A

Project Report

ON

Enhancing Heart Stroke Disease Prediction Accuracy Using Machine Learning Models

Submitted to

Rajiv Gandhi University of Knowledge Technologies,

RK Valley, KADAPA.

in partial fulfillment of the requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by

A.SUBBA RAYUDU	**R170979**
T.SAI YOGESH	**R170268**
O.VENKATA KRISHNAIAH	**R170452**

Under the Guidance of

B.Madhan Mohan,M.Tech

Assistant Proffesor,H.O.D. of E.C.E.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES,

RK VALLEY, KADAPA (DIST.), ANDHRA PRADESH, PINCODE -516330.

MARCH 2023-MAY 2023.

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES,

RK VALLEY, KADAPA (DIST.), ANDHRA PRADESH,

PINCODE -516330.March 2023-May 2023.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project report entitled **"Enhancing Heart Stroke Disease Prediction Accuracy Using Machine Learning Models"** a bonafide record of the project work done and submitted by

T.SAI YOGESH	**R170268**
O.VENKATA KRISHNAIAH	**R170452**
A.SUBBA RAYUDU	**R170979**

for the partial fulfillment of the requirements for the award of B.Tech. Degree in **ELECTRONICS AND COMMUNICATION ENGINEERING,** RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES , RK VALLEY.

Project Internal GUIDE

Mr. B.MADHAN MOHAN ,
Head of Department,
Department of E.C.E.,
RGUKT, RK Valley,KADAPA
A.P.,PINCODE :516330.

Head of the Department

Mr. B.MADHAN MOHAN ,
Head of Department,
Department of E.C.E.,
RGUKT, RK Valley, KADAPA
A.P.,PINCODE :516330.

External Viva-Voce Exam Held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We hereby declare that the project report entitled "**Enhancing Heart Stroke Disease Prediction Accuracy Using Machine Learning Models**" submitted to the Department of **ELECTRONICS AND COMMUNICATION ENGINEERING** in partial fulfillment of requirements for the award of the degree of **BACHELOR OF TECHNOLOGY.** This project is the result of our own effort and that it has not been submitted to any other University or Institution for the award of any degree or diploma other than specified above.

By,

A.SUBBA RAYUDU	R170979
T.SAI YOGESH	R170268
O.VENKATA KRISHNAIAH	R170452

ACKNOWLEDGEMENTS

We are thankful to our guide **Mr.B.Madhan Mohan** for his valuable guidance and encouragement. His helping attitude and suggestions have helped us in the successful completion of the project.

We would like to express our gratefulness and sincere thanks to **Mr.B.Madhan Mohan**, Head of the Department of **ELECTRONICS AND COMMUNICATION ENGINEERING**, for his kind help and encouragement during the course of our study and in the successful completion of the project work.

We have great pleasure in expressing our hearty thanks to our beloved Director **Dr. SANDHYA RANI** for spending her valuable time with us to complete this project.

Successful completion of any project cannot be done without proper support and encouragement. We sincerely thanks to the Management for providing all the necessary facilities during the course of study.

We would like to thank our parents and friends, who have the greatest contributions in all our achievements, for the great care and blessings in making us successful in all our endeavors.

By,

A.SUBBA RAYUDU R170979

T.SAI YOGESH R170268

O.VENKATA KRISHNAIAH R170452

ABSTRACT

Heart Stroke prediction is one of the most complicated tasks in medical field. In the modern era, approximately one person dies per minute due to heart Stroke. Data science plays a crucial role in processing huge amount of data in the field of healthcare. As heart stroke prediction is a complex task, there is a need to automate the prediction process to avoid risks associated with it and alert the patient well in advance.

The proposed work predicts the chances of Heart Stroke and classifies patient's risk level by implementing different data mining techniques such as KNN, Decision Tree and Random Forest.The trial results verify that Random Forest algorithm has achieved the highest accuracy of 90.16% compared to other ML algorithms implemented.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABSERVATION

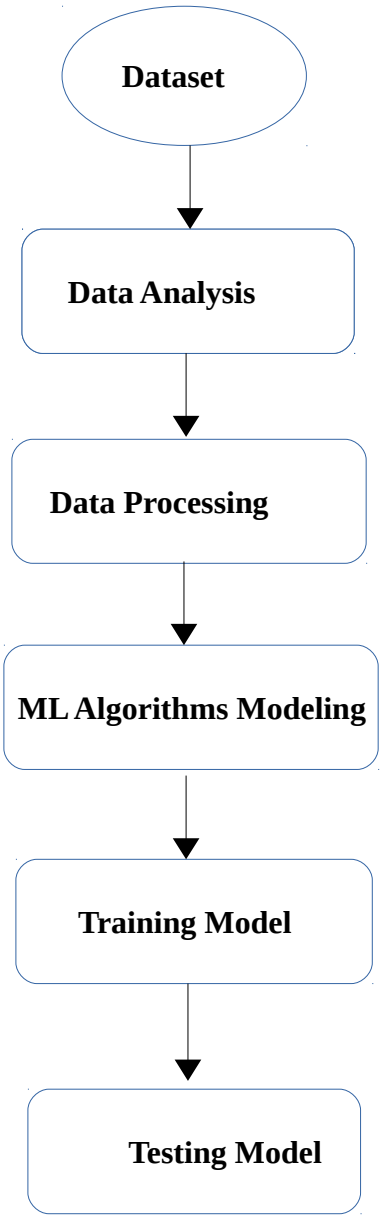
ML	Machine Learning
SVM	Support Vector Machine
CP	Chest Pain
KNN	K-Nearest Neighbours

1. INTRODUCTION

The major challenge in heart disease is its detection. There are instruments available which can predict heart disease but either it are expensive or are not efficient to calculate chance of heart disease in human. Early detection of cardiac diseases can decrease the mortality rate and overall complications. However, it is not possible to monitor patients everyday in all cases accurately and consultation of a patient for 24 hours by a doctor is not available since it requires more sapience, time and expertise. Since we have a good amount of data in today's world, we can use various machine learning algorithms to analyze the data for hidden patterns. The hidden patterns can be used for health diagnosis in medicinal data.

Machine learning proves to be effective in assisting in making decisions and predictions from the large quantity of data produced by the health care industry. This project aims to predict future Heart Disease by analyzing data of patients which classifies whether they have heart disease or not using machine-learning algorithm.Machine Learning techniques can be a boon in this regard. Even though heart disease can occur in different forms, there is a common set of core risk factors that influence whether someone will ultimately be at risk for heart disease or not. By collecting the data from various sources, classifying them under suitable headings & finally analysing to extract the desired data we can say that this technique can be very well adapted to do the prediction of heart disease.

1.1.Block Diagram:-



2. DATASET

Initially, we collect a dataset for our heart disease prediction system. After the collection of the dataset, we split the dataset into training data and testing data. The training dataset is used for prediction model learning and testing data is used for evaluating the prediction model. For this project, 70% of training data is used and 30% of data is used for testing. The dataset used for this project is Heart Disease UCI. The dataset consists of 76 attributes; out of which, 14 attributes are used for the system.

2.1. Importing Dataset of Patient Details

Importing the data set from Google drive to the Google Colab Online Notebook which supports to Graphics Processing Unit (GPU) to train our models faster if our execution platform is connected to a GPU manufactured by NVIDIA.

```
>>from google.colab import drive

>>drive.mount('/content/data')
```

OUTPUT:-Mounted at /content/data

2.2.Importing Important Libraries

Importing libraries also ensures code reusability, meaning that you can use the same libraries in multiple projects without having to rewrite the same code again and again. This helps to standardize your code and makes it easier to maintain and update over time.

2.2.1.Numpy

NumPy is a library for the python programming language, adding support for large,multi-dimensional arrays and matrices, along with a large collection of high level mathematical functions to operate on these arrays. The ancestor of NumPy, Numeric,was originally created by Jim with contributions from several other developers. In 2005, Travis created NumPy by incorporating features of the competing Numarray into Numeric, with extensive modifications. NumPy is open source software and has many contributors.

>>import numpy as np

2.2.2.Pandas

Pandas is a popular open-source Python library used for data manipulation, analysis, and cleaning. It provides powerful data structures for working with structured, tabular data, such as spreadsheets or SQL tables. Pandas is widely used in various fields such as finance, economics, statistics, social sciences, and engineering.

>>import pandas as pd

2.2.3.Matplotlib

Matplotlib is a popular open-source Python library used for data visualization. It provides a wide range of tools for creating different types of plots and charts, including line plots, scatter plots, bar plots, histograms, and many others.

>>import matplotlib.pyplot as plt

2.2.4.Seaborn

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn is a library in Python predominantly used for making statistical graphics. Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

>>import seaborn as sn

>>%matplotlib inline

2.2.5.OS

The os library in Python is a built-in module that provides a way to interact with the operating system. It provides a platform-independent way of working with files, directories, and processes.

```
>>import os

>>print(os.listdir())

>>['.config', 'heart.csv', 'sample_data']

#Importing and understanding our dataset

>>dataset = pd.read_csv("/content/heart.csv")

>>dataset.shape

>>(303, 14)
```

2.3.Input Data Attributes Selection

Gender (value 1: Male; value 0 : Female) Chest Pain Type (value 1: typical type 1 angina, value 2: typical type angina, value 3: non-angina pain; value 4: asymptomatic) Fasting Blood Sugar (value 1: > 120 mg/dl; value 0:< 120 mg/dl) Exang – exercise induced angina (value 1: yes; value 0: no) CA – number of major vessels colored by fluoroscopy (value 0 – 3) Thal (value 3: normal; value 6: fixed defect; value 7:reversible defect) Trest Blood Pressure (mm Hg on admission to the hospital) Serum Cholesterol (mg/dl) Thalach – maximum heart rate achieved Age in Year Height in cms Weight in Kgs.

```
>>dataset.sample(10)

>>dataset.describe()
```

Table.*2.3.Input Data Attributes Selection*

S.No.	Attribute	Description
1.	Age	Patient's age (29 to 77)
2.	Sex	Gender of patient(male-0 female-1)
3.	Cp	Chest pain type
4.	trestbps	Resting blood pressure(in mm Hg on admission to hospital ,values from 94 to 200)
5.	Chol	Serum cholesterol in mg/dl, values from 126 to 564)
6.	Fbs	Fasting blood sugar>120 mg/dl, true-1 false-0)
7.	Resting	Resting electrocardiographics result (0 to 1)
8.	Thail	Maximum heart rate achieved(71 to 202)
9.	Exang	Exercise included agina(1-yes 0-no)
10.	Oldpeak	ST depression introduced by exerciserelative to rest (0 to .2)
11.	Slope	The slop of the peak exercise ST segment (0 to 1)
12.	Ca	Number of major vessels (0-3)
13.	Thal	3-normal
14.	Targets	1 or 0

3.DATASET ANALYSIS

Dataset analysis is the process of examining and understanding the patterns and relationships within a dataset. It involves using statistical and visualization techniques to extract insights and knowledge from the data.Dataset analysis is a critical step in many fields, including data science, machine learning, and business intelligence.

3.1.Target Value Count

Target value count, also known as class distribution or class balance, refers to the distribution of target values or classes in a dataset. In many supervised learning problems, the goal is to predict the target variable or class label based on the input features. A target variable can have two or more classes, and the distribution of these classes can have a significant impact on the performance of a machine learning model.

```
>>Y = dataset["target"]

>>Y.value_counts()

>>sns.countplot(x=Y)

>>target_temp = dataset.target.value_counts()

>>print(target_temp)

1 165

0 138

Name: target, dtype: int64
```

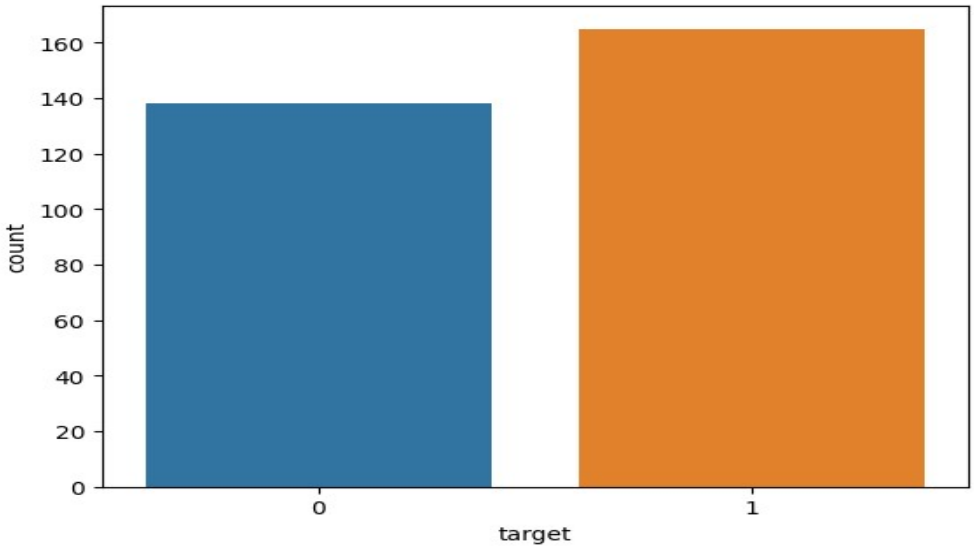


Fig.3.1.Target Value Count

>>print("% of Patients without heart problems :"+str(round(target_temp[0]*100/303,2)))

>>print("% of Patients with heart problems : "+str(round(target_temp[1]*100/303,2)))

>>Percentage of Patients without heart problems : 45.54%

>>Percentage of Patients with heart problems : 54.46%

3.2.Analysing the features of CP

Chest pain type (CP) is an important feature in the diagnosis of heart disease and is often included in datasets used for classification analysis. CP can be classified into four categories: typical angina, atypical angina, non-anginal pain, and asymptomatic.

Angina is a serious medical condition that requires prompt diagnosis and treatment. Treatment may include lifestyle changes, such as quitting smoking or adopting a healthier diet, medication to lower blood pressure and cholesterol levels, and procedures such as angioplasty or coronary artery bypass surgery.

>>dataset['cp'].unique()

>>sns.barplot(x=dataset['cp'],y=Y)

>><Axes: xlabel='cp', ylabel='target'>

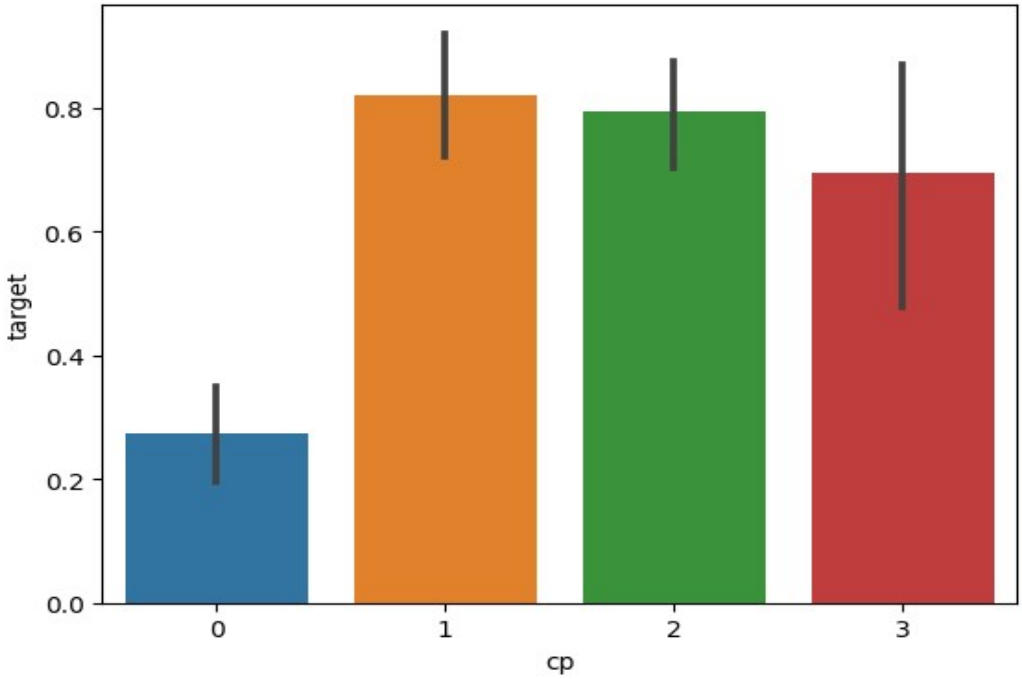


Fig.3.2.Analysing the features of CP

3.3.Analysing the feature of thal

The thal feature is a categorical variable that represents the results of a thallium stress test, which is a type of cardiac imaging test used to diagnose heart disease. The thal feature can take on three values: normal, fixed defect, and reversible defect.

>>dataset['thal'].unique()

>>array([1, 2, 3, 0])

>>sns.barplot(x=dataset['thal'],y=Y , alpha = 1)

>><Axes: xlabel='thal', ylabel='target'>

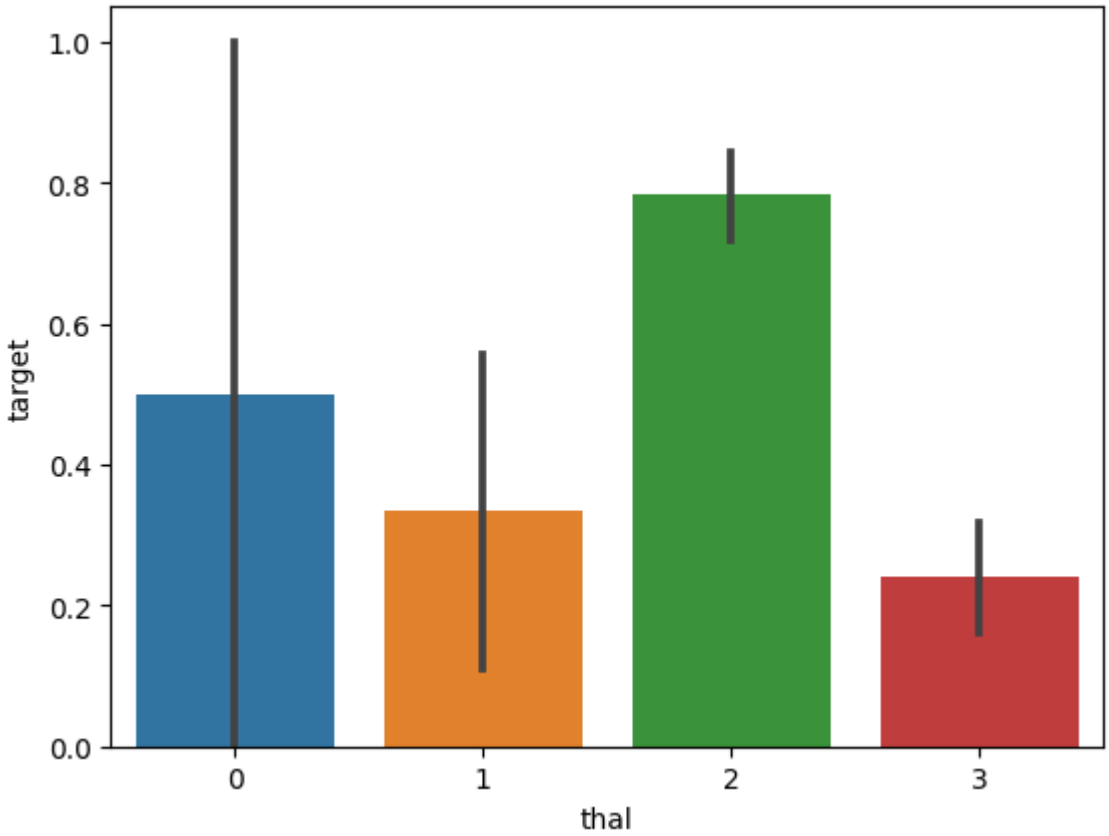


Fig.3.3.Analysing the feature of thal

3.4.Analysing of Gender

```
>>dataset['sex'].unique()

# The 0's (Females) are more likely to have heart diseases than males

>>array([1, 0])

>>sns.barplot(x = dataset['sex'],y = Y)

>><Axes: xlabel='sex', ylabel='target'>
```

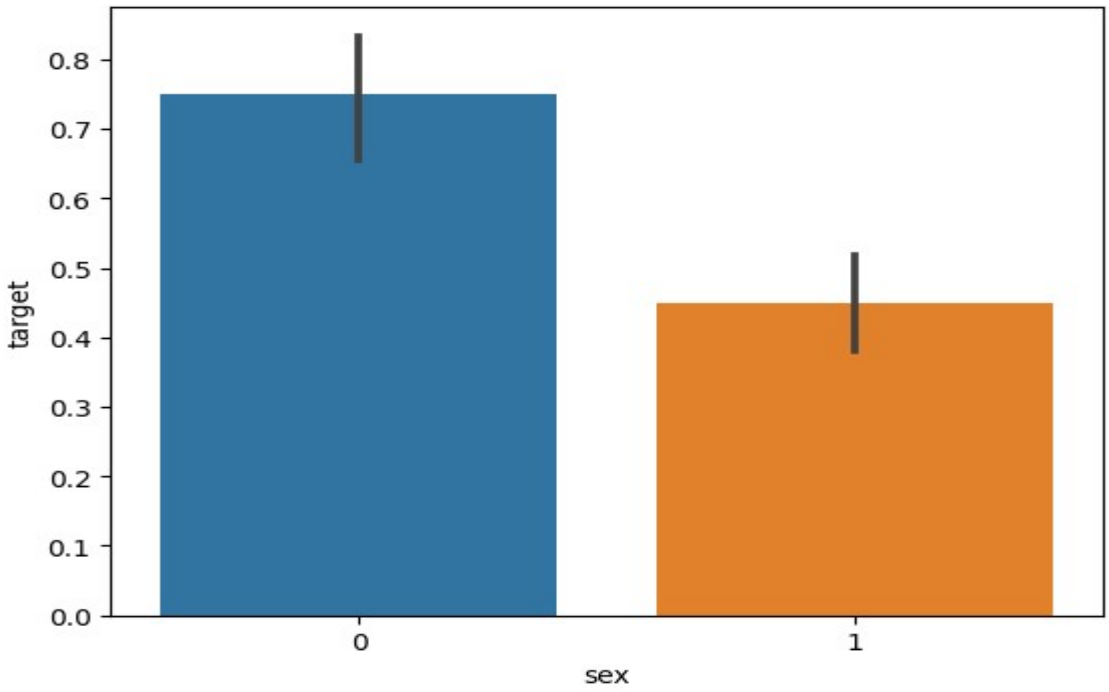


Fig.3.4.Analysing of Gender

3.5.Analysing of FBS

>>dataset['fbs'].describe

<bound 1 2 3 4

method NDFrame.describe of 0 0 0 0 0 ..

298 0

299 0

300 1

301 0

302 0

Name: fbs, Length: 303, dtype: int64> 1

>>dataset['fbs'].unique()

array([1, 0])

>>sns.barplot(x = dataset['fbs'], y = Y)

<Axes: xlabel='fbs', ylabel='target'>

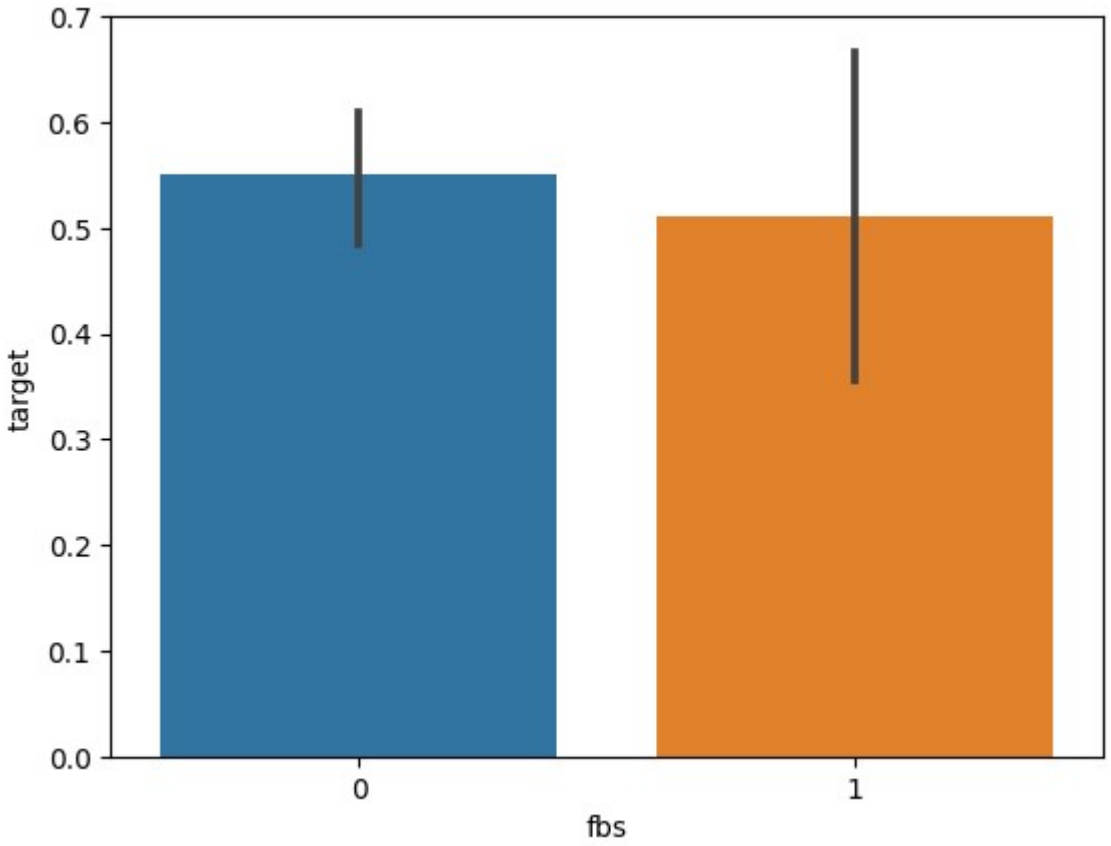


Fig.3.5.Analysing of FBS

3.6.Analysing of restecg

>>dataset['restecg'].unique()

array([0, 1, 2])

>>sns.barplot(x = dataset['restecg'], y = Y)

<Axes: xlabel='restecg', ylabel='target'>

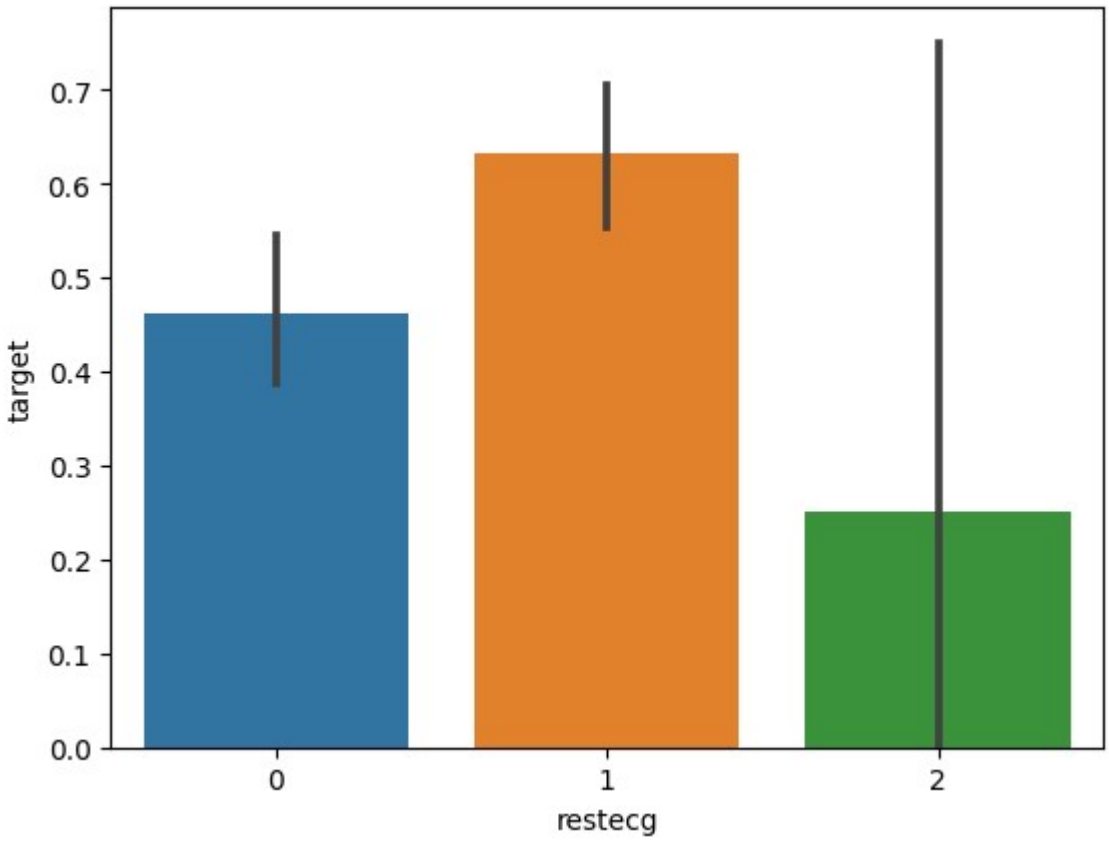


Fig.3.6.Analysing of restecg

3.7.Analysing the exang

dataset['exang'].unique()

array([0, 1])

sns.barplot(x=dataset['exang'],y=Y)

<Axes: xlabel='exang', ylabel='target'>

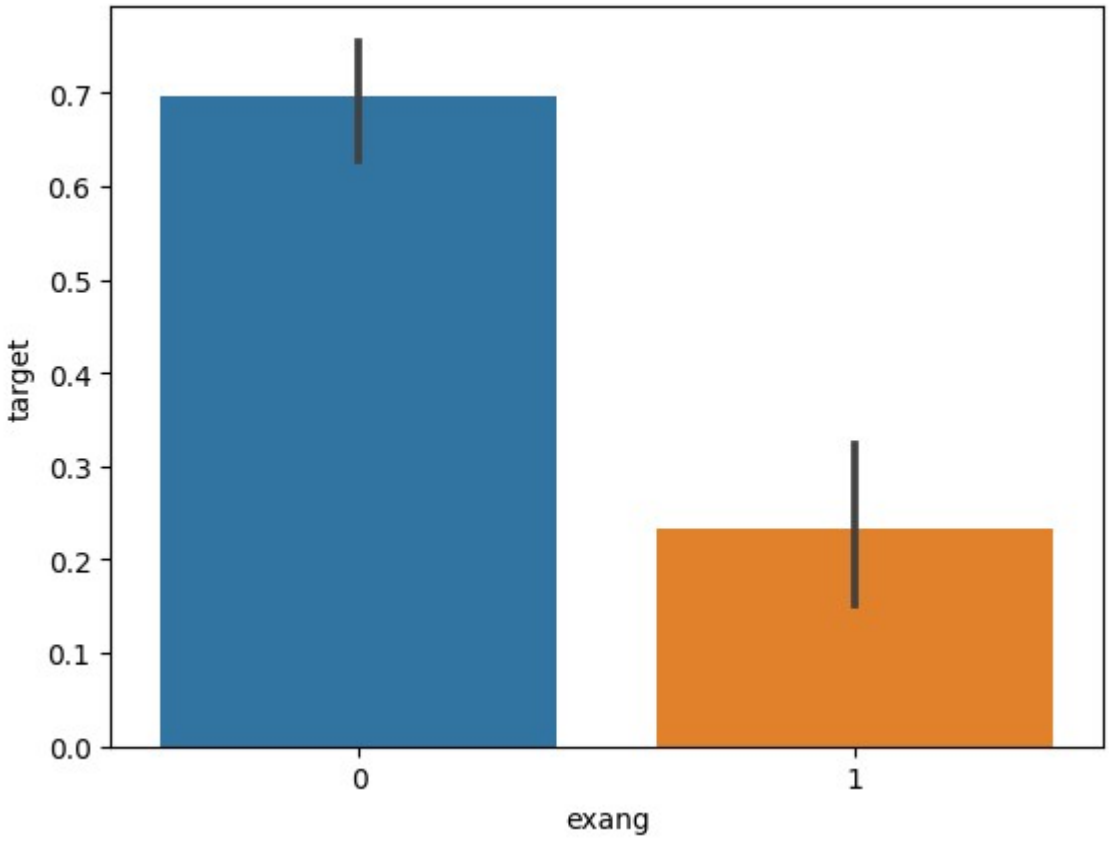


Fig.3.7.Analysing the exang

3.8.Analysing the slope

```
dataset['slope'].unique()
```

```
array([0, 2, 1])
```

```
sns.barplot(x=dataset['slope'],y=Y)
```

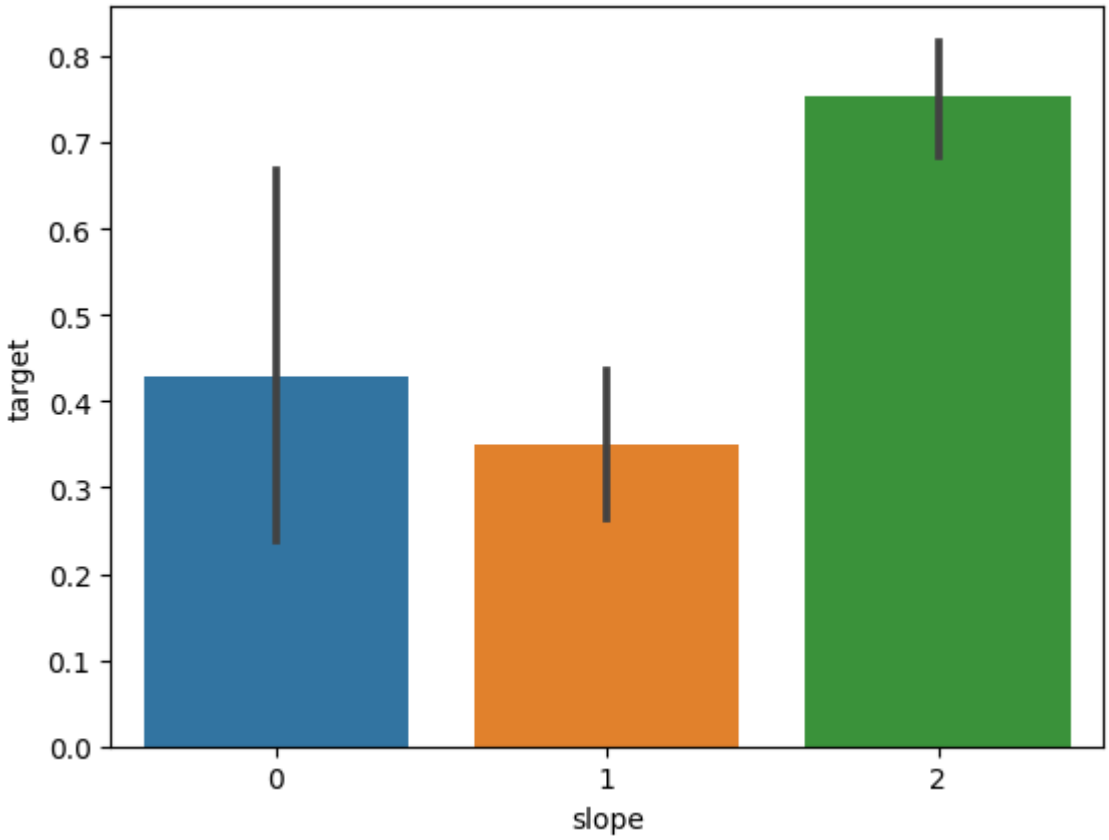


Fig.3.8.Analysing the slope

4.DATA PROCESSING

Data processing is the process of collecting, cleaning, transforming, and analyzing data to extract useful information and insights. It is an essential step in data analysis and machine learning, as it helps to ensure that the data is accurate, consistent, and in a suitable format for analysis.

4.1.Splitting data & Train test model

Splitting data into training and testing sets is an important step in machine learningto evaluate the performance of a model on new, unseen data. The goal of splitting the data is to estimate how well the model will perform on data it has not seen before, and to identify any issues or limitations with the model.

```
>>from sklearn.model_selection import train_test_split

>>predictors = dataset.drop('target',axis = 1)

>>target = dataset['target']

>>X_train, X_test,Y_train, Y_test = train_test_split(predictors, target, test_size=0.20 , random_state = 0)

>>X_train.shape , X_test.shape

((242, 13), (61, 13))

>>Y_train.shape , Y_test.shape

((242,), (61,))
```

5.MODEL FITTING

Model fitting is the process of finding the best set of model parameters that allow a model to best represent the relationship between the input variables and the output variable in a dataset. In machine learning,model fitting is used to train a model on a labeled dataset, allowing it to make predictions on new, unseen data.

5.1.ML Algorithms

Machine learning algorithms are used to automatically learn patterns and relationships in data, and to make predictions or decisions based on that learning. They are used in a wide range of applications, including image and speech recognition,natural language processing,recommendation systems,fraud detection, and medical diagnosis

5.1.1.Logistric Regression

Logistic regression is a statistical model used to predict binary outcomes, such as whether a customer will buy a product or not, whether a patient has a disease or not, or whether an email is spam or not. It is a type of regression analysis that estimates the probability of an event occurring based on one or more input variables.

```
>>from sklearn.linear_model import LogisticRegression

>>from sklearn.pipeline import make_pipeline

>>from sklearn.preprocessing import StandardScaler

>>from sklearn.metrics import accuracy_score

>>pipe = make_pipeline(StandardScaler(),LogisticRegression())

>>pipe.fit(X_train,Y_train)

>>Y_pred_lr = pipe.predict(X_test)

>>score_lr = round(accuracy_score(Y_pred_lr,Y_test)*100,2)
```

```
>>score_lr

>>k = pipe.score(X_test,Y_test)

>>print("The accuracy Score by Logistic Regression is : "+ str(round(k*100,2))+"%")
```

The accuracy Score by Logistic Regression is : 85.25%

5.1.2.Naive Bayes

Naive Bayes is a probabilistic machine learning algorithm used for classification tasks, such as text classification,spam detection, and sentiment analysis. It is based on Bayes' theorem and assumes that the input features are conditionally independent of each other, given the class label.

```
>>from sklearn.naive_bayes import GaussianNB

>>from sklearn.metrics import accuracy_score

>>nb = GaussianNB()

>>nb.fit(X_train,Y_train)

>>Y_pred_nb = nb.predict(X_test)

>>score_nb = round(accuracy_score(Y_pred_nb,Y_test)*100,2)

>>print("The accuracy score by Naive Bayes is :" +str(score_nb)+"%")
```

The accuracy score by Naive Bayes is :85.25%

5.1.3.SVM(Support Vector Machine)

Support Vector Machine (SVM) is a powerful and widely used machine learning algorithm for classification, regression and outlier detection tasks. SVMs are based on the idea of finding the hyperplane that best separates the data into different classes. The hyperplane is chosen in such a way that it maximizes the margin between the two classes, which is the distance between the hyperplane and the closest data points from each class.

```
>>from sklearn import svm

>>sv = svm.SVC(kernel = 'linear')

>>sv.fit(X_train, Y_train)

>>Y_pred_svm = sv.predict(X_test)

>>score_svm = round(accuracy_score(Y_pred_svm,Y_test)*100,2)

>>print("The accuracy score by Support Vector Machine is :"+str(score_svm)+"%")

>>The accuracy score by Support Vector Machine is :81.97%
```

5.1.4.K-Nearest Neighbors

K-Nearest Neighbors (KNN) is a non-parametric machine learning algorithm used for classification and regression tasks. It works by finding the K closest data points in the training set to a given test data point and making a prediction based on the labels or values of those K neighbors.

```
>>from sklearn.neighbors import KNeighborsClassifier

>>knn = KNeighborsClassifier(n_neighbors = 7)

>>knn.fit(X_train,Y_train)

>>Y_pred_knn = knn.predict(X_test)

>>Y_pred_knn.shape

(61,)

>>score_knn = round(accuracy_score(Y_pred_knn,Y_test)*100,2)

>>print("The accuracy score by KNN is : "+str(score_knn)+'%')

The accuracy score by KNN is : 67.21%
```

5.1.5.Decison Tree

Decision Tree is a popular machine learning algorithm for classification and regression tasks. It works by recursively partitioning the data into subsets based on the values of the input features until a stopping criterion is met. The resulting tree-like structure can be used to make predictions on new, unseen data.

```
>>from sklearn.tree import DecisionTreeClassifier

>>max_accuracy = 0

>>for i in range(200):

>>dt = DecisionTreeClassifier(random_state = i)

>>dt.fit(X_train,Y_train)

>>Y_pred_dt = dt.predict(X_test)

>>current_accuracy = round(accuracy_score(Y_pred_dt,Y_test)*100,2)

>>if (current_accuracy > max_accuracy):

>>max_accuracy = current_accuracy

>>best_i = i

>>dt = DecisionTreeClassifier(random_state = best_i)

>>dt.fit(X_train,Y_train)

>>Y_pred_dt = dt.predict(X_test)

>>Y_pred_dt[:]

array([0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0,

0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0,
```

1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1])

>>score_dt = round(accuracy_score(Y_pred_dt,Y_test)*100,2)

>>print("The accuracy score by Decision Tree is : "+ str(score_dt)+"%")

The accuracy score by Decision Tree is : 81.97%

5.1.6.Random Forest Algorithm

Random Forest is a popular machine learning algorithm used for classification, regression, and other tasks. It is an ensemble learning method that combines multiple decision trees to improve the accuracy and robustness of the model.

>>from sklearn.ensemble import RandomForestClassifier

>>max_accuracy = 0

>>for i in range(2000):

>>rf = RandomForestClassifier(random_state=i)

>>rf.fit(X_train,Y_train)

>>Y_pred_rf = rf.predict(X_test)

>>current_accuracy = round(accuracy_score(Y_pred_rf,Y_test)*100,2)

>>if current_accuracy > max_accuracy:

>>max_accuracy = current_accuracy

>>best_i = i

>>rf = RandomForestClassifier(random_state = best_i)

>>rf.fit(X_train,Y_train)

```
>>Y_pred_rf = rf.predict(X_test)

>>Y_pred_rf[:]

array([0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0,

0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0,

1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1])

>>score_rf = round(accuracy_score(Y_pred_rf,Y_test)*100,2)

>>print("The accuracy score by Random Forest is :"+str(score_rf)+"%")

The accuracy score by Random Forest is :90.16%
```

5.1.7.XG Boost

XGBoost (Extreme Gradient Boosting) is a popular machine learning algorithm used for classification, regression, and other tasks. It is an ensemble learning method that combines multiple weak models to create a strong model.

```
>>import xgboost as xgb

>>xgb_model = xgb.XGBClassifier(random_state = 42)

>>xgb_model.fit(X_train,Y_train)

>>Y_pred_xgb = xgb_model.predict(X_test)

>>Y_pred_xgb[:]

array([0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 0,

0, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0,

0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1])
```

```
>>score_xgb = round(accuracy_score(Y_pred_xgb,Y_test),2)*100

>>print("The accuracy score by XGB is:"+str(score_xgb)+"%")

The accuracy score by XGB is:79.0%
```

6.TESTING MODEL

Testing a model is an important step in machine learning to evaluate the performance of the model on new, unseen data. The goal of testing is to estimate how well the model will perform on data it has not seen before, and to identify any issues or limitations with the model.

6.1.Classification Analysis

Classification analysis is a type of machine learning analysis that is used to predict categorical outcomes, such as class labels or categories. It involves building a model that learns to classify new instances based on their features or attributes.

>>scores=[score_lr,score_nb,score_svm,score_knn,score_dt,score_rf,score_xgb]

>>algorithms = ['Logistic Regression','Naive Bayes','Support Vector Machines','K Nearest Neighbors','Decision Tree','Random For

>>for i in range(len(algorithms)):

>>print("The Accuracy score achieved by using " + algorithms[i] +' is : '+ str(scores[i])+ "%")

Logistic Regression is : 85.25%

Naive Bayes is : 85.25%

Support Vector Machines is : 81.97%

K Nearest Neighbors is : 67.21%

Decision Tree is : 81.97%

Random Forest is : 90.16%

XG Boost is : 79.0%

>>sns.set(rc = {'figure.figsize':(14,8)})

>>plt.xlabel("Algorithms")

>>plt.ylabel("Accuracy scores")

>>sns.barplot(x=algorithms,y=scores)

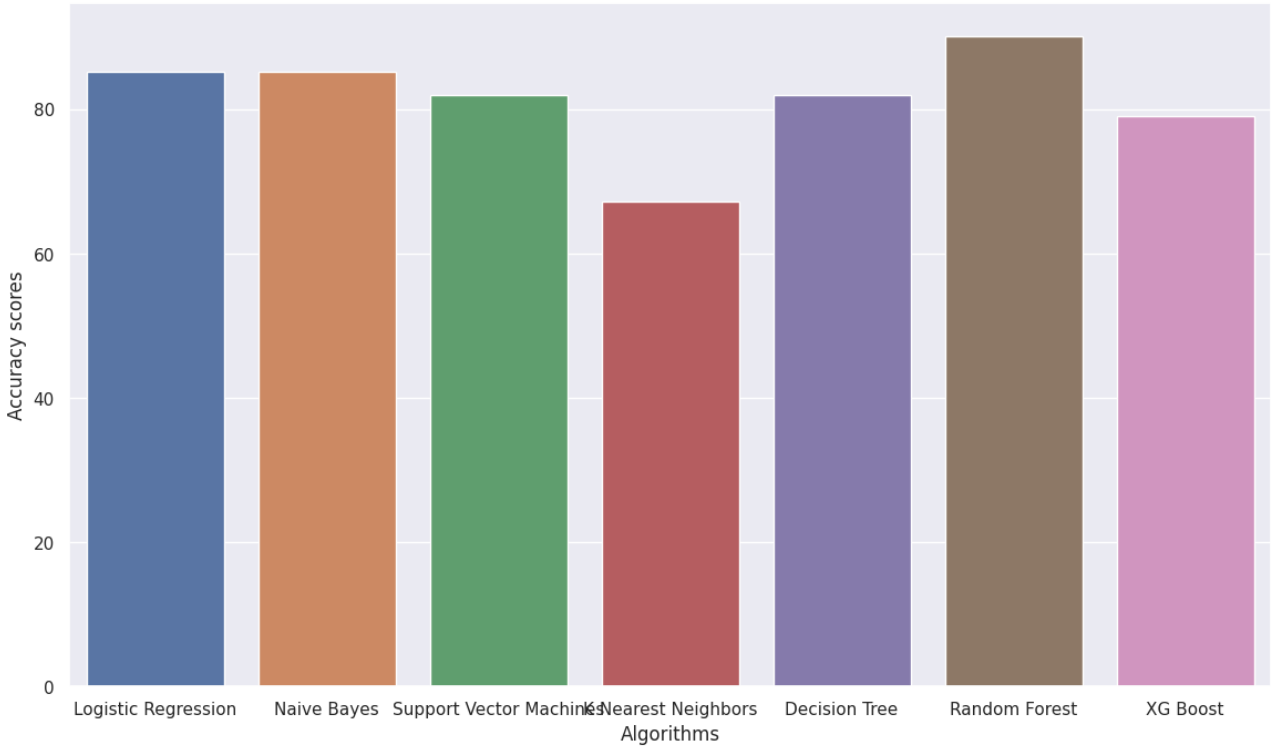


Fig.6.1.Classification Analysis

7.CONCLUSION

The early prognosis of heart disease can aid in making decisions on lifestyle changes in high-risk patients and in turn reduce the complications, which can be a great milestone in the field of medicine. The number of people facing heart diseases is on a raise each year. This prompts for its early diagnosis and treatment. The utilization of suitable technology support in this regard can prove to be highly beneficial to the medical fraternity and patients.The seven different machine learning algorithms used to measure the performance are SVM, Decision Tree, Random Forest, Naïve Bayes, LogisticRegression, K-Nearest Neighbors (KNN), and Extreme Gradient Boosting applied on the dataset to get 90% better accuracy in random forest algorithm.