**FEATURE PREDICTION OF heart ATTACKS identification with MACHINE LEARNING**

**Team Members**

Sai Susmitha Devi Komatlapalli - 700742098

Bhavana Prathyusha Madhugani - 700737914

Bhuvana Nandhimalla – 700744203

<https://github.com/SusmitharajuK/ml>

One of the most well-known causes of death worldwide is heart disease. Patients are frequently asymptomatic until a fatal incident happens, and in any case, while they are under observation, a trained workforce is necessary to notice a cardiac irregularity. Due to the usability of Electrocardiograms (ECG) in computerized design, there has lately been growing evidence of how Machine Learning may be used to spot such anomalies. With the help of recent innovations, we are now able to use this knowledge to create models that are prepared to analyze heartbeat instances and identify anomalies in them. In this study, expert cardiologists from diverse clinics and countries were able to identify seven different types of symptoms: typical, AF, tachycardia, bradycardia, arrhythmia, other, or boisterous.

In the contemporary clinical examination, identifying heart disease has grown to be a challenging clinical task. This conclusion is based on an accurate, minute analysis of the subject's diagnostic test data and the individual's medical history. With machine learning, the vast advancements in the area of deep learning aim to create smart automated systems that aid medical professionals in predicting and determining sickness. In a medical care scenario, like the one addressed in this composition, certainty modification is a crucial issue since it is crucial that the outcome of a neural network simulation can be trusted.

With various sorts of attributes from the dataset, the machine intelligence methodology will be used to forecast heart disease.

**Problem definition:**

It is still difficult to predict and identify the pace of complicated continuing conditions, such as cardiovascular deterioration. constrained observable evidence indicating the considerable expectation of perplexing ongoing conditions, such as Heart Failure. Notably, the current designs have demonstrated modest execution in vastly complicated cardiac datasets for risk anticipation in various conditions, including heart failure. The ability to demonstrate results in a language that people can understand. This has reduced their willingness to take a risk with significant statements and more thorough clinical reception, as well as their trustworthiness.

**Execution Plan:**

Cardiovascular disease detection using machine learning aims to identify heart disease at an early stage using the given features. The dataset that contains information about cardiovascular illnesses will be used in this endeavor. The preliminary processing will be carried out to the dataset, and the data that is noisy and has null values will be eliminated. Following data analysis and visualization, further computation will take place. To produce the forecast, a machine intelligence algorithm will be used.

Two components of the information set will be separated. 30% of the data from the initial dataset is used for testing, while the remaining 70% is used to train the deep learning algorithm.

**Contribution :**

The Machine Learning algorithm will be the Python-based application that contributes to finding out heart disease's early stage. It will be helpful for humans to detect it early and to take the necessary treatments at the correct time.

**Evaluation:**

The outcomes of the machine learning algorithm's prediction may be used to test the project evaluation. The preciseness of the algorithm result will be useful to assess the outcomes since the data generated by the Machine Learning algorithm will be utilized for foreseeing the disease. The dataset is evaluated using the algorithm's accuracy score for identifying heart disease.

The project may be performed immediately on any kind of hardware or software with an internet connection, thus the application will be constructed with the Google Colab Python Tool. No particular software has to be installed on the user machine. The Colab Tool aids in the development and execution of the application within the cloud server that hosts the Python library files. There are libraries for the Machine Learning algorithms built in.

**Techniques Applied:**

The assignment includes looking at a record of patients with cardiovascular conditions and managing the information appropriately. Then, several models were created, and predictions were generated using different Logistic Regression, KNN, Random Forest, and SVM model computations. The program uses the Sklearn and Keras machine learning libraries as well as other machine learning libraries.

**Dataset:**

In a database, the majority of the columns are overflowing with data. However, by using characteristic engineering, we will get even better outcomes. Importing frameworks and loading data forms the initial step. A fundamental knowledge of the information will then be required, including its form, survey, and the presence of any empty values in the data file. The first phase in making a forecast or beginning any Artificial Learning endeavor is getting acquainted with the data. There being no invalid values is a beneficial thing.

**Detailed Design of Features:**

The bulk of entries in a repository are filled to the brim with information. Yet, we will achieve superior results if we use feature modeling. The first phase entails bringing in data and integrating modules. The details form, research, and the existence of any unfilled fields in the input file must then be understood on a basic level. Acquiring familiar with the facts is the initial step in creating a prediction or starting any Artificial Learning project. The absence of unacceptable numbers is a good thing.

**Analysis of Heart Disease Prediction:**

The process will begin with the primary portion and go into each subsection to understand what impact it has on the goal segment. We will also carry out preliminary processing at the required phase and incorporate developing tasks. The goal of conducting top-to-bottom exploratory analysis is to gather and purify data for enhanced demonstration to achieve exceptional results and summed-up designs. Therefore, it should start with decomposing and preparing the information set for expectancy.

**Modules:**

1) Dataset collection

2) Data cleaning

3) Exploratory Data Analysis

4) Machine learning Modeling

5) Report

**1) Dataset collection:**

Visitors of various backgrounds provided data for a coronary illness collection with various sorts of features.

**2) Data Cleaning:**

The huge dataset has to be beforehand processed to create a solid dataset for additional trimming since it includes many erroneous and chaotic information. The first step in cleaning and processing the data is eliminating the values that are empty.

**3) Exploratory Data Analysis:**

An exploratory analysis is a technique to thoroughly examine and comprehend the link between the information and the dataset itself in order to make the phases of feature engineering and machine learning programming easy and efficient for forecasting. EDA assists in demonstrating the accuracy of our hypotheses. In simpler terms, performing hypothesis testing is helpful.4)

**4)Machine Learning Modeling:**

Finding the optimum algorithm and the appropriate hyperparameters to attain optimal precision is made easier with the aid of machine learning modeling. There are two versions of the dataset. The machine learning algorithm is trained using 70% of the records as training data. Testing is done on the remaining 30% of the dataset in order to forecast the procedure.

**5) Report:**

The Data is visualized based on the output of the Machine Learning algorithm and the data is mapped with different types of graphs to analyze and visualize the exact data to the user for the prediction. Marplot libraries are implemented to map the results based on user requirements

**SYSTEM SPECIFICAION**

**HARDWARE REQUIREMENTS:**

* Processor Intel(R) Pentium(R) CPU G2010 @
* Clock Speed 2.80GHz
* RAM 2.00 GB
* Hard Disk 1 TB HDD
* Monitor 15.6 Inches
* Mouse Logitech B100 Wired Optical Mouse
* Keyboard Full-size island-style keyboard with number

Keypad

* Display Card Super Video Graphics Adapter

**SOFTWARE REQUIREMENTS**

* Operating System : Windows 10
* Front-End Tool : Python in Google Colab

**SYSTEM ARCHITECTURE DIAGRAM**

Heart Disease prediction

Training, test data

Data Cleaning

Data preprocess

ML Algorithm implementation

Heart diseaseDataset

**ARCHITECTURE DIAGRAM**

The program receives the data set on cardiovascular disease as an input and does pre-processing. Once the training and test sets of data have been divided up and fed into the machine learning algorithm, the forecasting of heart attacks is completed.

**Load Packages:**

First step have to import the necessary packages to the application:

//--- Libray files

from keras.models import Sequential

from keras.layers import Dense

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

%matplotlib inline

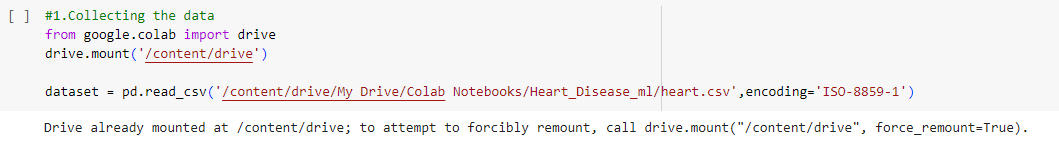
import os

print(os.listdir())

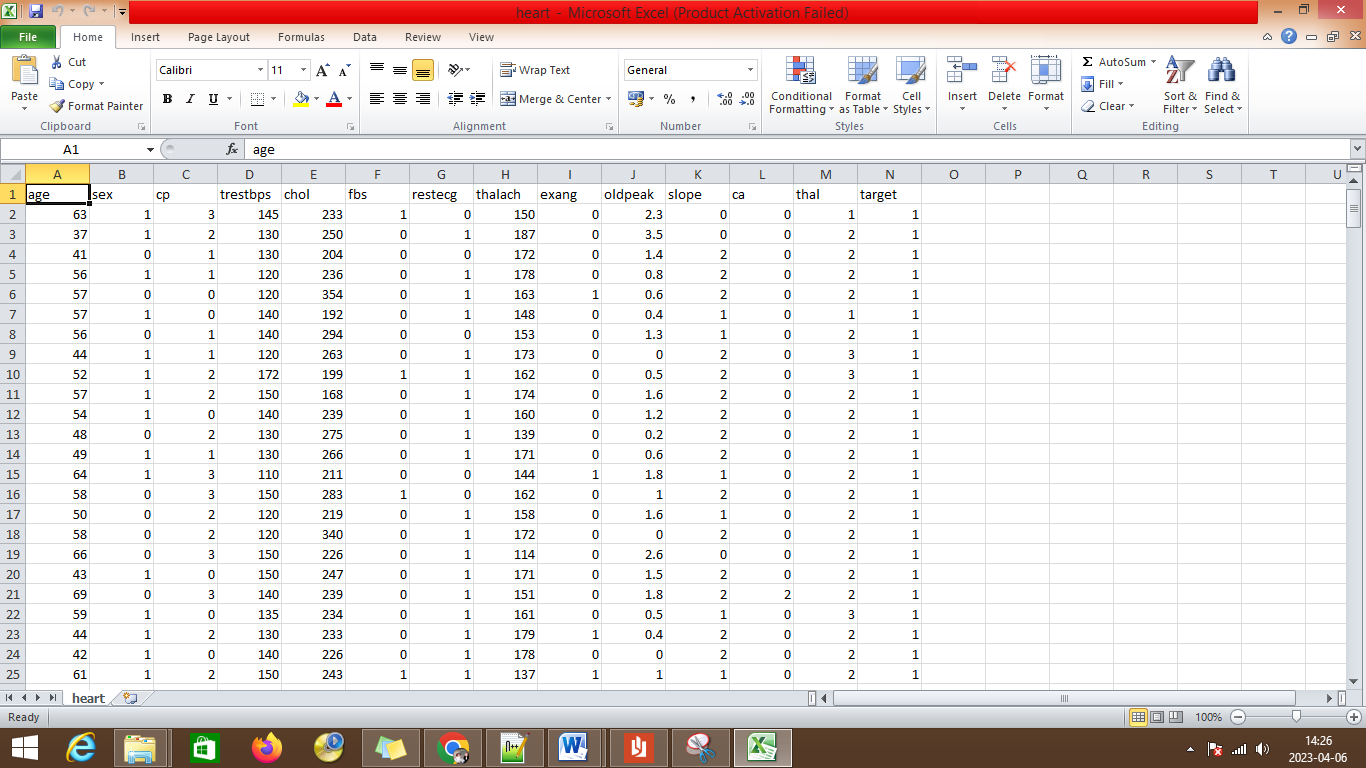
import warnings

warnings.filterwarnings('ignore')

The dataset would then be linked from Google Colab at this point. The dataset is initially uploaded to the Google Colab folder. The Python file should then establish a connection with the Google Colab folder's location.

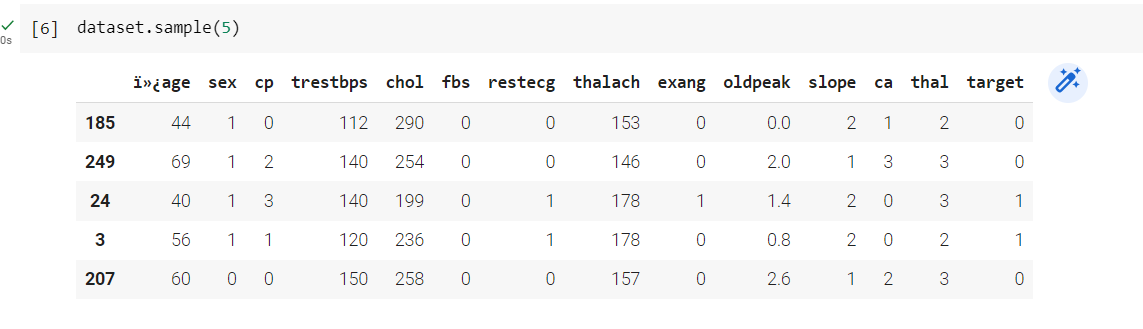
****

The information has an extremely straightforward design with elements. Each column is related to the heart disease attributes.



The label has been set with different label description features of:

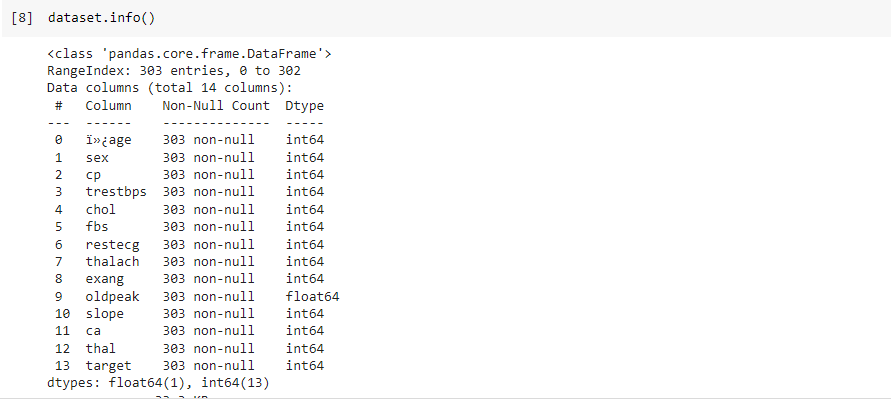
* age
* sex
* cp
* trestbps
* cholestral
* fbs
* restecg
* oldpeak
* slope
* ca
* thal
* target



The dictionary shows the records displayed with head values of the first 5 records from the dataset. The bar plot can be used to address the heart disease attributes.

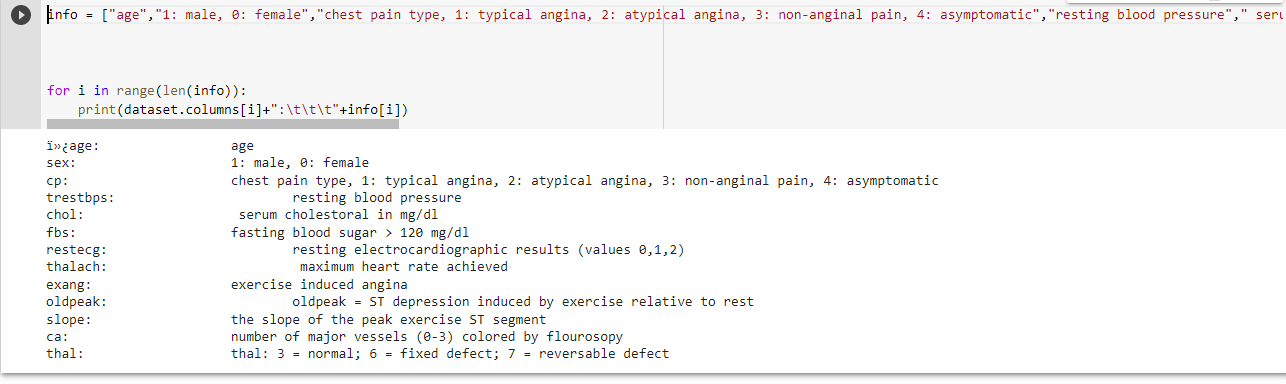


The describe function describes the coun, mean, max, and min statistical reports for the columns of age, sex, cp, chol, and other fields.



The information about the data is displayed with the data type.

**Attributes Exploration:**



The heart disease dataset attributes are explored for further analyzing.

Below are the values:

sex: 1: male, 0: female

cp: chest pain type, 1: typical angina, 2: atypical angina,

3: non-anginal pain, 4: asymptomatic

trestbps: resting blood pressure

chol: serum cholestoral in mg/dl

fbs: fasting blood sugar > 120 mg/dl

restecg: resting electrocardiographic results (values 0,1,2)

thalach: maximum heart rate achieved

exang: exercise induced angina

oldpeak: oldpeak = ST depression induced by exercise relative to rest

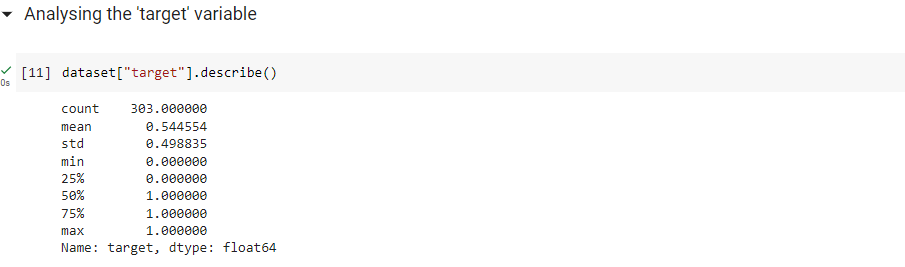
slope: the slope of the peak exercise ST segment

ca: number of major vessels (0-3) colored by flourosopy

thal: thal: 3 = normal; 6 = fixed defect; 7 = reversable defect

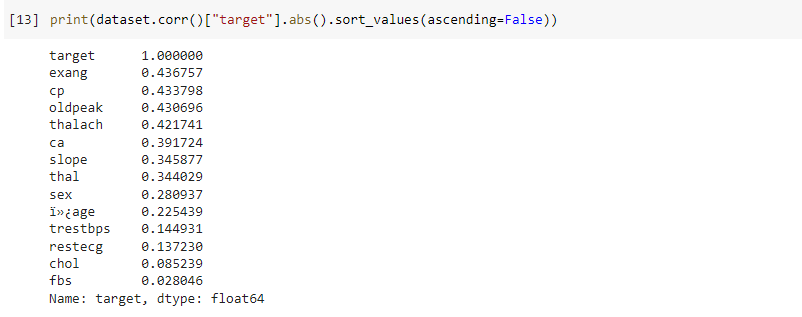
**Analysing :**

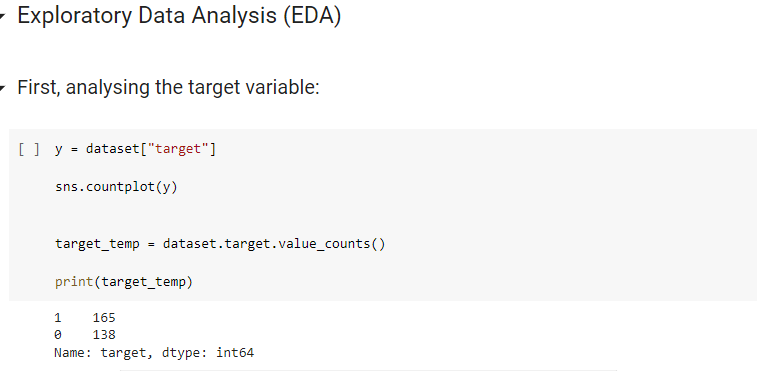
The target variables are analyzed



**Correlation of data:**

The correlation of data with respect to the attributes is linked and displayed.





The target variable is assigned to the y variable.

**Implemenation with Algorithms:**

The testing and training variables are split and passed into the algorithm for heart disease prediction.



The testing size is 20% and the training size is 80% of the dataset.

**Evaluation with Logistic Regression Algorithm:**

Logistic regression predicts the result of a dependent variable. Thus the result should be a downright or discrete worth. It tends to be either Yes or No, 0 or 1, and so on however rather than giving the specific worth as 0 and 1, it gives the probabilistic qualities which lie somewhere in the range of 0 and 1.

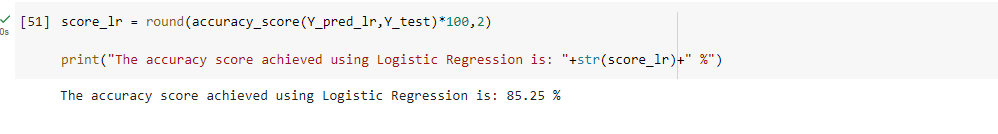
Logistic Regression is much like Linear Regression aside from how they are utilized. Direct Relapse is utilized for taking care of Regression problems, while Calculated relapse is utilized for tackling arrangement issues.

In Calculated relapse, rather than fitting a relapse line, we fit an "S" molded strategic capability, which predicts two greatest qualities (0 or 1).



**Results:**

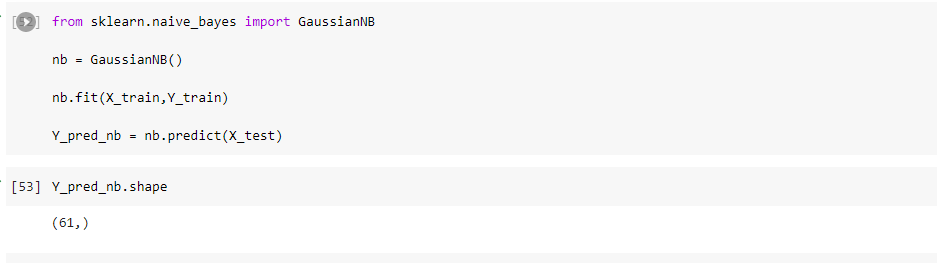
The results of the logistic regressin algorithm shows the accuracy of 85.25% with good performance in prediction.



**Evaluation with Naive Bayes:**

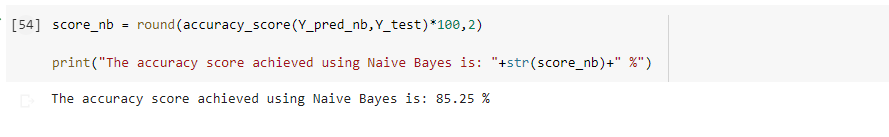
Information scientists use the AI algorithm Naive Bayes for characterisation. In light of the Bayes hypothesis, the naïve Bayes method operates. The Naive Bayes is a grouping computation that is based on the Bayes hypothesis, which is a method for calculating the likelihood of an event given its prior knowledge. The computation is deemed "innocent" since it disproves the theory that the highlights are exclusively separate from one another given the grade.

The Naive Bayes computation may be used to solve multi-class order problems as well as problems with two classes. It may also be used in other applications where there are several classes and varied features. It is typically used in message characterisation tasks like spam sorting or opinion analysis.

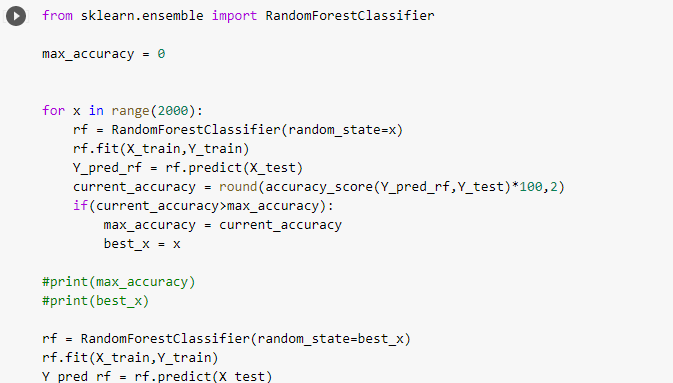


**Results:**

The results of the Naive Bayes algorithm show an accuracy of 85.25% with good performance in prediction.

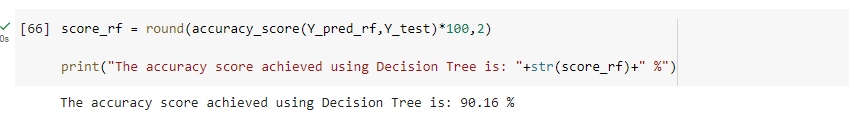


**Evaluation with Random Forest Algorithm:**

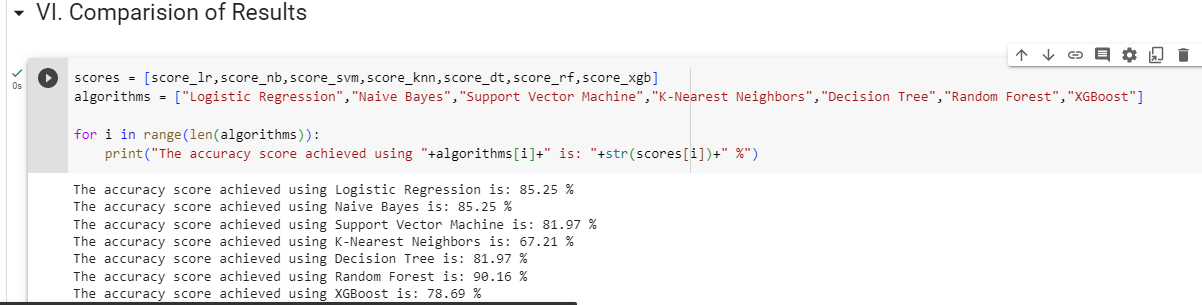


The random forest algorithm helps in prediction of the heart disease with the available dataset.

**Results:**



The results of the random forest algorithm show an accuracy of 90% with good performance in prediction.



**Conclusion and Future Enhancement:**

The accuracy levels of the heart disease were detected and compared using several sorts of algorithms. The machine learning method offers higher degrees of prediction accuracy. Future forecasts might make use of a huge dataset, which would help the system make predictions more accurately.