LR, RF, KNN, ANN, NNPS is 75.88%, 85.49%, 86.08%, 85.73%, 78%, 89% using 80% training and 20% testing combination respectively.

Keywords: Cardio Vascular Diseases (CVD), Decision Tree (DT), KNN, Logistic Regression (LR), Machine Learning, Random Forest (RF)

I. INTRODUCTION (12 BOLD)

The human heart is one of the most important part of the human body system [21]. The main function of the human heart is to controls the flow of blood in our body, anywhere heart failure can cause sudden death or collapse of some part of the human body. Heart disease remains the leading cause of death at the global level 20 years ago. This generation, of course now killing more people than ever before. [20]. It is estimated by the World Health Organization (WHO), that over 12 million people worldwide die annually due to heart disease. Cardiovascular disease is responsible for half of all deaths in the United States (US) and in other developed countries [7]. Heart disease may be due to ill health, smoking, alcohol, overeating can cause high blood pressure [7]. Further, the death toll due to heart disease has increased from 12.3 million to 17.3 between 1990 to 2013 i.e. approximately 40% higher death ratio in the specified amount of time [8]. Therefore, at first the accurate and timely diagnosis of heart disease is necessary to reduce health problems and death count.

The different machine learning techniques can play an important role to approximately predicate heart disease. The use of machine learning techniques nowadays are applying on the different medical database to perform the large and complex data analysis. The early prediction of heart disease can help make decisions about lifestyle changes in high-risk patients and reduce complications. Many researchers, more recently, have been using a few machine learning techniques to assist the health care industry and specialists in diagnosing heart-related diseases. The introduction of the paper should explain the nature of the problem, previous work, purpose, and the contribution of the paper. The contents of each section may be provided to understand easily about the paper.

Machine learning is widely used in the file of modern medical and diagnostic health sector in Presence of diseases using data models. Therefore, a lesson aims to identify the most important predictions for cardiovascular diseases and predict all risk through various machine learning models. An important challenge in today's health care system is that the provision of high quality, efficient, and effective health services diagnosis of disease [1]. The heart disease is diagnosed as a major source of death in recent years worldwide. The absolute accuracy of disease management depends on the right time to get that disease. In our proposed research work, we make an effort to find these heart diseases are still too young to avoid catastrophic consequences. In most cases, the medical database is discrete details. Therefore, making decisions to predict illness using different data becomes a difficult and challenging task. The main purpose of this paper is to provide a medical tool for getting heart disease at an early age. This will help too to provide effective treatment to patients and to avoid complications results. With the help of machine learning to find the hidden patterns are also different when analyzing the data provided. In this paper, we present a experimental performance analysis of various machine learning techniques such as DT, LR, KNN, RF, ANN, NNPC for predicting heart disease.

In short, the key idea of our proposed work in this paper is first we perform the data acquisition, after acquiring the dataset there are some missing values with reference to few attributes. Hence, we need to apply some pre-processing techniques like data cleaning. In data cleaning, missing values are replaced by corresponding mean values. Then after we train the model of heart disease prediction using 4 different machine learning techniques such as KNN, DT, LR, and RF. Further, to check the accuracy of our proposed model we perform testing. The training and testing of our model are performed by taking different combinations of training and testing datasets.

Main/key contributions of the proposed work in this this paper are described as follows.

1. To pre-process and select the best feature vectors using mean value and Pearson correlation techniques, respectively.

2. Training and testing the model with separate machine learning algorithms such as DT, LR, RF, KNN, ANN, and NNPS algorithms for heart disease predication.

3.Comparing the learning performance of different machine algorithms using confusion matrices, confidence interval, scores obtained by the algorithms, and computational time requirement.

The rest of the paper is organized as follows. Section II includes the background and related work. Section III describes the proposed heart disease prediction model using 4 different machine learning algorithms. Section IV explains test setup and results and analysis. At the end section, V highlights the conclusion and future works of the proposed heart disease prediction using machine learning.

III. RELATED WORK

Most of the researchers used Framingham machine learning databases to predict heart disease in their proposed work. Researchers in their experimental study achieved a different level of accuracy by using different machine learning algorithms. Further, existing works with reference to heart disease prediction with their limitations are described as follows.

A. Golande et al. [1] explores various machine learning methods that can be used to diagnose heart disease. In their proposed work, authors implemented DT, KNN, and K-Means algorithms for heart disease as a classification problem. The experimental results conclude that the accuracy obtained by the DT was significantly higher than was anticipated by the combination of various techniques and parameter correction. An important limitation of their proposed work was they did not use the data processing to clean up the data set.

F. S. Alotaibi et al. created a machine learning model and compared different machine learning algorithms [2] In their proposed work they used a quick miner tool to test the performance of different machine learning algorithms. In their proposed work the accuracy of the DT, LR, Naive Bayes and SVM classification algorithms were compared. The decision of the drug algorithm is very high. An important limitation of their approach was author did not use the data processing process to clean up the data set. T. R. Prince, et al. conducted research on machine learning algorithms divided into categories for predicting heart disease. The classification strategies used by the author are Naive Bayes, KNN, Decision tree, Neural network. Classification classifiers were analyzed by taking different numbers of symbols [3]. The key limitations of the proposed work were they did not perform any experimentation work.

Chen et al. in their proposed work compared the accuracy of different algorithms such as SVM, neural networks, Bayesian segregation, tree decision, and Logistic retrospective considerations [4]. In their tests, they found that SVM had a really high accuracy of 90.5%, neural networks 88.9%, Bayesian 82.2%, Decision tree 77.9% and logistic regression 73.9%. Shoumanetal et al. statistically significant risk factors for age, blood pressure, cholesterol, smoking, total cholesterol, diabetes, high blood pressure, genetics, obesity, metabolic syndrome [5]. The same paper also listed the Cleveland Heart Disease Database is a standard database for cardiovascular research as it is widely accepted. Detra noe tal [6]used logistic regression to obtain 77% accuracy of prediction. Further, in short, the existing works with reference to heart disease prediction are described in Table 1. Table 1 highlights the proposed methodology and limitations of the existing work.

Table1: Summary of Existing Works on CHDD

Writer Name	Proposed methodology	Limitations
Detrano et al.[6]	Logistic regression (77%)	There is only very less percentage of accuracy i.e., 77%
Cheung [9]	C4.5(81.11%) Naive Bayes (81.48%) BNNF (80.96%)	Author has performed naïve bayes algorithm acquired the better accuracy, but the drawback of this algorithm is the assumption of independent predictor features.
Tu et al. [10]	J4.8 Decision Tree (78.9%) Bagging Algorithm (81.41%)	Bagging is certainly a star classifier when we need to fight against the variance, create a more stable and robust model that can be run parallel. The only limitation is that it is computationally expensive since it requires a high number of estimators.

III. PROPOSED WORK

There are 8 different machine learning algorithms are implemented for heart disease predication. The detailed description of all the algorithms is described as follow.

1. Decision Tree (DT): The decision tree is a targeted machine learning algorithm [1]. This method is widely used for segregation problems. It works hard with continuous and differentiated signals. This algorithm divides a population into two or more identical sets based on the most important predictors. The decision of the tree algorithm first calculates the entry of each element. The database is then categorized with the help of variables or predictors with a high degree of information or a minimum of entropy. These two steps are repeated with the remaining signs.

$$Entropt(S) = (\sum_{i=1}^{c} -plog_2\, p_i) \qquad (1)$$

$$Gain(S, A) = Entropy(S) - \sum_{Value(A)} \frac{|S_v|}{|S|} Entopy(S_v) \qquad (2)$$

The entropy calculation is described by Equation 1, to calculate the entropy, we can decide to dataset partition into different branches of the decision tree. Where pi is a probability of class i in the given dataset. The calculation of information gain is defined by Equation 2. This algorithm will always try to maximize data gain. The qualification with the highest gain will be assessed categorized first. Where Sv is a subset of S with the value v element F. Figure 1 shows the basic structure of the decision tree. Decision trees consist of 3 kinds of nodes such as root nodes, intermediate nodes, and leaf nodes. In figure 1 root node represents the highest gain attribute among all the attributes in the given dataset. The intermediate nodes represent the feature vector/attributes of the given dataset. All the leaf nodes represent the labelled class.

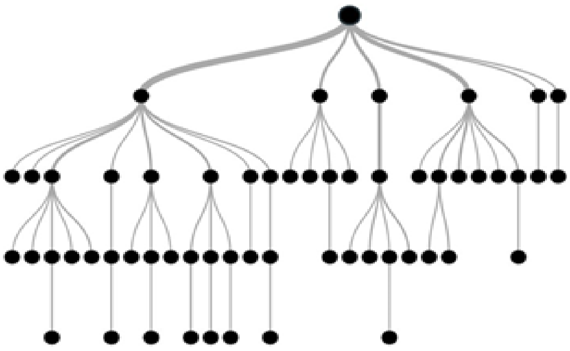


Figure 1: Flow chart of the Decision Tree

The study of the decision tree uses the decision tree as a predictive model that marks the recognition of an object to reach the conclusions of the intended value of the object. It is one of the most widely used prediction methods used in mathematics, data processing, and machine learning. Types of trees where the target variant can take a complete set of values called classified trees. In our proposed work first, we calculated the entropy (s) of the feature vectors. There are 15 feature vectors are given in the data set. Further, we calculated the gain (a) of each feature vector, then, we selected the highest gain feature vector as a root node. Hence, we generated a decision tree by applying a number of iterations in the same way.

2. K - Nearest Neighbor (KNN): K Nearest Neighbors is a simple algorithm, but it works amazingly well for doing just that saves all available cases and separates new data or a case based on the similarity scale [12]. It suggests that if a new point added to the sample is the same as the neighbor points, that point will belong to a particular category of points to a neighbor. Typically, the KNN algorithm is used in search apps where people want the same things. K in the KNN algorithm means the nearest number neighbors of a new point that needs to be predicted. The KNN algorithm is also known as a lazy learning because there is a small learning phase of the model because it is good for the ability to learn quickly. Instead, memorizes the training database and all work takes place during the forecast requested. This algorithm involves finding nearby k data points for training set in the data area where the target value is not available and provides values between the data obtained from it.

During the implementation of the KNN algorithm, we initialized the value of variable k by using the formula k=sqr(n), where n is the number of data points in the dataset. After that, we calculated Euclidean distance from queried input point to all other input points, and sort points based on their Euclidean distance value. Further, we selected the first k data points as a neighboring point of the queried input point. At the end, we assigned a class to the queried data point based on the majority class of neighboring points.

3. Logistic Regression Model (LRM): Logistic Regression is one of the tool to learn the algorithm configuration for data analysis where there is one or more independent variables that determine the outcome as well, and the variables are divided into two categories (DV) [16].

$$\sigma(z) = \frac{1}{1+e^{-z}} \qquad (3)$$

The logistic regression decision sigmoid function is described in Equation 3. Where, $\sigma(z)$=output between 0 and 1 (probability estimate), z is the input to the f unction, and e is the base of natural log.

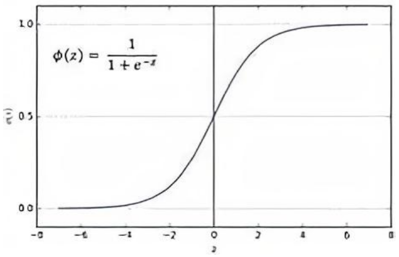


Figure 2: Logistic regression

Figure 2 shows the logistic regression. Reversal of order is also known as sigmoid activity that facilitates easy display on graphs. It even gives high accuracy. In this algorithm the starting data should be allowed and trained. From the training data we have to measure the best and closest coefficient and represent it. Independent variables n in the order model such as x1, x2, x3… xn

$$\log\left(\frac{P}{1-P}\right) = \beta_0 + \beta_1 X_1 + \cdots \ldots\ldots\ldots\ldots\ldots + \beta_n X_n \quad (4)$$

The logistic regression is described by Equation 4. Where P is the probability of an event that is a risk of CHD. Therefore, P always lies between 0 and 1. We applied multivariable logistic regression to predict heart disease, and we assigned 0.5 as a threshold value. The input value which is less than 0.5, we treated as binary 0, and the input value greater than 0.5 is treated as binary 1. Hence, in this way, we classify heart disease.

4. Random Forest (RF): Random Forest is also the most popular machine learning algorithm [1]. The RF can be used for regression and classification methods but it usually works best for classification. As the name suggests, the random process of forest looks at multiple decision trees before giving a result. So, basically a collection of decision trees. This process is based on the belief that a large number of trees will turn to the right decision. For classification, it uses the voting system and determines the category but in going back means that every single release of trees of decision. Works great with high-volume data stocks size.

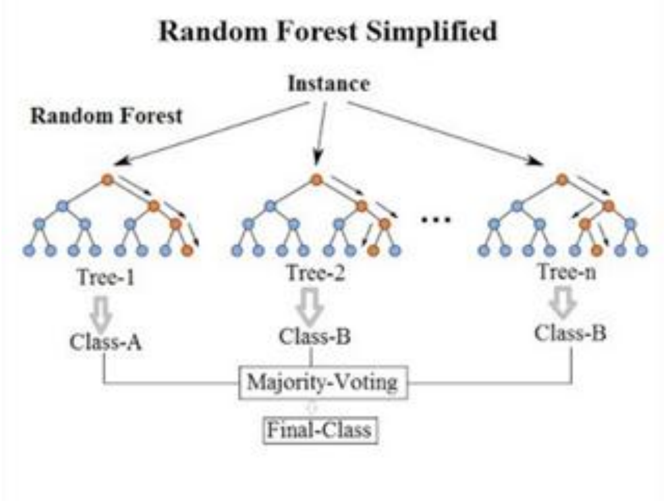


Figure 3: Random Forest.

Figure 3 shows the random forest. A random forest algorithm is one of the most effective ways to differentiate. This section moves the idea of a decision tree to another level. It creates a forest of trees where each tree is formed with a random selection of elements from perfect features. RF contains many trees of decision. Each decision tree provides a vote showing the decision about the category of the item. There are three important barriers to planning in a random forest such as

a) Number of trees (n tree)

b) Minor node size

c) Number of the features used to separate every node of a tree.

Creating the number of trees in the RF depends on the number of input data points in the data set. Hence, we selected 10 sets of random samples and designed the random tree for each sample dataset. Further, we trained all the decision trees by the respective sample datasets. After that, we performed voting and selected the most voted result. The

5. Artificial Neural Network:- The ANN is a computer model based on neural biological networks. ANN is based on the observation of the human brain. The human brain is the most complex structure of neurons. Analogically ANN is an integrated set of three units as an input, hidden, and output unit. In clinical trials, patient risk factors or attributes are used as an inclusion. The effectiveness of the neural implant network was proven medically. ANN is used to predict heart disease. Here the insertion layer consisting of 8 neurons corresponds to 8 key signals. There is only one output variant that takes a value of either 0 or 1. The value of 0 indicates that the person does not have a heart attack and the number 1 represents the person having heart disease. In he

6. Neural Network using Pearson correlation:-Correlation is the measure of a direct relationship of two or more variants. Through correlation, we can predict changes from one to another after using the feature selection combination that is good variables are highly correlated with the target. In addition, the variables should be in line with the target but should not be related among them. If two variables are made true, we can predict one another. Further, if two features are faulty, the model only needs one of them, as a second does not add any additional details. We will use the Pearson correlation here.

7. Confusion Matrix:- This is used to show a summary of the prediction results including right and wrong in the classification problem. Moreover, this applies not only to mistakes but also to types of errors. The parts of the confusion matrix show the following parameters.

I. True Positives (TP): cases predicted yes, and they are infected.

II. True Negatives (TN): cases predicted to be negative and free from the disease.

III. False Positives (FP): cases predicted yes, but no disease (Type I error).

IV. False Negatives (FN): cases that predict no, but actually have a disease (type II error).

Table 2: Different terms to test model performance.

Terms	Formula
Model accuracy (all in all, how often the classifier corrects)	(TP+TN)/(TP+TN+TP+FN)
Misclassification Rate (overall, frequency or error rate)	(FP+FN)/(TP+TN+FP+FN)
Sensitivity or Real Measure (True Positive rate) (If yes, how often do you say yes)	TP/(TP+FN)
Specification or Real Negative Rating (If not, how often does it predict)	TN/(TN+FP)

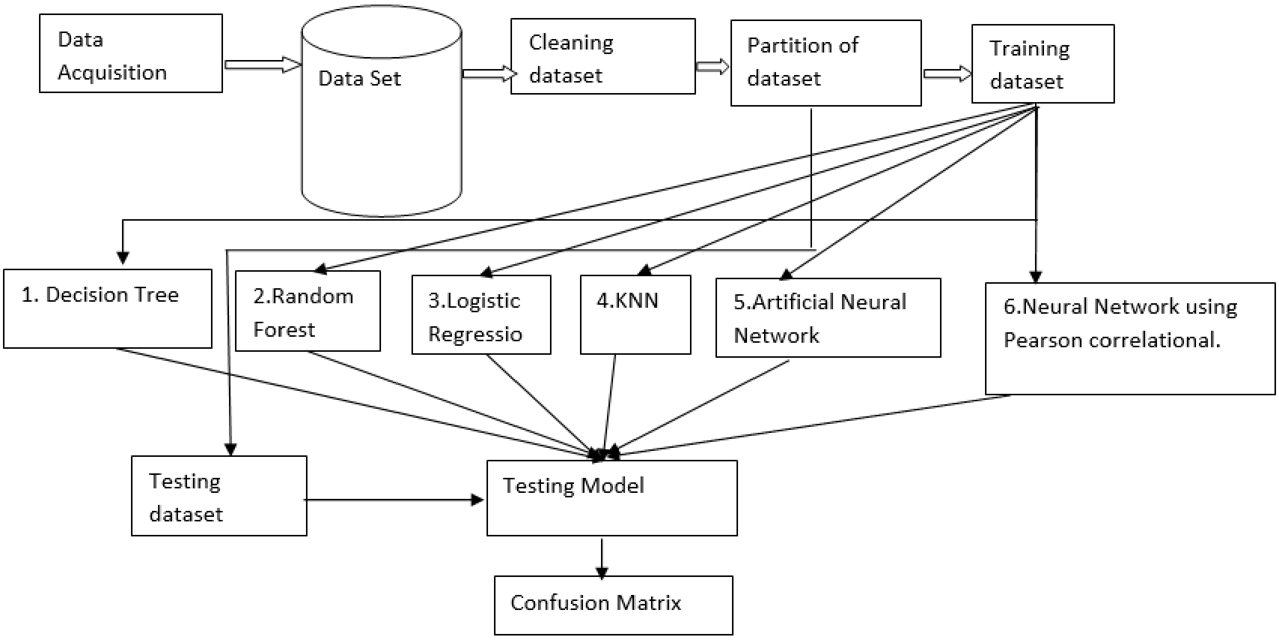


Figure 4: Flow Chart of Proposed Heart Disease Prediction.

8. Proposed model for heart disease predication: -Figure 4 shows the flow chart of the proposed work. It shows the different steps followed to build classification models in machine learning.

As shown in figure 4 first, we need to perform the data acquisition. After acquiring the dataset, we need to apply some pre-processing techniques like data cleaning. In data cleaning, missing values are replaced by corresponding mean values. The following step is to train the data and test on different models with different splitting percentages. Collect all the accuracies and computational time obtained after testing and compare all the approaches to predict the best model. The detailed descriptions of different steps are described as follows.

a. **Data Acquisition: -** The dataset which is used for implementing different machine learning algorithms analysis is downloaded from the Kaggle website (https://www.kaggle.com), from the ongoing research of the heart of Framingham, Massachusetts. The purpose of this classification study was to predict whether a patient had a 10-year risk of future heart disease. The Framingham database has 4238 records of patient’s data with 15 attributes.

b. **Data Pre-Processing:** In order to build an accurate ML model, data processing is required. This section includes data extraction from the Cleveland Heart Disease Dataset (CHDD) in the same format. The step involves data conversion, which includes the removal of missing fields, the normalization of data, and the removal of deviations.

c. **Data Cleaning:** The data we want to process will not be clean, which may be noise or may contain missing values in our process. We cannot get good results. To get the best and perfect results we need to end all this, the process of eliminating all this is data cleaning.

d. **Missing values:** We will fill in the missing values and we can remove the noise by using certain techniques such as filling in mean values or standard values in non-existent areas.

e. **Reduction:** When working on data it may be complex and difficult to understand at times to make it understandable to the system, we will reduce the format required to achieve the best results.

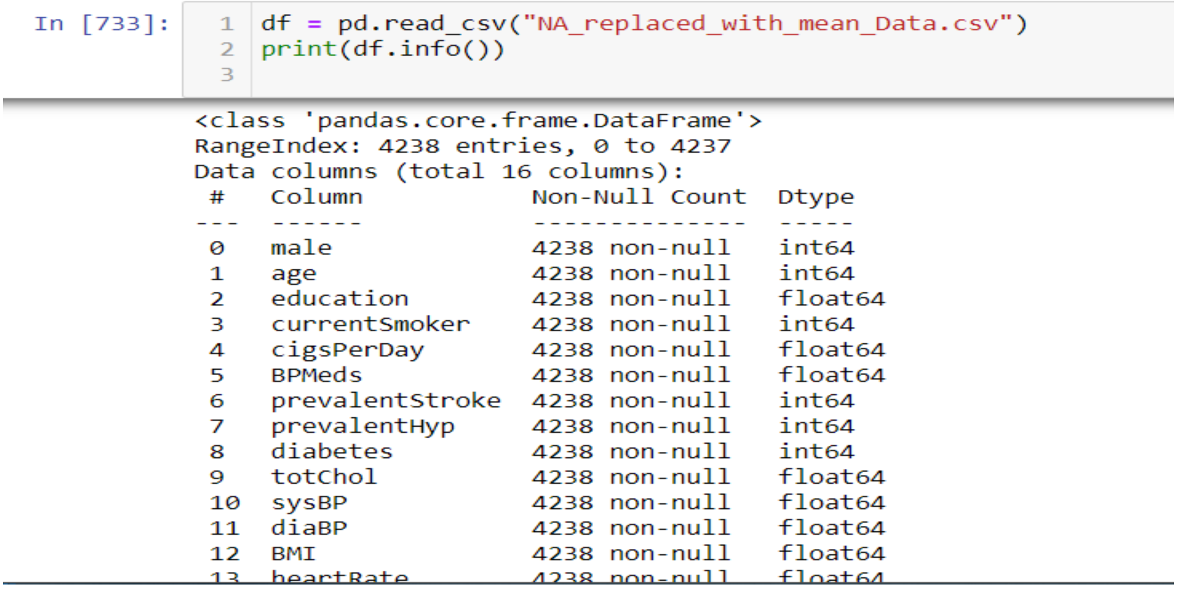
f. **Training and Testing:** After performing data acquit ion and pre-processing the training and testing are performed by implementing different machine learning algorithm such as DT, LR, KNN, RF by taking different combinations of training and testing data set such as 80%-20%,70%-30%,60%-40%.

V. EXPERIMENTAL SETUP AND RESULTS ANALYSIS

To check the performance of the heart disease prediction using 8 different machine learning algorithms we performed an extensive experiment. The experimental setup and results and analysis sections are described as follows:

(a) Experimental Setup: We conducted our experiment on Intel core i5 processor with 8 GB RAM size. Further, we used Jupiter Notebook with Python 3. 7.4. The key motivation to use python language for the implementation of different machine algorithms because python provides different build-in packages such as NumPy, Pandas, SciPy, Matplotlib, Standard Scaler, etc. These packages are very important from the implementation and performance measurement point of view. Firstly, we collected the dataset from the Kaggle website known as the Framingham dataset. The number of feature vectors in this dataset is 16, 15 among them are feature vectors and 1 class. The 4238-total number of observed values for a single feature vector was given in the data set. To get better accuracy. we used Pearson Correlation for feature selection, and we got 3 key features. Later, we trained and tested the data with three different splits of 60-40%, 70-30%, and 80-20%. The above-mentioned steps are to test with 6 different models such as DT, RF, LR, KNN, ANN, and NNPS.

(b). Results and Analysis: The data set consists of 15 feature vectors and a predicted numbers. The machine learning model is based on DV identification. It used binary LR, KNN, DT, RF, ANN, NNPS which is one of the categories due to the specific variation of Classical Data Analysis performed using Jupyter Notebook. Following are the steps which have been taken to evaluate machine learning models. First of all, we loaded data and different libraries. It includes cardiac predictive data using the Framingham CSV file in Jupyter Notebook to build machine learning models. In addition, libraries required are used as supporting applications.

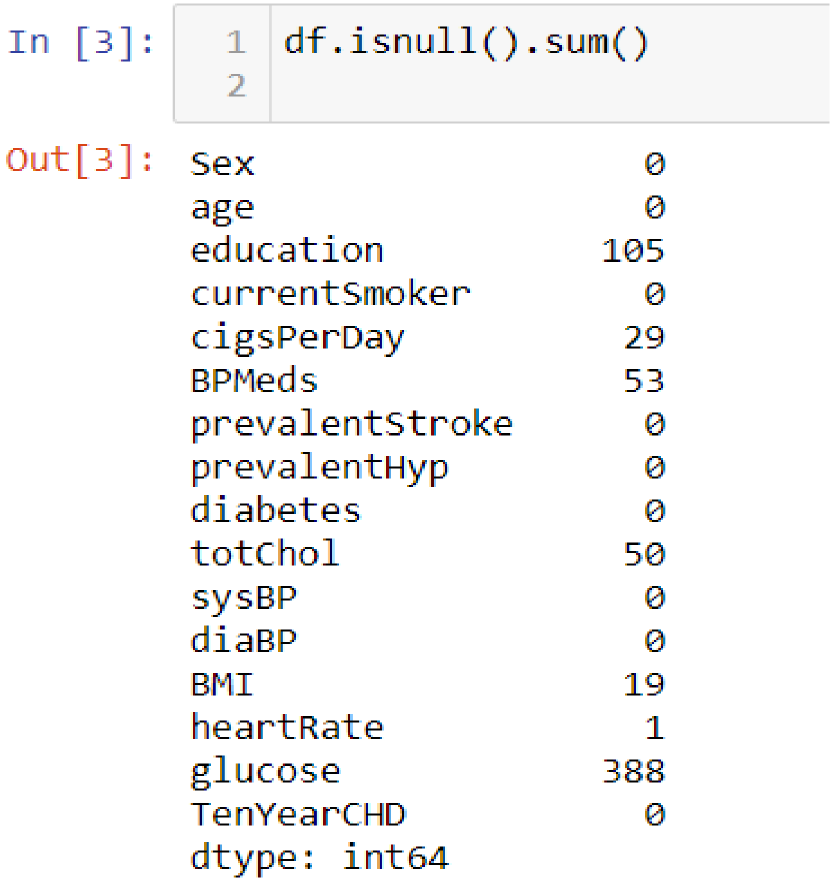


```
In [733]:  df = pd.read_csv("NA_replaced_with_mean_Data.csv")
            print(df.info())

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4238 entries, 0 to 4237
Data columns (total 16 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   male             4238 non-null   int64  
 1   age              4238 non-null   int64  
 2   education        4238 non-null   float64
 3   currentSmoker    4238 non-null   int64  
 4   cigsPerDay       4238 non-null   float64
 5   BPMeds           4238 non-null   float64
 6   prevalentStroke  4238 non-null   int64  
 7   prevalentHyp     4238 non-null   int64  
 8   diabetes         4238 non-null   int64  
 9   totChol          4238 non-null   float64
 10  sysBP            4238 non-null   float64
 11  diaBP            4238 non-null   float64
 12  BMI              4238 non-null   float64
 13  heartRate        4238 non-null   float64
```

Figure 5: Data exploration.

Figure 5 shows an overview of data types, number of rows (4238), number of columns (16) of all the features. Further, we identify the missing values of the given dataset. In addition, the number of missing values is determined by cleaning up the existing database. The approximate total of the values lost based on symbols is given below.

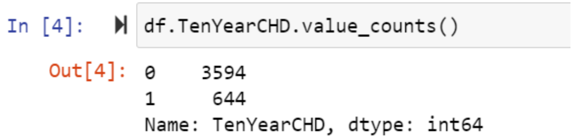


```
In [3]:   df.isnull().sum()

Out[3]:   Sex                  0
          age                  0
          education          105
          currentSmoker        0
          cigsPerDay          29
          BPMeds              53
          prevalentStroke      0
          prevalentHyp         0
          diabetes             0
          totChol             50
          sysBP                0
          diaBP                0
          BMI                 19
          heartRate            1
          glucose            388
          TenYearCHD           0
          dtype: int64
```

Figure 6: Summarization of Missing values

Figure 6 shows the summarization of the missing values in the given dataset. The dataset contains some missing values in the education, cigs per day, BPMeds, totChol, BMI, heart rate, and glucose columns. Machine learning models have no way to reasonably deal with missing values, we choose to replace the missing values with corresponding means. Thereafter the total amount of missing values in the column is determined using the Pandas Data frame. The total number of lines with missing values is 494, as only 12 percent of lost data values are substituted for corresponding definition values.

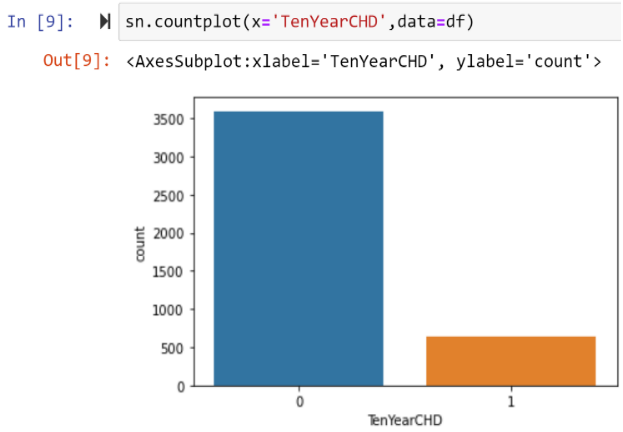


```
In [4]:  df.TenYearCHD.value_counts()

Out[4]:  0    3594
         1     644
         Name: TenYearCHD, dtype: int64
```

Figure 7: No of People Affected by Heart Disease.

Figure 7 shows the no of people contrived by heart disease. Where binary 0 describes the people, who are not affected by heart disease and 1 describes the people who are affected by heart disease. Figure 8 shows the descriptive figures related to the 10-year risk of CHD.

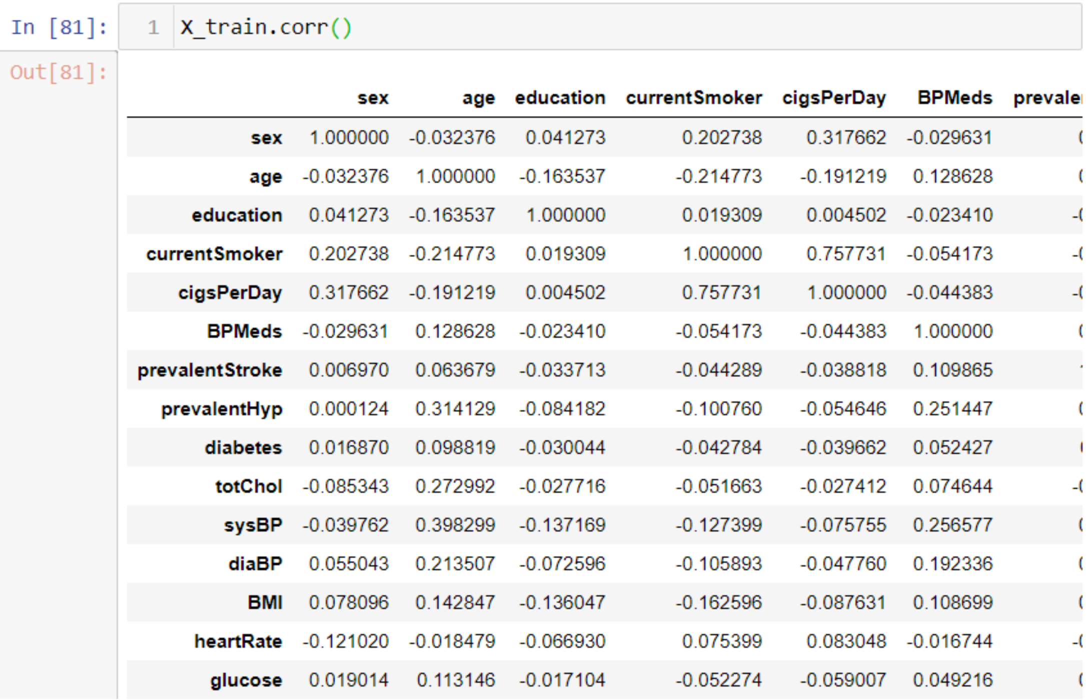


```
In [9]:  sn.countplot(x='TenYearCHD',data=df)

Out[9]:  <AxesSubplot:xlabel='TenYearCHD', ylabel='count'>
```

Figure 8: Graph on Ten Years CHD

According to figure 8, there are 3179 patients without heart disease and 572 patients at risk of heart disease. Figure 9 shows the correlation values for each feature vector. In addition, feature selection is a process of reducing the number of input variables when making a predictive model. It means the process of selecting the most relevant features in our data that you can provide to our model. By reducing the number of features we can often speed up training and increase the accuracy of the model or both.



```
In [81]:  1  X_train.corr()

Out[81]:
                       sex       age  education  currentSmoker  cigsPerDay    BPMeds  prevale
            sex   1.000000 -0.032376   0.041273       0.202738    0.317662 -0.029631        (
            age  -0.032376  1.000000  -0.163537      -0.214773   -0.191219  0.128628        (
      education   0.041273 -0.163537   1.000000       0.019309    0.004502 -0.023410       -(
  currentSmoker   0.202738 -0.214773   0.019309       1.000000    0.757731 -0.054173       -(
     cigsPerDay   0.317662 -0.191219   0.004502       0.757731    1.000000 -0.044383       -(
         BPMeds  -0.029631  0.128628  -0.023410      -0.054173   -0.044383  1.000000        (
prevalentStroke   0.006970  0.063679  -0.033713      -0.044289   -0.038818  0.109865
   prevalentHyp   0.000124  0.314129  -0.084182      -0.100760   -0.054646  0.251447        (
       diabetes   0.016870  0.098819  -0.030044      -0.042784   -0.039662  0.052427        (
        totChol  -0.085343  0.272992  -0.027716      -0.051663   -0.027412  0.074644       -(
          sysBP  -0.039762  0.398299  -0.137169      -0.127399   -0.075755  0.256577        (
          diaBP   0.055043  0.213507  -0.072596      -0.105893   -0.047760  0.192336        (
            BMI   0.078096  0.142847  -0.136047      -0.162596   -0.087631  0.108699        (
      heartRate  -0.121020 -0.018479  -0.066930       0.075399    0.083048 -0.016744       -(
        glucose   0.019014  0.113146  -0.017104      -0.052274   -0.059007  0.049216        (
```

Figure 9: Correlation values for each feature.

Figure 9 shows the correlation values of each feature vectors. We can observe that each feature calculates correlation with all the remaining features. To do the feature selection we applied Pearson Correlation Technique. We should always try to do a train test split so that we will be able to prevent the over-fitting part. All functionalities that are planned with respect to correlation are done only on the training dataset. And from the test dataset whatever things we did in training, we did similar things on the test dataset. Suppose in our training dataset, we found that there are four features that are highly correlated. We will remove those directly from X_train. Then in X_test, we won't do a correlation re-test again. Because in X_test we will directly remove those features themselves. Firstly, we did a Pearson correlation and drew a heatmap. Pearson correlation ranges between -1 and +1. We have calculated correlation for X_train before but by seeing those values, we cannot clearly understand. Hence, to make it into a more visualized format we took the help of a heatmap with correlation.

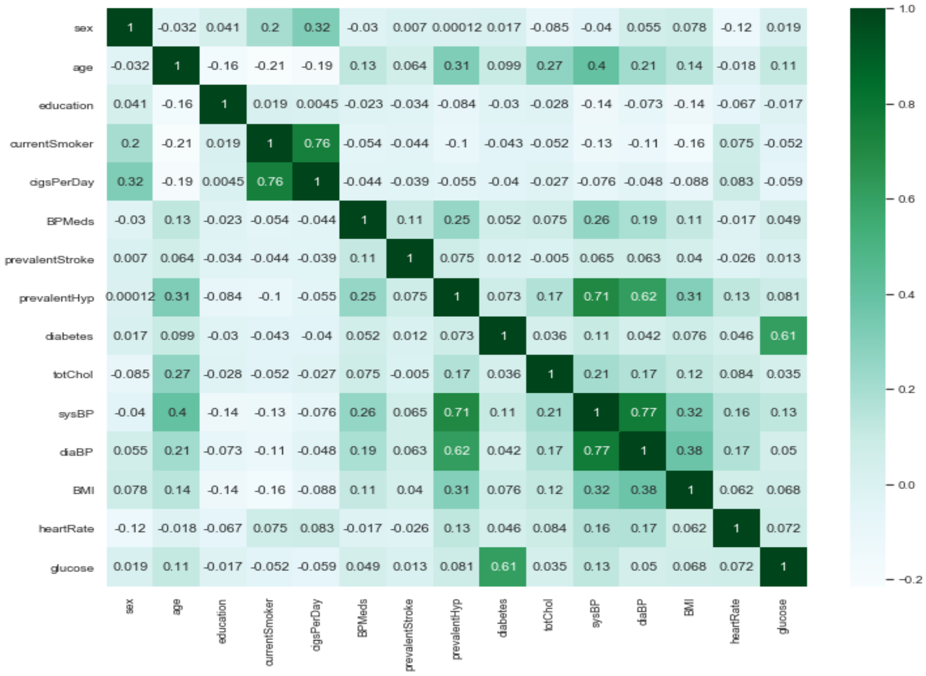


Figure 10: Correlation Heat Map.

Figure 10 shows the heat map of the feature vectors in the dataset. Figure 10 describes the statistical measure of the strength of the relationship between the relative movements of two variables. We used a heatmap to visualize the correlation coefficient between the attributes of the dataset. In addition, we have used a function called an adjustment to select the most relevant features. It will remove the first feature associated with any other feature. This function takes a dataset like X_train with a threshold value of 70%. We say that the threshold value should at least 70% correlated with other features then only we will remove a feature. For example, from the above heatmap, if we compare sys BP with diabetes, they have a 0.77 correlation which is 77%. So, one of the features will be removed from the dataset. In this way, we compare each feature with all the other features.

The information provided is considered to be included in various ML algorithms such as RF, LR, DT, KNN, ANN, and NNPC. Then we divided the input database into 60% of the training database and the remaining 40% in the test database. Database Training is a database used for model training. The test site is used to test the performance of the appropriate model. Different metrics are used as precision, and computational complexity as described further. The various algorithms tested in this paper are listed below.

60 Training & 40% Testing Dataset:-

In the case of LR the accuracy of the model is 85.4% in 0.0078 seconds. Time is calculated in seconds. The confusion matrix of the logistic regression for the dataset is described as follows. According to the outcome of the confusion matrix, positive prediction (1447+3) =1450, an incorrect prediction (242+4) =246. Therefore, True Positives:1447, True Negatives:3, False Positives:4 (Type I error), False Negatives:242(Type II error). Further, the accuracy achieved by the decision is 76% in 0.00324 seconds. The detailed description of the confusion matrix of the data set is Correct prediction (1222+74) =1296, Incorrect prediction. (229+171) =400. Therefore, True Positives:1222, True Negatives:74, False Positives:229(Type I error), False Negatives:171(Type II error). We chose the decision tree to check the heart decision prediction selected because they are fast, reliable, easy to translate, and very little data preparation is required.

Further, we applied random forest for heart disease prediction. The confusion matrix of the random forest for the heart diseases prediction for the given data set is described as. The accuracy of the heart disease prediction in the case of random forest is achieved by 86% with 0.011 seconds simulation time. In the case of random forest, we observed the following values as Correct prediction (1443+17) =1460, Incorrect prediction (8+228) =236. Therefore, True Positives:1443, True Negatives:17, False, Positives:8(Type I error), False Negatives:228(Type II error). The performance of the ANN wth reference to heart disease predication is described as, we checked the performance of the heart disease prediction by applying the KNN machine learning algorithm. Further, we achieved accuracy for KNN is 85.7% in 0.00158 seconds. The confusion matrix for KNN is described as follows.

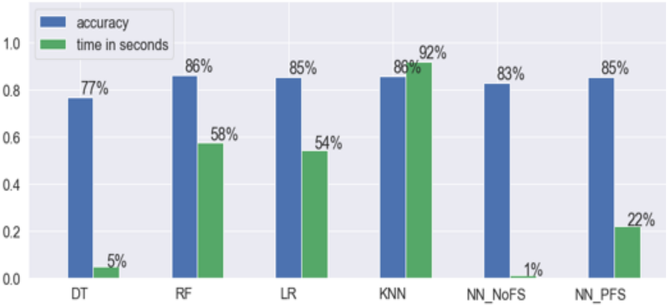
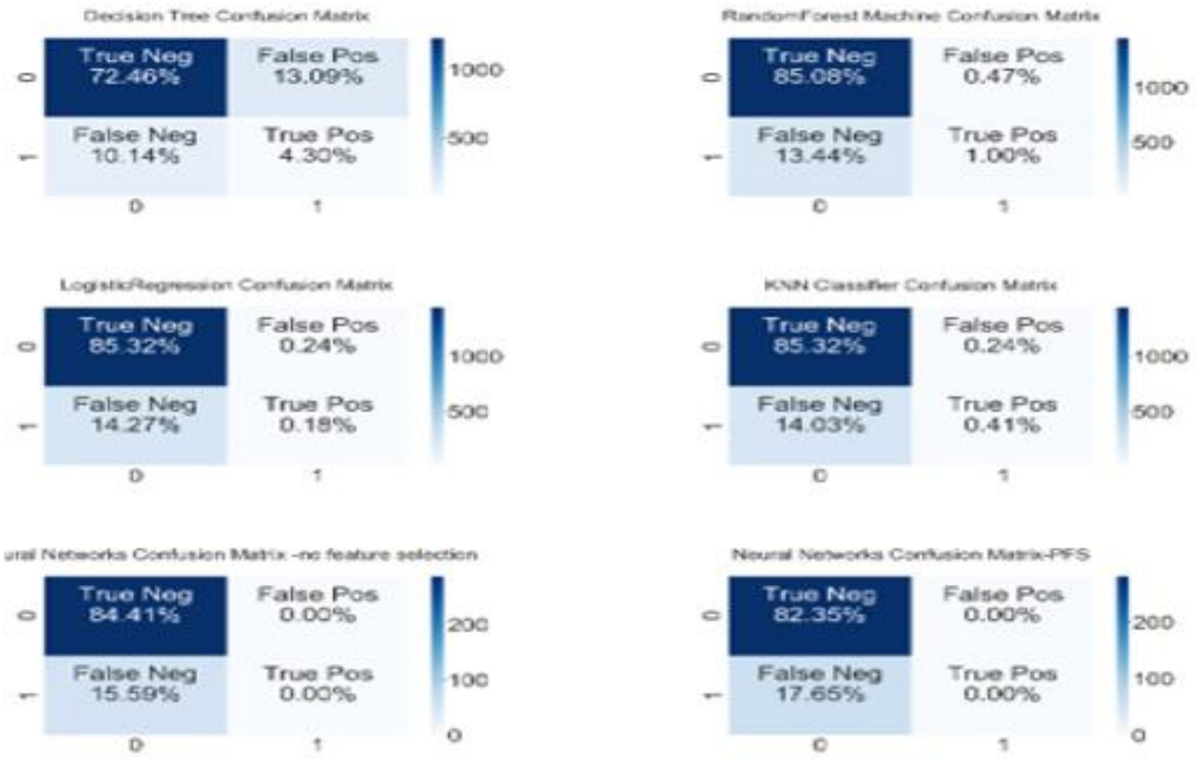


Figure 11: Accuracy and computational Time graph for 60-40 split

In case of KNN we observed the following values as Correct prediction (1447+7) =1454, Incorrect prediction (4+238) =242. Therefore, True Positives:1447, True Negatives:7, False Positives:4(Type I error), False Negatives:238(Type II error)

Figure 11 shows the accuracy and computation time of different machine learning algorithms by splitting data set into 60% training and 40% testing. As we can see in figure 11, random forest performed better in terms of accuracy by 86 % over other machine learning algorithms. Further, in terms of computational time, KNN performed well with 92% time efficiency over other machine learning algorithms.


Decision Tree Confusion Matrix

RandomForest Machine Confusion Matrix

LogisticRegression Confusion Matrix

KNN Classifier Confusion Matrix

ural Networks Confusion Matrix -no feature selection

Neural Networks Confusion Matrix-PFS

Figure 12: Confusion Matrix for 60% Training & -40% Testing Split.

Figure 12 shows the confusing matrix of six machine learning models. In figure 12 True Pos describes that the percentage of patients is known in the database as heart disease affects patients and is best predicted under this subsection (1). However, a class of False Neg class was initially for heart disease with affected patients but was incorrectly predicted (0) - they could not be attacked. In the same way among all patients who do not attack the well-predicted True Neg category. And the False Pos category was also recorded incorrectly.

70% Training & 30% Testing Dataset:-

After conducting the experiment on 60%-40 data set, we splited the data set into 70% training and 30% testing. Figure 13 shows the accuracy and computation time for six different machine learning algorithms.

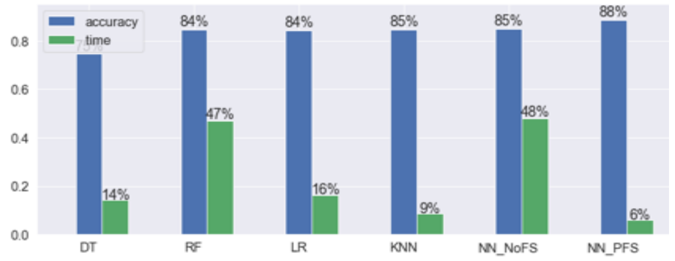
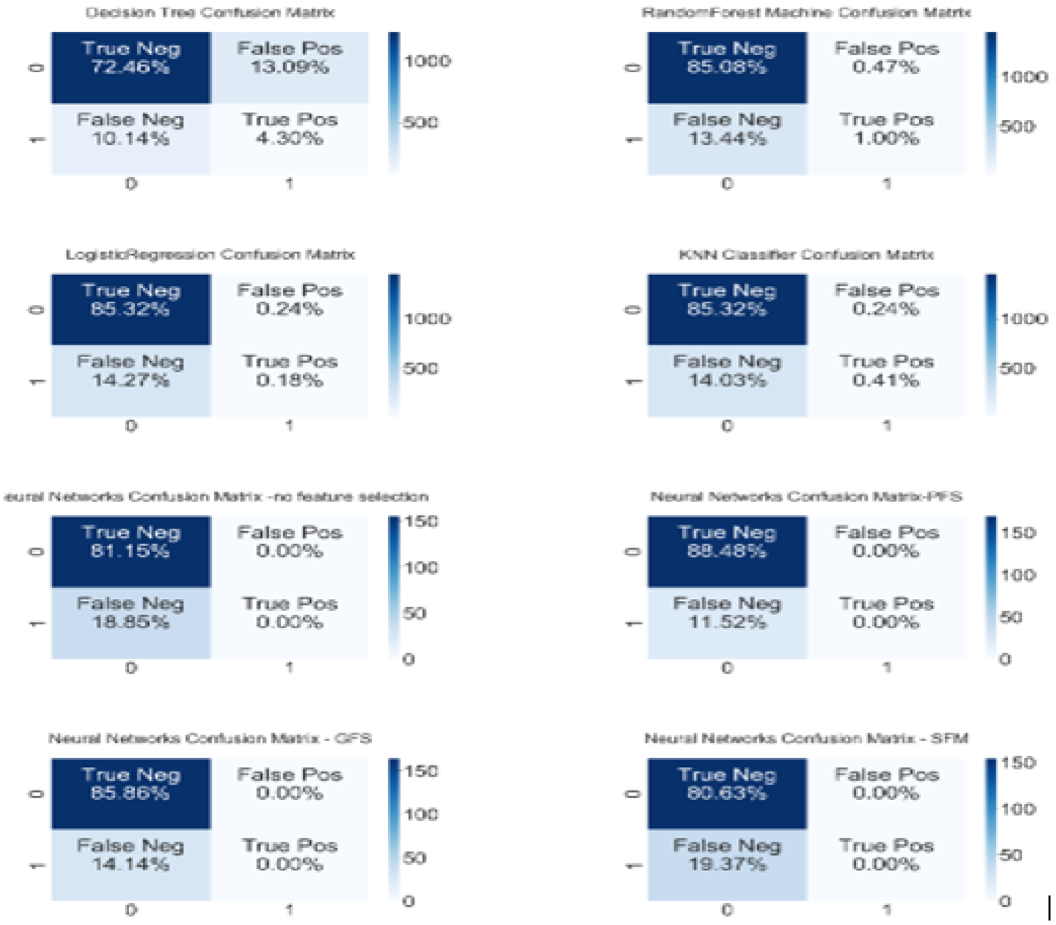


Figure 13: Accuracy and Computational Time graph for 70-30 split.

In figure 13 we can observe that Neural Network Pearson Coefficient (NNPC) gives the highest accuracy of 88%. Further, the computational time required to NNPS is 6% . The worst performance on 70%-30% split of data set is achieved by DT with 78%. Figure 14 shows the confusion matrix of all machine learning algorithms in the data mode of 70% -30. In addition, for all True Pos dividers, it indicates that the percentage of patients known in the database as heart disease affects patients and is best predicted under this subsection (1). However, a class of False Neg class was initially for heart disease with affected patients but was incorrectly predicted (0) - they could not be attacked. In the same way among all patients who do not attack the well-predicted True Neg category. And the False Pos category was also recorded incorrectly.


Decision Tree Confusion Matrix

RandomForest Machine Confusion Matrix

LogisticRegression Confusion Matrix

KNN Classifier Confusion Matrix

eural Networks Confusion Matrix -no feature selection

Neural Networks Confusion Matrix-PFS

Neural Networks Confusion Matrix - GFS

Neural Networks Confusion Matrix - SFM

Figure 14: confusion matrix for 70-30 split.

80% Training & 20% Testing Dataset:-

At the end, we conducted our experiment by taking an 80%traing -20% testing dataset for all 6 machine learning algorithms.

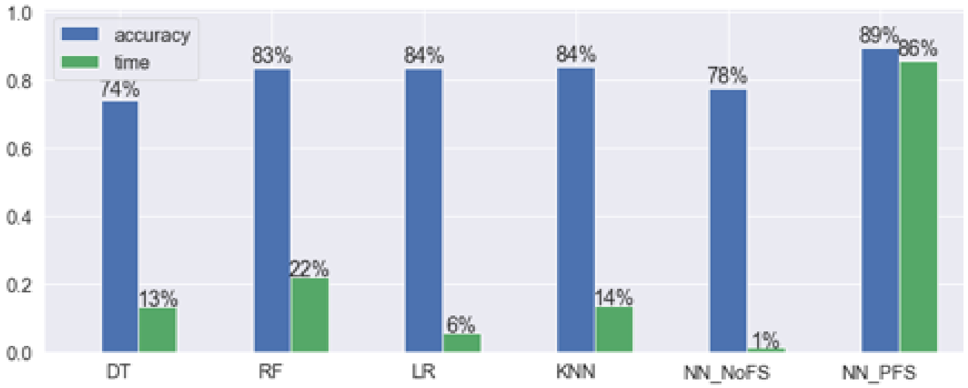


Figure 1: Accuracy and Computational Time graph for 80-20 split.

Figure 15 shows the accuracy and computational time of different machine learning algorithms with respect to heart disease prediction. Further, we can observe from figure 15 that the NNPC gives highest amount of accuracy i.e. 89% with 86% computation accuracy. The accuracy of the NNPC algorithm is increased by 1% in the case of 80%-20% as compared to 70%-30%. The worst performance in terms of accuracy is achieved by DT i.e. 74%.

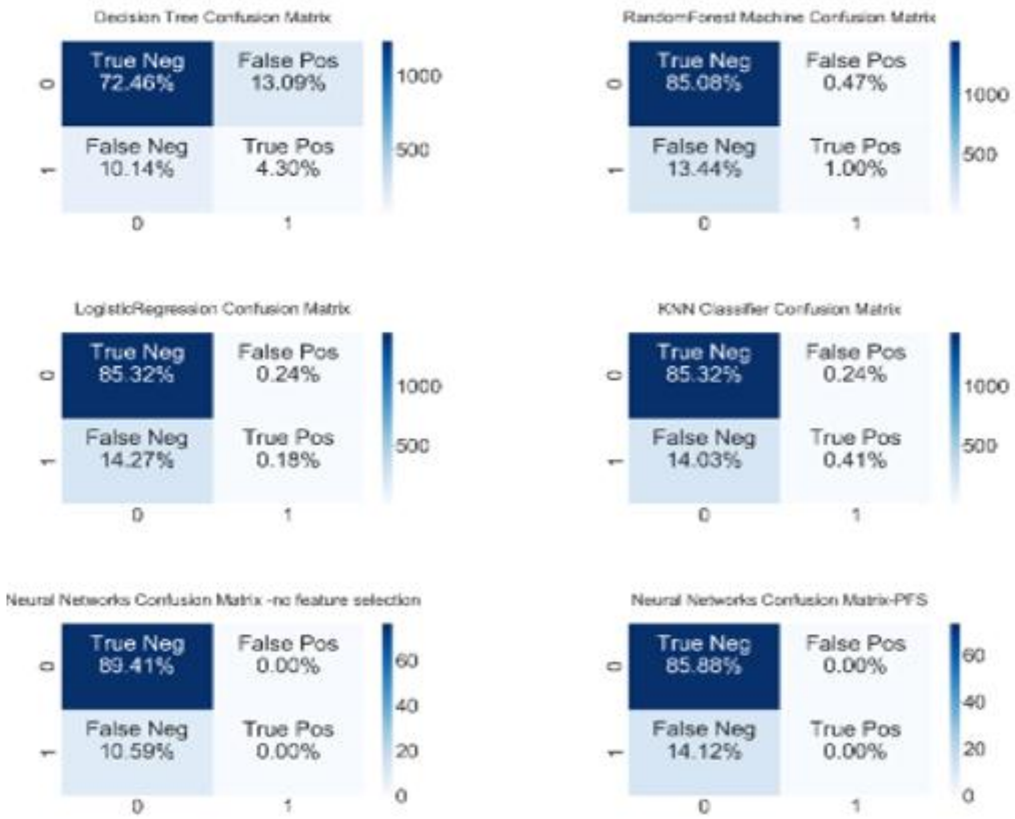


Figure 16: confusion matrix for 80-20 split.

Figure 16 show the confusion matrix of four algorithms by taking 80%-20 dataset. Of all the dividers True Pos indicates that the percentage of patients is known in the database as heart disease affects patients and is best predicted under this subsection (1). However, a class of False Neg class was initially for heart disease with affected patients but was incorrectly predicted (0) - they could not be attacked. In the same way among all the patients who are not attacked True Neg class category predicted correctly. And False Pos category is also recorded wrongly.

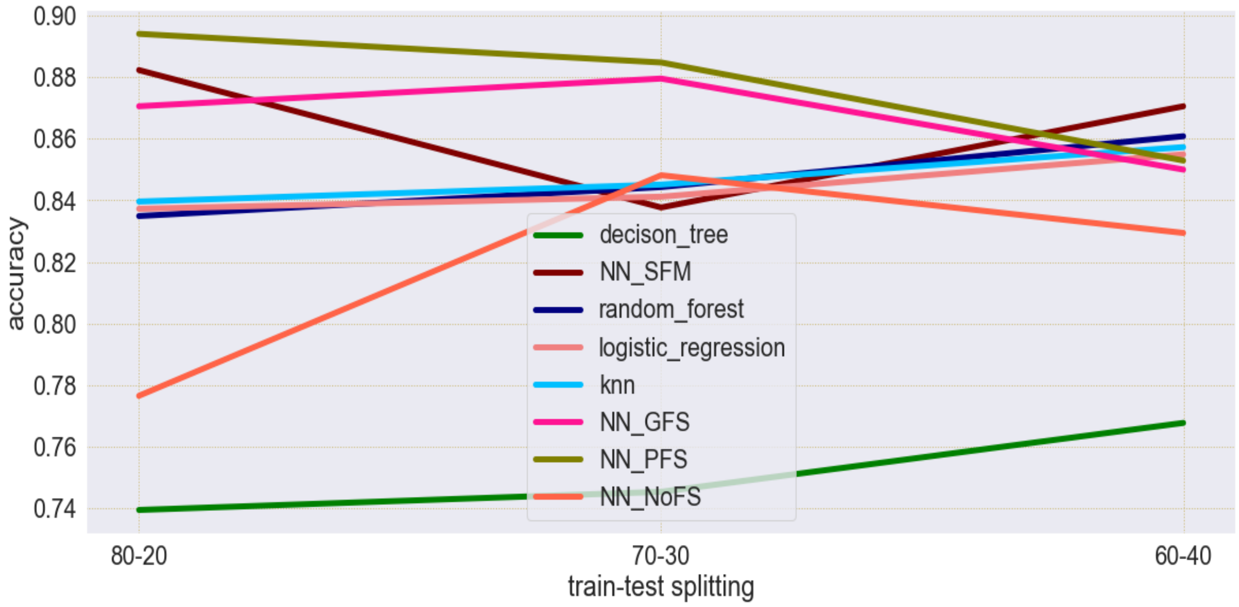


Figure 17: Comparison of final accuracies

Figure 17 shows overall compression of six machine learning algorithms such as Decision Tree Random Forest, logistic, and KNN machine learning algorithms for the heart disease prediction. As we can see in figure 18 the Random Forest curve increased gradually compared to all other models for every split. Whereas the performance of the decision tree is very low and the other two models performed equally and close to the random forest.

V. CONCLUSION AND FUTURE WORKS

In this research work, we highlighted the 10-year CHD using 15 feature vectors. The key attributes/feature vectors are selected using the Pearson correlation. The experimental accuracy achieved by the DT, LR, RF,KNN, ANN, NNPC is 75.88%, 85.49%, 86.08%, 85.73%, 85%, 89% respectively. The experimental result describes that the NNPC algorithm with 89.% accuracy is the most efficient algorithm for heart disease prediction. In future we will work on a hybrid model of machine learning with different feature selection techniques to improve the accuracy of the model. The future direction of work is to develop a more robust model by selecting different feature vector selection, data preprocessing techniques with different hybrid feasible hybrid combinations.

REFERENCE:

[1] Avinash Golande, Pavan Kumar T, Predicting Heart Disease Using Machine Learning Methods, International Journal of Recent Technology and Engineering, Vol 8, pages 944-950,2019.

[2] Fahd Saleh Alotaibi, Application of the Machine Learning Model for Predicting Cardiovascular Disease, (JACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 6, 2019.

[3] Theresa PrincyR, J. Thomas, Human Heart Disease Guarantee Program Using Data Mining Techniques, International Conference on Regional Energy and Computer Technology, Bangalore, 2016.

[4] Jianxin Chen, Guangcheng Xi, Yanwei Xing, Jing Chen, and Jie Wang 2007 “Predicting Disease by NEIS.

[5] Ms. Shouman, Tim Turner, Rob Stocker 2012 "Using Data Recording Methods for Cardiovascular Diagnosis and Treatment" Electronics, Communication and Computers (JECECC), 2012 Conference in Japan-Egypt March 2012, pages 173-177.

[6] Robert Detrano, Andras Janosi, Walter Steinbrunn, MatthiasFisterer, Johann-Jakob Schmid, Sarbjit Sandhu, Kern H. Guppy, Stella Lee, Victor Froelicher 1989 Journal of Cardiology, pages 304-310.15.

[7] T. Nagamani, S. Logeswari, B. Gomathy, Cardiovascular Guessing using Mining Data and MepReduce Algorithm, International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-3, January 2019.

[8] Mozaffarian, D., Benjamin, E., Go, A., Arnett, D., Blaha, M. Cushman, M. et al. (2015). Heart Disease and Stroke Statistics 2015, Update. Circulation, 131 (4). doi: 10.1161 / cir.00000000000000002.

[9] Cheung, N 2001 "Mechanical learning methods for medical analysis" School of Information Technology and Electrical Engineering, B.Sc. Thesis, University of Queensland.

[10] My Chau Tu, Dongil Shin, Dong Kyoo Shin 2009 "Successful Diagnosis of Cardiovascular Disease through the Wrapping Method " Biomedical Engineering and Informatics, 2009. BMEI '09. Second World Conference, pages 1-4.

[11] Lamia Abed Noor Muhammed 2012 "Using Data Mining Process to Diagnose Heart Disease" Electronics, Communications and Computers (JEC-ECC), 2012 Conference of Japan-Egypt on March 2012, pp. 173-177.

[12] Seyedamin Pouriyeh, Sara Vahid, Giovanna Sannino, Giuseppe De Pietro, Hamid Arabnia, Juan Gutierrez et al. "Complete Research and Comparison of Mechanical Learning Methods in the Heart Disease Center", 22nd IEEE Symposium on Computers and Communication (ISCC 2017): Workshops - ICTS4eHealth 2017.

[13] Hanen Bouali and Jalel Akaichi et al. “A study comparing different classification techniques, Cardiovascular Disease Using Case.”, 13th 2014 World Conference on Mechanical Learning and Applications.

[14] Simge EKIZ and PakizeErdogmusetal. "Comparative Study of the Diagnosis of Cardiovascular Diseases", 978-1-5386-0440-3 / 17 / $ 31.00 © 2017 IEEE.

[15] S. Rajathi and Drs. Radhamani et al. “Predicting and Analyzing Rheumatic Heart Disease using KNN and ACO Separation”, 2016.

[16] Miguel-Hurtado, O., Isimenywa, R., Stevenage, S., Neil, G., & Black, S. (2016). Comparing Machine Learning Classifiers with Linear / Logistics Regression to look at the relationship between hand size and personality traits. FIRST PLOS, 11 (11), at 0165521. doi: 10.1371 / journal.pone.0165521.

[17] Peng, C., Lee, K., & Ingersoll, G. (2002). Introduction to Logistic Regression Analysis and Reporting. Academic research journal, 96 (1), 3-14. doi: 10.1080 / 00220670209598786.

[18] Shan Xu, Tiangong Zhu, Zhen Zang, Daoxian Wang, Junfeng Hu noXiaohui Duan et al. “How to Estimate Cardiovascular Risk Assessment Based on CFS Subset Assessment and Random Forest Planning Framework”, 2017 IEEE 2nd International Conference on Big Data Analysis..

[19] Ahmad Shahin, Walid Moudani, FadiChakik, Mohamad Khalil et al. ”Data Extraction in Health Care Information Systems: Studies in Northern Lebanon”, ISBN: 978-1-4799-3166-8 © 2014 IEEE.

[20] WHO Identifies the Main Causes of Death and Disability Worldwide: 2000-2019., WHO, https://www.who.int/news/item/09-12-2020-who-reveals-leading-cuses-of-death - and disability-worldwide-2000-2019. Accessed Jan. 18. 2021.

[21] Top 12 heart functions with parts, location, and drawing, https://www.organsofthebody.com/heart. Accessed Jan. 18. 2021.

[22] Gerona, Aurelian, Hands-on Machine Learning and Scikit-Learn and TensorFlow: Ideas, Tools, and Strategies for Building Intelligent Programs. The first edition, O'Reilly Media, 2017