**A Project Report**

**On**

**ANTICIPATING CORONARY HEART DISEASE RISK WITH ADVANCED ANALYTICS**

A project Report submitted in Partial Fulfillment of the requirements for the Award of the Degree of

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted by**

**SHOVANJEET KAR : 2001206067**

**FATMA DARAKSHA :2001206017**

**ANANYA ANUCHHANDA :2001206037**

**SWARNALATA SAMAL :2001206072**

UNDER THE GUIDENCE OF

**Assist. Prof. PRATAP KUMAR CHAMPATI**

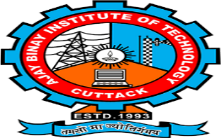
**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



**AJAY BINAY INSTITUTE OF TECHNOLOGY**

**CUTTACK-753014**

**(2020-2024) BATCH**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK**

**CERTIFICATE OF APPROVAL**

The foregoing project named “**Anticipating Coronary Heart Disease Risk with Advanced Analytics**” is a Bonafede work carried out by

**SHOVANJEET KAR :2001206067**

**FATMA DARAKSHA :2001206017**

**ANANYA ANUCHHANDA :2001206037**

**SWARNALATA SAMAL :2001206072**

In the partial fulfillment for the award of the degree of Bachelor of Technology in Computer Science & Engineering. Ajay Binay Institute of Technology, Cuttack of Biju Pattnaik University of Technology, Odisha in the year 2024-2025 is an authentic work carried out under our guidance and supervision.

The matter embodied in this project has not been submitted to any other university/institute for the award of any degree to the best of our knowledge.

**Prof. Dr. Rajesh Kumar Sahoo**

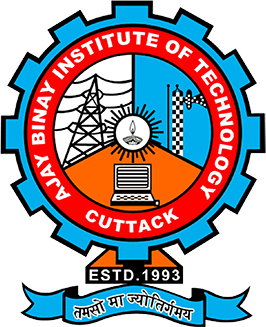
Prof., Head, Department of Computer Science and Engineering

Countersigned by:

**Assist. Prof. Pratap Kumar Champati**

Prof., Head, Department of Computer Science and Engineering ABIT,

Cuttack-753014



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK**

**ACKNOWLEDGEMENT**

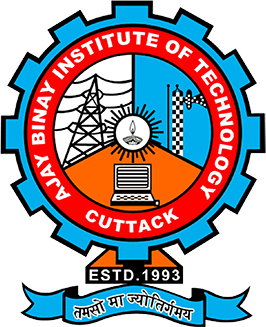
The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mentioning of the people whose constant guidance and encouragement made it possible. We take pleasure in presenting before you, our project, which is result of studied blend of both research and knowledge.

We express my earnest gratitude to **Assist. Prof. PRATAP KUMAR CHAMPATI** our project guide, for her constant support, encouragement, and guidance. We are grateful for his cooperation and valuable suggestions.

We feel to avail ourselves of this opportunity to express our deep sense of gratitude to **Assist. Prof. PRATAP KUMAR CHAMPATI**, for the facilities made available and instructions given to us in accomplishing this project successfully.

Finally, we express our gratitude to all other members who are involved either directly or indirectly for the completion of this project.

**SHOVANJEET KAR : 2001206067**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK**

**CERTIFICATE**

This is to certify that the project report entitled “**Anticipating Coronary Heart Disease Risk with Advanced Analytics**” is the work done by:

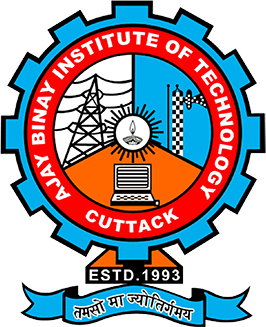
**SHOVANJEET KAR : 2001206067**

Of B.Tech (Computer Science and Engineering) of ABIT, Cuttack under BPUT, Odisha submitted in partial fulfillment for the award of degree.

We are satisfied that they have worked sincerely and with proper care.

**Signature of Supervisor/Guide Signature of Examiner Signature of H.O.D**

**Assist. Prof. Pratap Kumar Champati Prof. Dr. Rajesh Kumar Sahoo**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**AJAY BINAY INSTITUTE OF TECHNOLOGY, CUTTACK**

**DECLARATION**

We do hereby declare that the project entitled, “**Anticipating Coronary Heart Disease Risk with Advanced Analytics**” submitted in the Department of Computer Science & Engineering, Ajay Binay Institute of Technology, Cuttack of Biju Pattnaik University of Technology, Odisha in partial fulfillment of requirement for the award of 8th semester B. Tech Degree in Computer Science and Engineering is an authentic work carried out by us during 2024-2025 under the supervision of **Prof. PRATAP KUMAR CHAMPATI.** The matter presented in this report has not been submitted by us in any other University/Institute for the award of B. tech Degree.

**Submitted By**

**SHOVANJEET KAR : 2001206067**

**Anticipating Coronary Heart Disease Risk with Advanced Analytics**

**ABSTRACT**

Heart disease is one of the most significant causes of mortality in the world today. Prediction of cardiovascular disease is a critical challenge in the area of clinical data analysis. The diagnosis of heart disease through traditional medical history has been considered as not reliable in many aspects. The health care industry produces a huge amount of data and these data is not always made use to the full extent and is often underutilized. Using this huge amount of data, a disease can be detected, predicted or even cured. In this project, we have developed a Machine Learning based diagnosis system for heart disease prediction by taking the help of medical data such as Blood pressure, cholesterol, body mass index, smoking habit, heart rate etc as input and then these features are modelled for prediction. This model can then be used to predict future medical data. Here we perform the comparative analysis of the algorithms like decision tree, Naïve Bayes, Logistic Regression, KNearest Neighbors .Then the accuracy of the model using each of the algorithms will be calculated and the one having the best accuracy will be taken as the model of predicting the heart disease and finally a GUI will be designed to predict the disease.

Tools used: - Anaconda Framework

Editor: - Jupiter Notebook

Packages Used: -Pandas, numpy, matplotlib, seaborn, sklearn

Contents

1.INTRODUCTION 11

1.1 What is coronary heart disease?....................................................................11

[1.1.1 Causes 12](#_Toc66443453)

[1.1.2 Symptoms 13](#_Toc66443455)

[1.2 What is Machine learning? 14](#_Toc66443456)

[1.2.1 Application of Machine learning 14](#_Toc66443457)

[1.2.2 Types of Machine Learning 16](#_Toc66443458)

[1.3 Supervised Learning 16](#_Toc66443459)

[Algorithms](#_Toc66443460) 19

[1.3.1 Logistic regression: 19](#_Toc66443461)

[Pros and Cons:](#_Toc66443462)

[Multiclass Classification](#_Toc66443464)

[1.3.2Naïve Bayes Classifier 27](#_Toc66443470)

[What is Bayes Theorem?](#_Toc66443471)

[Gaussian Naive Bayes](#_Toc66443472)

[MultiNomial Naive Bayes 29](#_Toc66443473)

[Bernoulli Naive Bayes 29](#_Toc66443474)

[Pros and Cons of Naïve Bayes: 29](#_Toc66443476)

[1.3.3 Decision Tree 30](#_Toc66443477)

[1.3.4 K-Nearest Neighbor(KNN): 33](#_Toc66443481)

[1.3.5 Support Vector Machine Algorithm 40](#_Toc66443482)

[1.3.6 Random Forest Algorithm 47](#_Toc66443483)

1.4 Unsupervised Learning……………………………………………………..47

[1.4.1 Clustering 48](#_Toc66443485)

[1.5 Reinforcement Learning 48](#_Toc66443486)

[1.6 Objective 49](#_Toc66443487)

2. Literature Survey………………………………………………………………………………………….50

[3.Problem definition and algorithm 51](#_Toc66443488)

[3.1 Task definition 51](#_Toc66443489)

[3.2 Heartdisease data set 51](#_Toc66443490)

[4. Experimental set up and evaluation 54](#_Toc66443491)

[4.1 Components to be installed: 55](#_Toc66443492)

[4.2 Methodology 56](#_Toc66443493)

[4.2.1 Collect the dataset: 57](#_Toc66443494)

[4.2.2 Load the dataset*:* 57](#_Toc66443495)

[4.2.3 Describe the dataset*:* 61](#_Toc66443496)

[4.2.4 Heat Map 61](#_Toc66443498)

[4.2.5 Analysis of the dataset for cleaning purpose: 62](#_Toc66443500)

[Male and Female countplot 48](#_Toc66443504)

[3.3 Model building 52](#_Toc66443505)

[3.3.1 Split the dataset for training and testing: 52](#_Toc66443507)

[3.4Train the dataset using algorithms 53](#_Toc66443508)

[3.4.1 Using KNN Algorithm 53](#_Toc66443509)

[3.4.2 Using Decision Tree Algorithm 54](#_Toc66443510)

[3.4.3 Using Logistic Regression Algorithm 55](#_Toc66443511)

[3.4.4 Using Naïve Bayes Algorithm 56](#_Toc66443512)

[3.4.5 Using Random Forest Algorithm 57](#_Toc66443512)

[3.4.6 Using SVM Algorithm 57](#_Toc66443512)

[3.5 Compare the accuracy of all the Algorithm 57](#_Toc66443513)

[3.6 Graphical Representation 58](#_Toc66443514)

[5.Project Code 62](#_Toc66443515)

[6.GUI Implementation 70](#_Toc66443516)

7. Conclusion……………………………………………………………………………………………………73

8. Future Work………………………………………………………………………………………………….73

9. Reference……………………………………………………………………………………………………..74

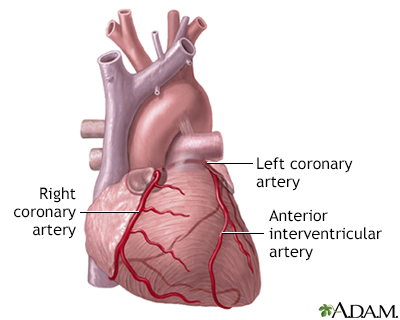
**1.INTRODUCTION**

The Heart is one of the most important organs in the human body. It is the centre of the circulatory system. The heart functions as a pump that propels blood to different parts of the human body through a network of blood vessels, supplying a constant supply of oxygen as well as other vital nutritional components. If the heart ever stops functioning and ceases to pump blood, the body will shut down and within very less time a person will expire.

## 1.1 **What is** coronary heart disease?

Coronary heart disease (CHD) is also called as coronary artery disease. The coronary arteries are the blood vessels that carry blood to your heart. Coronary artery disease is the narrowing or blockage of the coronary arteries. This condition is usually caused by atherosclerosis. Atherosclerosis is the build-up of cholesterol and fatty deposits (called plaques) inside the arteries. These plaques can clog the arteries or damage the arteries, which limits or stops blood flow to the heart muscle.

If the heart does not get enough blood, it cannot get the oxygen and nutrients it needs to work properly. This can cause chest pain (angina) or a heart attack.

Coronary artery disease (CAD) is the most common type of heart disease in the United States. It is sometimes called ischemic heart disease.

coronary arteries supply blood to the heart muscle. The right coronary artery supplies both the left and right heart, the left coronary artery supplies the left heart.

### 1.1.1 Causes

Coronary artery disease is thought to begin with damage or injury to the inner layer of a coronary artery, sometimes as early as childhood. The damage may be caused by various factors, including:

* Smoking
* High blood pressure
* High cholesterol
* Diabetes or insulin resistance
* Not being active (sedentary lifestyle)

Once the inner wall of an artery is damaged, fatty deposits (plaque) made of cholesterol and other cellular waste products tend to collect at the site of injury. This process is called atherosclerosis. If the plaque surface breaks or ruptures, blood cells called platelets clump together at the site to try to repair the artery. This clump can block the artery, leading to a heart attack.

**1.1.2 Symptoms**

An acute coronary event, such as a heart attack, may cause the following symptoms:

* Cold sweats
* Dizziness
* Light-headedness
* Nausea or a feeling of indigestion
* Neck pain
* Shortness of breath, especially with activity
* Sleep disturbances
* Weakness

The usage of information technology in health care industry is increasing day by day to aid doctors in decision making activities. It helps doctors and physicians in disease management, medications and discovery of patterns and relationships among diagnosis data. Current approaches to predict cardiovascular risk fail to identify many people who would benefit from preventive treatment, while others receive unnecessary intervention. Machine-learning offers opportunity to improve accuracy by exploiting complex interactions between risk factors. We assessed whether machine-learning can improve cardiovascular risk prediction.

## 1.2 What is Machine learning?

Machine learning is a subfield of artificial intelligence (AI).Artificial intelligence (AI) refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. Machine learning is the set of different methods that can be used to find patterns from the dataset and then use those patterns to predict future conditions or to make efficient decisions under some conditions. Machine learning introduces different algorithms that can be used to make understanding of the current situations to machine and then based upon that machines can take decisions. Machine learning can make decisions independently at its own.

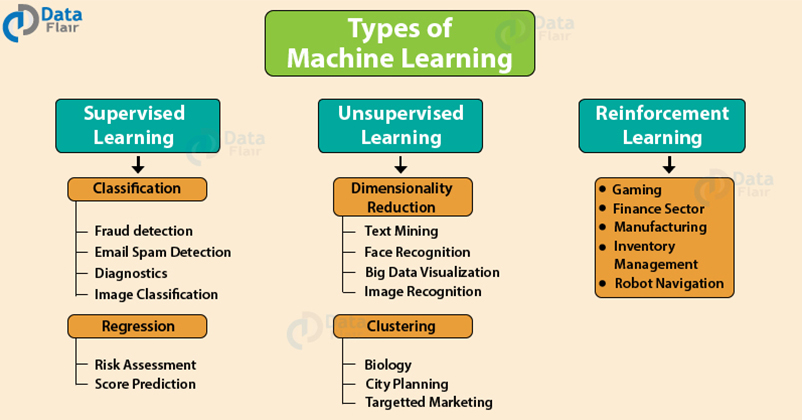
### 1.2.1 Application of Machine learning

Applications of Machine Learning include:

* **Web Search Engine:** One of the reasons why search engines like google, Bing etc work so well is because the system has learnt how to rank pages through a complex learning algorithm.
* **Photo tagging Applications:** Be it Facebook or any other photo tagging application, the ability to tag friends makes it even more happening. It is all possible because of a face recognition algorithm that runs behind the application.
* **Spam Detector:** Our mail agent like Gmail or Hotmail does a lot of hard work for us in classifying the mails and moving the spam mails to spam folder. This is again achieved by a spam classifier running in the back end of mail application.
* YouTube, which recommends videos based on our previous watched videos with the help of Machine Learning algorithm
* Self-customizing programs E.g., Amazon, Netflix product recommendations
* It is used in banking and financial sector, healthcare, retail, publishing and social media etc.
* It is used by Google and Facebook to push relevant advertisements based on users search history.

Machine learning is a multidisciplinary field. It can be applied in various areas such as: robotics, data mining, finance, health care, banking, fraud detection etc.

### 1.2.2 Types of Machine Learning

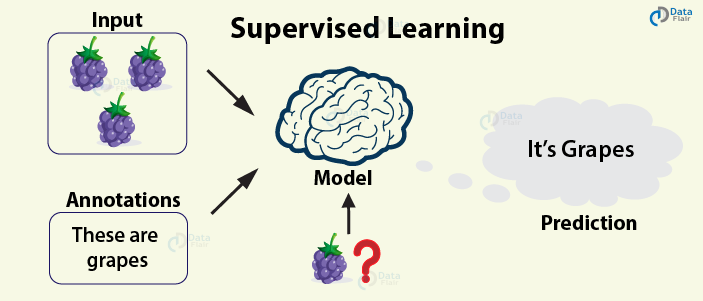


There are three important types of Machine Learning Algorithms i.e: –

* Supervised Learning
* Unsupervised Learning
* Reinforcement Learning

## 1.3 Supervised Learning

Supervised Learning is the most popular paradigm for performing machine learning operations. It is widely used for data where there is a precise mapping between input-output data. The dataset, in this case, is labelled, meaning that the algorithm identifies the features explicitly and carries out predictions or classification accordingly. As the training period progresses, the algorithm is able to identify the relationships between the two variables such that we can predict a new outcome.



Resulting Supervised learning algorithms are task-oriented. As we provide it with more and more examples, it is able to learn more properly so that it can undertake the task and yield us the output more accurately.

Steps

In order to solve a given problem of supervised learning, one has to perform the following steps:

Determine the type of training examples. Before doing anything else, the user should decide what kind of data is to be used as a training set. In case of handwriting analysis, for example, this might be a single handwritten character, an entire handwritten word, or an entire line of handwriting.

Gather a training set. The training set needs to be representative of the real-world use of the function. Thus, a set of input objects is gathered and corresponding outputs are also gathered, either from human experts or from measurements.

Determine the input feature representation of the learned function. The accuracy of the learned function depends strongly on how the input object is represented. Typically, the input object is transformed into a feature vector, which contains a number of features that are descriptive of the object. The number of features should not be too large, because of the curse of dimensionality; but should contain enough information to accurately predict the output.

Determine the structure of the learned function and corresponding learning algorithm. For example, the engineer may choose to use support vector machines or decision trees.

Complete the design. Run the learning algorithm on the gathered training set. Some supervised learning algorithms require the user to determine certain control parameters. These parameters may be adjusted by optimizing performance on a subset (called a *validation* set) of the training set, or via cross-validation.

Evaluate the accuracy of the learned function. After parameter adjustment and learning, the performance of the resulting function should be measured on a test set that is separate from the training set.

Algorithm choice

A wide range of supervised learning algorithms are available, each with its strengths and weaknesses. There is no single learning algorithm that works best on all supervised learning problems (see the No free lunch theorem).

When considering a new application, the engineer can compare multiple learning algorithms and experimentally determine which one works best on the problem at hand (see cross validation). Tuning the performance of a learning algorithm can be very time-consuming. Given fixed resources, it is often better to spend more time collecting additional training data and more informative features than it is to spend extra time tuning the learning algorithms.

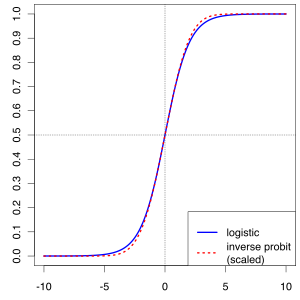
### Algorithms

The most widely used learning algorithms are:

* Logistic regression
* Naive Bayes
* Decision trees
* K-nearest neighbor algorithm
* Support vector machine algorithm
* Random forest

Some of the algorithms that come under supervised learning are as follows

### 1.3.1 Logistic regression:



In statistics, the logistic model (or logit model) is a widely used statistical model that, in its basic form, uses a logistic function to model a binary dependent variable; many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model; it is a form of binomial regression. Mathematically, a binary logistic model has a dependent variable with two possible values, such as pass/fail, win/lose, alive/dead or healthy/sick; these are represented by an indicator variable, where the two values are labeled "0" and "1". In the logistic model, the log-odds (the logarithm of the odds) for the value labeled "1" is a linear combination of one or more independent variables ("predictors"); the independent variables can each be a binary variable (two classes, coded by an indicator variable) or a continuous variable (any real value).

The corresponding probability of the value labeled "1" can vary between 0 (certainly the value "0") and 1 (certainly the value "1"), hence the labeling; the function that converts log-odds to probability is the logistic function, hence the name. The unit of measurement for the log-odds scale is called a *logit*, from *logistic unit*, hence the alternative names.

Logistic regression was developed by statistician David Cox in 1958. The binary logistic regression model has extensions to more than two levels of the dependent variable: categorical outputs with more than two values are modeled by multinomial logistic regression, and if the multiple categories are ordered, by ordinal logistic regression, for example the proportional odds ordinal logistic model.

**Applications**

Logistic regression is used in various fields, including machine learning, most medical fields, and social sciences. For example, the Trauma and Injury Severity Score (TRISS), which is widely used to predict mortality in injured patients, was originally developed by Boyd et al. using logistic regression. Many other medical scales used to assess severity of a patient have been developed using logistic regression. Logistic regression may be used to predict the risk of developing a given disease (e.g. diabetes; coronary heart disease), based on observed characteristics of the patient (age, sex, body mass index, results of various blood tests, etc.). Another example might be to predict whether an Indian voter will vote BJP or Trinamool Congress or Left Front or Congress, based on age, income, sex, race, state of residence, votes in previous elections, etc. The technique can also be used in engineering, especially for predicting the probability of failure of a given process, system or product. It is also used in marketing applications such as prediction of a customer's propensity to purchase a product or halt a subscription, etc.

Logistic regression can be seen as a special case of the generalized linear model and thus analogous to linear regression. The model of logistic regression, however, is based on quite different assumptions (about the relationship between dependent and independent variables) from those of linear regression. In particular the key differences between these two models can be seen in the following two features of logistic regression. First, the conditional distribution {\displaystyle y\mid x}is a Bernoulli distribution rather than a Gaussian distribution, because the dependent variable is binary. Second, the predicted values are probabilities and are therefore restricted to (0, 1) through the logistic distribution function because logistic regression predicts the probability of particular outcomes rather than the outcomes themselves.

Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications. Logistic Regression is used when the dependent variable(target) is categorical

For example,

**.** To predict whether an email is spam (1) or (0)

**.** Whether the tumour is malignant (1) or not (0)

Consider a scenario where we need to classify whether an email is spam or not. If we use linear regression for this problem, there is a need for setting up a threshold based on which classification can be done. Say if the actual class is malignant, predicted continuous value 0.4 and the threshold value is 0.5, the data point will be classified as not malignant which can lead to serious consequence in real time.

From this example, it can be inferred that linear regression is not suitable for classification problem. Linear regression is unbounded, and this brings logistic regression into picture. Their value strictly ranges from 0 to 1.

**Simple Logistic Regression**

Model

Output = 0 or 1

Hypothesis => Z = WX + B

hΘ(x) = sigmoid (Z)

Sigmoid Function



Figure 2: Sigmoid Activation Function

If ‘Z’ goes to infinity, Y(predicted) will become 1 and if ‘Z’ goes to negative infinity, Y(predicted) will become 0.

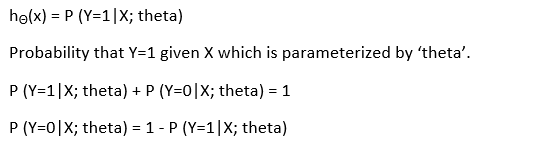
**Analysis of the hypothesis**

The output from the hypothesis is the estimated probability. This is used to infer how confident can predicted value be actual value when given an input X. Consider the below example,

X = [x0 x1] = [1 IP-Address]

Based on the x1 value, let’s say we obtained the estimated probability to be 0.8. This tells that there is 80% chance that an email will be spam.

Mathematically this can be written as,



This justifies the name ‘logistic regression’. Data is fit into linear regression model, which then be acted upon by a logistic function predicting the target categorical dependent variable.

**Types of Logistic Regression**

**1. Binary Logistic Regression**

The categorical response has only two 2 possible outcomes. Example: Spam or Not

**2. Multinomial Logistic Regression**

Three or more categories without ordering. Example: Predicting which food is preferred more (Veg, Non-Veg, Vegan)

**3. Ordinal Logistic Regression**

Three or more categories with ordering.

Example: Movie rating from 1 to 5

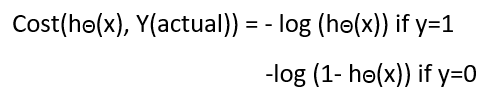
**Decision Boundary**

To predict which class a data belongs, a threshold can be set. Based upon this threshold, the obtained estimated probability is classified into classes.

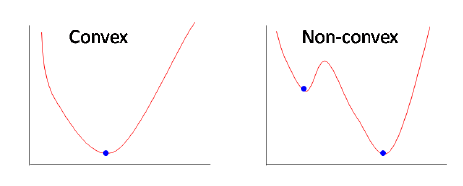
Say, if predicted value ≥ 0.5, then classify email as spam else as not spam.

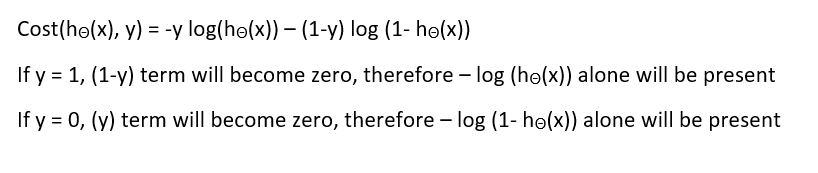
Decision boundary can be linear or non-linear. Polynomial order can be increased to get complex decision boundary.

**Cost Function**



Linear regression uses mean squared error as its cost function. If this is used for logistic regression, then it will be a non-convex function of parameters (theta). Gradient descent will converge into global minimum only if the function is convex.





**Pros and Cons:**

**Pros:**

It is a widely used technique because it is very efficient, does not require too many computational resources, it’s highly interpretable, it doesn’t require input features to be

scaled, it doesn’t require any tuning, it’s easy to regularize, and it outputs well-calibrated predicted probabilities.

Like linear regression, logistic regression does work better when you remove attributes that are unrelated to the output variable as well as attributes that are very similar (correlated) to each other. Therefore Feature Engineering plays an important role in regards to the performance of Logistic and also Linear Regression. Another advantage of Logistic Regression is that it is incredibly easy to implement and very efficient to train. I typically start with a Logistic Regression model as a benchmark and try using more complex algorithms from there on.

Because of its simplicity and the fact that it can be implemented relatively easy and quick, Logistic Regression is also a good baseline that you can use to measure the performance of other more complex Algorithms.

**Cons:**

A disadvantage of it is that we can’t solve non-linear problems with logistic regression since its decision surface is linear.

In other words: You should think about using logistic regression when your Y variable takes on only two values (e.g. when you are facing a classification problem). Note that you could also use Logistic Regression for multiclass classification, which will be discussed in the next section

**Multiclass Classification**

Out there are algorithms that can deal by themselves with predicting multiple classes, like Random Forest classifiers or the Naive Bayes Classifier. There are also algorithms that can’t do that, like Logistic Regression, but with some tricks, you can predict multiple classes with it too. This is a classification task where our Algorithm should tell us which number is on an image.

1. **One-versus-All (OvA)**

With this strategy, you train 10 binary classifiers, one for each number. This simply means training one classifier to detect 0s, one to detect 1s, one to detect 2s and so on. When you then want to classify an image, you just look at which classifier has the best decision score

1. **One-versus-One (OvO)**

Here you train a binary classifier for every pair of digits. This means training a classifier that can distinguish between 0s and 1s, one that can distinguish between 0s and 2s, one that can distinguish between 1s and 2s etc. If there are N classes, you would need to train NxN(N-1)/2 classifiers, which are 45 in the case of the MNIST dataset.

### 1.3.2Naïve Bayes Classifier

* Naive Bayes classifier is a straightforward and powerful algorithm for the [classification](https://dataaspirant.com/2016/09/24/classification-clustering-alogrithms/) task. Even if we are working on a data set with millions of records with some attributes, it is suggested to try Naive Bayes approach.
* Naive Bayes classifier gives great results when we use it for textual data analysis. Such as Natural Language Processing.
* To understand the naive Bayes classifier, we need to understand the Bayes theorem. So, let’s first discuss the Bayes Theorem.

**What is Bayes Theorem?**

Bayes theorem named after Rev. Thomas Bayes. It works on conditional probability. Conditional probability is the probability that something will happen, given that something else has already occurred. Using the conditional probability, we can calculate the probability of an event using its prior knowledge.

Below is the formula for calculating the conditional probability.

\textrm{P(H \textbar E) = }  \frac{\textrm{ P(E \textbar H) * P(H)}} {\textrm{P(E)}}

where

* P(H) is the probability of hypothesis H being true. This is known as the prior probability.
* P(E) is the probability of the evidence (regardless of the hypothesis).
* P(E|H) is the probability of the evidence given that hypothesis is true.
* P(H|E) is the probability of the hypothesis given that the evidence is there**.**
* Let’s consider an example to understand how the above formula of Bayes theorem works.

**Types of Naive Bayes Classifier Algorithms**

**Gaussian Naive Bayes**

When attribute values are continuous, an assumption is made that the values associated with each class are distributed according to Gaussian i.e., Normal Distribution.

If in our data, an attribute say “x” contains continuous data. We first segment the data by the class and then compute mean \mu_{y}  & Variance {\sigma_{y}}^{2}  of each class.  
P(x_i \mid y) &= \frac{1}{\sqrt{2\pi\sigma^2_y}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma^2_y}\right)

**Multinomial Naive Bayes**

Multinomial Naive Bayes is preferred to use on data that is multinomially distributed. It is one of the standard classic algorithms. Which is used in text categorization (classification). Each event in text classification represents the occurrence of a word in a document.

**Bernoulli Naive Bayes**

Bernoulli Naive Bayes is used on the data that is distributed according to multivariate Bernoulli distributions. i.e., multiple features can be there, but each one is assumed to be a binary-valued (Bernoulli, Boolean) variable. So, it requires features to be binary valued.

**Pros and Cons of Naïve Bayes**:

**Pros:**

* Naive Bayes Algorithm is a fast, highly scalable algorithm.
* Naive Bayes can be used for Binary and Multiclass classification. It provides different types of Naive Bayes Algorithms like GaussianNB, MultinomialNB, BernoulliNB.

. It is a simple algorithm that depends on doing a bunch of counts.

Great choice for Text Classification problems. It’s a popular choice for spam email classification.

* It can be easily train on small dataset

**Cons-**

* It considers all the features to be unrelated, so it cannot learn the relationship between features. E.g., Let’s say Remo is going to a part. While cloth selection for the party, Remo is looking at his cupboard. Remo likes to wear a white colour shirt. In Jeans, he likes to wear a brown Jeans, But Remo doesn’t like wearing a white shirt with Brown Jeans. Naive Bayes can learn individual features importance but can’t determine the relationship among features.

### 

### 1.3.3 Decision Tree Algorithm

A decision tree is a [flowchart](https://en.wikipedia.org/wiki/Flowchart)-like structure in which each internal node represents a "test" on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represent classification rules.

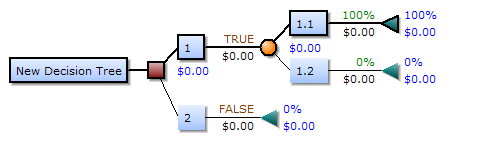
In [decision analysis](https://en.wikipedia.org/wiki/Decision_analysis), a decision tree and the closely related [influence diagram](https://en.wikipedia.org/wiki/Influence_diagram) are used as a visual and analytical decision support tool, where the [expected values](https://en.wikipedia.org/wiki/Expected_value) (or [expected utility](https://en.wikipedia.org/wiki/Expected_utility)) of competing alternatives are calculated.

A decision tree consists of three types of nodes:[[1]](https://en.wikipedia.org/wiki/Decision_tree#cite_note-1)

1. Decision nodes – typically represented by squares
2. Chance nodes – typically represented by circles
3. End nodes – typically represented by triangles

Decision trees are commonly used in [operations research](https://en.wikipedia.org/wiki/Operations_research) and [operations management](https://en.wikipedia.org/wiki/Operations_management). If, in practice, decisions have to be taken online with no recall under incomplete knowledge, a decision tree should be paralleled by a [probability](https://en.wikipedia.org/wiki/Probability) model as a best choice model or online selection model [algorithm](https://en.wikipedia.org/wiki/Algorithm). Another use of decision trees is as a descriptive means for calculating [conditional probabilities](https://en.wikipedia.org/wiki/Conditional_probability).

Decision trees, [influence diagrams](https://en.wikipedia.org/wiki/Influence_diagrams), [utility functions](https://en.wikipedia.org/wiki/Utility_function), and other [decision analysis](https://en.wikipedia.org/wiki/Decision_analysis) tools and methods are taught to undergraduate students in schools of business, health economics, and public health, and are examples of operations research or [management science](https://en.wikipedia.org/wiki/Management_science) methods.

[](https://en.wikipedia.org/wiki/File:Decision-Tree-Elements.png)

Drawn from left to right, a decision tree has only burst nodes (splitting paths) but no sink nodes (converging paths). Therefore, used manually, they can grow very big and are then often hard to draw fully by hand. Traditionally, decision trees have been created manually — as the aside example shows — although increasingly, specialized software is employed.

**Decision rule**

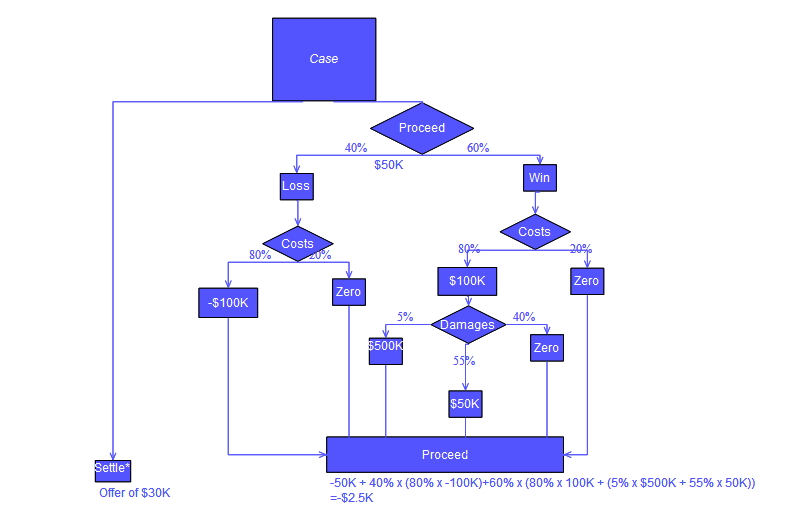
The decision tree can be [linearized](https://en.wikipedia.org/wiki/Linearization) into decision rules,[[2]](https://en.wikipedia.org/wiki/Decision_tree#cite_note-2) where the outcome is the contents of the leaf node, and the conditions along the path form a conjunction in the if clause. In general, the rules have the form:

if condition1 and condition2 and condition3 then outcome.

Decision rules can be generated by constructing [association rules](https://en.wikipedia.org/wiki/Association_rule_learning) with the target variable on the right. They can also denote [temporal](https://en.wikipedia.org/wiki/Time) or [causal](https://en.wikipedia.org/wiki/Causal) relations.

**Decision tree using flowchart symbols**

Commonly a decision tree is drawn using [flowchart](https://en.wikipedia.org/wiki/Flowchart) symbols as it is easier for many to read and understand.

[](https://en.wikipedia.org/wiki/File:DecisionCalcs.jpg)

**Pros and Cons:**

**Pros:**

Among decision support tools, decision trees (and [influence diagrams](https://en.wikipedia.org/wiki/Influence_diagrams)) have several advantages. Decision trees:

* Are simple to understand and interpret. People are able to understand decision tree models after a brief explanation.
* Have value even with little hard data. Important insights can be generated based on experts describing a situation (its alternatives, probabilities, and costs) and their preferences for outcomes.
* Help determine worst, best and expected values for different scenarios.
* Use a [white box](https://en.wikipedia.org/wiki/White_box_(software_engineering)) model. If a given result is provided by a model.
* Can be combined with other decision techniques.

**Cons:**

* They are unstable, meaning that a small change in the data can lead to a large change in the structure of the optimal decision tree.
* They are often relatively inaccurate. Many other predictors perform better with similar data. This can be remedied by replacing a single decision tree with a [random forest](https://en.wikipedia.org/wiki/Random_forest) of decision trees, but a random forest is not as easy to interpret as a single decision tree.
* For data including categorical variables with different number of levels, [information gain in decision trees](https://en.wikipedia.org/wiki/Information_gain_in_decision_trees) is biased in favor of those attributes with more levels.[[7]](https://en.wikipedia.org/wiki/Decision_tree#cite_note-7)
* Calculations can get very complex, particularly if many values are uncertain and/or if many outcomes are linked.

### 1.3.4 K-Nearest Neighbor Algorithm (KNN):

KNN is known as an instance-based learning, or also popular as a lazy learning, where the role is only estimated locally and all calculation is postponed until classification. The k-NN algorithm is the simplest among all machine learning algorithms. It studies all existing cases and classifies into new cases based on common factor. (e.g., distance functions). Features for which it is very popular is its simplicity of interpretation and short calculation time even with such ease, it can provide greatly feasible results. The neighbors are taken from a set of objects for which the object property value (for k-NN regression) or the classes (for kNN classification) are recognized. This can be supposed of as the training set for the algorithm, though no explicit training step is essential.

* In pattern recognition, the *k*-nearest neighbors’ algorithm (*k*-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the *k* closest training examples in the feature space. The output depends on whether *k*-NN is used for classification or regression.
* In *k-NN classification*, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor.
* In *k-NN regression*, the output is the property value for the object. This value is the average of the values of its *k* nearest neighbors.
* *k*-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The *k*-NN algorithm is among the simplest of all machine learning algorithms.
* Both for classification and regression, a useful technique can be used to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones.



* Example of k-NN classification. The test sample (green circle) should be classified either to the first class of blue squares or to the second class of red triangles. If k = 3 (solid line circle) it is assigned to the second class because there are 2 triangles and only 1 square inside the inner circle. If k = 5 (dashed line circle) it is assigned to the first class (3 squares vs. 2 triangles inside the outer circle).
* The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.
* In the classification phase, *k* is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the *k* training samples nearest to that query point.
* A commonly used distance metric for continuous variables is Euclidean distance. For discrete variables, such as for text classification, another metric can be used, such as the overlap metric (or Hamming distance). In the context of gene expression microarray data, for example, *k*-NN has also been employed with correlation coefficients such as Pearson and Spearman.
* The best choice of *k* depends upon the data; generally, larger values of *k* reduce effect of the noise on the classification,but make boundaries between classes less distinct.
* A good *k* can be selected by various heuristic techniques (see hyper parameter optimization). The special case where the class is predicted to be the class of the closest training sample (i.e., when *k* = 1) is called the nearest neighbor algorithm.
* The accuracy of the *k*-NN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the feature scales are not consistent with their importance

**How does KNN work?**

In the classification setting, the K-nearest neighbor algorithm essentially boils down to forming a majority vote between the K most similar instances to a given “unseen” observation. Similarity is defined according to a distance metric between two data points. A popular choice is the Euclidean distance given by

d(x,x′)=(x1−x′1)2+(x2−x′2)2+…+(xn−x′n)2−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−√d(x,x′)=(x1−x1′)2+(x2−x2′)2+…+(xn−xn′)2

but other measures can be more suitable for a given setting and include the Manhattan, Chebyshev and Hamming distance.

More formally, given a positive integer K, an unseen observation xx and a similarity metric dd, KNN classifier performs the following two steps:

It runs through the whole dataset computing dd between xx and each training observation. We’ll call the K points in the training data that are closest to xx the set AA. Note that K is usually odd to prevent tie situations.

It then estimates the conditional probability for each class, that is, the fraction of points in AA with that given class label. (Note I(x)I(x) is the indicator function which evaluates to 11 when the argument xx is true and 00otherwise)

P(y=j|X=x) =1K∑i∈AI(y(i)=j) P(y=j|X=x) =1K∑i∈AI(y(i)=j)

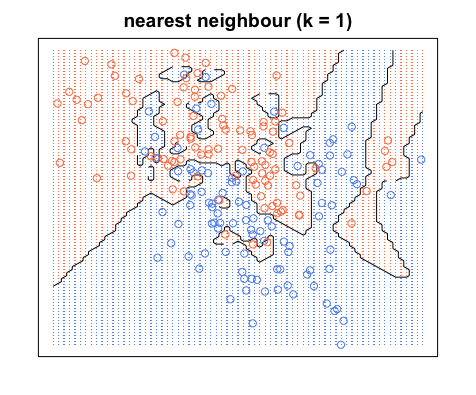
Finally, our input xx gets assigned to the class with the largest probability.

KNN searches the memorized training observations for the K instances that most closely resemble the new instance and assigns to it the their most common class.

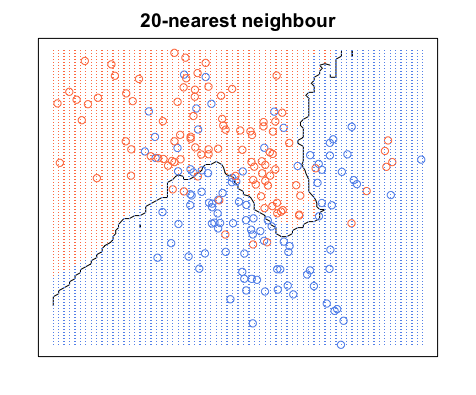
An alternate way of understanding KNN is by thinking about it as calculating a decision boundary (i.e., boundaries for more than 2 classes) which is then used to classify new points.

Like most machine learning algorithms, the K in KNN is a hyper parameter that we, as a designer, must pick in order to get the best possible fit for the data set. Intuitively, we can think of K as controlling the shape of the decision boundary.

When K is small, we are restraining the region of a given prediction and forcing our classifier to be “more blind” to the overall distrib9ollution. A small value for K provides the most flexible fit, which will have low bias but high variance. Graphically, our decision boundary will be more jagged.



On the other hand, a higher K averages more voters in each prediction and hence is more resilient to outliers. Larger values of K will have smoother decision boundaries which mean lower variance but increased bias.



**Pros and Cons of KNN**

**Pros**

* One of the most attractive features of the K-nearest neighbor algorithm is that is simple to understand and easy to implement. With zero to little training time, it can be a useful tool for off-the-bat analysis of some data set you are planning to run more complex algorithms on. Furthermore, KNN works just as easily with multiclass data sets whereas other algorithms are hardcoded for the binary setting. Finally, as we mentioned earlier, the non-parametric nature of KNN gives it an edge in certain settings where the data may be highly “unusual”.

**Cons**

* One of the obvious drawbacks of the KNN algorithm is the computationally expensive testing phase which is impractical in industry settings. Note the rigid dichotomy between KNN and the more sophisticated Neural Network which has a lengthy training phase albeit a **very fast** testing phase. Furthermore, KNN can suffer from skewed class distributions. For example, if a certain class is very frequent in the training set, it will tend to dominate the majority voting of the new example (large number = more common). Finally, the accuracy of KNN can be. severely degraded with high-dimension data because there is little difference between the nearest and farthest neighbor

### 1.3.5 Support Vector Machine Algorithm

Support Vector Machines are a type of supervised machine learning algorithm that provides analysis of data for classification and regression analysis. While they can be used for regression, SVM is mostly used for classification. We carry out plotting in the n-dimensional space. The value of each feature is also the value of the specified coordinate. Then, we find the ideal hyperplane that differentiates between the two classes.

These support vectors are the coordinate representations of individual observation. It is a frontier method for segregating the two classes.

**How SVM work:**

The objective of the support vector machine algorithm is to find a decision boundary or hyper plane in an N-dimensional space (N — the number of features) that distinctly classifies the data points.



Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

“Select the hyper-plane which segregates the two classes better”. In this scenario, hyper-plane “B” has excellently performed this job.

**A hyper plane** is a line that divides the points in input variable space according to their class.., either class 0 or class 1.

In two-dimensions you can visualize this as **a line.**

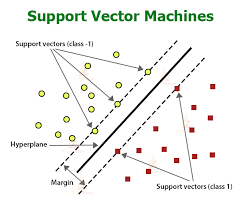
For 3 or more dimensional space it is called as **hyper plane**.

**What is Margin?**

The distance between the nearest data points (both class) and the decision boundary (hyper plane) is called as **Margin.**

In svm we have to maximize distance of closest points on surface from decision boundary (hyper plane)

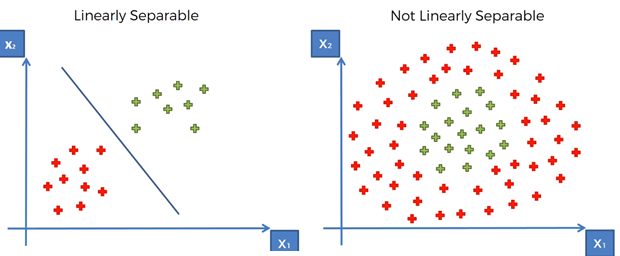
Here we can see that the margin for hyper plane C is high in comparison **to** B and C .Hence the right hyper plane is C. If we select hyper plane with low margin then there is high chance of miss-classification.

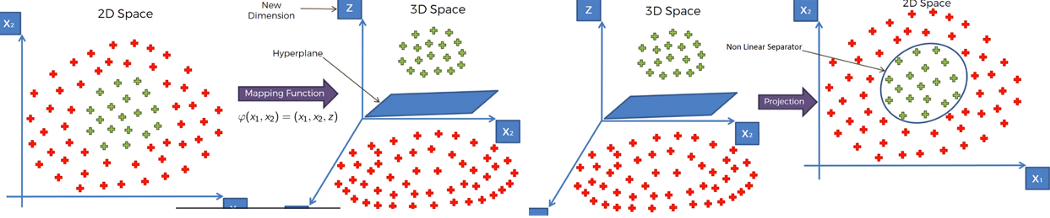


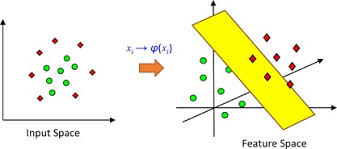
**What is Support vector?**

The nearest positive points and negative points passing through the marginal plane are called as support vectors. There should be minimum of 2 support vectors but can be more.

These support vectors determine the equation of hyper plane. Normally the no of support vectors is small.

****

****

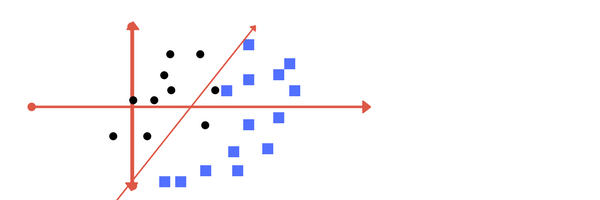
****

**Tuning parameters for svm:**

Tuning parameters are used to improve accuracy of SVM

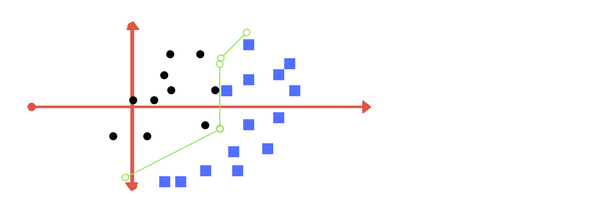
**C ---- regularization parameter**

The Regularization parameter (often termed as C parameter in python’s sklearn library) tells the SVM optimization how much you want to avoid misclassifying each training example.



**Low regularization value.**

It has some misclassification due to lower regularization value.



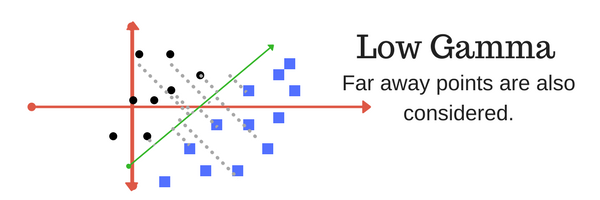
**High regularization value**

**Kernel**

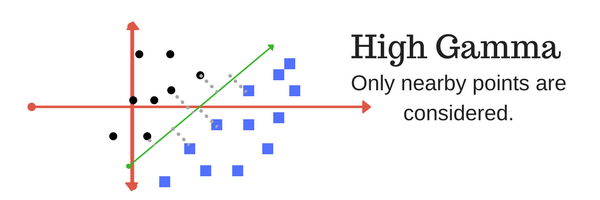
SVM uses a technique called the kernel in which kernel takes a low dimensional input space and transforms it into a higher dimensional space. In simple words, kernel converts non-separable problems into separable problems by adding more dimensions to it. It makes SVM more powerful, flexible and accurate.

**Gamma -**

With low gamma, points far away from separation line are considered in calculation.



High gamma means the points close to separation line are considered in calculation.



**Advantages:**

Works well on small and cleaner data set

**Drawback:**

Not suitable for larger data sets as the training time can be high.

### 1.3.6 Random Forest Algorithm

Random Forest classifiers are a type of ensemble learning method that is used for classification, regression and other tasks that can be performed with the help of the decision trees. These decision trees can be constructed at the training time and the output of the class can be either classification or regression. With the help of these random forests, one can correct the habit of overfitting to the training set.

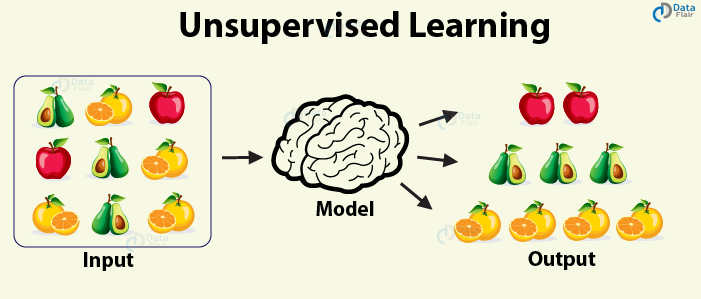
Some of the advantages and disadvantages of random forest classifiers are as follows:

**Advantages –** Random Forest Classifiers facilitate the reduction in the over-fitting of the model and these classifiers are more accurate than the decision trees in several cases.

**Disadvantages –** Random forests exhibit real-time prediction but that is slow in nature. They are also difficult to implement and have a complex algorithm.

## 1.4 Unsupervised Learning

In the case of unsupervised learning algorithm, the data is not explicitly labeled into different classes, that is, there are no labels. The model is able to learn from the data by finding implicit patterns. Unsupervised Learning algorithms identify the data based on their densities, structures, similar segments, and other similar features. Unsupervised Learning Algorithms are based on Hebbian Learning. Cluster analysis is one of the most widely used techniques in supervised learning. Let us look at some of the important algorithms that come under Unsupervised Learning.

*[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/sites/2/2019/07/unsupervised-learning.png)*

### 1.4.1 Clustering

**Clustering**, also known as cluster analysis, is a technique of grouping similar sets of objects in the same group that is different from the objects in another group. Some of the essential clustering techniques are as follows –

**a. K-means**

The aim of the [k-means clustering algorithm](https://data-flair.training/blogs/k-means-clustering-tutorial/) is to partition the n observations in the data into k clusters such that each observation belongs to the cluster with the nearest mean. This serves as the prototype of the cluster.

**b. DBSCAN**

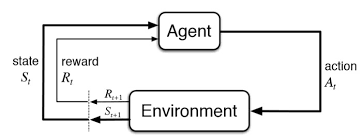
This is a clustering method that groups the data based on the density. It groups together the points that are given in the space and marks the outliers in the low-density region.

**c. Hierarchical clustering**

In this form of clustering, a hierarchy of clusters is built.

## 1.5 Reinforcement Learning

Reinforcement Learning covers more area of [**Artificial Intelligence**](https://data-flair.training/blogs/ai-tutorials-home/) which allows machines to interact with their dynamic environment in order to reach their goals. With this, machines and software agents are able to evaluate the ideal behavior in a specific context. With the help of this reward feedback, agents are able to learn the behavior and improve it in the longer run. This simple feedback reward is known as a reinforcement signal.



The agent in the environment is required to take actions that are based on the current state. This type of learning is different from Supervised Learning in the sense that the training data in the former has output mapping provided such that the model is capable of learning the correct answer. Whereas, in the case of reinforcement learning, there is no answer key provided to the agent when they have to perform a particular task. When there is no training dataset, it learns from its own experience.

## 1.6 Objective

Heart problems have gained a lot of interest in medical research because of their impact on human health where early diagnosis is critical to delaying the development of heart disease, the world's leading cause of death. thus, it is much needed to predict the possibility of occurrence of heart disease based on their attributes.

This research aims into a variety of machine learning classification algorithms for predicting heart disease using Framingham dataset.

# 2. Literature Survey

**Current approaches To predict heart disease**

1. Heart Disease Prediction Using Machine Learning and Data Mining Techniques: Application of Framingham Dataset.Turkish Journal of Computer and Mathematics Education.Vol.12 No.14(2021), 4864- 4870.Walaa Adel Mahmoud , Prof. Dr. Mohamed Aborizka ,Prof. Dr. Fathy Ahmed Elsayed Amer2

In this research work The 10-fold cross-validation resampling is used to validate the prediction model. 5 different machine learning algorithms (KNearest Neighbors,Logistic Regression,decision Tree,Random Forest and support vector machine )are used in this work. Accuracy scores of each algorithm are evaluated.

KNearestNeighbors-83.95 %

SupportVectorMachine-84.5 %

DecisionTree-84.82 %

LogisticRegression-84.89 %

RandomForest-85.05 %

The Random Forest algorithm is found to have highest accuracy.

**Remark**

In this approach the features are not standardized .

**2.** Machine Learning-Based Classification Algorithms for the Prediction of Coronary Heart Diseases.

Kelvin Kwakye & Emmanuel Dadzie

**Remark**

In this approach missing values are handled by mean values

# 3. Problem definition

## 3.1 Task definition

In this project the task is to predict whether a person is going to have coronary heart disease or not in upcoming 10 years by using 6 popular machine learning algorithms. These are logistic regression, K Nearest neighbours, Naïve bayes , Decision tree ,Support vector machine and random Forest algorithm. After implementing these algorithms, the best model will be found for heart disease prediction. A graphical user interface will be designed for to make prediction for new dataset.

## 3.2 Heart disease data set

This dataset is located at Kaggle website. The dataset contains 16 attributes in all, of which 15 are independent factors and one variable i.e., TenYearCHD is the dependent variable or target variable. This data set contains total 4240 data. All attributes in data set are numeric.

The 16 attributes that are defined here are as follows

**Demographic:**

* Sex: male or female (1-male and 0-female)
* Age: Age of the patient;(Continuous - Although the recorded ages have been truncated to whole numbers, the concept of age is continuous)
* Education: no further information provided

**Behavioural:**

* Current Smoker: whether or not the patient is a current smoker (Nominal)
* Cigs Per Day: the number of cigarettes that the person smoked on average in one day (can be considered continuous as one can have any number of cigarettes, even half a cigarette.)

**Information on medical history:**

* BP Meds: whether or not the patient was on blood pressure medication (Nominal)
* Prevalent Stroke: whether or not the patient had previously had a stroke (Nominal)
* Prevalent Hyp: whether or not the patient was hypertensive (Nominal)
* Diabetes: whether or not the patient had diabetes (Nominal)

**Information on current medical condition:**

* Tot Chol: total cholesterol level (Continuous)
* Sys BP: systolic blood pressure (Continuous)
* Dia BP: diastolic blood pressure (Continuous)
* BMI: Body Mass Index (Continuous)
* Heart Rate: heart rate (Continuous - In medical research, variables such as heart rate though in fact discrete, yet are considered continuous because of large number of possible values.)
* Glucose: glucose level (Continuous)

**Target variable to predict:**

* 10year risk of coronary heart disease (CHD) - (binary: “1”, means “Yes”, “0” means “No”)

|  |  |
| --- | --- |
| **Attribute & description** | |
| Age | (32-70) |
| Sex | 0=Female, 1=Male |
| Education | It takes values as:  1=High School,  2=High School or GED,  3=College or Vocational School, 4=College |
| Current Smoker | 0=No  1=Yes |
| Cigs Per Day | Number of Cigarettes smoked Per Day (0-70) |
| BP Meds | 0=No  1=Yes |
| Prevalent Stroke | 0=No  1=Yes |
| Prevalent Hyp | 0=No  1=Yes |
| Diabetes | 0=No  1=Yes |
| Tot Chol | Serum Cholesterol (107-696) (mg/dl) |
| Sys BP | (83.5-295) (mm/hg) |
| Día BP | (48-142.5) (mm/hg) |
| Body Mass  Index | (15.54-56.8) |
| Heart Rate | Heart Rate achieved (44-143) |
| Glucose | (40-394) (mg/dl) |
| 10-year CHD | 0=Healthy, 1=Diseases |

# 4. Experimental set up and evaluation

**Python:**

Python is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language), [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). Created by [Guido van Rossum](https://en.wikipedia.org/wiki/Guido_van_Rossum) and first released in 1991, Python has a design philosophy that emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability), notably using [significant whitespace](https://en.wikipedia.org/wiki/Significant_whitespace). It provides constructs that enable clear programming on both small and large scales. In July 2018, Van Rossum stepped down as the leader in the language community.

Python features a [dynamic type](https://en.wikipedia.org/wiki/Dynamic_type) system and automatic management. It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), [imperative](https://en.wikipedia.org/wiki/Imperative_programming), [functional](https://en.wikipedia.org/wiki/Functional_programming) and [procedural](https://en.wikipedia.org/wiki/Procedural_programming), and has a large and comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library).[[29]](https://en.wikipedia.org/wiki/Python_(programming_language)#cite_note-About-29)

Python interpreters are available for many [operating systems](https://en.wikipedia.org/wiki/Operating_system). [C Python](https://en.wikipedia.org/wiki/CPython), the [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation) of Python, is [open source](https://en.wikipedia.org/wiki/Open-source_software) software and has a community-based development model, as do nearly all of Python's other implementations. Python and C Python are managed by the non-profit [Python Software Foundation](https://en.wikipedia.org/wiki/Python_Software_Foundation).

Here, we have used Python 3.8 as it is the stable version as on now.

**Python Packages:**

They are simply directories, but with a twist. Each package in Python is a directory which MUST contain a special file called \_\_init\_\_.py. This file can be empty, and it indicates that the directory it contains is a Python package, so it can be imported the same way a module can be imported.   
Packages are a way of structuring Python’s module namespace by using "dotted module names". A.B stands for a sub module named B in a package named A. Two different packages like P1 and P2 can both have modules with the same name, let's say A, for example. The sub module A of the package P1 and the sub module A of the package P2 can be totally different. A package is imported like a "normal" module.

## 4.1 Components to be installed:

Anaconda (python 3.9)

Python Packages to be installed

* **pandas**- used to provide fast, flexible and expressive data structures to make working with “relational” or “labeled” data both easy and intuitive.
* **numpy**: To work with arrays
* **tkinter**- used for developing Graphical User Interface (GUI).
* **sklearn-** efficient for data mining and data analysis. It contains a number of algorithms which can be implemented directly when required.
* **matplotlib**- contains modules which are used for plotting graphs.
* **Seaborn** – used for Graphical Presentation

## 4.2 Methodology

**Data preprocessing**

* **Remove unnecessary columns**
* **handle missing values and outliers**

**Raw dataset**

### 

**Exploratory Data Analysis**



**Split data set**

**Testing (20%)**

**Training (80%)**

**PERFORMANCE EVALUATION**

**Accuracy and F1 Score**

**Algorithms**

* LR
* KNN
* DT
* SVM
* NB
* RF

**Best Algorithm**

**Make Prediction**

**4.2.1Data Preprocessing**

Data preprocessing is a group of techniques that are applied on the data to improve the quality of the data. Raw data can be converted to useful data. It uses various techniques such as handling missing values, detecting outliers, removing un wanted attributes etc.

Once data is cleaned then it can be used in machine learning models for training. Data preprocessing is one of the most important steps involved in predictive modelling.

**Steps followed in data preprocessing**

* Education column is removed as it has no relation with heart disease
* Most corelated features are found by using correlation matrix(heatmap).

Sysbp and Diabp are highly corelated

Current smoker and Cigsperday are highly corelated.so only one of them is selected.

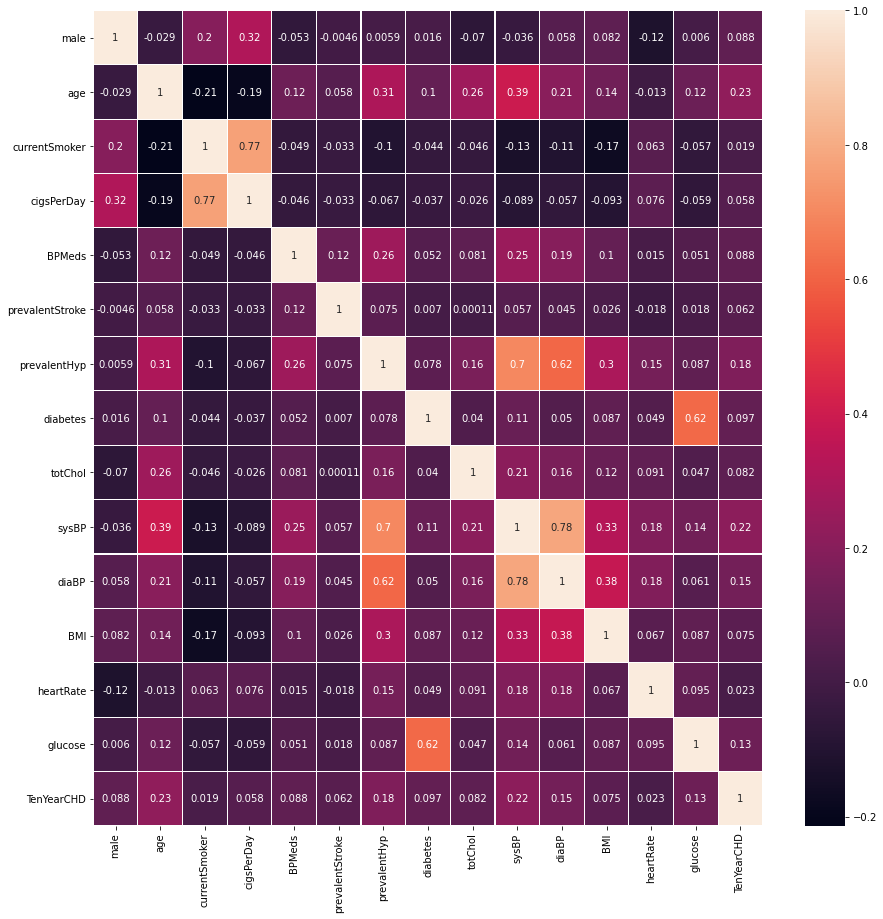
* Missing values are handled by imputation .
* Outliers found and removed.
* Duplicated rows removed.
* Features have been standardized by using Standard Scaler .

**4.2.2Explarotery analysis**

**Exploratory Data Analysis (EDA)**is an approach to analyse the data using visual techniques. It is used to discover trends, patterns, or to check assumptions with the help of statistical summary and graphical representations.

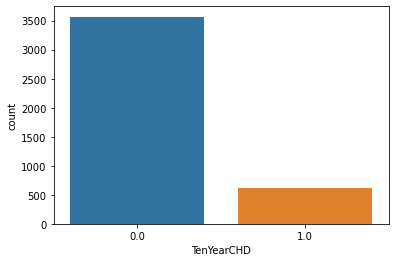
**Corelation Matrix :**

Corelation Matrix is used to show the relationship among the features. It is shown by heatmap

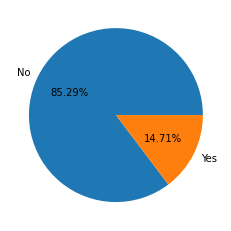
****

(Heat Map)

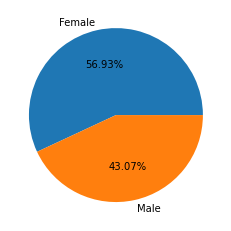
**Bar Chart showing Number of People Suffer From heart disease:**



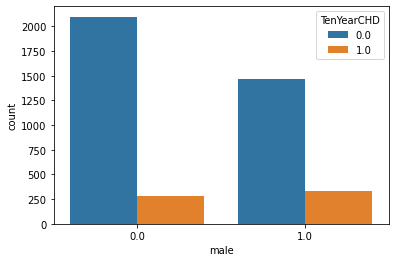
**Pie Chart showning percentage of people with and wit out heart disease**

****

**Pie chart showing male and female percentage**



**Male and Female having disease or not**

****

**4.2.3 Splitting dataset**

Data set is splitted into training and testing subsets .

For training 80% data and for testing 20% of data is selected .

**4.2.3 Algorithms**

For building machine learning model following algorithms

Are used.

**A**.KNearest Neighbors

**B**.Logistic Regression

**C.**Decision Tree

**D.**Naive Bayes

**E.**Support Vector Machine

**F.**Random Forest

# 4.2.4 Performance Evaluation Criteria

Commonly, the performance of ML prediction systems is frequently assessed using metrics based on the classification algorithm. The accuracy, confusion matrix, and F-measure are used to evaluate the prediction findings in this research.

# Confusion Matrix

A confusion matrix is used to evaluate algorithms in ML. A matrix for a binary classification problem is a square of two by two as illustrated in following Table , where the column represents the algorithm's prediction and the row reflects the real value of the class label, where true-positive (TP) is numerous positive samples accurately predicted. False-negative (FN) refers to numerous positive samples that were anticipated incorrectly. False- positive (FP) refers to a situation in which numerous negative samples are incorrectly classified as positive. A true-negative (TN) is a collection of negative samples that have been accurately anticipated .

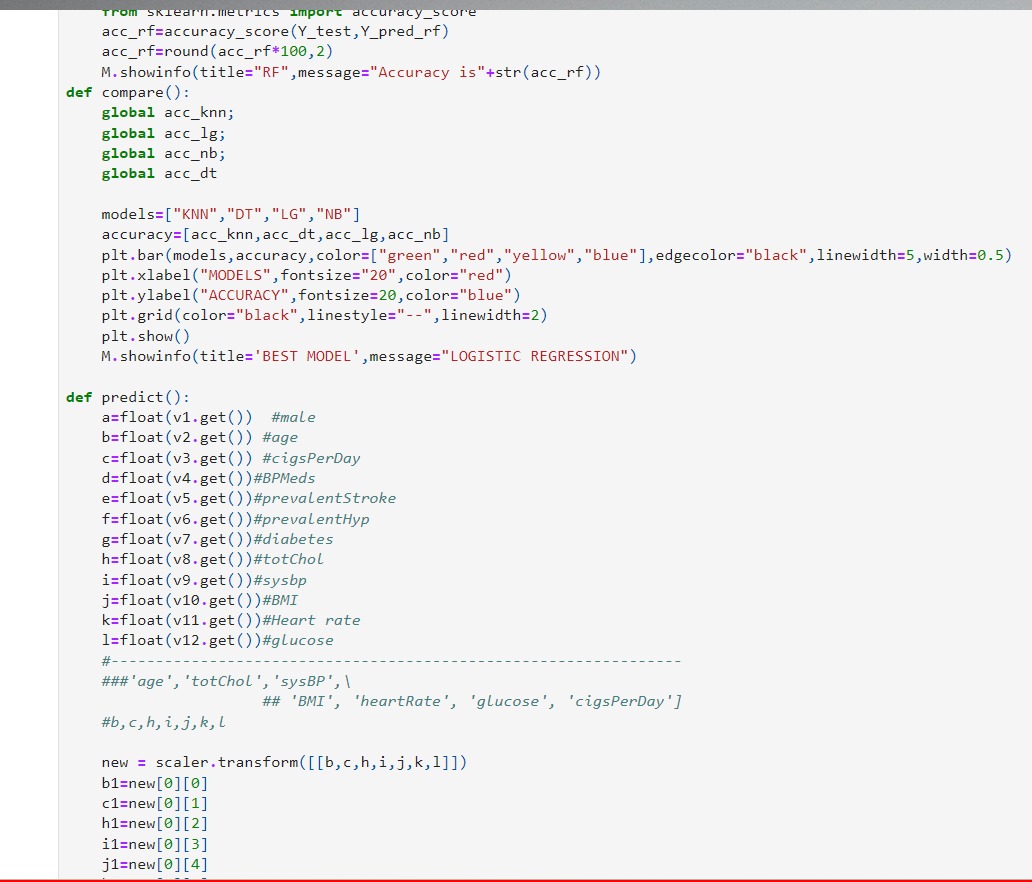
|  |  |  |
| --- | --- | --- |
| **Actual** | **Predict** | |
| Positive | Negative |
| **Positive** | TP | FP |
| **Negative** | FN | TN |

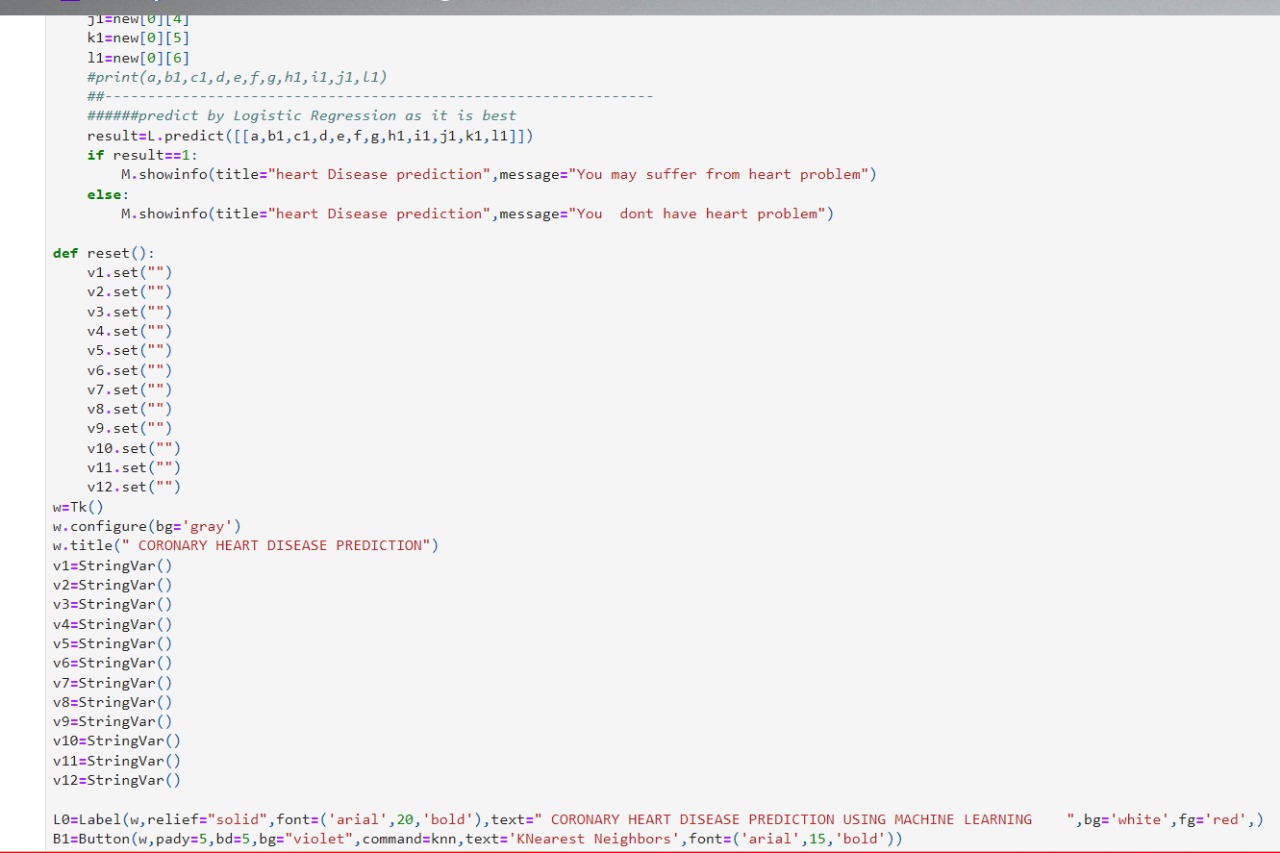
**5.PROJECT CODE**

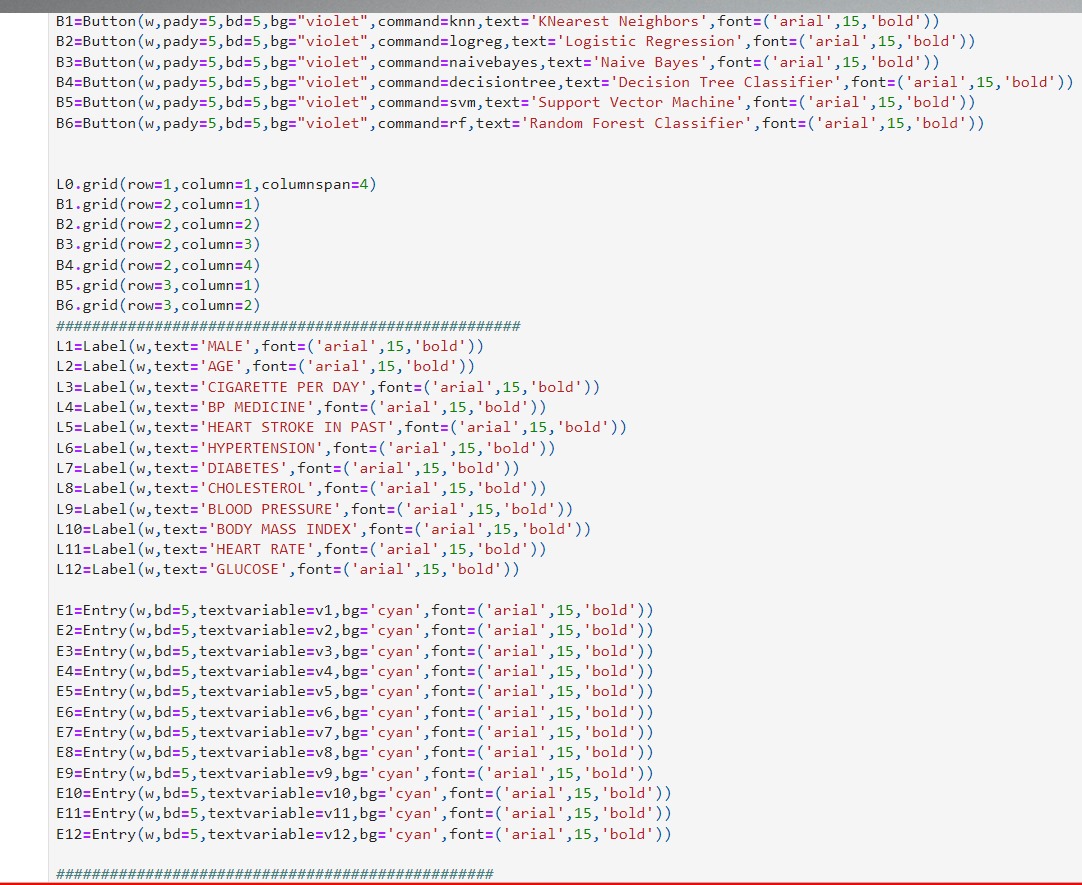








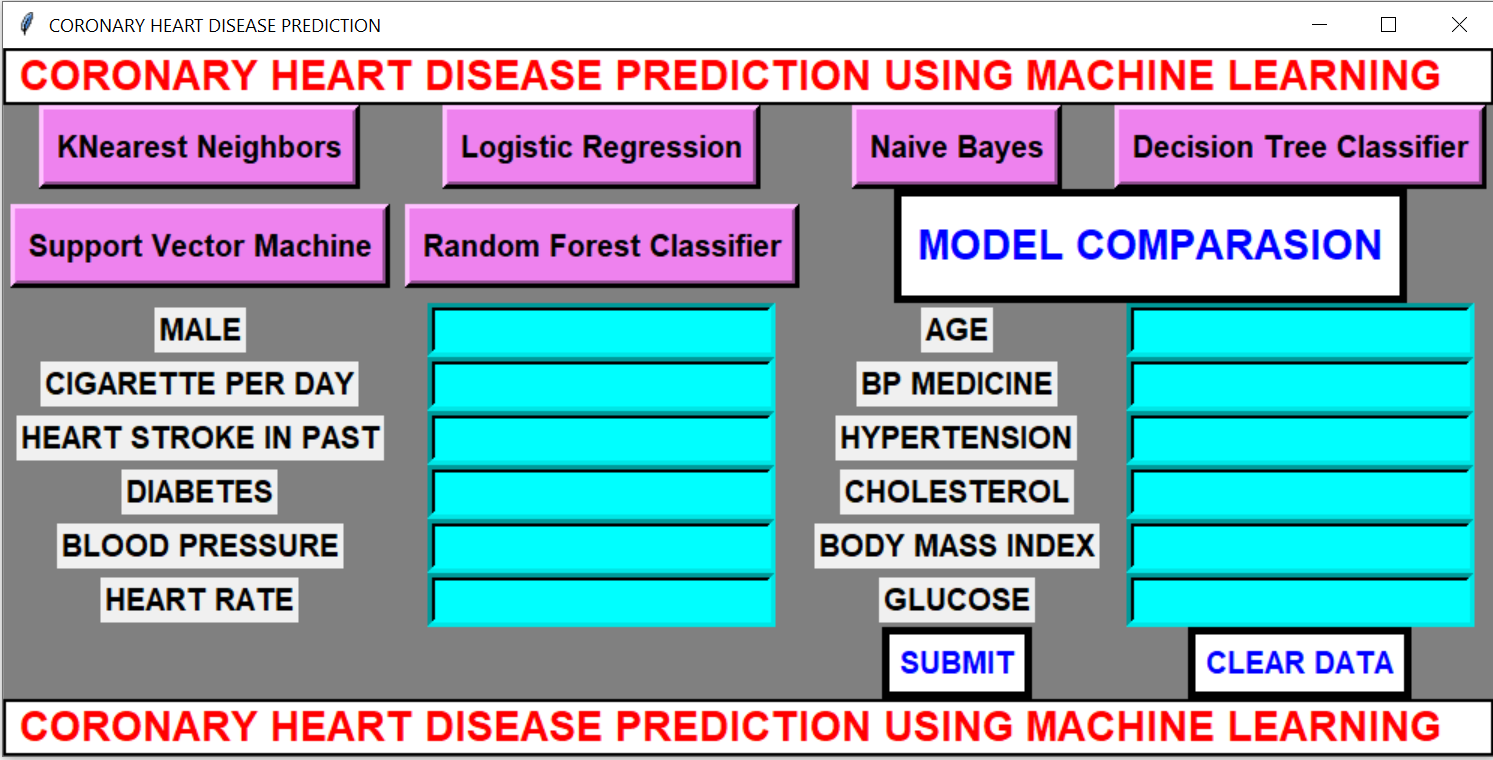


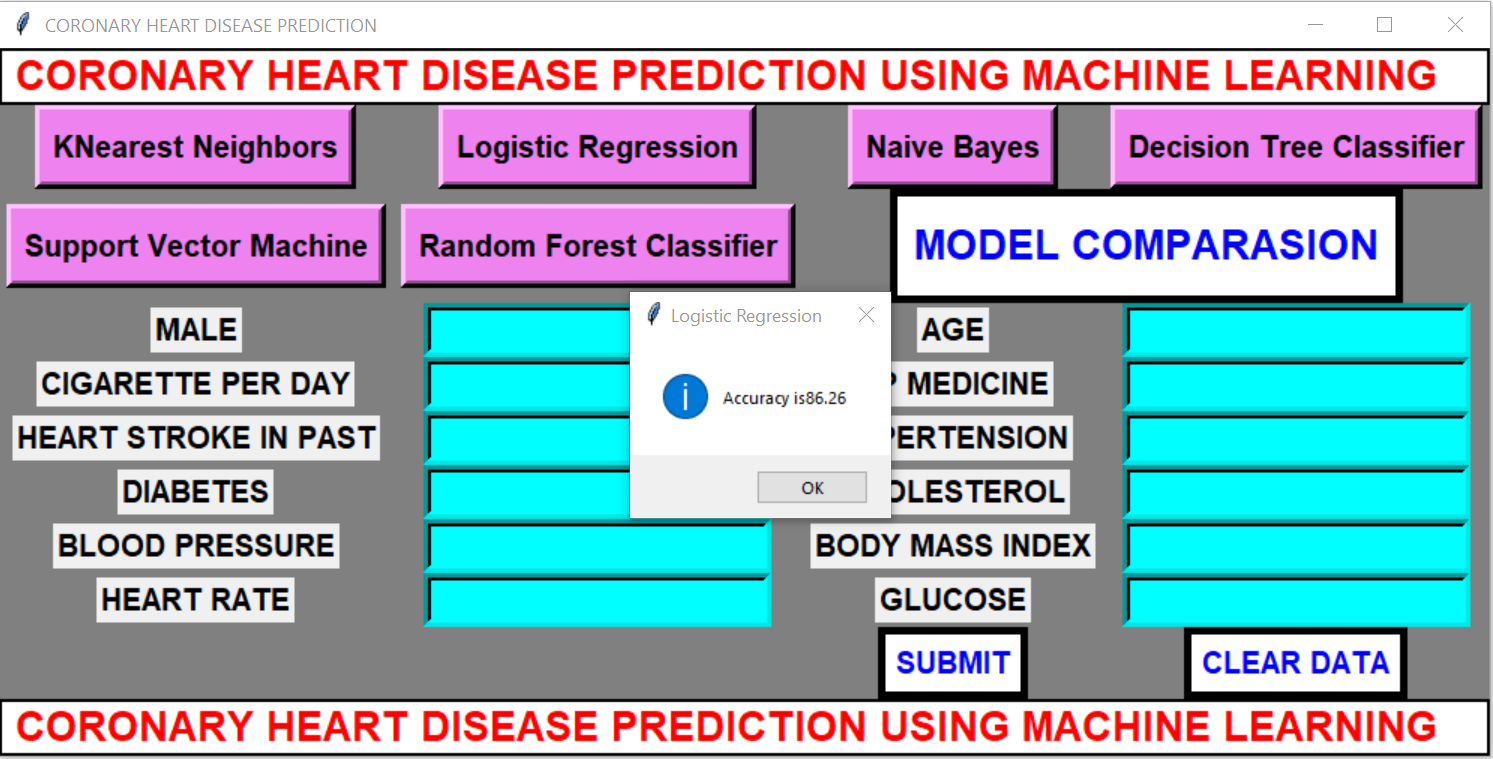




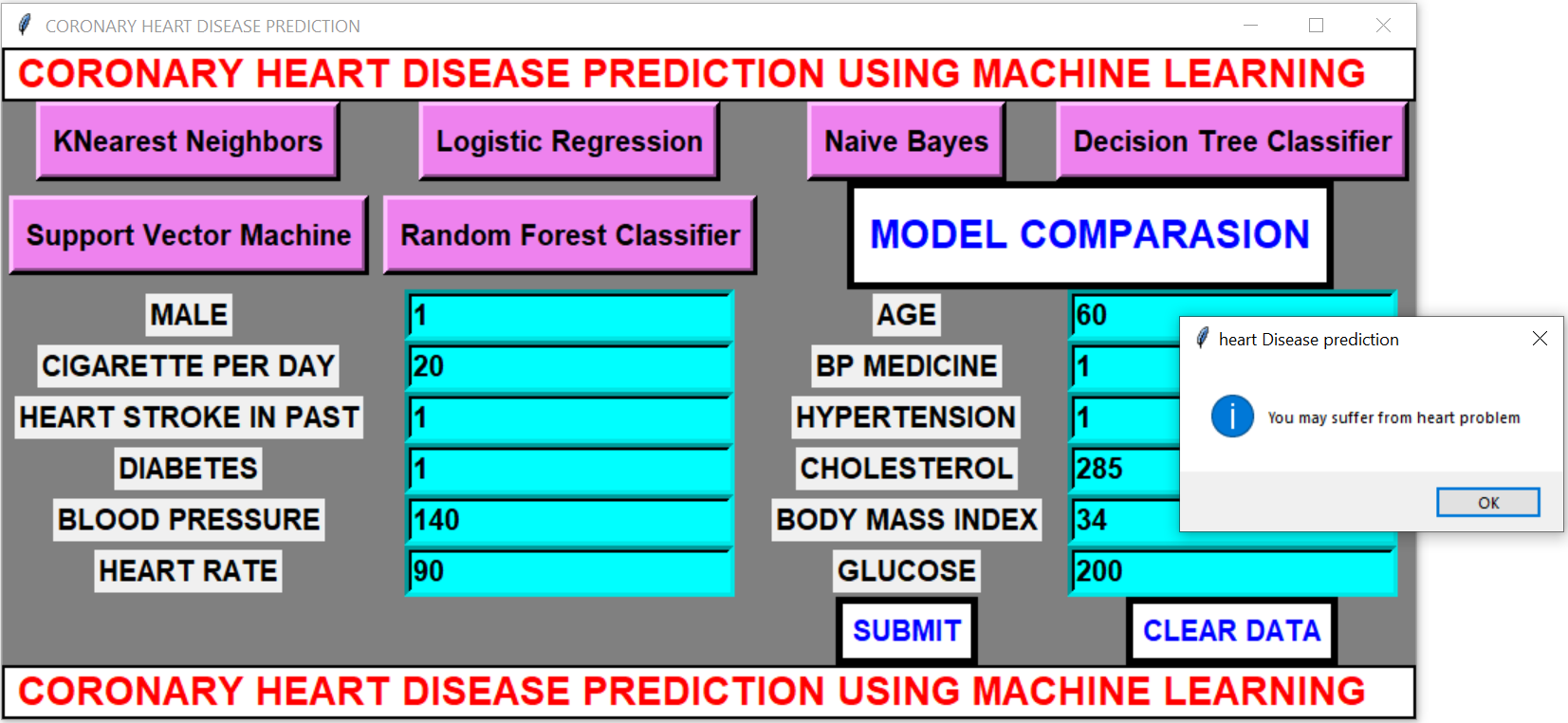
6. GUI Implementation

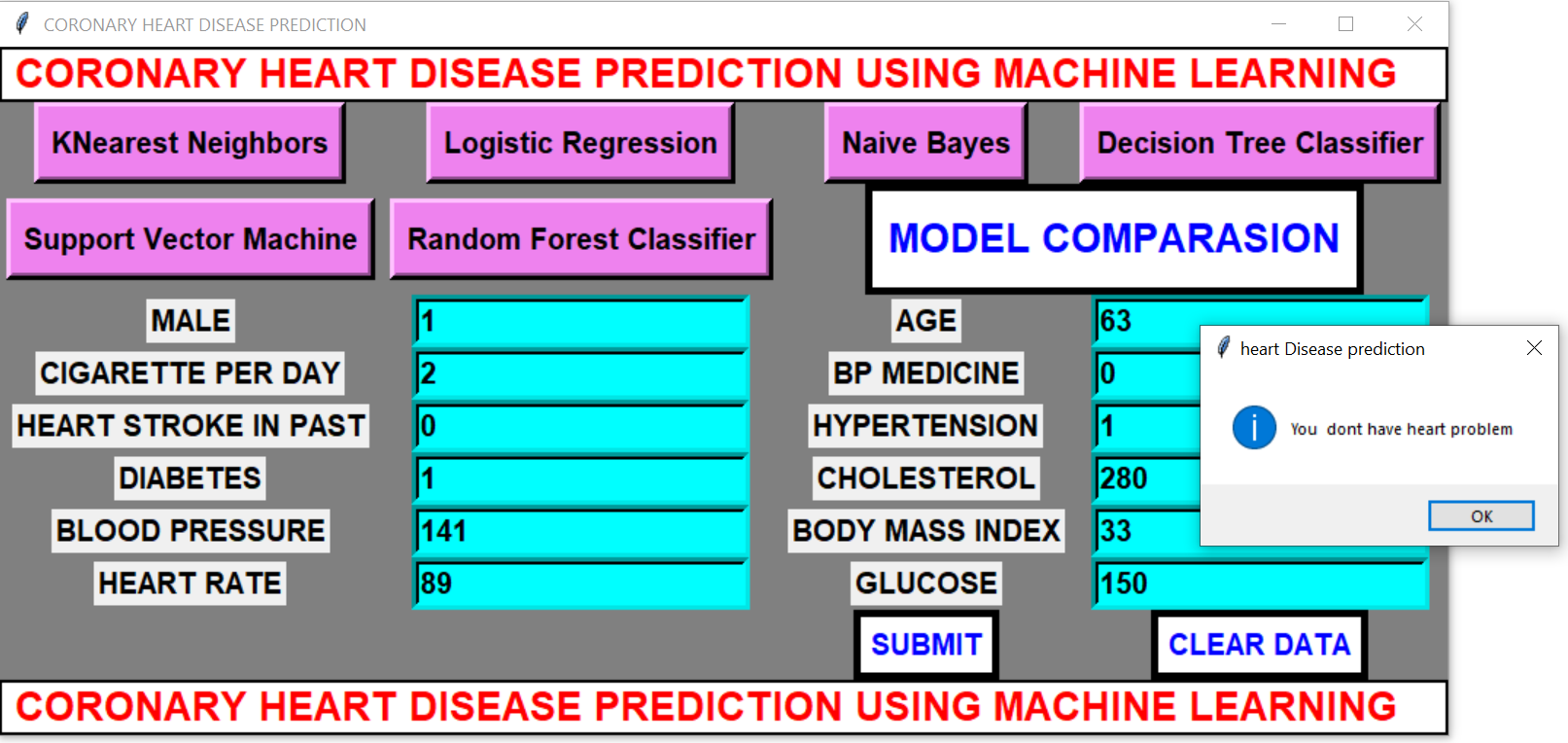
A GUI is developed in python by using tkinter package to generate a simple interface which takes input for all the values necessary for evaluation. After the input is taken from the user, prompt appears which decides whether a person has a presence of heart disease or not.





**Accuracy of Logistic Regression**





**7. Conclusion**

* In this project six different machine learning algorithms are applied for prediction of heart disease.
* They are Logistic Regression, K-nearest Neighbours, Naïve Bayes, Decision Tree, Random Forest and SVM .
* Accuracy of Logistic Regression is 86.26%, K-nearest Neighbours is 85.42%, Naïve Bayes is 83.63%, Decision Tree is 83.03%, Random forest is 86.14% and SVM is 85.9%.
* From the above we conclude that Logistic Regression is the best model for coronary heart disease prediction.

**8. Future Work**

* In this project 6 machine learning algorithms are applied for prediction of coronary heart disease .It was found that Logistic Regression performs better than other algorithms. Also a GUI is developed for prediction of heart disease. In future artificial neural networks algorithms will be used to get better accuracy. Also a web based system will be developed so that people from different parts of world can use it.

**9.Reference**

1. Heart Disease Prediction Using Machine Learning and Data Mining Techniques: Application of Framingham Dataset.Turkish Journal of Computer and Mathematics Education.Vol.12 No.14(2021), 4864- 4870.Walaa Adel Mahmoud , Prof. Dr. Mohamed Aborizka ,Prof. Dr. Fathy Ahmed Elsayed Amer2 .
2. J. S. Sonawane and D. R. Patil, ”Prediction of heart disease using multilayer perceptron neural network,” International Conference on Information Communication and Embedded Systems (ICICES2014), pp. 1–6, Feb, 2014. doi: 10.1109/ICICES.2014.7033860.
3. S. A. Pattekari and A. Parveen, ”Prediction system for heart disease using Naive Bayes,” International Journal of Advanced Computer and Mathematical Sciences, vol. 3, no. 3, pp. 290–294, 2012.
4. K. H. Miao, J. H. Miao, and G. J. Miao, ”Diagnosing Coronary Heart Disease Using Ensemble Machine Learning,” (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 7, no. 10, pp. 30–39, 2016. doi: 10.14569/IJACSA.2016.071004.
5. K. H. Miao and J. H. Miao, ”Coronary Heart Disease Diagnosis using Deep Neural Networks,” (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 9, no. 10, pp. 1-8, 2018. doi: 10.14569/IJACSA.2018.091001.
6. R. Das, I. Turkoglu, and A. Sengur, ”Effective diagnosis of heart disease through neural networks ensembles,” Expert Systems with Applications, vol. 36, no. 4, pp. 7675–7680, May, 2009. doi: https://doi.org/10.1016/j.eswa.2008.09.013.
7. S. Bashir, U. Qamar, F. H. Khan, and M. Y. Javed, ”An Efficient Rule- Based Classification of Diabetes Using ID3, C4.5, & CART Ensem- bles,” in 2014 12th International Conference on Frontiers of Information Technology, pp. 226–231, Dec 2014. doi: 10.1109/FIT.2014.50.
8. T. Panch, P. Szolovits, and R. Atun, ”Artificial intelligence, machine learning and health systems,” Journal of Global Health, vol. 8, Decem- ber. 2018. doi: doi: 10.7189/jogh.08.020303.
9. G. D. Magoulas and A. Prentza, ”Machine Learning in Medical Applica- tions,” Machine Learning and Its Applications, vol. 2049, pp. 300–307, 2001. doi: https://doi.org/10.1007/3-540-44673-7 19
10. S. A. Alasadi and W. Bhaya , ”Review of data preprocessing techniques in data mining,” Journal of Engineering and Applied Sciences, vol. 12, no. 16, pp. 4102–4107, Sep 2017.
11. A. Gupta and V. Khathuria,” Framingham heart study,” International Journal on Future Revolution in Computer Science & Communication Engineering, vol. 4, no. 11, pp. 55–58, 2018.
12. K.V. Nagendra and M. Ussenaiah, ”Analysis of classification algorithms on heart diseases data using association rule mining, ” International Journal of Computational Engineering Research(IJCER), vol. 08, no. 6, pp. 39–46, 2018.
13. K. V. Nagendra and M. Ussenaiah, ”Support vector machine and neural network classification improved by bagging,” International Journal on Future Revolution in Computer Science & Communication Engineering, vol. 4, no. 2, pp. 125–130, 2018.
14. J. Beunza, E. Puertas, E. Garc´ıa-Ovejero, G. Villalba, E. Condes,

G. Condes, and M. F. Landecho ”Comparison of machine learning algorithms for clinical event prediction(risk of coronary heart disease),” Journal of Biomedical Informatics, vol. 97, p. 103257, Sep.2017. doi:https://doi.org/10.1016/j.jbi.2019.103257.

1. A. S. T. Nishadi, ”Predicting heart diseases in logistic regression of machine learning algorithms by python jupyterlab, ” International Journal of Advanced Research and Publications, vol. 3,no. 8, pp. 69–74, Aug 2019.
2. A. Bhardwaj, A. Kundra, B. Gandhi, S. Kumar, A. Rehalia, and

M. Gupta, ”Prediction of heart attack using machine learning,” IITM Journal of Management and IT, vol. 10, no. 1, pp. 20–24, 2019.

1. A. Valle, A. Cinaud, V. Blachier, H. Lelong, M. E. Safar, and J. Blacher ”Coronary heart disease diagnosis by artificial neural networks including aortic pulse wave velocity index and clinical parameters,” Journal of Hypertension, vol. 37, no. 8, pp. 1682–1688, Aug. 2019. doi:10.1097/HJH.0000000000002075.
2. K. Lim, B. M. Lee, U. Kang, and Y. Lee ”An optimized DBN- based coronary heart disease risk prediction,” International Journal of Computers Communications & Control, vol. 13,no. 4, pp. 492–502, Jul 2018. doi: https://doi.org/10.15837/ijccc.2018.4.3269 .
3. J.K. Kim and S. Kang, ”Neural network-based coronary heart disease risk prediction using feature correlation analysis, ” Jour- nal of Healthcare Engineering, vol. 2017, pp. 1–13, 2017. doi: https://doi.org/10.1155/2017/2780501.
4. N.S. Rajliwall, G. Chetty, and R. Davey, ”Chronic disease risk moni- toring based on an innovative predictive modelling framework,” 2017 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1–8, 2017. doi: 10.1109/SSCI.2017.8285257, 1-8, 2017.
5. N. S. Rajliwall, R. Davey, and G. Chetty, ”Machine learning based mod- els for cardiovascular risk prediction,” 2018 International Conference on Machine Learning and Data Engineering(iCMLDE), pp. 142–148, 2018. doi: 10.1109/iCMLDE.2018.00034, 142– 148, 2018.
6. I. D. Mienye, Y. Sun, and Z Wang, ”Improved sparse autoencoder based artificial neural network approach for prediction of heart disease,”, Informatics in Medicine Unlocked, vol. 18, 100307, 2020.
7. P. Puvar, N. Patel, A. Shah, R. Solanki, and D. Rana, ”Heart Disease Detection using Ensemble Learning Approach,” International Research Journal of Engineering and Technology (IRJET), vol. 8, no. 5, pp. 1414- 1418, May, 2021.
8. N. K. Sharma, M. Vemula, and V. Tadiboyina, ”An Experimental Study of Heart Disease Prediction Using Different Supervised Machine Learning Algorithms”, International Journal of Engineering Research and Technology, vol. 14, no. 3, pp. 227-240, 2021.
9. H. A. G. Elsayed and L. Syed, ”An automatic early risk classification of hard coronary heart diseases using framingham scoring model,” in Proceedings of the Second International Conference on Internet of things, Data and Cloud Computing. ACM, pp. 1-8, Mar 2017. doi: https://doi.org/10.1145/3018896.3036384
10. P. Jonsson and C. Wohlin, ”An evaluation of k-nearest neighbour impu- tation using likertdata,” 10th International Symposium on Software Met- rics, 2004. Proceedings.IEEE, pp. 108-118, 2004. doi: 10.1109/MET- RIC.2004.1357895.
11. G. Aksu, C. O. Gu¨zeller, and M.T Eser, ”The Effect of the Nor- malization Method Used in Different Sample Sizes on the Success of Artificial Neural Network Model”, International Journal of Assess- ment Tools in Education, vol. 6, no. 2, , pp. 170–192, 2019. doi: https://doi.org/10.21449/ijate.479404.
12. S. Prasad, ”Some notes on z- scores and t- scores,” International Journal of scientific research and management (IJSRM), vol. 3, no. 4, pp. 2608- 2610, 2015.
13. K. Potdar, T. S. pardawala, and C. D. pai, ”A comparative study of categorical variable encoding techniques for neural network classifiers,” International Journal of Computer Applications, vol. 175 ,no.4, pp. 7–9, Oct, 2017.
14. H. LIU, F. Hussain, C. L.TAN, and M. DASH, ”Discretization: An Enabling Technique,” Data Mining and Knowledge Discovery, vol.6, pp. 393–423, 2002. doi: https://doi.org/10.1023/A:1016304305535.
15. D. Ramesh, P. Suraj, and L. Saini, ”Big data analytics in healthcare: A survey approach,” in 2016 International Conference on Microelec- tronics, Computing and Communications (Mi-croCom), pp. 1–6, Jan, 2016.
16. C. B. C. Latha and S. C. Jeeva, ”Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques,” Informatics in Medicine Unlocked, vol. 16, p.100203, 2019. doi: https://doi.org/10.1016/j.imu.2019.100203.
17. A. H. Khaleel, G. A. Al-Suhail, and B. M. Hussan, ”A weighted voting of k-nearest neighbor algorithm for diabetes mellitus,” International Journal of Computer Science and Mobile Computing, vol. 6, no. 1, pp. 43–51, 2017.
18. N. Ali, D. Neagu, and P. Trundle, ”Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets,” SN Applied Sci- ences, vol 1, no. 12, Nov, 2019.
19. Y. Zhang, Z. Lin, Y. Kang, R. Ning, and Y. Meng, ”A feed-forward neural network model for the accurate prediction of diabetes mellitus,” INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH, vol.7, no.8, pp. 151–155, Aug, 2018.
20. H. Yan, et al., ”A multilayer perceptron-based medical decision support system for heart disease diagnosis,” Expert Systems with Applications, vol.30, no.2, pp. 272 – 281, 2006