

# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



## PROJECT WORK-4 REPORT on

## “HEART DISEASE PREDICTION”

*Submitted by*

**Shivanshu Pande (1BM18CS151)**

**Samarth C Shetty(1BM18CS141)**

**Shweta Patil(1BM18CS156)**

**Rohan Siwach (1BM18CS131)**

*Under the Guidance of*

**Prof. Rekha G S**

**Assistant Professor, BMSCE**

*in partial fulfilment for the award of the degree of*

**BACHELOR OF ENGINEERING**

*in*

**COMPUTER SCIENCE AND ENGINEERING**



**B. M. S. COLLEGE OF ENGINEERING**

**BENGALURU-560019 April-2022 to July-2022**

**(Autonomous Institution under VTU)**

**B. M. S. College of Engineering,**  
**Bull Temple Road, Bangalore 560019**  
(Affiliated To Visvesvaraya Technological University, Belgaum)  
**Department of Computer Science and Engineering**



**CERTIFICATE**

This is to certify that the project work entitled “**Heart Disease Prediction**” carried out by **Shivanshu Pande(1BM19CS151)**, **Samarth C Shetty(1BM18CS141)**, **Shweta Patil(1BM18CS156)** AND **Rohan Siwach(1BM18CS131)** who are bonafide students of **B. M. S. College of Engineering**. It is in partial fulfilment for the award of **Bachelor of Engineering in Computer Science and Engineering** of the Visveswaraya Technological University, Belgaum during the year 2021. The project report has been approved as it satisfies the academic requirements in respect of **Project Work-4 (20CS6PWPW4)** work prescribed for the said degree.

Signature of the Guide  
**Prof. Rekha G S**  
Assistant Professor  
BMSCE, Bengaluru

Signature of the HOD  
**Dr. Jyothi S Nayak**  
Professor & Head, Dept. of CSE  
BMSCE, Bengaluru

External Viva

Name of the Examiner

Signature with date

1. \_\_\_\_\_

\_\_\_\_\_

2. \_\_\_\_\_

\_\_\_\_\_

**B. M. S. COLLEGE OF ENGINEERING DEPARTMENT OF COMPUTER  
SCIENCE AND ENGINEERING**



***DECLARATION***

We, **Shivanshu Pande(1BM19CS151), Samarth C Shetty(1BM18CS141), Shweta Patil(1BM18CS156) AND Rohan Siwach(1BM18CS131)**, students of 5th Semester, B.E, Department of Computer Science and Engineering, B. M. S. College of Engineering, Bangalore, here by declare that, this Project Work-4 entitled "**Heart Disease Prediction**" has been carried out by us under the guidance of Prof. Rekha G S, Assistant Professor, Department of CSE, B. M. S. College of Engineering, Bangalore during the academic semester Mar-2021-Jun-2021

We also declare that to the best of our knowledge and belief, the development reported here is not from part of any other report by any other students.

Signature

**Shivanshu Pande(1BM19CS151),**

**Samarth C Shetty(1BM18CS141)**

**Shweta Patil(1BM18CS156)**

**Rohan Siwach(1BM18CS131)**

# Chapter1 Introduction

The project we have taken up is based on Data Science and ML. The topic we have chosen heart disease prediction using machine learning algorithms. This project aims to establish a relationship between a medical Heart Attack and its direct relation on 14 prominent attributes using principles of Data Analysis and Visualization and then based on Mathematical Logistic Regression of variables find the predicted probability of Heart Attack using classification algorithms and concept like KNN, SVM, CNN. The tools used in this project are NumPy ,Pandas, MATLAB and matplotlib for data analysis Keras ,Python and Scikitlearn and TensorFlow for the ML part.

## 1.1 Motivation

Heart Attack is the primary reason for deaths in adults. Our project can help predict the people who are likely to be diagnosed with a heart disease by help of their medical history.By detecting someone at a risk of Heart Attack preventive measures can be taken also this will result in lesser casualty ,better lifespan and even revenue for Medical Institutions.

## 1.2 Scope of Project

This is a multivariate type of dataset which means providing or involving a variety of separate mathematical or statistical variables, multivariate numerical data analysis. It is composed of 14 attributes which are age, sex, chest pain type, resting blood pressure, serum cholesterol etc. We have Considered a total of 303 cases for a basic Model creation.

The goal of our heart disease prediction project is to determine if a patient should be diagnosed with heart disease or not, which is a binary outcome, so:

Positive result = 1, the patient will be diagnosed with heart disease.  
Negative result = 0, the patient will not be diagnosed with heart disease

### **1.3 Problem Statement**

A dataset is selected from the UCI repository with patient's medical history and attributes. By using this dataset, we predict whether the patient can have a heart disease or not. To predict this, we use 14 medical attributes of a patient and classify him if the patient is likely to have a heart disease. These medical attributes are trained under three algorithms: Logistic regression, KNN and Random Forest Classifier

## **Chapter 2 Literature Survey**

Main Reference Paper[25]

“Machine Learning is a way of Manipulating and extraction of implicit, previously unknown/known and potential useful information about data” [1]. Machine Learning is a very vast and diverse field and its scope and implementation is increasing day by day. Machine learning Incorporates various classifiers of Supervised, Unsupervised and Ensemble Learning which are used to predict and Find the Accuracy of the given dataset. We can use that knowledge in our project of HDPS as it will help a lot of people. Cardiovascular diseases are very common these days, they describe a range of conditions that could affect your heart. World health organization estimates that 17.9 million global deaths from (cardiovascular diseases) CVDs [2]. It is the primary reason of deaths in adults. Our project can help predict the people who are likely to diagnose with a heart disease by help of their medical history [6]. It recognizes who all are having any symptoms of heart disease such as chest pain or high blood pressure and can help in diagnosing disease with less medical tests and effective treatments, so that they can be cured accordingly. This project focuses on mainly three data mining techniques namely: (1) Logistic regression, (2) KNN and (3) Random Forest Classifier. The accuracy of our project is 87.5% for which is better than

previous system where only one data mining technique is used. So, using more data mining techniques increased the HDPS accuracy and efficiency. Logistic regression falls under the category of supervised learning. Only discrete values are used in logistic regression.

A cardiovascular disease detection model has been developed using three ML classification modelling techniques. This project predicts people with cardiovascular disease by extracting the patient medical history that leads to a fatal heart disease from a dataset that includes patients' medical history such as chest pain, sugar level, blood pressure, etc. This Heart Disease detection system assists a patient based on his/her clinical information of them been diagnosed with a previous heart disease. The algorithms used in building the given model are Logistic regression, Random Forest Classifier and KNN [22]. The accuracy of our model is 87.5%. Use of more training data ensures the higher chances of the model to accurately predict whether the given person has a heart disease or not [9]. By using this computer aided techniques, we can predict the patient fast and better and the cost can be reduced very much. There are a number of medical

A quiet Significant amount of work related to the diagnosis of Cardiovascular Heart disease using Machine Learning algorithms has motivated this work. This paper contains a brief literature survey. An efficient cardiovascular disease prediction has been made by using various algorithms some of them include Logistic Regression, KNN, Random Forest Classifier Etc. It can be seen in Results that each algorithm has its strength to register the defined objectives [7].

In this research [1], In this paper the problem of constraining and summarizing different algorithms of data mining used in the field of medical prediction are discussed. The focus is on using different algorithms and combinations of several target attributes for intelligent and effective heart attack prediction using data mining. For predicting heart attack, significantly 15 attributes are listed and with basic data mining technique other approaches

e.g. ANN, Time Series, Clustering and Association Rules, soft computing approaches etc. can also be incorporated. The outcome of

predictive data mining technique on the same dataset reveals that Decision Tree outperforms and some time Bayesian classification is having similar accuracy as of decision tree but other predictive methods like KNN, NeuralNetworks, Classification based on clustering are not performing well. The second conclusion is that the accuracy of the Decision Tree and Bayesian Classification further improves after applying genetic algorithm to reduce the actual data size to get the optimal subset of attribute sufficient for heart disease prediction

[3] In this research work association rules are mined on a medical data set to improve heart disease diagnosis. Each rule represents a simple predictive pattern that describes a subset of the data set projected on a subset of attributes. From a medical perspective, association rules relate combinations of binary target attributes (absence/existence of artery disease) and subsets of independent attributes (risk factors and heart muscle health measurements). Association rules have important advantages over traditional supervised machine learning or statistical algorithms (e.g. decision trees [8], [30], logistic regression [18], support vector machines [18]): they have a straightforward interpretation based on the probability of occurrence of a pattern and the conditional probability between two patterns (medical measurements and risk factors relationship to specific artery narrowing

In this study [2/9], The overall objective of our work is to predict more accurately the presence of heart disease. In this paper, two more input attributes obesity and smoking are used to get more accurate results. Three data mining classification techniques were applied namely Decision trees, Naive Bayes & Neural Networks. From results it has been seen that Neural Networks provides accurate results as compare to Decision trees & Naive Bayes.

This paper [4],In this paper we are proposing a heart disease prediction system using naïve bayes and k-means clustering. We are using k-means clustering for increasing the efficiency of the output. This is the most effective model to predict patients with heart disease. This model could

answer complex queries, each with its own strength with respect to ease of model interpretation, access to detailed information and accuracy .

[5]The objective of the proposed research is to make more accurate prediction of heart disease for a patient. Subsequently, three classifiers like Naïve Bayes, DT-GI and SVM are used to predict the heart disease of a patient given dataset of 13 attributes. Inconsistencies and missing values were also resolved before the model construction. Moreover, the prediction of heart disease is also computed for the proposed Ensemble technique using majority vote based technique. Observations exhibit that the accuracy of the proposed ensemble technique is much higher than the rest of techniques. The technique can be extended to identify the intensity of heart disease. Fuzzy learning models can be applied to predict the intensity of cardiac disease. Moreover, same framework can be used for multidisease prediction such as diabetes, breast cancer and liver disease diagnosis.

[6] The objectives of this study were to develop a coronary heart disease (CHD) risk model among the Korean Heart Study (KHS) population and compare it with the Framingham CHD risk score. The present study provides the first evidence that the Framingham risk function overestimates the risk of CHD in the Korean population where CHD incidence is low. The Korean CHD risk model is well-calculated alternations which can be used to predict an individual's risk of CHD and provides a useful guide to identify the groups at high risk for CHD among Koreans.

[7] Current guidelines do not support the use of genetic profiles in risk assessment of coronary heart disease (CHD). However, new single nucleotide polymorphisms associated with CHD and intermediate cardiovascular traits have recently been discovered. We aimed to compare several multilocus genetic risk score (MGRS) in terms of association with CHD and to evaluate clinical use.



This paper [10] This research paper intends to provide a survey of current techniques of knowledge discovery in databases using data mining techniques that are in use in today's medical research particularly in Heart Disease Prediction. Number of experiment has been conducted to compare the performance of predictive data mining technique on the same dataset and the outcome reveals that Decision Tree outperforms and some time Bayesian classification is having similar accuracy as of decision tree but other predictive methods like KNN, Neural Networks, Classification based on clustering are not performing well. The second conclusion is that the accuracy of the Decision Tree and Bayesian Classification further improves after applying genetic algorithm to reduce the actual data size to get the optimal subset of attribute sufficient for heart disease prediction.

In this paper [12], From our studies, we have managed to achieve our research objectives. Available dataset of Heart disease from UCI Machine Learning Repository has been studied and preprocessed and cleaned out to prepare it for classification process. Coactive Neuro-fuzzy modeling was proposed as a dependable and robust method developed to identify a nonlinear relationship and mapping between the different attributes. It has been shown that of GA is a very useful technique for auto-tuning of the CANFIS parameters and selection of optimal feature set. The fact is that computers cannot replace humans and by comparing the computer-aided detection results with the pathologic findings, doctors can learn more about the best way to evaluate areas that computer aided detection highlights.

This paper [13] is to perform a long term and continuous tracking of the electrocardiogram (ECG) of individuals to prevent hazardous or fatal heart related events. The ECG will help detect myocardial infarction, more commonly known as a heart attack, potentially hours before the user

would have sought medical treatment. This paper demonstrates that the transmission of Bluetooth signals in a BAN is possible with a simple antenna design. Through the use of creeping wave propagation, the signal can be made to reach locations on the body that are inaccessible without scattering off nearby objects. It has been experimentally shown in this paper that this mode of transmission meets the minimum requirement for Bluetooth signals. Thus, any Bluetooth device should be able to maintain contact with the GG BAN.

[20] The early diagnosis of MI can save life and can help to provide timely treatment. Thus, it is necessary to go for annual health checkups. The ECG is the primary tool to diagnose the electrical activity of the heart. Any abnormalities present in the heart activity is reflected in the ECG signals.

However, it is challenging and time-consuming to visually assess the ECG signals. Therefore, implementing a CAD system in clinical settings will ensure an objective and fast diagnosis of MI. In this work, we proposed a novel method to automatically diagnose MI using 11-layer deep CNN. We have used two different datasets (with and without noise) to evaluate the effectiveness of our proposed method. We have achieved an average accuracy, sensitivity, and specificity of 93.53%, 93.71%, and 92.83% respectively for ECG beats with noise. Our proposed system attained high-performance results even though there are noises present in the ECG beats. This suggests that our system can recognize the class of the ECG signals even with the presence of noise in the signal. Also, we obtained an average accuracy, sensitivity, and specificity of 95.22%, 95.49%, and 94.19% for ECG beats without noise. This shows that the overall performance of our proposed system is good enough and hence, can be introduced in clinical settings. Our proposed system can assist doctors in their diagnosis.

Like the above such 15 base papers all 24 papers have been ensemble and used to conduct the project.

# Chapter 3 Design

## 3.1 High Level Design

It is a data flow diagram that explains the architecture that will be utilized for the development of a system.

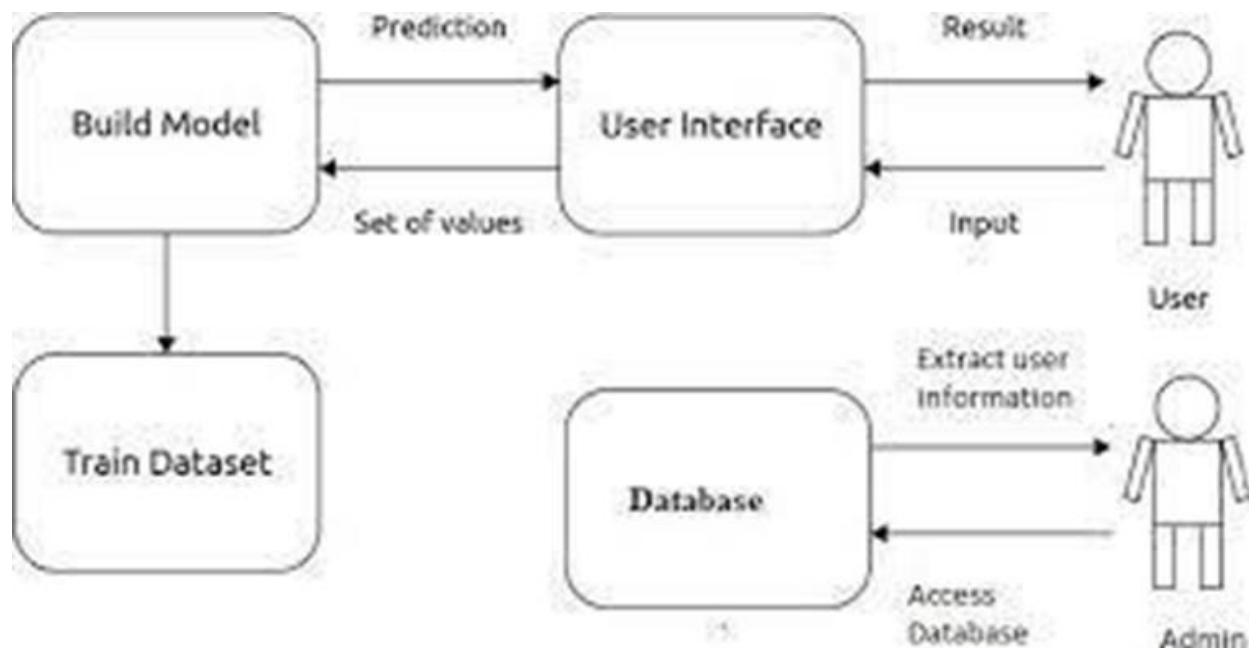


Figure 3.1: High Level Design

## 3.2 Sequence Diagram

It is the type of interaction architecture that explains the how the order in which a certain group of objects are implemented.

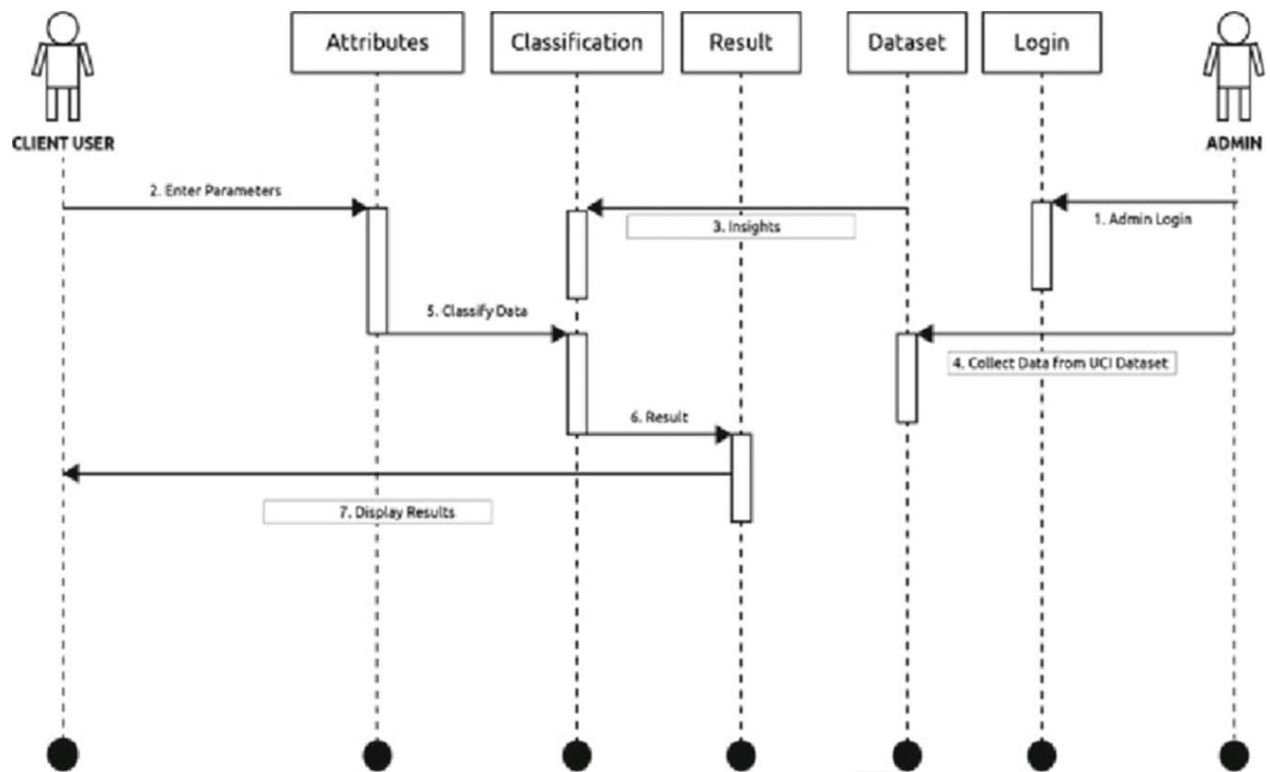


Figure 3.2: Sequence Diagram

### 3.3 Use Case Diagram

Use case diagrams provide us with the behavior of a system and helps us to attain therequirements and needs of a working system.

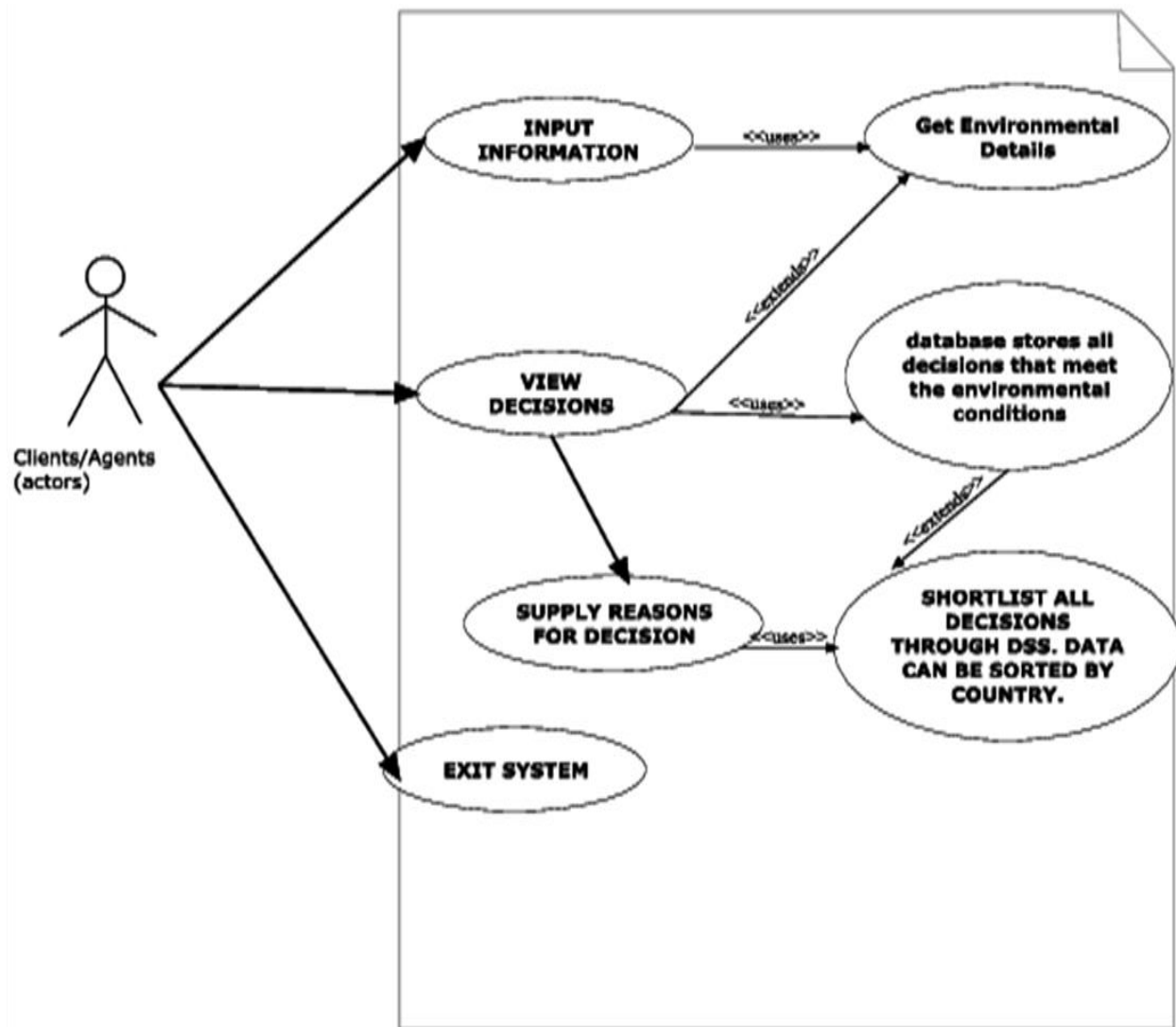


Figure 3.3: Use Case Diagram

# Chapter 4 Implementation

## 4.1 Proposed Methodology

In order to attain the purpose set for the venture, the primary system is to do enough historical past study, so the studies papers are considered for attaining a large amount of fundamental expertise withinside the field. The complete venture is primarily based totally on a large dataset of patients taken from the UCI database, so we selected a quantitative studies method.

### 4.1.1 KNN

K-Nearest Neighbors deduces the similarity among the brand new statistics and to be had statistics and applies the brand new case or statistics into the class this is maximum much like the to be had categories. It shops all of the to be had statistics and classifies a brand new statistics factor primarily based totally at the similarity.

- Select the range K of the neighbours
- Calculate the Euclidean distance of K number of neighbors
- Take the K nearest neighbors as per the calculated Euclidean distance.
- Among these k neighbors, count the number of the data points in each cate- gory.
- Assign the new data points to that category for which the number of the neighbor is maximum.

### 4.1.2 Logistic Regression algorithm

Logistic regression is an example of supervised learning. It is used to calculate or predict the probability of a binary (yes/no) event occurring.

Logistic Regression steps. Below are the steps:

- Data Preprocessing step
- Fitting Logistic Regression to the Training set
- Predicting the test result
- Test accuracy of the result(Creation of Confusion matrix)
- Visualizing the test set result.

Here we use the sigmoid concept for the logistic regression model creation:-

The sigmoid function is a mathematical function used to map the predicted values to probabilities. It maps any real value into another value within a range of 0 and 1. The value of the logistic regression must be between 0 and 1, which cannot go beyond this limit, so it forms a curve like the "S" form. The S-form curve is called the Sigmoid function or the logistic function. In logistic regression, we use the concept of the threshold value, which defines the probability of either 0 or 1. Such as values above the threshold value tends to 1, and a value below the threshold values tends to 0.

### **4.1.3 Neural Network**

Neural Networks procedures statistics in a completely comparable manner to what the human mind does. These networks absolutely examine from the examples that we offer them with, you can't software them to carry out a selected task. They will examine only from beyond studies in addition to examples, that is why you don't want to offer all of

the statistics concerning any unique task.

1. Information is fed into the input layer which transfers it to the hidden layer.
2. The interconnections between the two layers assign weights to each input randomly.
3. A bias added to every input after weights are multiplied with them individually.
4. The weighted sum is transferred to the activation function.
5. The activation function determines which nodes it should fire for feature extraction.
6. The model applies an application function to the output layer to deliver the output.
7. Weights are adjusted, and the output is back-propagated to minimize error.

The interconnections between the two layers assign weights to each input randomly. A bias added to every input after weights are multiplied with them individually. The weighted sum is transferred to the activation function. The activation function determines which nodes it should fire for feature extraction. The model applies an application function to the output layer to deliver the output. Weights are adjusted, and the output is back-propagated to minimize error.

#### 4.1.4 SVM

Support Vector Machine or SVM is used for Classification in addition to Regression problems. The purpose of the set of rules is to create the fine line or selection boundary which can segregate n-dimensional area into instructions so that we are able to, without problems place the new statistics factor into the perfect classes so that we can easily put the new data in the correct category in the future. SVM chooses the extreme factors/vectors that assist in developing the hyperplane. We can without problems look at the “margins” in the discriminative classifiers. SVM will pick the line that



maximizes the margin. Next, we are able to use Scikit-Learn's support vector classifier to educate an SVM version in these statistics.

- We can easily observe the “margins” within the discriminative classifiers. SVM will choose the line that maximizes the margin.
- Next, we will use Scikit-Learn's support vector classifier to train an SVM model on this data.

#### **4.1.5 Random Forest Classifier**

The random forest is a classification algorithm consisting of many decision trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree.

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction.

We can understand the working of Random Forest algorithm with the help of following steps –

- Step 1 – First, start with the selection of random samples from a given dataset.
- Step 2 – Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
- Step 3 – In this step, voting will be performed for every predicted result.

Step 4 – At last, select the most voted prediction result as the final prediction result

#### **4.1.6 Decision tree classifier**

Decision trees use multiple algorithms to decide to split a node into two or more sub-nodes. The creation of sub-nodes increases the homogeneity of

resultant sub-nodes. In other words, we can say that the purity of the node increases with respect to the target variable. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features.

The decision rules are generally in form of if-then-else statements. deeper the tree, the more complex the rules and fitter the model

A general algorithm for a decision tree can be described as follows:

1. Pick the best attribute/feature. The best attribute is one which best splits or separates the data.
2. Ask the relevant question.
3. Follow the answer path.
4. Go to step 1 until you arrive to the answer.

The best split is one which separates two different labels into two sets.

## **4.2 Tools and Technologies Used**

Our local computers. Therefore minimum 4 GB (64-bit) RAM and 20 GB (64-bit) hard disk space will be needed which can run on Windows or Mac OS.

Anaconda Navigator - Jupyter Notebook

UCI API/DataOrg API - for UCI dataset

Python

Kaggle

Machine Learning Algorithms

TensorFlow Software-Extension.

## 4.3 Testing

The initial data was split into 67-33 proportion where the 33% of the data was used as testing data. The rest 67% was used as the main dataset

```
X=df.iloc[:,0:13].values
y=df['output'].values
from sklearn.model_selection import train_test_split
X_train, X_test,y_train, y_test=train_test_split(X,y,test_size=0.33,random_state=40)

from sklearn.model_selection import cross_val_score, GridSearchCV
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
model1=lr.fit(X_train,y_train)
prediction1=model1.predict(X_test)
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test,prediction1)
cm
sns.heatmap(cm, annot=True,cmap='winter',linewidths=0.3, linecolor='black',annot_kws={"size": 20})
TP=cm[0][0]
```

Figure 4.1: Logistic Regression

```
from sklearn.model_selection import RandomizedSearchCV
from sklearn.tree import DecisionTreeClassifier

tree_model = DecisionTreeClassifier(max_depth=5,criterion='entropy')
cv_scores = cross_val_score(tree_model, X, y, cv=10, scoring='accuracy')
m=tree_model.fit(X, y)
prediction=m.predict(X_test)
cm= confusion_matrix(y_test,prediction)
sns.heatmap(cm, annot=True,cmap='winter',linewidths=0.3, linecolor='black',annot_kws={"size": 20})
print(classification_report(y_test, prediction))

TP=cm[0][0]
```

Figure 4.2: Decision Tree classifier

```
[ ] from sklearn.neighbors import KNeighborsClassifier

model = KNeighborsClassifier(n_neighbors=5)
error = []
# Calculating error for K values between 1 and 30
for i in range(1, 30):
    knn = KNeighborsClassifier(n_neighbors=i)
    knn.fit(x_train, y_train)
    pred_i = knn.predict(x_test)
    error.append(np.mean(pred_i != y_test))
plt.figure(figsize=(12, 6))
plt.plot(range(1, 30), error, color='red', linestyle='dashed', marker='o',
         markerfacecolor='blue', markersize=10)
plt.title('Error Rate K Value')
plt.xlabel('K Value')
plt.ylabel('Mean Error')
print("Minimum error:-",min(error),"at K =",error.index(min(error))+1)
```

Figure 4.3: KNN Algorithm-We got k value as 7[shown in Results]

```
➤ classifier= KNeighborsClassifier(n_neighbors=7)
classifier.fit(x_train, y_train)
y_pred= classifier.predict(x_test)
from sklearn.metrics import confusion_matrix
cm= confusion_matrix(y_test, y_pred)
print(cm)
```

Figure 4.4: KNN Algorithm[contd]

```
[ ] from sklearn.metrics import accuracy_score
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier(n_estimators=500,criterion='entropy',max_depth=8,min_samples_split=5)
model3 = rfc.fit(X_train, y_train)
prediction3 = model3.predict(X_test)
cm3=confusion_matrix(y_test, prediction3)
sns.heatmap(cm3, annot=True,cmap='winter',linewidths=0.3, linecolor='black',annot_kws={"size": 20})
TP=cm3[0][0]
TN=cm3[1][1]
FN=cm3[1][0]
FP=cm3[0][1]
print(round(accuracy_score(prediction3,y_test)*100,2))
print('Testing Accuracy for Random Forest:',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Random Forest:',(TP/(TP+FN)))
print('Testing Specificity for Random Forest:',(TN/(TN+FP)))
print('Testing Precision for Random Forest:',(TP/(TP+FP)))
```

Figure 4.5: Random Forest Classifier

```
from sklearn.svm import SVC
svm=SVC(C=12, kernel='linear')
model4=svm.fit(X_train,y_train)
prediction4=model4.predict(X_test)
cm4= confusion_matrix(y_test,prediction4)
sns.heatmap(cm4, annot=True, cmap='winter', linewidths=0.3, linecolor='black', annot_kws={"size": 20})
TP=cm4[0][0]
TN=cm4[1][1]
FN=cm4[1][0]
FP=cm4[0][1]

print('Testing Accuracy for SVM:', (TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Random Forest:', (TP/(TP+FN)))
print('Testing Specificity for Random Forest:', (TN/(TN+FP)))
print('Testing Precision for Random Forest:', (TP/(TP+FP)))
```

Figure 4.6: SVM Classifier

```
[ ] from keras.models import Sequential
    from keras.layers import Dense
    from keras.optimizers import Adam

    # define a function to build the keras model
    def create_model():
        # create model
        model = Sequential()
        model.add(Dense(8, input_dim=13, kernel_initializer='normal', activation='relu'))
        model.add(Dense(4, kernel_initializer='normal', activation='relu'))
        model.add(Dense(2, activation='softmax'))

        # compile model
        adam = Adam(lr=0.001)
        model.compile(loss='categorical_crossentropy', optimizer=adam, metrics=['accuracy'])
        return model

    model = create_model()

    print(model.summary())
```

Figure 4.7 CNN-NEURAL NETWORK

## Chapter 5

# Results and Discussion

In our case, we used 6 Machine Learning algorithms and conducted a comparative analysis amongst the algorithms. Upon conducting the analysis,

Decision Tree Classifier accuracy was found to be 97% [HIGHEST] and Neural Network accuracy was about 90-91% , then was Support Vector Machine [svm] at 88% accuracy followedby Logistic Regression 89% accuracy , RandomForestClassifier[85%] and KNN Algorithm[86.84%] both at almost same accuracy.


```
[ ] # fit the model to the training data
model.fit(X_train, Y_train, epochs=2268, batch_size=9, verbose = 1)

27/27 [=====] - 0s 1ms/step - loss: 0.2861 - accuracy: 0.8802
Epoch 1525/2268
27/27 [=====] - 0s 1ms/step - loss: 0.2849 - accuracy: 0.8884
27/27 [=====] - 0s 2ms/step - loss: 0.2598 - accuracy: 0.9050
Epoch 2268/2268
27/27 [=====] - 0s 2ms/step - loss: 0.2451 - accuracy: 0.9174
<keras.callbacks.History at 0x22d59eb2d90>
```

Figure 5.1: Neural Networks

KNN:-

The accuracy of the KNN model was found to be approximately 86.84% [h] [h]



```
[[26  7]
 [ 3 40]]
```

```
[ ] from sklearn.metrics import accuracy_score
accuracy_score(y_test, y_pred)
```

```
0.868421052631579
```

Figure 5.2: KNN

Minimum error:- 0.13157894736842105 at K = 7

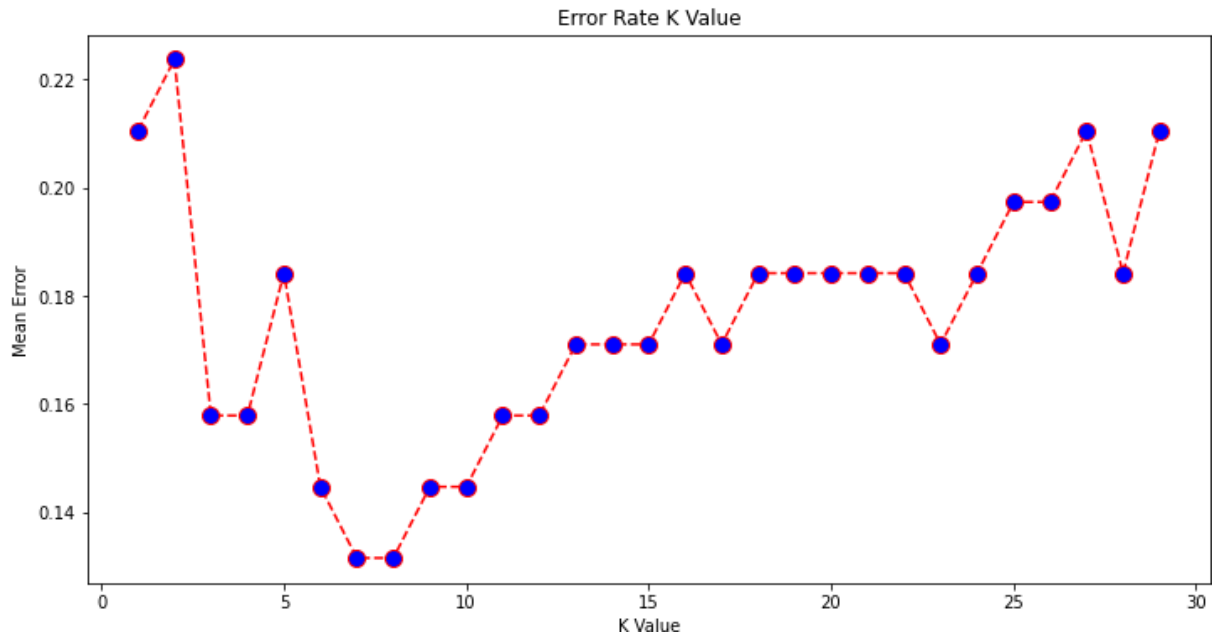


Figure 5.3: KNN Error rate.

The accuracy of the Logistic Regression model was found to be approximately 89%

```
print('Testing Accuracy for Logistic Regression:',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Logistic Regression:',(TP/(TP+FN)))
print('Testing Specificity for Logistic Regression:',(TN/(TN+FP)))
print('Testing Precision for Logistic Regression:',(TP/(TP+FP)))
```

C:\ANACONDA\lib\site-packages\sklearn\linear\_model\\_logistic.py:763: ConvergenceWarning: lbfgs failed to converge STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max\_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html>

Please also refer to the documentation for alternative solver options:

[https://scikit-learn.org/stable/modules/linear\\_model.html#logistic-regression](https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)

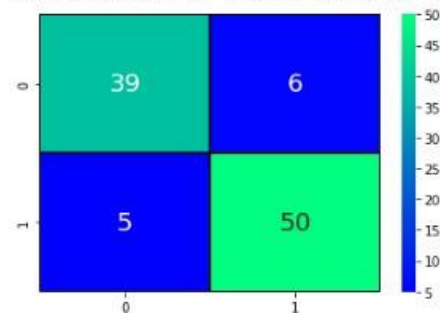
```
n_iter_i = _check_optimize_result(
```

```
Testing Accuracy for Logistic Regression: 0.89
```

```
Testing Sensitivity for Logistic Regression: 0.8863636363636364
```

```
Testing Specificity for Logistic Regression: 0.8928571428571429
```

```
Testing Precision for Logistic Regression: 0.8666666666666667
```



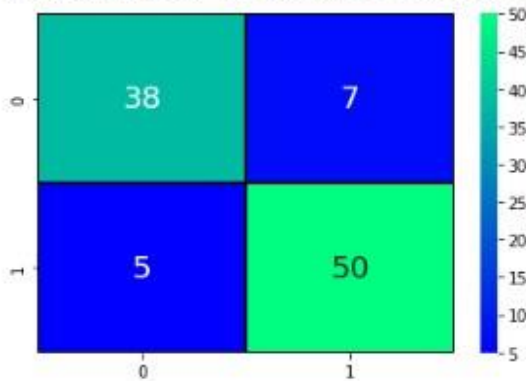


#### 5.4: Logistic Regression The accuracy of the SVM model was found to be approximately

```
TN=cm4[1][1]
FN=cm4[1][0]
FP=cm4[0][1]

print('Testing Accuracy for SVM:',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Random Forest:',(TP/(TP+FN)))
print('Testing Specificity for Random Forest:',(TN/(TN+FP)))
print('Testing Precision for Random Forest:',(TP/(TP+FP)))
```

Testing Accuracy for SVM: 0.88  
 Testing Sensitivity for Random Forest: 0.8837209302325582  
 Testing Specificity for Random Forest: 0.8771929824561403  
 Testing Precision for Random Forest: 0.8444444444444444



88%Figure 5.5: SVM -Linear

```
model3 = rfc.fit(X_train, y_train)
prediction3 = model3.predict(X_test)
cm3=confusion_matrix(y_test, prediction3)
sns.heatmap(cm3, annot=True,cmap='winter',linewidths=0.3, linecolor='black',annot_kws={"size": 20})
TP=cm3[0][0]
TN=cm3[1][1]
FN=cm3[1][0]
FP=cm3[0][1]
print(round(accuracy_score(prediction3,y_test)*100,2))
print('Testing Accuracy for Random Forest:',(TP+TN)/(TP+TN+FN+FP))
print('Testing Sensitivity for Random Forest:',(TP/(TP+FN)))
print('Testing Specificity for Random Forest:',(TN/(TN+FP)))
print('Testing Precision for Random Forest:',(TP/(TP+FP)))
```

85.0  
 Testing Accuracy for Random Forest: 0.85  
 Testing Sensitivity for Random Forest: 0.8409090909090909  
 Testing Specificity for Random Forest: 0.8571428571428571  
 Testing Precision for Random Forest: 0.8222222222222222

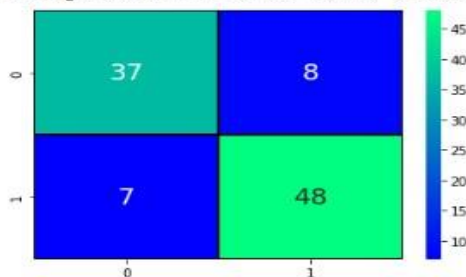


Figure 5.6: Random Forest Classifier





Figure 5.7: Decision Tree Classifier

## Chapter 6

# Conclusion and Future Work

In conclusion, the comparative analysis between 6 Machine Learning models were conducted. The 6 algorithms taken for the comparative analysis are:

**KNN Algorithm**

- **Logistic Regression Algorithm**

**Neural Networks**

- **Support Vector Machine Classifier (SVM)**

**Random Forest Classifier**

- **Decision Tree Classifier**

In the comparative analysis conducted, after collecting the raw data of more than 300+ dataset with sufficient attributes and acceptable ratio of Binary Target Function, it was found that the Decision Tree and CNN gave the highest accuracy out of all the models with a prediction rate of 97% and 90.2%[avg].

1. **Use more data for processing** The data that we've used for the data processing is unarguably an excellent dataset however 302 datasets is too small for a concise and stable model and thus will not give us a very high or highly accurate prediction rate in long term Real-Time-Application. In order to get more accurate results we need to use more datasets[at least in order of 1000s] rather than the data taken direct from the database[UGI-Dataset].
2. **Use newer Data as Medical /Scientific Study is involved** Also this Dataset Despite being recent is still years old and in medical Field things keep changing and so must data and its attributes ,attribute correlation .

## Bibliography

- [1] [1] Soni J, Ansari U, Sharma D & Soni S (2011). Predictive data mining for medical diagnosis: an overview of heart disease prediction. *International Journal of Computer Applications*, 17(8), 43-8
- [2] [2] Dangare C S & Apte S S (2012). Improved study of heart disease prediction system using data mining classification techniques. *International Journal of Computer Applications*, 47(10), 44-8.
- [3] [3] Ordonez C (2006). Association rule discovery with the train and test approach for heart disease prediction. *IEEE Transactions on Information Technology in Biomedicine*, 10(2), 334-43.
- [4] [4] Shinde R, Arjun S, Patil P & Waghmare J (2015). An intelligent heart disease prediction system using k-means clustering and Naïve Bayes algorithm. *International Journal of Computer Science and Information Technologies*, 6(1), 637-9.
- [5] [5] Bashir S, Qamar U & Javed M Y (2014, November). An ensemble-based decision support framework for intelligent heart disease diagnosis. In *International Conference on Information Society (i-Society 2014)* (pp. 259-64). IEEE. ICCRDA 2020 IOP Conf. Series: Materials Science and Engineering 1022 (2021) 012072 IOP Publishing doi:10.1088/1757-899X/1022/1/012072 9
- [6] [6] Jee S H, Jang Y, Oh D J, Oh B H, Lee S H, Park S W & Yun Y D (2014). A coronary heart disease prediction model: the Korean Heart Study. *BMJ open*, 4(5), e005025.
- [7] [7] Ganna A, Magnusson P K, Pedersen N L, de Faire U, Reilly M, Ärnlöv J & Ingelsson E (2013). Multilocus genetic risk scores for coronary heart disease prediction. *Arteriosclerosis, thrombosis, and vascular biology*, 33(9), 2267-72.
- [8] [8] Jabbar M A, Deekshatulu B L & Chandra P (2013, March). Heart disease prediction using lazy associative classification. In *2013 International Mutli-Conference on Automation, Computing, Communication, Control*.

- [9] [9] Dangare Chaitrali S and Sulabha S Apte. "Improved study of heart disease prediction system using data mining classification techniques." *International Journal of Computer Applications* 47.10 (2012): 44-8.
- [10] [10] Soni Jyoti. "Predictive data mining for medical diagnosis: An overview of heartdisease prediction." *International Journal of Computer Applications* 17.8 (2011): 43-8.
- [11] [11] Chen A H, Huang S Y, Hong P S, Cheng C H & Lin E J (2011, September). HDPS: Heart disease prediction system. In *2011 Computing in Cardiology* (pp. 557-60). IEEE.
- [12] [12] Parthiban, Latha and R Subramanian. "Intelligent heart disease prediction system using CANFIS and genetic algorithm." *International Journal of Biological, Biomedical and Medical Sciences* 3.3 (2008).
- [13] [13] Wolgast G, Ehrenborg C, Israelsson A, Helander J, Johansson E & Manefjord H (2016). Wireless body area network for heart attack detection [Education Corner]. *IEEE antennas and propagation magazine*, 58(5), 84-92.
- [14] [14] Patel S & Chauhan Y (2014). Heart attack detection and medical attention using motion sensing device -kinect. *International Journal of Scientific and Research Publications*, 4(1), 1-4.
- [15] [15] Zhang Y, Fogoros R, Thompson J, Kenknight B H, Pederson M J, Patangay A & Mazar S T (2011). U.S. Patent No. 8,014,863. Washington, DC: U.S. Patent and Trademark Office.
- [16] [16] Raihan M, Mondal S, More A, Sagor M O F, Sikder G, Majumder M A & Ghosh K (2016, December). Smartphone based ischemic heart disease (heart attack) risk prediction using clinical data and data mining approaches, a prototype design. In *2016 19th International Conference on Computer and Information Technology (ICCIT)* (pp. 299-303). IEEE.
- [17] [17] Buechler K F & McPherson P H (1999). U.S. Patent No. 5,947,124. Washington, DC: U.S. Patent and Trademark Office.
- [18] [18] Takci H (2018). Improvement of heart attack prediction by the feature selection methods. *Turkish Journal of Electrical Engineering & Computer Sciences*, 26(1), 1-10.

- [19] [19] Worthen W J, Evans S M, Winter S C & Balding D (2002). U.S. Patent No. 6,432, 124. Washington, DC: U.S. Patent and Trademark Office.
- [20] [20] Acharya U R, Fujita H, Oh S L, Hagiwara Y, Tan J H & Adam M (2017). Application of deep ICCRDA 2020 IOP Conf. Series: Materials Science and Engineering 1022 (2021) 012072 IOP Publishing doi:10.1088/1757-899X/1022/1/012072 10 convolutional neural network for automated detection of myocardial infarction using ECG signals. Information Sciences, 415, 190-8.
- [21] [21] Brown N, Young T, Gray D, Skene A M & Hampton J R (1997). Inpatient deaths from acute myocardial infarction, 1982-92: analysis of data in the Nottingham heart attack register. BMJ, 315(7101), 159-64.
- [22] [22] Piller L B, Davis B R, Cutler J A, Cushman W C, Wright J T, Williamson J D & Haywood L J (2002). Validation of heart failure events in the Antihypertensive and Lipid Lowering Treatment to Prevent Heart Attack Trial (ALLHAT) participants assigned to doxazosin and chlorthalidone. Current controlled trials in cardiovascular medicine, 3(1), 10.
- [23] [23] Folsom A R, Prineas R J, Kaye S A & Soler J T (1989). Body fat distribution and self-reported prevalence of hypertension, heart attack, and other heart disease in older women. International journal of epidemiology, 18(2), 361-7.
- [24] [24] Kiyasu J Y (1982). U.S. Patent No. 4,338,396. Washington, DC: U.S. Patent and Trademark Office.
- [25] Harshit Jindal. Heart disease prediction using machine learning algorithms. IOP Conference Series: Materials Science and Engineering