CARDIOVASCULAR DISEASE PREDICTION

Project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

Saloni Sharma (191003)

Dhruv Batra(191026)

UNDER THE GUIDANCE OF

Dr. Alok Kumar

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY,
WAKNAGHAT

May 2023

TABLE OF CONTENTS

DECLARATION

We hereby declare that the work reported in the B.Tech Project Report entitled **"Cardiovascular Disease Prediction"** submitted at **Jaypee University of Information Technology, Waknaghat, India** is an authentic record of our work carried out under the supervision of **Dr Alok Kumar.** We have not submitted this work elsewhere for any other degree or diploma.

Saloni Sharma
191003

Dhruv Batra
191026

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr Alok Kumar
Date:

Head of the Department/Project Coordinator

ACKNOWLEDGEMENT

Firstly, we express my heartfelt gratitude and gratefulness to almighty God for his divine blessings to make it possible to complete the project work successfully.

We are really grateful and wish our profound indebtedness to our supervisor **Dr. Alok Kumar**, Assistant Professor (SG), department of ECE, Jaypee University of Information Technology, Waknaghat. Deep knowledge and keen interest of our supervisor in the field of Machine Learning to carry out this project . His endless patience , scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, reading many inferior drafts and correcting them at all stages have made it possible to complete this project successfully.

We would also generously welcome each one of those individuals who have helped us straightforwardly or in a roundabout way in making this project a win. In this unique situation, we might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated our undertaking.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

LIST OF ABBREVIATIONS

Abbreviation	Word
CVD	Cardiovascular Disease
ML	Machine Learning
NB	Naive Bayes
CNN	Convolution Neural Networks
SVM	Support Vector Machine
XGBoost	Extreme Gradient
RF	Random Forest
LR	Logistic Regression
MLP	Multilayer Perceptron
GBDT	Gradient Boosting Decision Tree
DT	Decision Tree
KNN-K	Nearest Neighbour
ROC	Receiver Operating Characteristic

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

An illness that negatively impacts the heart and blood vessels is referred to as having a cardiovascular disease. Due to the fact that it is one of the leading sources of mortality worldwide, early prediction is required. Prediction and classification issues are frequently addressed using machine learning. Therefore, we attempted to create a system that can identify cardiovascular disease in its early stages so the person can be informed beforehand, which will aid in an early diagnosis. We had 12 features , which included the age, gender, the type of chest pain, resting blood press. , cholest., maximum heart rate, resting Bp, fasting blood sugar, exercise angina, ST slope and old peak, we have taken a dataset from the IEEE Data Port. We then reduced the features, after finding the correlation between them, and employed six machine learning methods, including SVM, decision tree, random forests, KN-neighbour, XG Boost and multilayer perceptron and also combined certain models together then evaluated the model on basis of it’s accuracy, sensitivity , precision, F1 score , log loss and Mathew’s correlation coefficient . We concluded that the random forest and MLP Model gave the highest accuracy of 91% .

CHAPTER 1

INTRODUCTION

As per the record of the World Health Organization (WHO), 17.9 million demises worldwide in 2019 were attributed to cardiovascular disease, or 32% of all fatalities. [1] Heart disease, is one of the main reasons people die throughout the world. Cardiovascular disease is viewed as one of the most important subjects, in data analysis. The frequency of cardiovascular disease has been increasing at fast pace worldwide, since a few years ago. Numerous researches have been conducted, in this field in an effort to pinpoint the most crucial, heart disease risk factors and accurately calculate the total risk. Early identification of CVD is crucial since heart disease is also known as the "silent killer" because it may kill a person without showing any overt symptoms.

1.1 CARDIOVASCULAR SYSTEM

The cardiovascular system is also called as the, circulatory system, which carries oxygen, waste materials and nutrients, in every part of the body. The heart, blood arteries, and blood make up this system. A brief description of each element is given below:

- The Heart-The muscular organ that pumps blood, all over the body is the heart. It is the size of a closed fist and is situated in the chest, a little bit towards the left. It is subdivided into the four chambers that are, right atrium, the right ventricle, left atrium, and , the left ventricle..
- Blood vessels: Blood circulates throughout the body through blood vessels, which are known as the tubes. They comprise of the capillaries, the arteries and the veins. The veins returns the deoxygenated blood into the heart, while, the arteries transport oxygenated blood away from the heart. The minute blood vessels, called the capillaries, which are, in charge of transferring oxygen, waste materials and nutrients from the blood to the tissues of the body.
- Blood: The WBC, the platelets, RBC and the plasma make up the fluid which is known as the blood.WBC are part of the immune sys. and it helps to resist against the infections, whereas the red blood cells, are in charge of

- carrying the oxygen, whereas, the plasma is a fluid that travels all over the body and transfers the nutrients, hormones, and waste products, and platelets are in charge of the blood clotting.

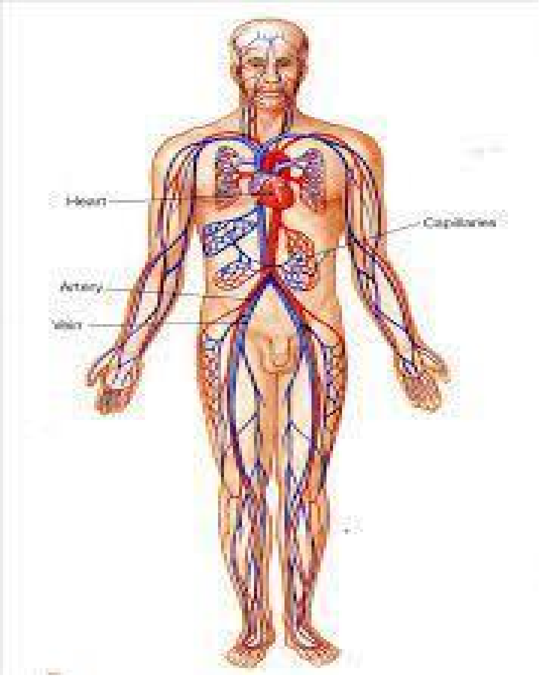


Figure 1.1: Circulatory System

1.2 STUDY

This study aims to forecast the future cases of heart disease by examining person data that applies a ML algorithm to categories whether a person has heart disease or not. Machine learning aids in the prediction of data from datasets supplied by the healthcare industry. In this respect, ML techniques are of significant support. Inspite the fact that heart disease can show itself in a numerous ways, there is a usual set of basic possible factors that decide whether or not someone would eventually be at risk for this condition. By obtaining data from diverse sources, categorizing them, and using different ML algorithms on them, we can say that this strategy may be extremely well suited to produce the correct prediction.

There are 11 characteristics and a target variable in this dataset. It contains 5 numerical and 6 nominal variables. Each feature is fully described as follows:

This dataset has a target variable and 11 characteristic. There are 6 nominal and 5 numerical variables in it, as given below:

1. Age: It shows the person's age in years, it is numeric.

2. Gender: It tells the gender of the person (M (male) as 1, F(female) as 0),it is nominal.

3. Chest Pain Type: It tells us about the type of the chest pain felt by person sorted into:

1 is typical 2 is typical angina, 3 is non-angina pain, 4 is asymptomatic ,it is Nominal.

4. Resting BP: Level of the bp when the person is at rest, it is measured in mm/HG , it is number.

5. Cholesterol: It is measured in mg/dl , it is numeric.

6. Fasting blood sugar: It is the blood sugar levels on fasting that is greater than 120 mg/dl that is given as 1 in case of a true value and 0 in case of false.

7. Resting ECG: It is presented in 3 values: 0 as Normal, 1 as Abnormality in ST-T wave, 2 as left ventricular hypertrophy, it is nominal

8. Mx heart rate: It is the max heart rate that is achieved, it is num.

9. Exercise angina: It is angina induced by doing exercise, 0: NO, 1:Yes , it is nominal

10. Old peak : It is exercise induced ST-depression in contrast with the state of rest, it is numeric

11. ST slope: ST segment determined in terms of slope during peak exercise:0: Normal 1: Up sloping 2: Flat 3: Down sloping , it is nominal

12. Target: It is the target variable, if it predicts 1, it means that the person is suffering from heart risk and 0 means that the person is normal.[3]

1.3 PROBLEM STATEMENT

Finding a cure for heart disease is the main challenge. Although there are technologies that can predict the disease of the heart, they either cost a lot of money or are inefficient at predicting the heart disease in a human. Early diagnosis and sensing of heart diseases can decrease the death rate and overall implications. It is not always possible for to accurately monitor the patients every day and confer with the doctor for 24 hours because it requires more intelligence, time, and knowledge.

With the availability of modern data, we may use different types of machine learning algorithms to look for patterns that are hidden. To identify health problems, hidden patterns in medical data may be used.

1.4 MOTIVATION

The aim of this project is to suggest a methodology for diagnosing CVD in patients in its early phases. The objective of this project is to determine the most effectual classification scheme for diagnosing heart disease in patients. As we know that ,we can efficiently detect the factors of the cardiovascular disease after getting ourselves examined at the hospital, but after that , for a person it is inconvenient to check every value and range of the healthcare report to see if they are more prone to heart disease or not, so we developed a model that will assist in checking and predicting if the person is more prone to heart disease or not , they just have to put every value of the healthcare report and it will predict the chances of the person of having CVD .

CHAPTER 2

LITERATURE SURVEY

In [1]J.Liu, X.Dong , H.ZhaoandY.Tianet al. ,proposed a paper in which they employed RF, linear regression, MLP, GBDT,CatBoost and LightGBM, which provided accuracy 90%. The degree of correlation, between the implemented models was the next thing they looked at. They used LR as the meta learner and LightGBM and ET as the base learners after receiving the results. Their model has a 91.22% accuracy rate. In [2] J.Emakhu and S.Shrestha et al., proposed a paper in which discretization was applied during the preparation stage to classify the data from a medical database using ensemble classifiers. The University of California provided the data that was used in the article (UCI). They assessed the model's accuracy, sensitivity, and specificity performance. They were 87.04% accurate overall. In [3] Katarya, R., Meena, S.K. et al, proposed a paper that covered the various category of heart disease that a person may have and its symptoms. They examined a number of machine learning techniques on the dataset from UCI and found that random forest outperformed the others. In [4] A. Agarwal, A. Rajdhan, D. Ravi, Dr. P.Ghuli, A. Rajdhan,etal,proposed a paper that utilized several data mining approaches, including NB, Decision Tree, LR, and RF, in their proposed study, patient risk levels are categorized, and the possibility of acquiring heart disease is projected. The effectiveness of several machine learning techniques is therefore compared in this study. The trial results show that, when analyzed to other ML algorithms used, the Random Forest approach has the best accuracy (90.16%). In [5] Goel, Rati,et al.,proposed a work ,where they predicted heart disease by using different features in the given data set. They combined a data set consistiting of 13 features and 383 single values to analyze the patients performance and used NB,DT,RF,KNN,LR, SVM, they achieved highest accuracy of 86% using SVM algorithm. In [6] G. Deepthi ,C.Shivani, K.Nagavinith ,K.Hanudeep ,et al. ,proposed a work where they compared various classifiers such as the decision tree, the NB, the LR, the SVM, and the Random Forest, and they proposed an ensemble classifier that performed hybrid categorized by taking both strong and robust classifiers, and they then proposed a classifier such as the Ada-boost and XG-boost that provided an accuracy of 81%.In [7] Khamparia,M. Shabaz ,S. Pande,andP.Singh,R.Bharti,et al ,proposed a work in which they used a dataset with 14 key features ,isolation Forest was used to manage the dataset's irrelevant features, and data was additionally standardized to produce better results. This paper also discusses how the learning can be coupled with some multimedia technology, such as mobile. Their deep learning method produced an accuracy of 94.2%. In [8] S.Singhal , H. Kumar , V.Passricha ,et al. ,proposed a work ,in which they used 13 features from the Cleveland dataset as input in the research to propose a novel CNN-based heart disease prediction model. With a modified back propagation approach, the suggested model is trained. By predicting both the absence and presence of heart disease, their model is said to provide results that are 95% accurate.In[9] A. Tiwari, A.Chugh, A.Sharma ,et al ,proposed a work ,in which they combined data from Hungary, Cleveland, Long Beach VA, Switzerland, and Statlog, the dataset included important variables such chest pain ,serum cholesterol, fasting blood sugar, and more. The model is evaluated using ROC, AUC curve, specificity, F1-score, sensitivity, MCC, and accuracy. The study suggests a system using a combined ensemble classifier that makes use of the ExtraTrees Classifier, RF, XGBoost, and other machine learning methods. Their model's accuracy rate was 92.34%.In[10]P. Anbuselvan Et al. , proposed a work, in which they investigated the models of Logistic Regression, NB, Support Vector Machine, K-Nearest Neighbors, and DT, that are part of supervised learning. The ensemble approach of XGBoost, the Random Forest technique, and Tree were all used to conduct a comparative examination of the most effective algorithm. When differentiated to other algorithms, random forest is shown to have the best accuracy (86.89%).

CHAPTER 3

PROPOSED SYSTEM MODEL

3.1 METHODOLOGY

- We used IEEE dataset, in which 11 features were preselected.
- We imported the libraries for data wrangling , preprocessing , data visualization, model validation, and for machine learning algorithms.
- Then we loaded the dataset.
- Then data cleaning and preprocessing was done.
- Then the distribution of the CVD was done to check the heart patients and normal patients.
- After that distribution of rest ECG, chest pain type, age, ST slope and age was done of heart patients and normal patients.
- Outliers were detected and removed after the above step.
- Then we checked the correlation with diabetes.
- Then we reduced the features again to 5 features.
- Then we applied the SMOTE technique to balance out the dataset.
- Then training and testing was done.
- Then we cross validated the model that performed well and filtered the top performing models.
- Then model building was done using SVM , KNN , decision tree, random forest , MLP and XGB and ensemble modeling.

- Then we evaluated our model on based upon it's acc., sensi., prec, F1 score, ROC, log loss and Mathew correlation coefficient.

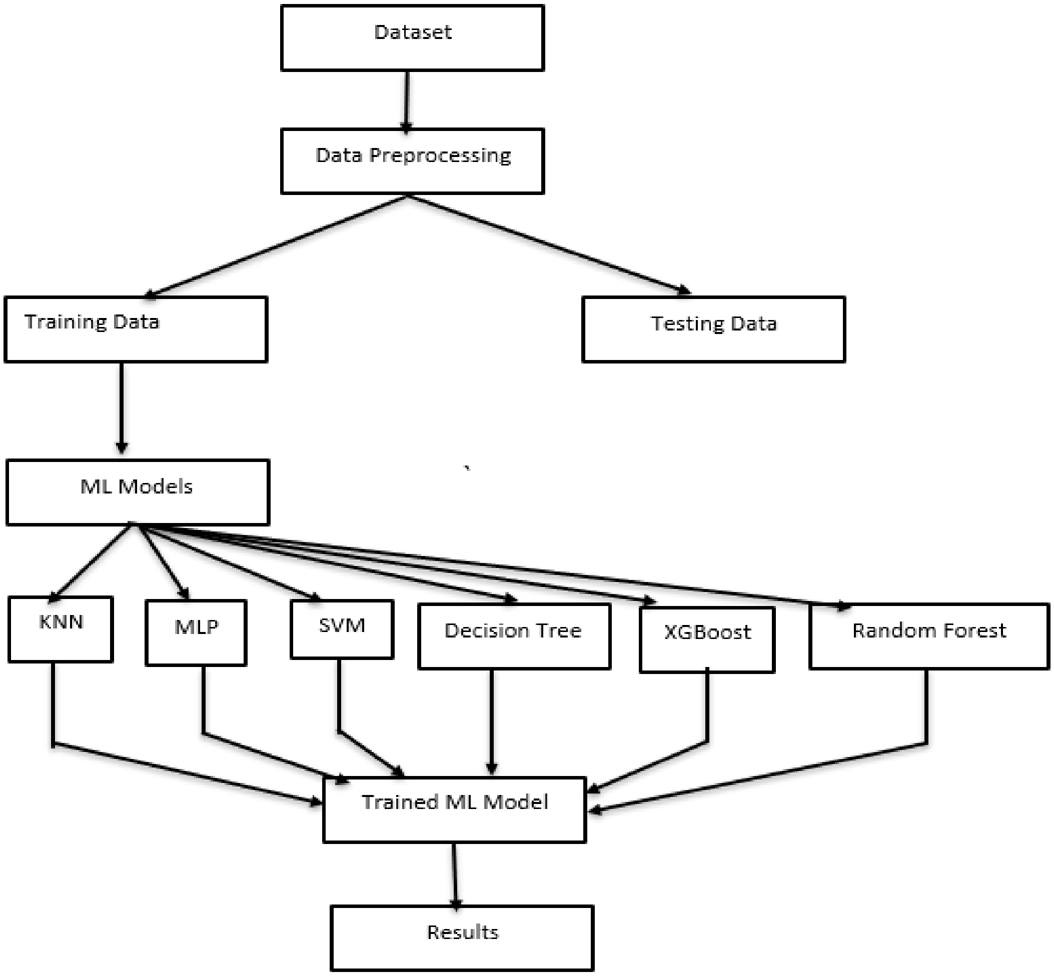


Figure 3.1: System Architecture

3.2 MACHINE LEARNING TECHNIQUES

A classification challenge in machine learning occurs when a label of a class is anticipated for a specific example of the given input data.

3.2.1 Supervised learning

The way of teaching computers using "labeled" trained data and then applying data to forecast results is referred to as supervised learning. Data that has so far been allocated the right o/p is referred to as "labeled data." In supervised learning, machines receive data for training that then behave as the supervisor, coaching them on how to correctly predict the outcome. It uses the same principle that a student might discover when being instructed by a teacher. The process of supervised learning entails providing the appropriate data for the input and output to the machine learning model. An algorithm for supervised learning seeks to identify a mapping func. that will connect the i/p variable (x) with the o/p variable (y).[5]

3.2.2 Unsupervised learning

Unsupervised learning, in contrast to supervised learning, cannot be utilized to directly address a regression or classification issue since we lack the associated output data. The goals of unsupervised learning include discovering a dataset's underlying structure, grouping data based on similarities, and representing the dataset in a compressed format. It aids in extracting insightful information from the data.[5]

3.2.3 Reinforcement learning

In contrary to supervised learning, reinforcement learning depends on the reinforcement agent to determine how to accomplish the job at hand. An answer key is part of the training data in supervised learning. Since the answer is already known, the model is trained using it in supervised learning. It is necessary for it to gain knowledge from its experience in the absence of a training dataset.[5]

3.3 ALGORITHMS USED

3.3.1 Random Forest

With random forest, problems with classification and regression can both be resolved. It uses supervised machine learning to its advantage. A group of decision trees known as a random forest uses numerous decision trees for training and prediction. Every sample that is taken from the given dataset is used in recurrent sampling, and the decision tree is made for every sample. Predictions are built on the intersection of all the selected decision trees. It is known that adding more trees will produce a better and more precise result. It is employed in both classification and regression, as previously indicated. In classification, the data class is decided using a voting system, but in prediction , the mean of all the data is derived. [3]

3.3.2 Decision Tree

It is a supervised technique that is employed to regress & explain given data. The fundamental layout of the decision tree is depicted in Figure 3.2. A decision tree is used to distribute data in a structure that is tree like; after calculating the entropy of each characteristic, the root is divided into sub-trees or the different branches that are built on the maximal info gain and a few control. The attributes are used in this method's recursive execution, and the leaf node delivers the outcome.

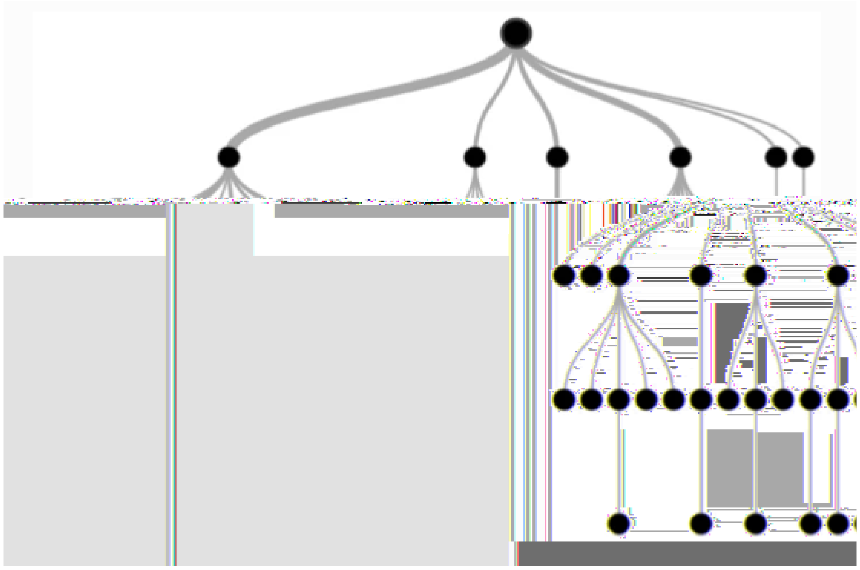


Figure 3.2: Decision Tree Structure

$$Entropy(X) =- \sum_{i=1}^{n} p(x_i) log_b p(x_i) \text{ -----------------(3.1)}$$

In Eqn. (4), the class entropy is computed for the given data set X, the set of classes n, and the correlation of numbers of items in class ‘n’ to the total number of components in the set’ X’.

(IG)Info Gain = (CE) class entro. –(EC)entro characteristics --------(3.2)

The class entropy in the information gain equation is removed from each branch's entropy (3.2). Decision trees are useful for absolute and data that is continuous, like yes or no and 0 or 1. Because every attribute is taken into account and assessed using a tree-like structure, it offers outstanding accuracy. Models are simple to follow and rules are easily developed. However, this method could be over-fit if a tree has several branches. [3]

3.3.3 K-Nearest Neighbor(KNN)

Classification and regression issues can be solved using the supervised machine learning method K-nearest Neighbor (KNN). When a data point's target value is missing in a KNN model, the nearest k points of data are found in the training set, and their average value is used in its place. While the mode of the k labels are allocated in category, the mean of the k labels is retrieved in regression. When prior understanding of the data is lacking, categorization is performed using this basic algorithm. The closest data points can be located using distance metrics like the Manhattan distance or the Euclidean distance. Even with noisy huge data sets and erroneous forecasts, it can produce superior results.[3]

3.3.4 Support Vector Machine

SVM comes under the supervised learning technique (labeled data is used). The SVM builds a subspace that is as broad as is practical to distinguish different categories of data or to retain same data of one sort on one side and alike data of the other type on the other side of the margin. SVM can be used for prediction and classification. Based on the likelihood of changing cardiac disease on one side of a margin versus none on the other, heart prediction can be divided into disease classes.

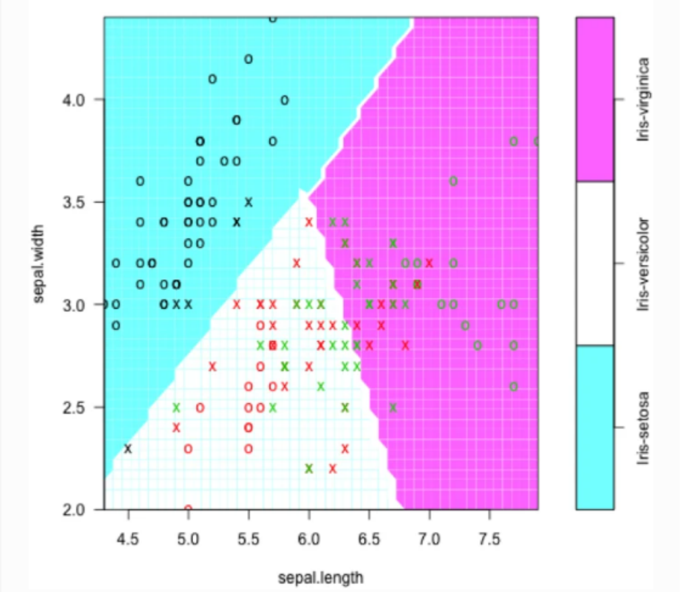


Figure 3.3: SVM classification plot

The two types of SVM are linear and nonlinear, respectively. In contrast to linear SVM, which can split the data into different groups using a line, non-linear SVM is created when a line cannot be used to separate the data into various groups. When complex data makes it hard to separate the data using linear SVM, non-linear SVM is utilized. A kernel function can be used to translate various classes of data into the high dimensions, where they can subsequently be converted into linear separable classes.[3] A definition of the kernel function equation is:

3.3.5 Multi-Layer Perceptron (MLP)

Instead of using only one hidden layer, an ANN known as a multi-layer perceptron employs multiple hidden layers with back propagation (consisting of three and more layers as well as the i/p and o/p layer). MLP is a network (there is no cycle set up in linking of the connections).It functions by having input from other perceptrons, assigning weights to each node, and transferring that data to the buried layer. Also fed from secret is the output layer. The expected value and actual output are utilized to build the forecast value, which is calculated either during i/p processing or by back propagation. Back propagation between the hidden layer and the o/p t layer is carried out until the error value is reduced, with the intention being to reduce the error value.[3]

3.3.6 XGBoost

It is the gradient boosted decision tree and provides side to side tree boosting. In this method, decision trees are constructed back to back. In XGBoost, weights play an important role. All the variables that are independent will receive weights, after that they are then taken into a decision tree that will predict the outcomes. The second decision tree is allotted the variables that the first tree will predict incorrectly by ascending the weight of those variables. The ensemble of these individual classifiers and predictors produces a strong, better and precise model. Regression, classification, ranking, and user-defined prediction problems are all possible applications.[9]

3.4 Ensemble Learning

Ensemble methods in machine learning merge the knowledge gathered from various learning models to enable more precise and enhanced decision-making.

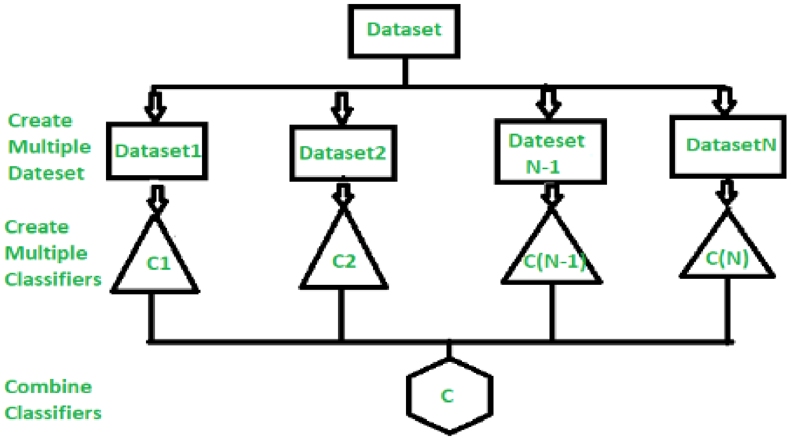


Figure 3.4: Ensemble learning Technique

Ensemble learning methods can be broadly classified into three categories namely bagging, stacking, and boosting. It is essential to possess comprehensive knowledge of all three methods.

The following methods are explained below:

- Bagging is a technique in machine learning that requires creating multiple decision trees on various subsets of the same dataset. The predictions made by these trees are then averaged to produce an overall prediction.
- Stacking, on the other hand, involves training multiple models on the same dataset using different algorithms and then using distinct models to learn the optimal way to merge the predictions.
- Boosting is a method that includes adding ensemble members continually to correct the errors made by the previous models and outputs a weighted average of the predictions.

CHAPTER 4

Feature Selection

4.1 Feature Selection Methods

Feature selection is the method of lowering the quantity, of variables used as the input ,given ,while creating a predictive model. In some of the situations, decreasing the no . of i/p variables perhaps enhance the efficiency of the model and , also reducing the computing cost of modeling, whereas in the statistical based feature selection methods, involves choosing the input variables having the strongest relationships to the target variable ,after statistically analyzing the relationships b/w each i/p variable and the o/p variable.

- There are mainly two sort of feature selection techniques ,that are, super. and unsuperver., furthermore ,the supervised procedure may be subdivided into wrapper, filter and intrinsic.
- The feature selection methods, that are filter based, use statistical methods to rate the dependence or correlation among the input variables that may be filtered to choose the most pertinent characteristics.
- Statistical measures for the feature selection should be properly chosen based on the type of data of the variable as input and the variable as output or response.

4.2 Feature Selection Methods

Feature selection methods, are used to basically decrease the num. of variables used as input to the one that are ought to be very convenient to the model in order to forecast, the target variable more accurately.

Feature selection is essentially concentrated on eliminating the non-informative or redundant predictors from the model.

Some predictive modeling issues have a huge num. of variables that can slow the happening and training of the models and need a great amount of system memory. In addition to it, the performance of certain models can deteriorate when including the input variables that are not important to the target variable.

Feature Selection: Choose a portion of the dataset’s input characteristics.

- Unsupervised- It doesn't use the variable that is the target, for e.g. ,clear away the unnecessary variables like correlation
- Supervised- It uses the variable that is the target, for e.g., remove the irrelevant variables.
- Wrapper- It looks for the features subsets that performs well like RFE.
- Filter- Choose feature subsets depending on how they relate to the target like statistical methods and feature relevant methods.
- Intrinsic- Automatic feature selection during training algo, like ,decision trees
- Dimensionality Reduction-A lower dimensional feature space will be projected from the input data.

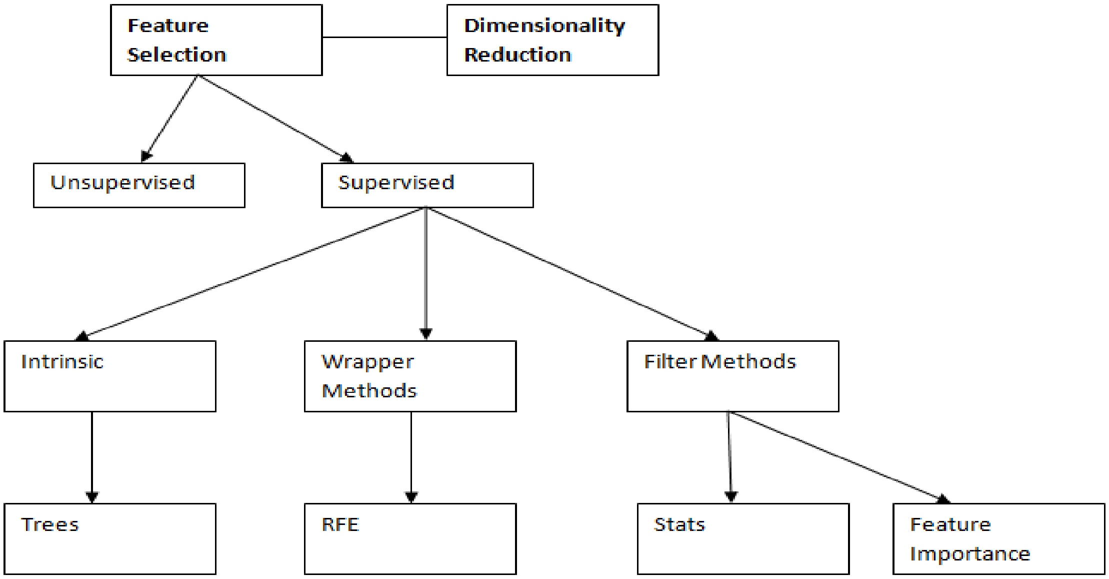


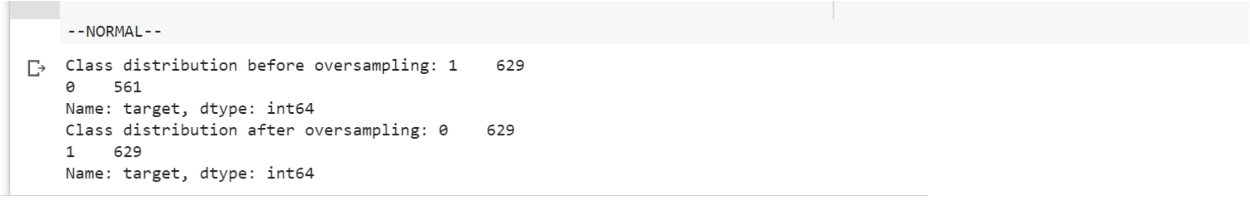
Figure 4.1: Feature Selection Technique

4.3 Statistics for Filter-Based Feature Selection Methods

When selecting filter features, the statistical measures used are typically based on the correlation between input and output variables. The choice of analytic measures is largely determined by the types of data being analyzed. The most common data types include numerical (e.g. height) and categorical (e.g. labels), which can be further categorized as integer or floating point for numerical variables, and boolean, ordinal, or nominal for categorical variables.Some common data types for input variables include numerical variables, which consist of integer and floating point variables, and categorical variables, which include boolean, ordinal, and nominal variables.

4.4 SMOTE

SMOTE is a technique which includes the oversampling, in which the synthetic samples are created for the class which are in the minority. The problem of overfitting that arises due to random oversampling can be controlled by this algorithm. It basically, focuses on the space between the features to create new instances with aid of the interpolation in middle of positive instances.



```
--NORMAL--
Class distribution before oversampling: 1    629
0    561
Name: target, dtype: int64
Class distribution after oversampling: 0    629
1    629
Name: target, dtype: int64
```

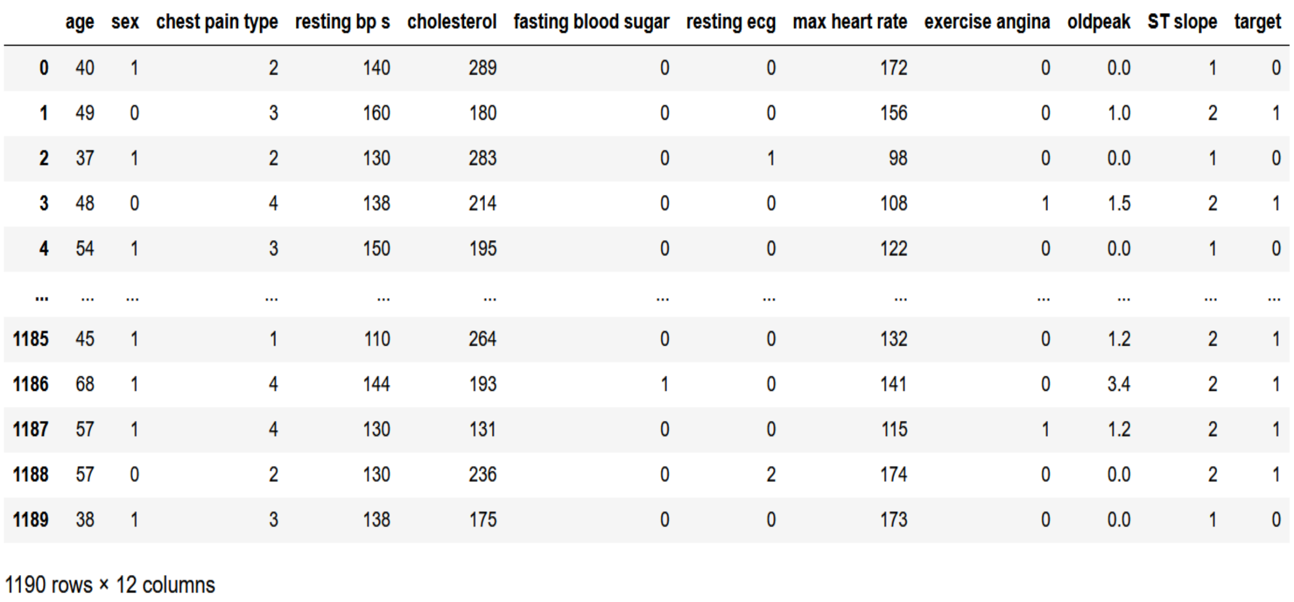
Figure 4.2: SMOTE Applied on reduced dataset

CHAPTER 5

EXPERIMENTAL ANALYSIS

5.1 Results

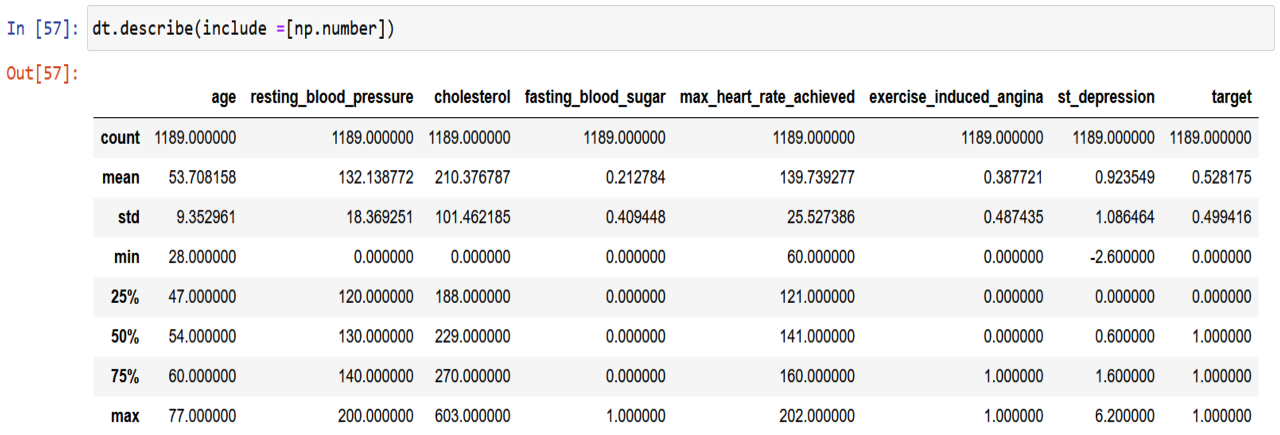
We first loaded the dataset , that had 1190 rows and 12 columns.



	age	sex	chest pain type	resting bp s	cholesterol	fasting blood sugar	resting ecg	max heart rate	exercise angina	oldpeak	ST slope	target
0	40	1	2	140	289	0	0	172	0	0.0	1	0
1	49	0	3	160	180	0	0	156	0	1.0	2	1
2	37	1	2	130	283	0	1	98	0	0.0	1	0
3	48	0	4	138	214	0	0	108	1	1.5	2	1
4	54	1	3	150	195	0	0	122	0	0.0	1	0
...	...	...	...	...	...	...	...	...	...	...	...	...
1185	45	1	1	110	264	0	0	132	0	1.2	2	1
1186	68	1	4	144	193	1	0	141	0	3.4	2	1
1187	57	1	4	130	131	0	0	115	1	1.2	2	1
1188	57	0	2	130	236	0	2	174	0	0.0	2	1
1189	38	1	3	138	175	0	0	173	0	0.0	1	0

1190 rows × 12 columns

Table 5.1: List of loaded dataset



```
In [57]: dt.describe(include =[np.number])
Out[57]:
```

	age	resting_blood_pressure	cholesterol	fasting_blood_sugar	max_heart_rate_achieved	exercise_induced_angina	st_depression	target
count	1189.000000	1189.000000	1189.000000	1189.000000	1189.000000	1189.000000	1189.000000	1189.000000
mean	53.708158	132.138772	210.376787	0.212784	139.739277	0.387721	0.923549	0.528175
std	9.352961	18.369251	101.462185	0.409448	25.527386	0.487435	1.086464	0.499416
min	28.000000	0.000000	0.000000	0.000000	60.000000	0.000000	-2.600000	0.000000
25%	47.000000	120.000000	188.000000	0.000000	121.000000	0.000000	0.000000	0.000000
50%	54.000000	130.000000	229.000000	0.000000	141.000000	0.000000	0.600000	1.000000
75%	60.000000	140.000000	270.000000	0.000000	160.000000	1.000000	1.600000	1.000000
max	77.000000	200.000000	603.000000	1.000000	202.000000	1.000000	6.200000	1.000000

Table 5.2: Exploratory Data Analysis

The below dataset has balanced the 561 patients that are normal and 628 patients with the heart disease.

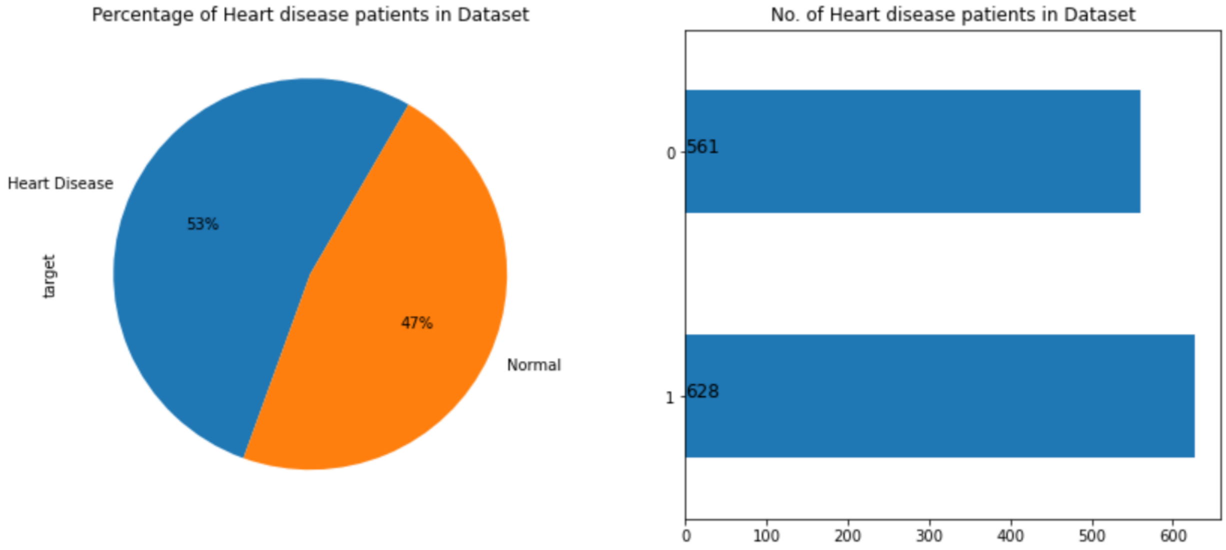


Figure 5.3: Distribution of Heart Disease

From the below plot we observe that the percentage of male patients is higher than that of the female patients and average age of having risk of heart disease is around 55 years.

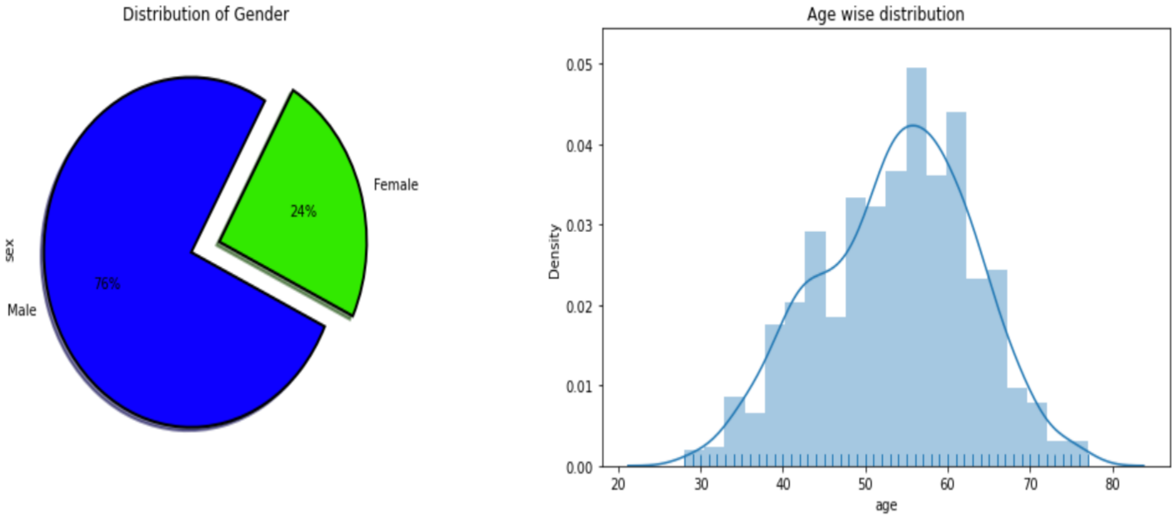


Figure 5.4: Gender and age wise Distribution

From the below four plots we observe that more male patients are likely to having heart disease than the female patients and mean age , is around the age of 58 to 60 years for the heart patients

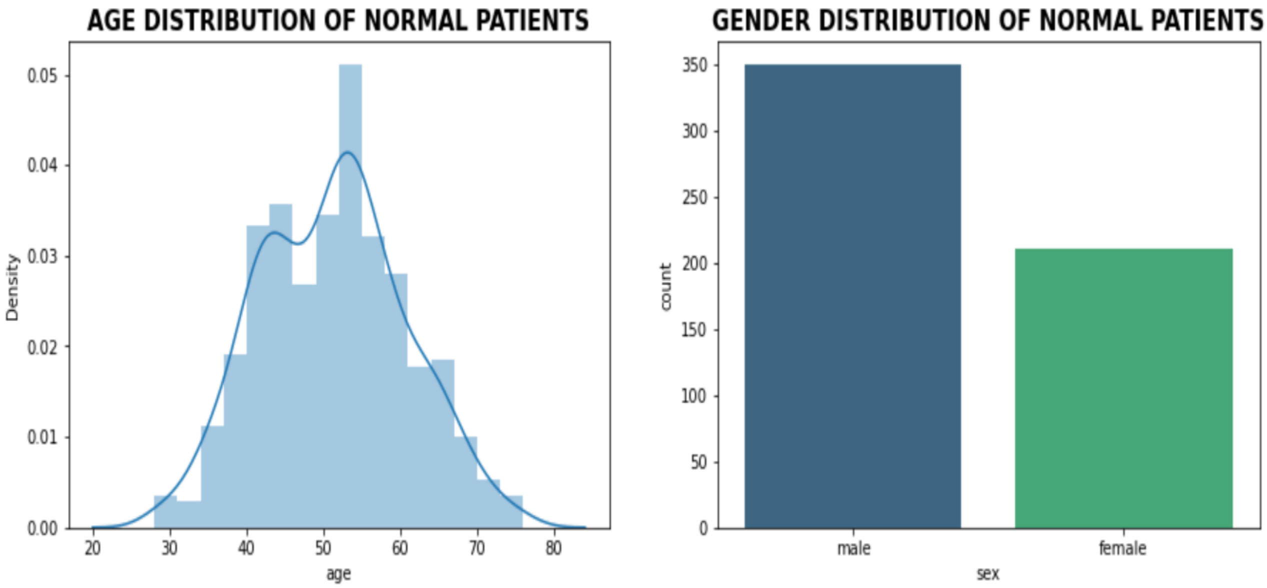


Figure 5.5: Age and gender wise distribution of normal patients

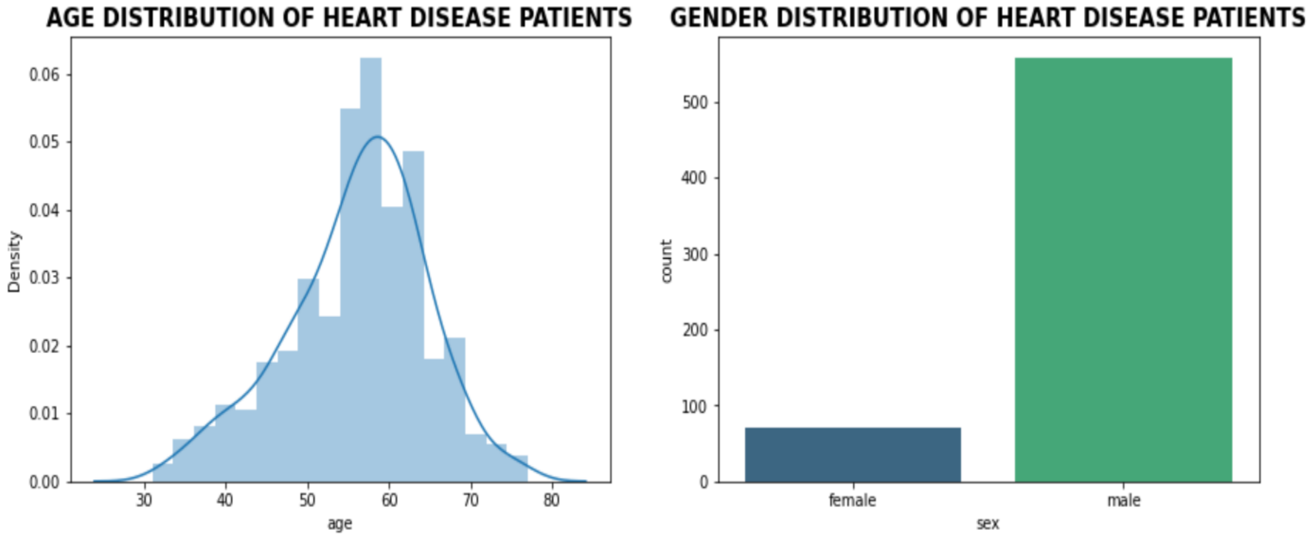


Figure 5.6: Age and gender wise distribution of Heart Disease Patients

Below plots tells us about the distribution of chest pain type, rest ECG and ST slope of normal patients and the heart patients

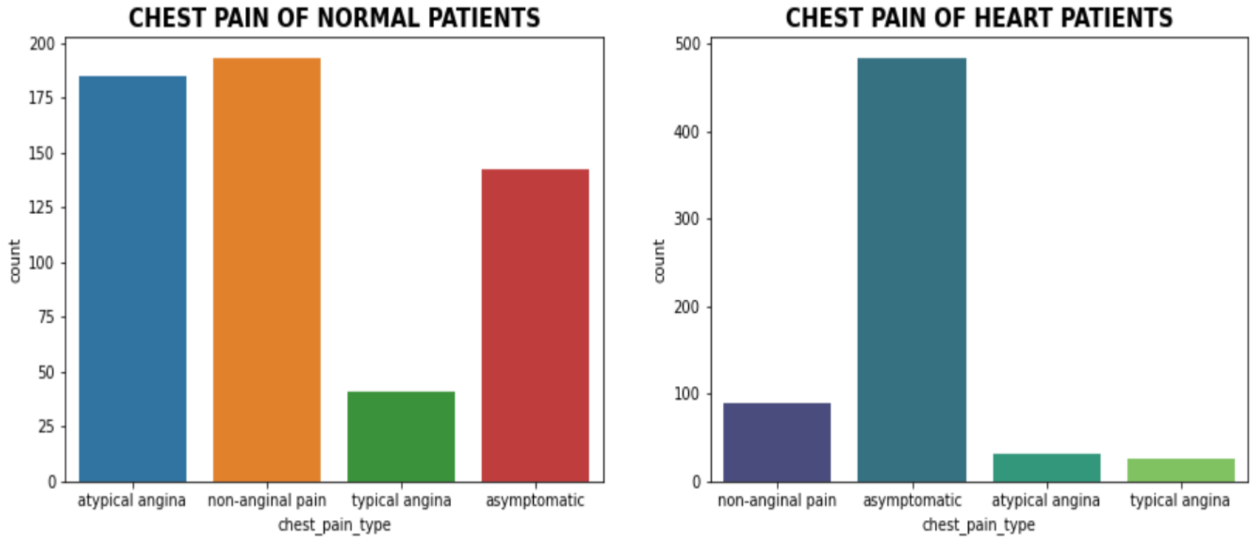


Figure 5.7: Distribution of Chest Pain Type

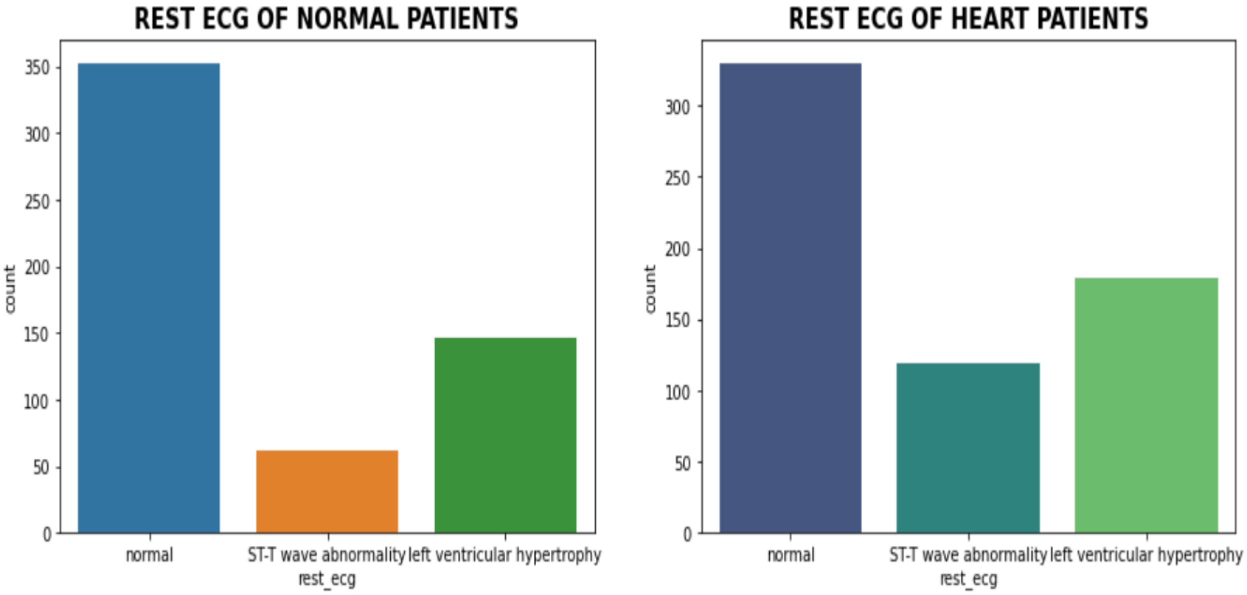


Figure 5.8: Distribution of Rest ECG

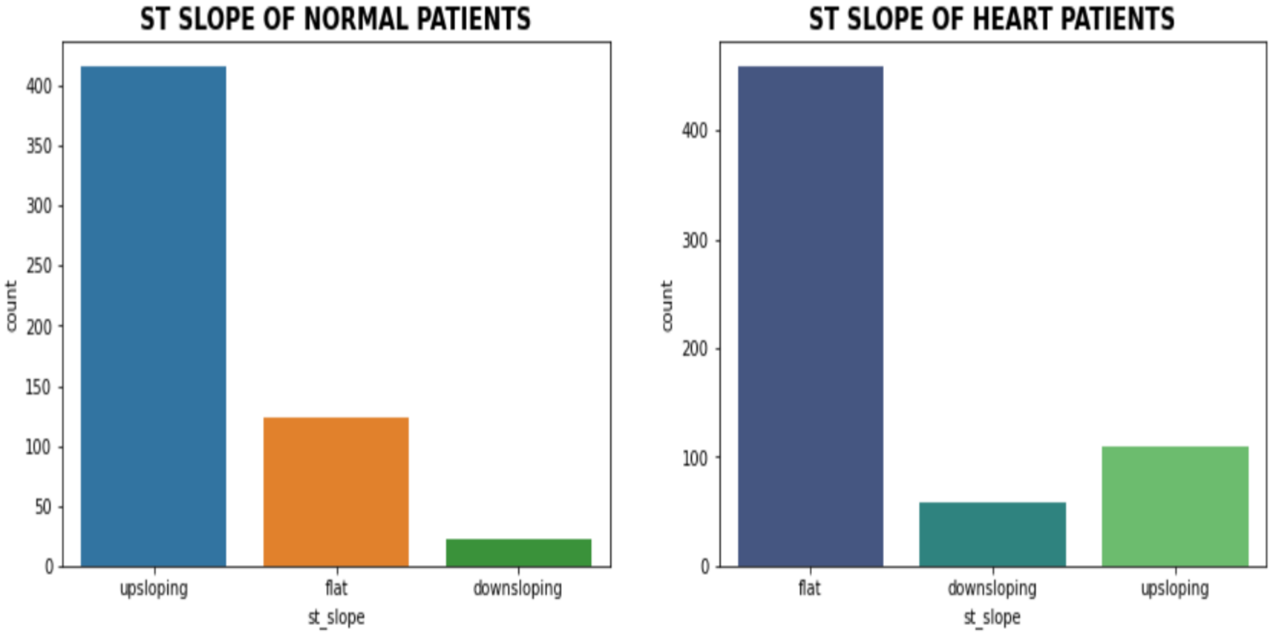


Figure 5.9: Distribution of ST Slope

We are checking the correlation with the diabetes, since, if a person has high blood glucose from diabetes, it can harm their blood vessels and the nerves that is controlling the heart and the blood vessels, which can eventually lead to heart disease.

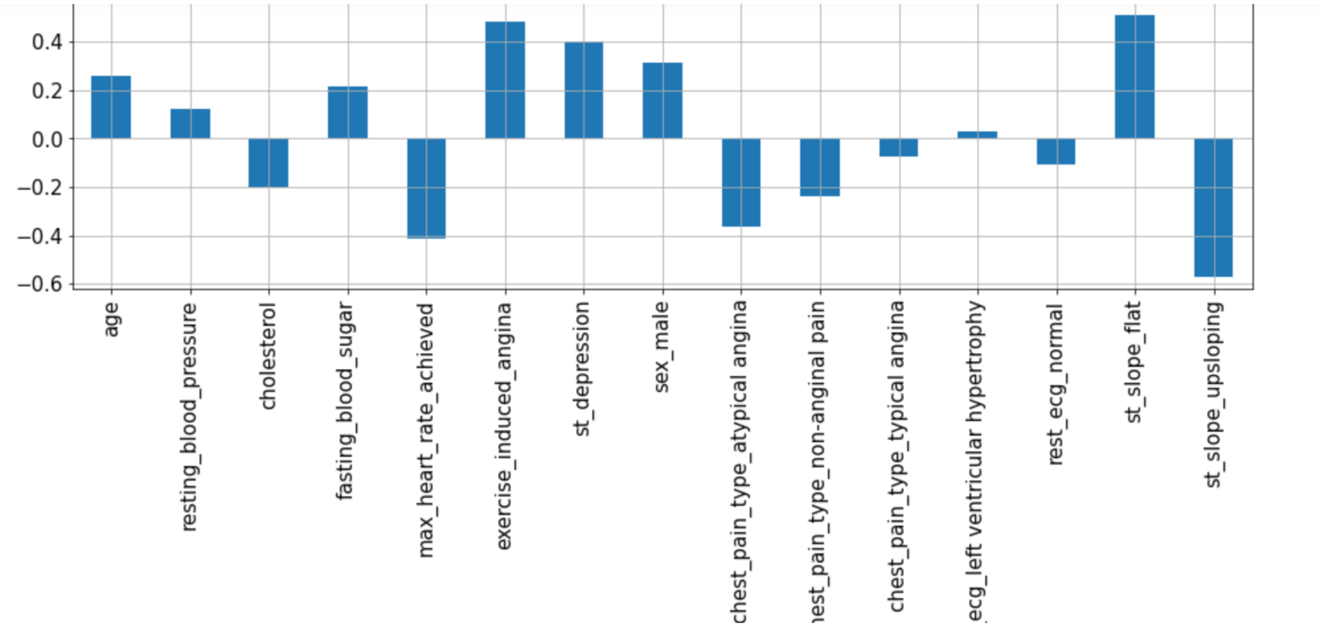
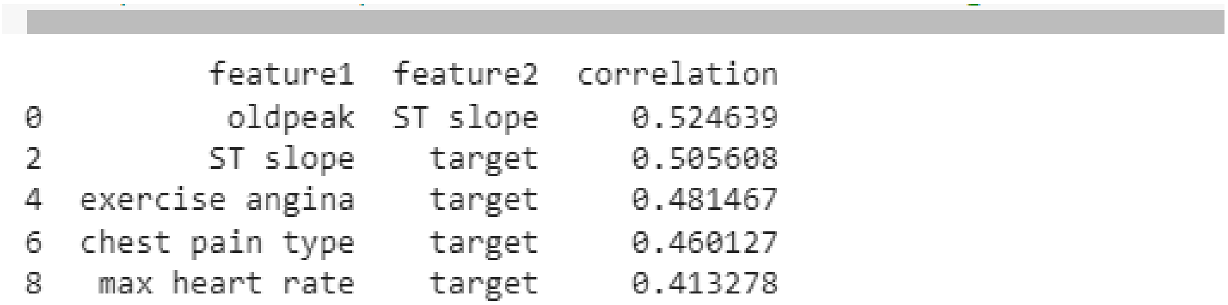


Figure 5.10: Correlation with diabetes

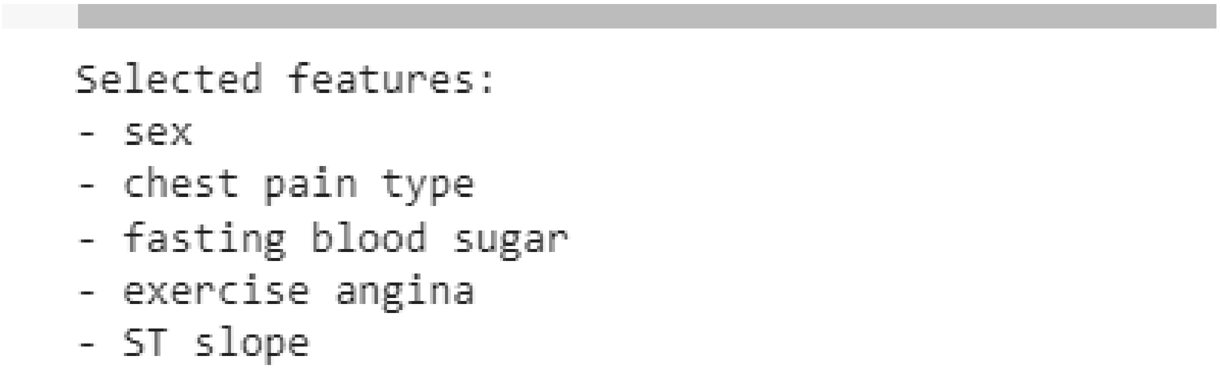
We computed the correlation matrix, and , then found out the most correlated features and then printed the top 5 most correlated features with their correlation coefficients.



```
          feature1  feature2  correlation
0          oldpeak  ST slope     0.524639
2         ST slope    target     0.505608
4  exercise angina    target     0.481467
6  chest pain type    target     0.460127
8   max heart rate    target     0.413278
```

Figure 5.11: Correlation between features

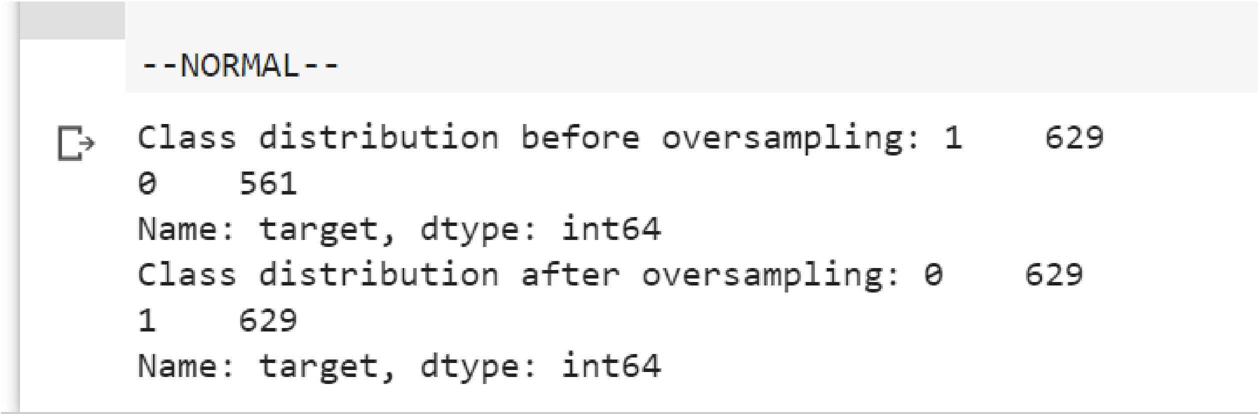
We the created a logistic regression object, and, RFE object with the parameter set to 5 and furthermore reduced the features.



```
Selected features:
- sex
- chest pain type
- fasting blood sugar
- exercise angina
- ST slope
```

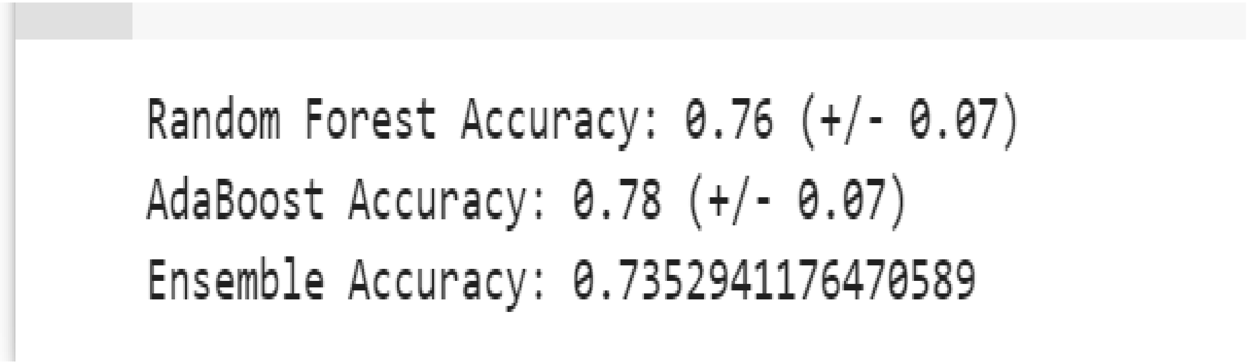
Figure 5.12: Reduced features

We applied SMOTE to balance our new dataset of the reduced features:



```
--NORMAL--
Class distribution before oversampling: 1    629
0    561
Name: target, dtype: int64
Class distribution after oversampling: 0    629
1    629
Name: target, dtype: int64
```

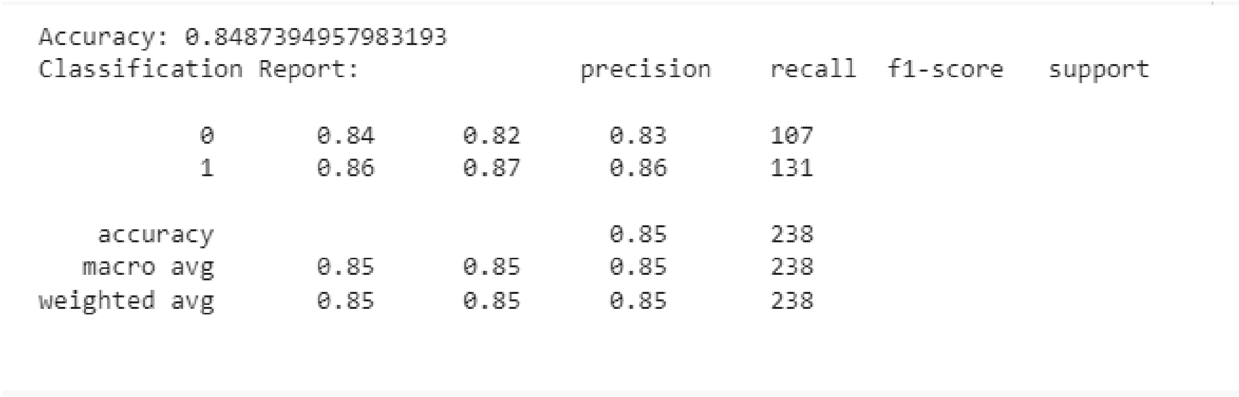
Figure 5.13: SMOTE Technique



```
Random Forest Accuracy: 0.76 (+/- 0.07)
AdaBoost Accuracy: 0.78 (+/- 0.07)
Ensemble Accuracy: 0.7352941176470589
```

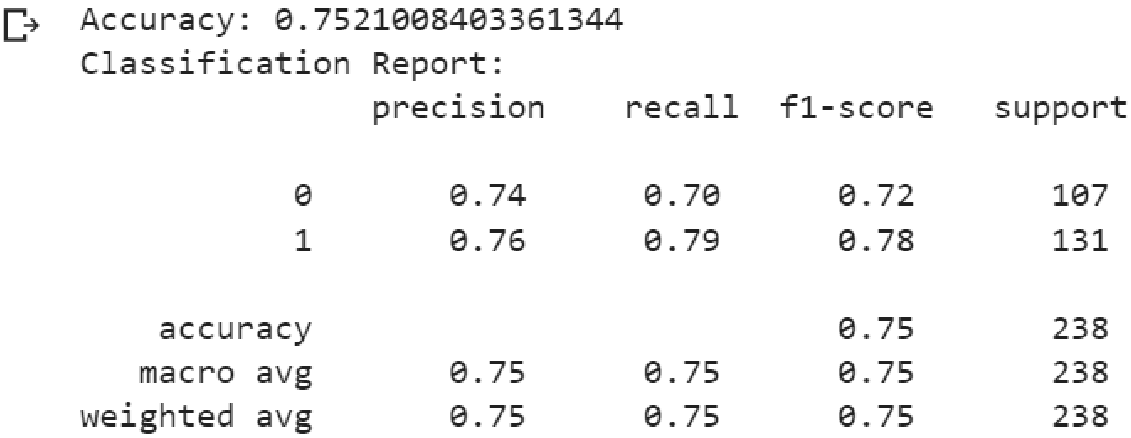
Figure 5.14: Ensemble Model

We created different models of Machine Learning and combined some of the models to get better accuracy.



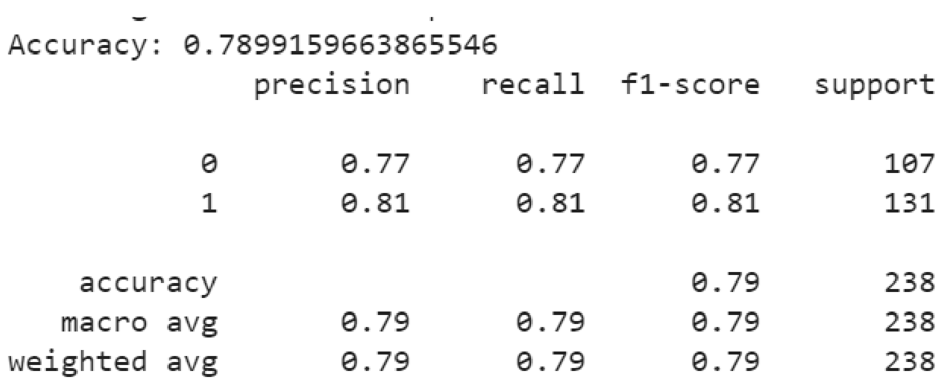
```
Accuracy: 0.8487394957983193
Classification Report:               precision    recall  f1-score   support

           0       0.84      0.82      0.83       107
           1       0.86      0.87      0.86       131

    accuracy                           0.85       238
   macro avg       0.85      0.85      0.85       238
weighted avg       0.85      0.85      0.85       238
```

Figure 5.15: SVM Model



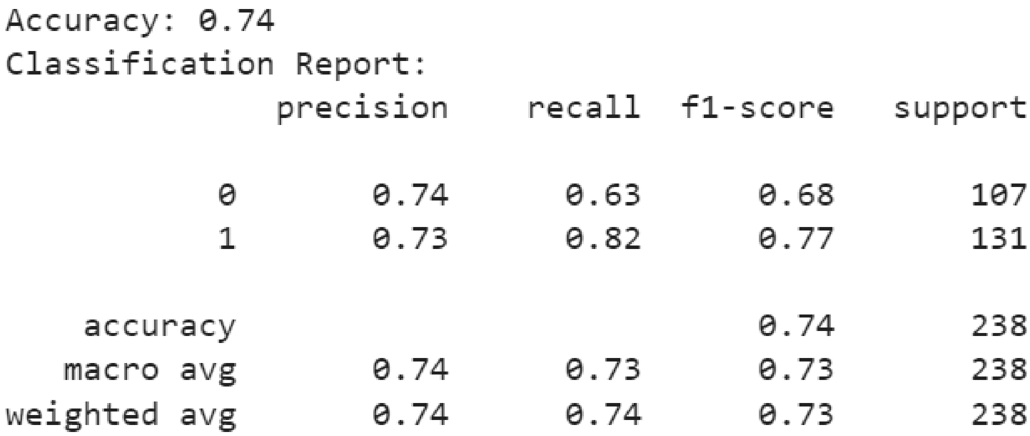
```
Accuracy: 0.7521008403361344
Classification Report:
              precision    recall  f1-score   support

           0       0.74      0.70      0.72       107
           1       0.76      0.79      0.78       131

    accuracy                           0.75       238
   macro avg       0.75      0.75      0.75       238
weighted avg       0.75      0.75      0.75       238
```

Figure 5.16: KNN Model



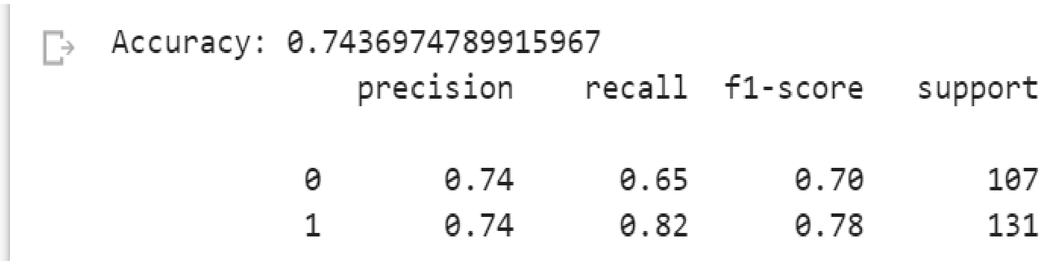
```
Accuracy: 0.7899159663865546
              precision    recall  f1-score   support

           0       0.77      0.77      0.77       107
           1       0.81      0.81      0.81       131

    accuracy                           0.79       238
   macro avg       0.79      0.79      0.79       238
weighted avg       0.79      0.79      0.79       238
```

Figure 5.17: MLP Model



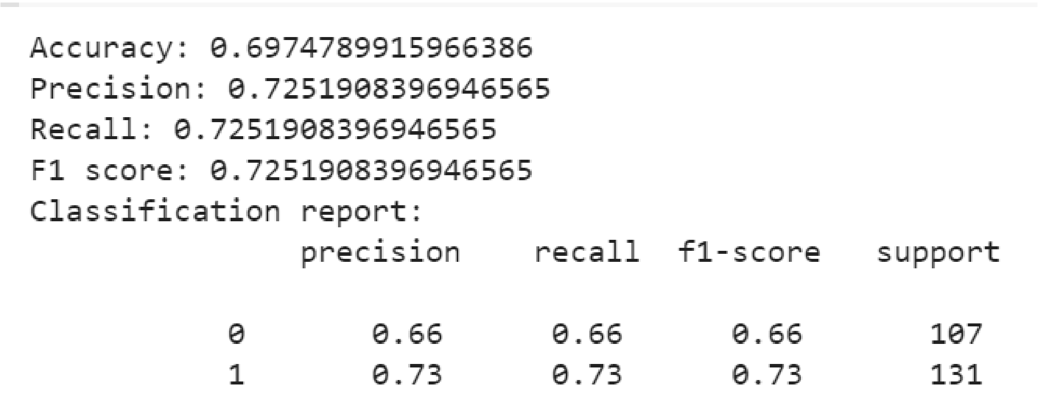
```
Accuracy: 0.74
Classification Report:
              precision    recall  f1-score   support

           0       0.74      0.63      0.68       107
           1       0.73      0.82      0.77       131

    accuracy                           0.74       238
   macro avg       0.74      0.73      0.73       238
weighted avg       0.74      0.74      0.73       238
```

Figure 5.18: Random Forest Model



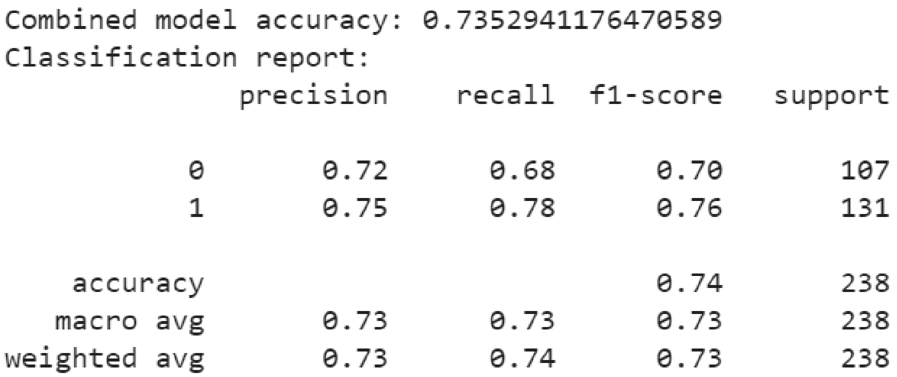
```
Accuracy: 0.7436974789915967
              precision    recall  f1-score   support

           0       0.74      0.65      0.70       107
           1       0.74      0.82      0.78       131
```

Figure 5.18: XGB Model



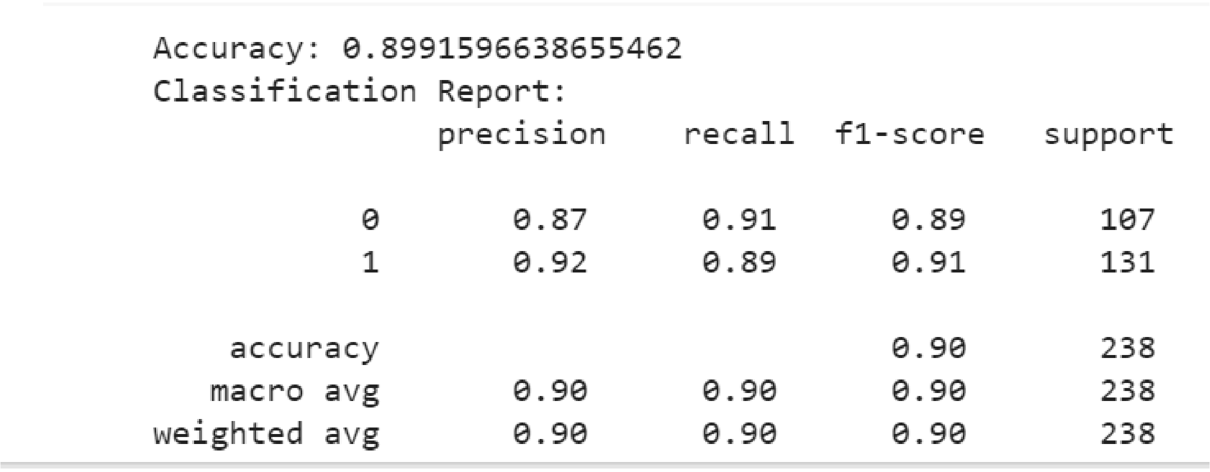
```
Accuracy: 0.6974789915966386
Precision: 0.7251908396946565
Recall: 0.7251908396946565
F1 score: 0.7251908396946565
Classification report:
              precision    recall  f1-score   support

           0       0.66      0.66      0.66       107
           1       0.73      0.73      0.73       131
```

Figure 5.19: Decision Tree



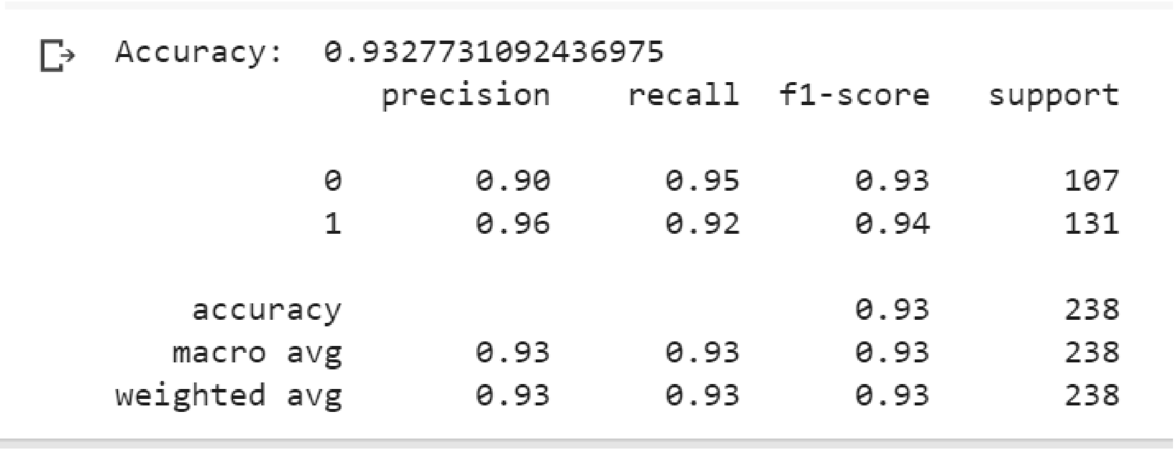
```
Combined model accuracy: 0.7352941176470589
Classification report:
              precision    recall  f1-score   support

           0       0.72      0.68      0.70       107
           1       0.75      0.78      0.76       131

    accuracy                           0.74       238
   macro avg       0.73      0.73      0.73       238
weighted avg       0.73      0.74      0.73       238
```

Figure 5.20: Random Forest and XGB Model



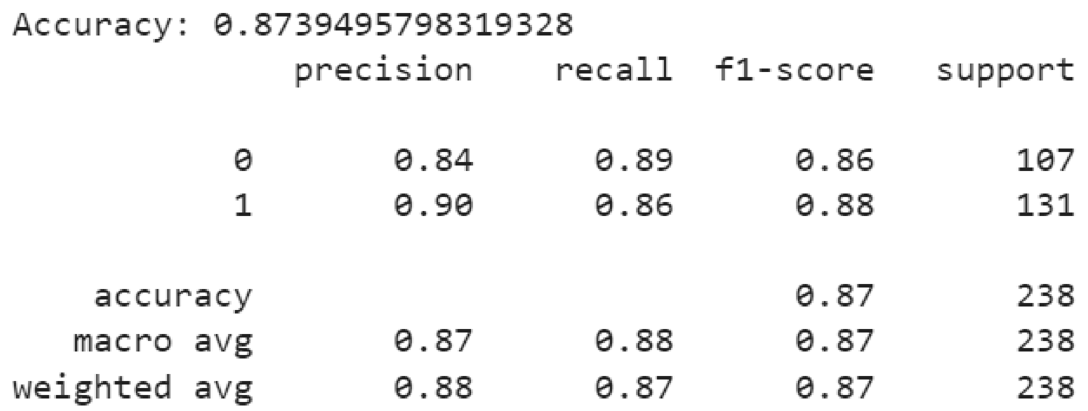
```
Accuracy: 0.8991596638655462
Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.91      0.89       107
           1       0.92      0.89      0.91       131

    accuracy                           0.90       238
   macro avg       0.90      0.90      0.90       238
weighted avg       0.90      0.90      0.90       238
```

Figure 5.21: MLP and Random Forest Model



```
Accuracy:  0.9327731092436975
              precision    recall  f1-score   support

           0       0.90      0.95      0.93       107
           1       0.96      0.92      0.94       131

    accuracy                           0.93       238
   macro avg       0.93      0.93      0.93       238
weighted avg       0.93      0.93      0.93       238
```

Figure 5.22: Random Forest, XGB and MLP Model



```
Accuracy: 0.8739495798319328
              precision    recall  f1-score   support

           0       0.84      0.89      0.86       107
           1       0.90      0.86      0.88       131

    accuracy                           0.87       238
   macro avg       0.87      0.88      0.87       238
weighted avg       0.88      0.87      0.87       238
```

Figure 5.23: Random Forest, KNN and MLP Model

5.2 Performance Analysis

The project uses SVM, MLP, random forest, KNN, XGB and decision trees among other machine learning techniques to predict CVD. For the prediction of CVD, 12 variables are considered , that are gender, the type of chest pain, cholesterol, resting ECG , maximum heart rate etc. The accuracy of each algo is calculated, and the method that provides the best accuracy is chosen for the prediction of heart disease. Numerous assessment criteria, including accuracy, precision, ROC, sensitivity, specificity, log loss, Mathew correlation coefficient and f1-score, are taken into consideration while estimating the experiment.

TP is True positive

FP is False Positive

FN is False Negative

TN is True Negative

- Accuracy- The ratio of all the correct predicted values is known as the accuracy of that model.[9] It is written as:

$$Accuracy = (TP + TN) \div (\mathrm{TP} + \mathrm{FN} + \mathrm{TN} + \mathrm{FP}) \qquad (5.1)$$

- Precision- It is the part of precisely positive results that we got to all of the results that are positive that the algorithm has anticipated.[9] It is written as:

$$Precision = (TP) \div (\mathrm{TP} + \mathrm{FP}) \qquad (5.2)$$

- F1 Score-Precision and Recall's harmonic mean, or F1 Score, is calculated. It gauges the test's precision. This measure has a range of 0 to 1.[9]

$$\mathrm{F1} = (2 \times \mathrm{precision} \times \mathrm{recall}) \div (\mathrm{precision} + \mathrm{recall}) \qquad (5.3)$$

- Sensitivity :It discusses the potential for a classifier to correctly forecast a favorable outcome in the presence of disease.[9]

$$Sensitivity = (TP) \div (\mathrm{TP} + \mathrm{FN}) \qquad (5.4)$$

- Specificity or True negative rate (TNR): It talks about how likely it is for a classifier to predict a positive outcome when there is a sickness. [9]

$$Specificity = (TN) \div (\mathrm{TN} + \mathrm{FP}) \qquad (5.5)$$

- Mathews correlation coefficient (MCC):MCC 1/4 0 signifies a forecast that is as accurate as chance, MCC 1/4 1 denotes absolute discrepancy between actual and projected values, and MCC 1/4 + 1 denotes an unflawed prediction. [9].

$$\mathrm{MCC} = (\mathrm{TP}\times\mathrm{TN} - \mathrm{FP}\times\mathrm{FN}) / \sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})} \qquad (5.6)$$

- AU-ROC : AU-ROC is a helpful and popular performance metric for classification issues. TPR vs. FPR at various threshold levels are used to plot it. The AU-ROC is a great performance comparison metric because it assesses performance across a wide range of class distributions and error levels.[9]

$$\text{AU-ROC} = 1 \div (2\times(TP \div (TP+FN)) + (TN \div (TN+FP))) \qquad (5.7)$$

- Log loss function: The log loss function transforms a theoretical claim into a real-world claim. Continuous issue iteration by inquiring, modeling the issue using the chosen approach, and testing is required to create a highly accurate predictor.

$$Hp(q) = (-\ 1 \div N)\times \sum_{i=1}^{N} (yi\,.\,log(p(yi))\ +\ (1\ -\ yi).\,log(1\ -\ p(yi))$$

(5.8)

Where p(y) is the estimated probability of the data point being 1 for all N points, and y is the label (0 and 1 for binary).[9]

CHAPTER 6

CONCLUSION

As we know cardiovascular disease is an ailment that takes hold of the heart negatively, which is the foremost cause of death worldwide, detecting the disease in early stages is necessary. Therefore, we have developed a model using machine learning algorithms that will predict if a person is more prone to having a heart disease or not .We have used six algorithms that are Random Forest, Decision Tree, KNN, SVM, XGB, and MLP in our project.

We then combined several models like, MLP and KNN Model, Random Forest and XGB Model, MLP and Random Forest Model, Random Forest, XGB and MLP Model and Random Forest, KNN and MLP Model.

After evaluating the efficacy of each of the six machine learning approaches, a model is developed that will predict the CVD in advance. Therefore, the goal is to use a variety of evaluation metrics, including those that accurately predict the disease, such as accuracy, precision, sensitivity, specificity, log loss, MCC, and f1-score. When methods are examined, we see that the Random Forest, and MLP Model gave the highest accuracy of 91%.

In future we would like to make this model available as a website and as an app.

REFERENCES

[1] Jimin Liu, Xueyu Dong , HuiqiZhaoandYinhua Tian , *“Predictive Classifier for Cardiovascular Disease Based on Stacking Model Fusion”* , College of Intelligence Equipment, Shandong University of Science and Technology, Tai’an 271000, China, 2022.

[2] Joshua Emakhu and Sujeet Shrestha ,*“Prediction System for Heart Disease Based on Ensemble Classifiers”*, Proceedings of the 5th NA International Conference on Industrial Engineering and Operations ManagementDetroit, Michigan, USA, August 10 - 14, 2020

[3] Katarya, R., Meena, S.K. ,*“Machine Learning Techniques for Heart Disease Prediction: A Comparative Study and Analysis”* , Health Technol. 11, 87–97 (2021).

[4] ApurbRajdhan ,Avi Agarwal , Milan Sai , Dundigalla Ravi, Dr. Poonam Ghuli, 2020,*“ Heart Disease Prediction using Machine Learning”,*INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) ,Vol 09, Issue 04 (April 2020)

[5] Goel, Rati, *“Heart Disease Prediction Using Various Algorithms of Machine Learning”* (July 12, 2021). Proceedings of the International Conference on Innovative Computing & Communication (ICICC) 2021

[6] Gunturu Deepthi,Cherukuri Shivani, KoruproluNagavinith,KesuboyinaHanudeep*, “HEART DISEASE PREDICTION USING MACHINE LEARNING ALGORITHMS”,*2020 International Conference on Electrical and Electronics Engineering (ICE3-2020)

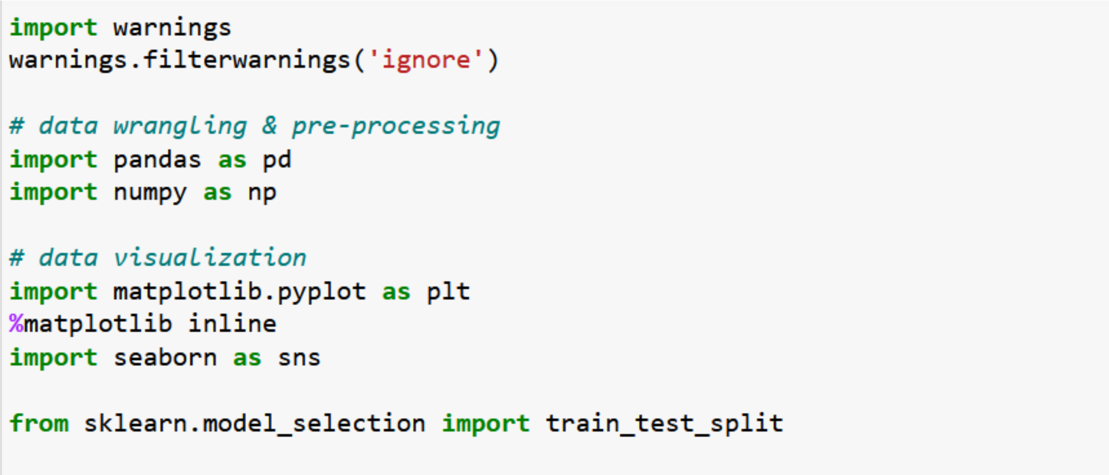
[7] Rohit Bharti, Aditya Khamparia, Mohammad Shabaz, Gaurav Dhiman, Sagar Pande, Parneet Singh, *"Prediction of Heart Disease Using a Combination of Machine Learning and Deep Learning",* Computational Intelligence and Neuroscience, vol. 2021, Article ID 8387680, 11 pages, 2021.

[8] ShubhanshiSinghal , Harish Kumar , Vishal Passricha , *“Prediction of Heart Disease using CNN”* , American International Journal of Research in Science, Technology, Engineering & Mathematics, 23(1), June-August,2018

[9] Achyut Tiwari, Aryan Chugh, Aman Sharma,*“Ensemble framework for cardiovascular disease prediction”,* Computers in Biology and Medicine,Volume 146,2022

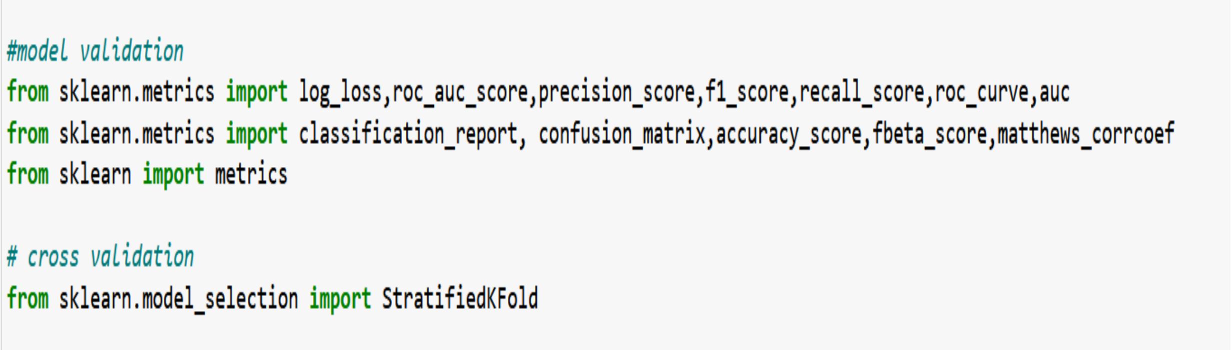
[10]Pooja Anbuselvan, *“Heart Disease Prediction using Machine Learning Techniques”*,International Journal of Engineering Research & Technology (IJERT),Vol. 9 Issue 11, November-2020

APPENDIX



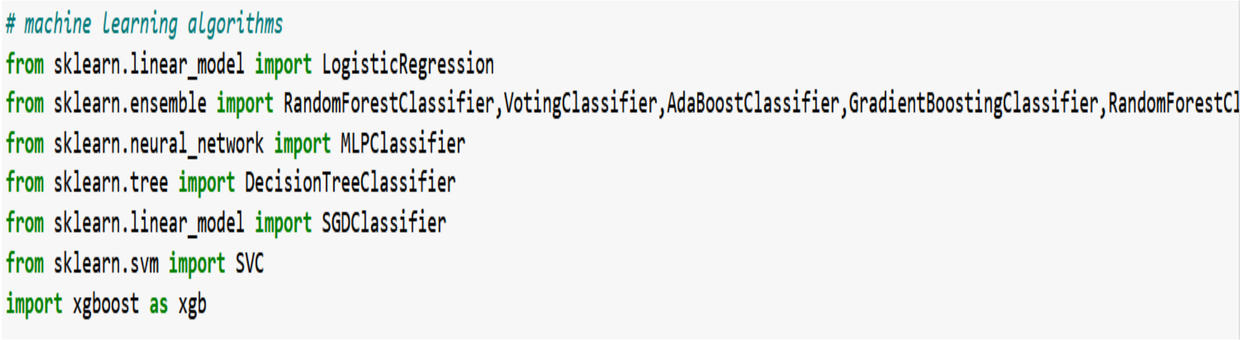
```
import warnings
warnings.filterwarnings('ignore')

# data wrangling & pre-processing
import pandas as pd
import numpy as np

# data visualization
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns

from sklearn.model_selection import train_test_split
```

Figure A.1: Imported Libraries



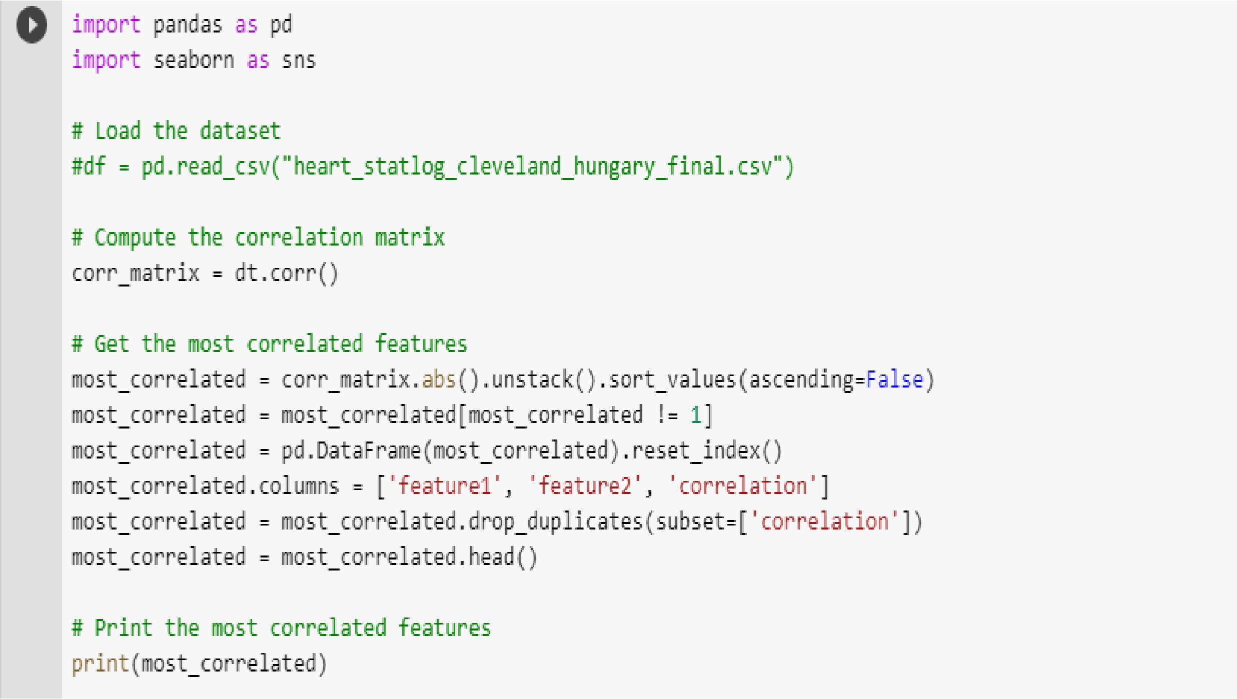
```
#model validation
from sklearn.metrics import log_loss,roc_auc_score,precision_score,f1_score,recall_score,roc_curve,auc
from sklearn.metrics import classification_report, confusion_matrix,accuracy_score,fbeta_score,matthews_corrcoef
from sklearn import metrics

# cross validation
from sklearn.model_selection import StratifiedKFold
```

Figure A.2: Imported for model evaluation



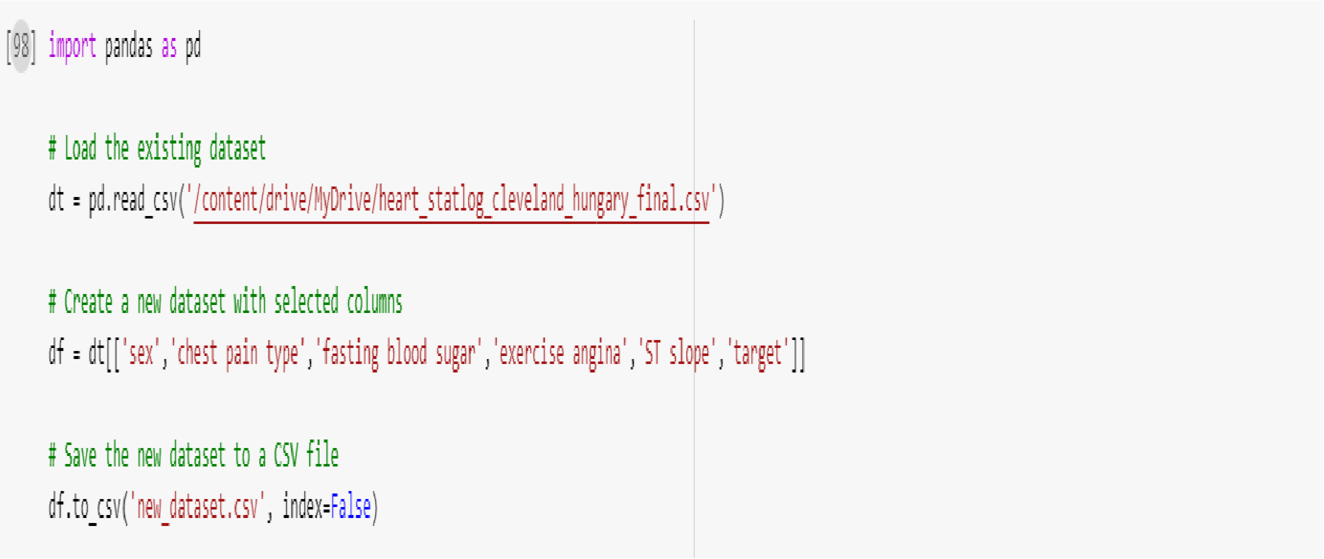
```
# machine learning algorithms
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier,VotingClassifier,AdaBoostClassifier,GradientBoostingClassifier,RandomForestCl
from sklearn.neural_network import MLPClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import SGDClassifier
from sklearn.svm import SVC
import xgboost as xgb
```

Figure A.3: Algorithms Used



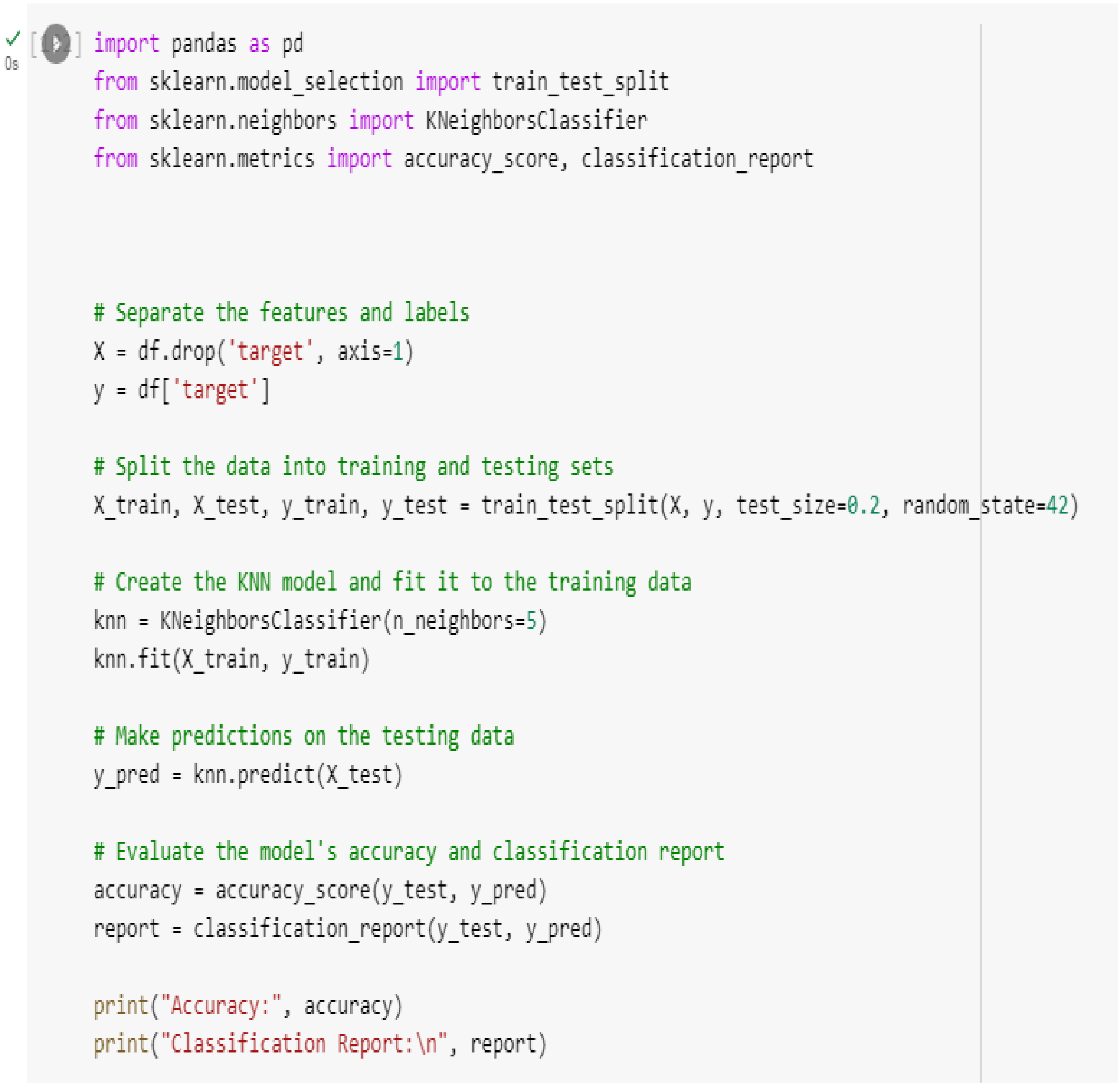
```
import pandas as pd
import seaborn as sns

# Load the dataset
#df = pd.read_csv("heart_statlog_cleveland_hungary_final.csv")

# Compute the correlation matrix
corr_matrix = dt.corr()

# Get the most correlated features
most_correlated = corr_matrix.abs().unstack().sort_values(ascending=False)
most_correlated = most_correlated[most_correlated != 1]
most_correlated = pd.DataFrame(most_correlated).reset_index()
most_correlated.columns = ['feature1', 'feature2', 'correlation']
most_correlated = most_correlated.drop_duplicates(subset=['correlation'])
most_correlated = most_correlated.head()

# Print the most correlated features
print(most_correlated)
```

Figure A.4: Correlated features



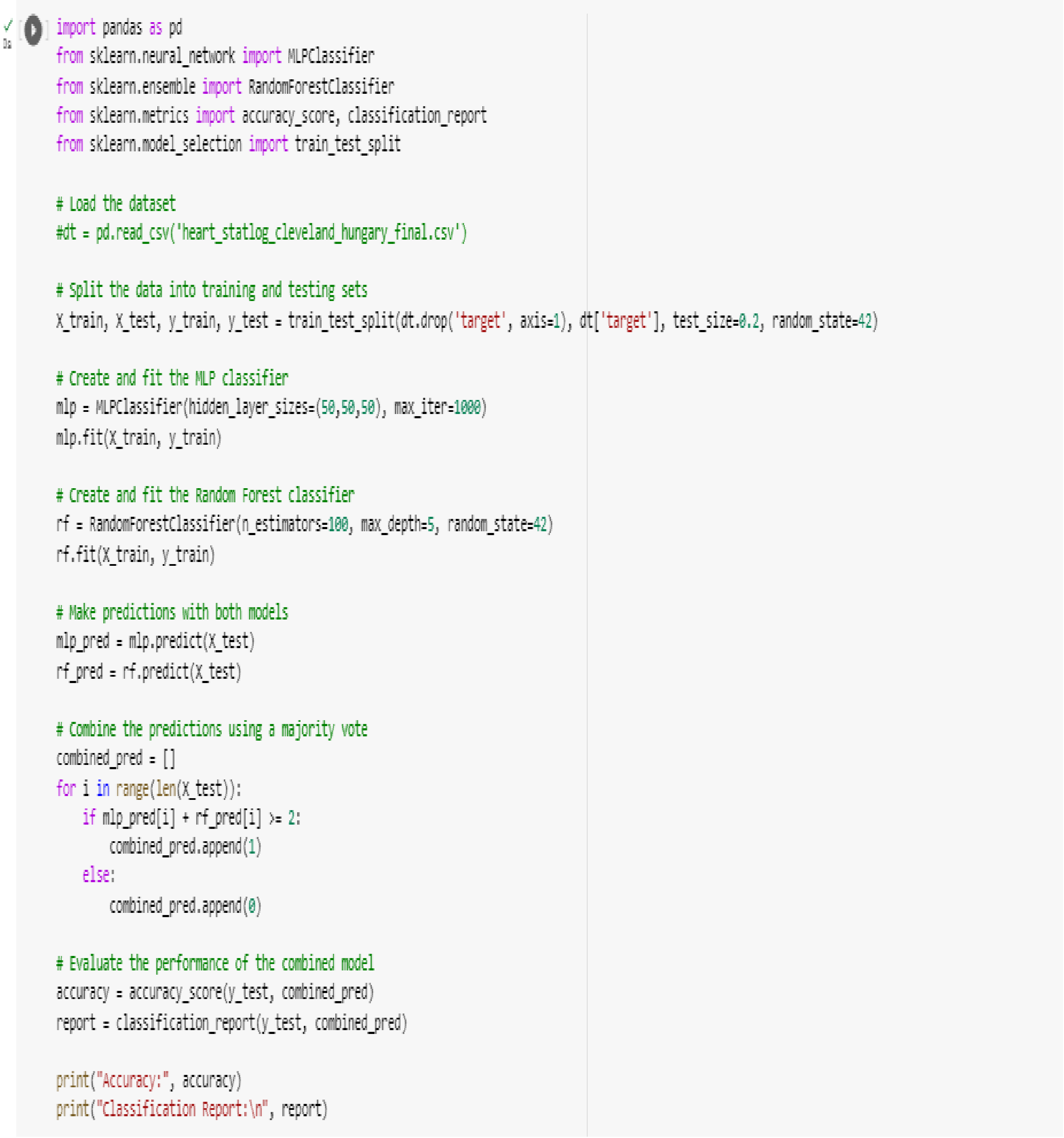
```
import pandas as pd

# Load the existing dataset
dt = pd.read_csv('/content/drive/MyDrive/heart_statlog_cleveland_hungary_final.csv')

# Create a new dataset with selected columns
df = dt[['sex','chest pain type','fasting blood sugar','exercise angina','ST slope','target']]

# Save the new dataset to a CSV file
df.to_csv('new_dataset.csv', index=False)
```

Figure A.5: New Dataframe



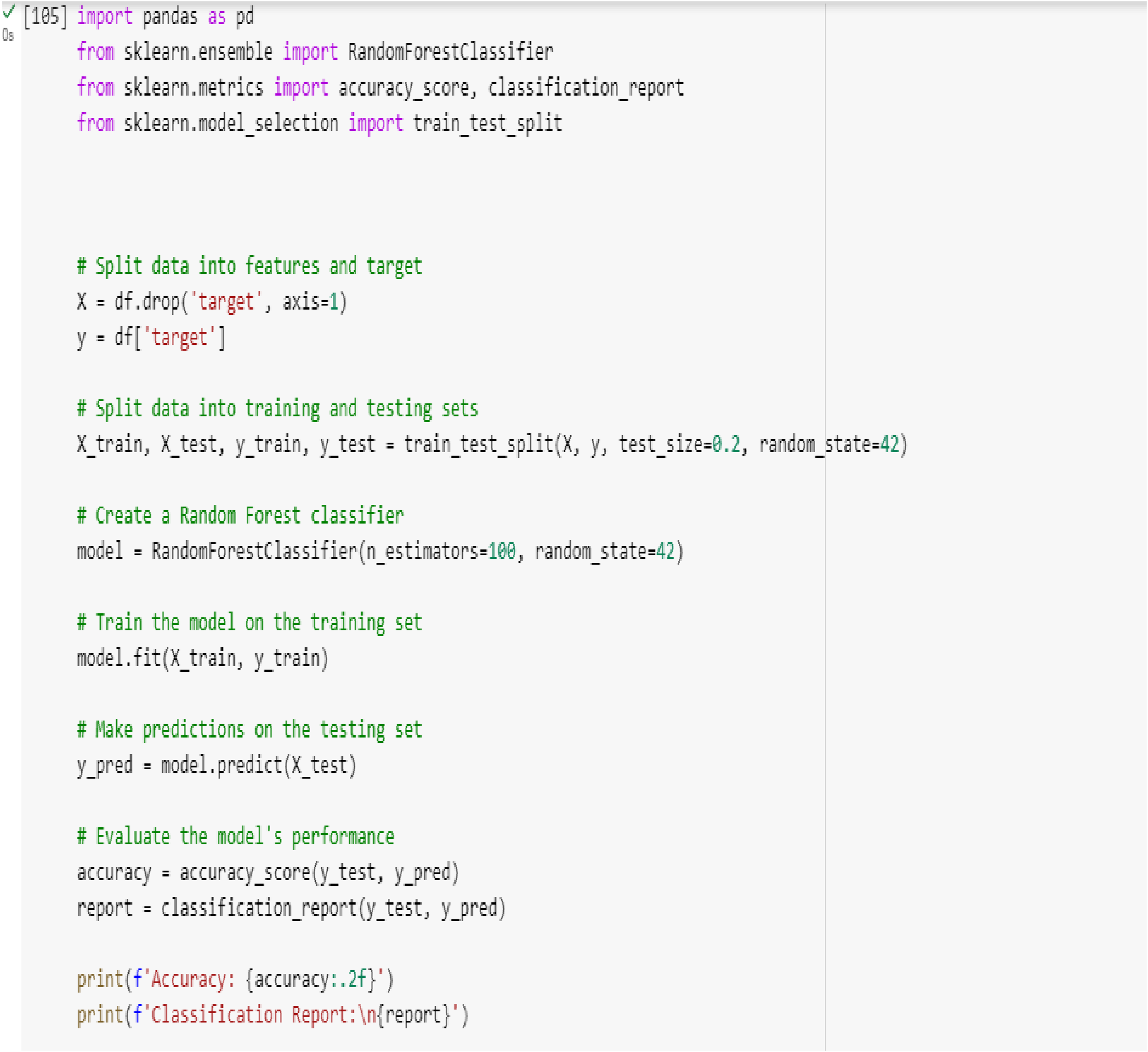
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, classification_report



# Separate the features and labels
X = df.drop('target', axis=1)
y = df['target']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create the KNN model and fit it to the training data
knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train, y_train)

# Make predictions on the testing data
y_pred = knn.predict(X_test)

# Evaluate the model's accuracy and classification report
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
print("Classification Report:\n", report)
```

Figure A.6: KNN Model



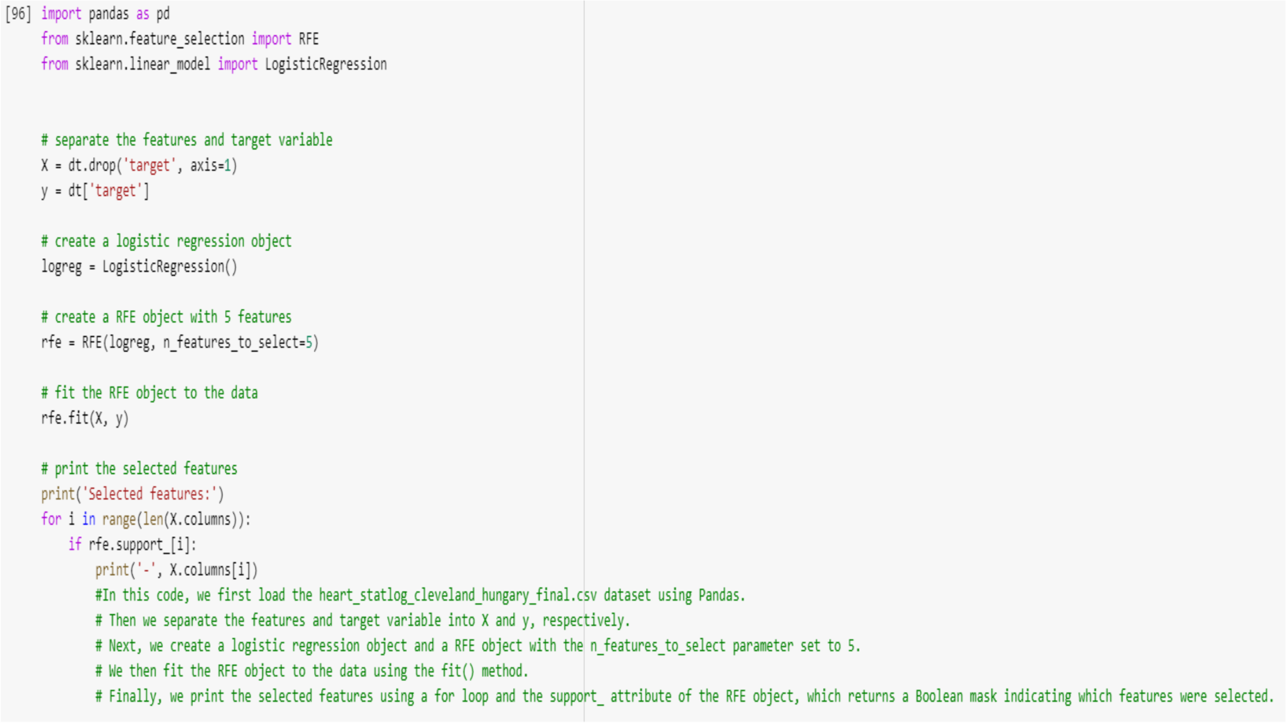
```
import pandas as pd
from sklearn.neural_network import MLPClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.model_selection import train_test_split

# Load the dataset
#dt = pd.read_csv('heart_statlog_cleveland_hungary_final.csv')

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(dt.drop('target', axis=1), dt['target'], test_size=0.2, random_state=42)

# Create and fit the MLP classifier
mlp = MLPClassifier(hidden_layer_sizes=(50,50,50), max_iter=1000)
mlp.fit(X_train, y_train)

# Create and fit the Random Forest classifier
rf = RandomForestClassifier(n_estimators=100, max_depth=5, random_state=42)
rf.fit(X_train, y_train)

# Make predictions with both models
mlp_pred = mlp.predict(X_test)
rf_pred = rf.predict(X_test)

# Combine the predictions using a majority vote
combined_pred = []
for i in range(len(X_test)):
    if mlp_pred[i] + rf_pred[i] >= 2:
        combined_pred.append(1)
    else:
        combined_pred.append(0)

# Evaluate the performance of the combined model
accuracy = accuracy_score(y_test, combined_pred)
report = classification_report(y_test, combined_pred)

print("Accuracy:", accuracy)
print("Classification Report:\n", report)
```

Figure A.7: MLP and Random forest Model



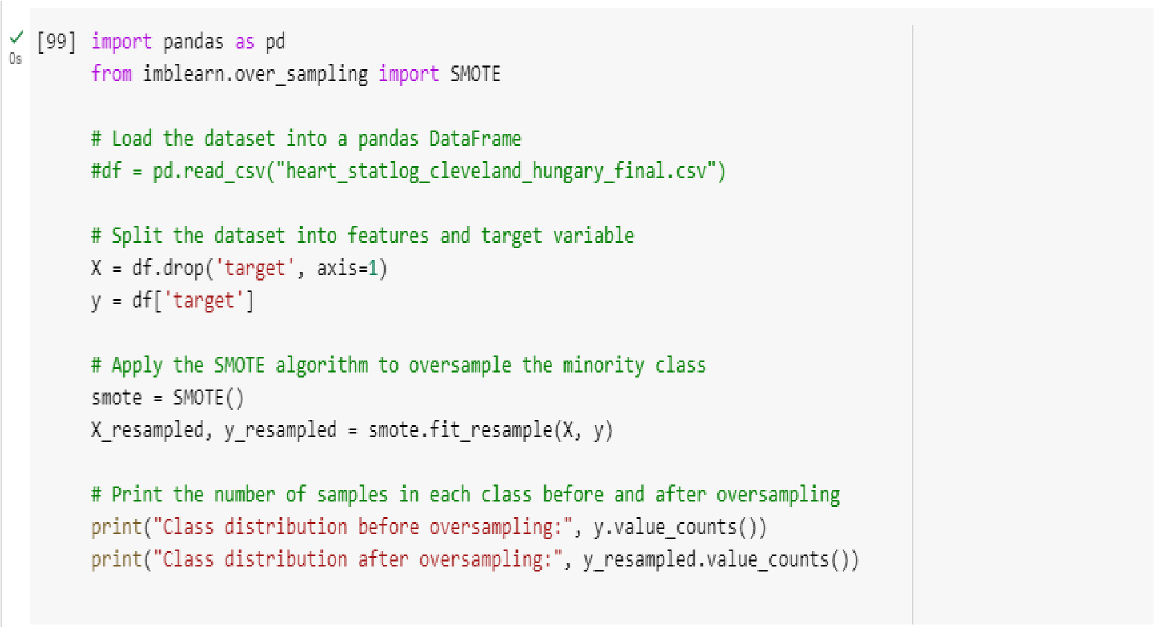
```
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.model_selection import train_test_split



# Split data into features and target
X = df.drop('target', axis=1)
y = df['target']

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a Random Forest classifier
model = RandomForestClassifier(n_estimators=100, random_state=42)

# Train the model on the training set
model.fit(X_train, y_train)

# Make predictions on the testing set
y_pred = model.predict(X_test)

# Evaluate the model's performance
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

print(f'Accuracy: {accuracy:.2f}')
print(f'Classification Report:\n{report}')
```

Figure A.8: Random Forest



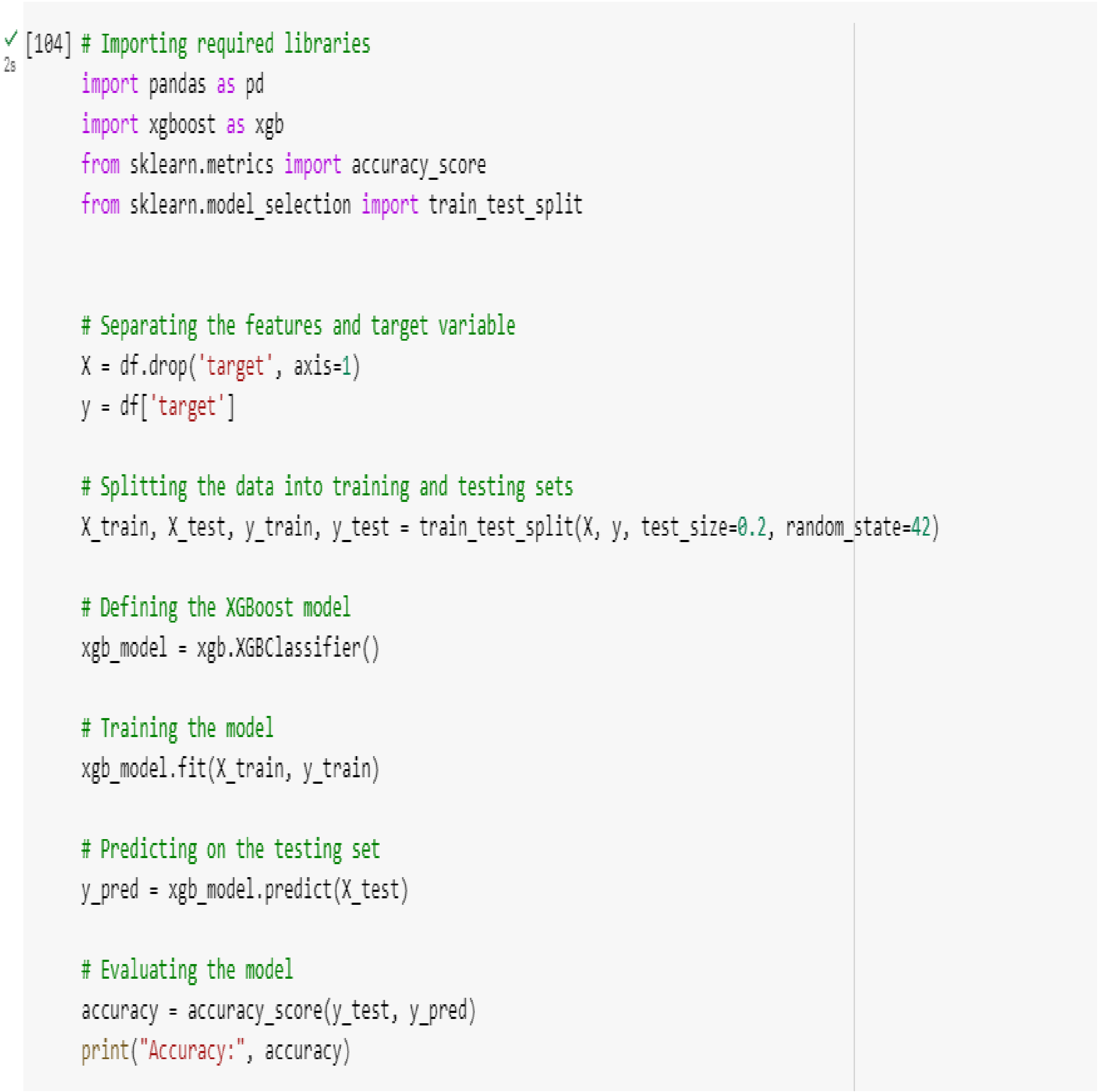
```
[96] import pandas as pd
     from sklearn.feature_selection import RFE
     from sklearn.linear_model import LogisticRegression


     # separate the features and target variable
     X = dt.drop('target', axis=1)
     y = dt['target']

     # create a logistic regression object
     logreg = LogisticRegression()

     # create a RFE object with 5 features
     rfe = RFE(logreg, n_features_to_select=5)

     # fit the RFE object to the data
     rfe.fit(X, y)

     # print the selected features
     print('Selected features:')
     for i in range(len(X.columns)):
         if rfe.support_[i]:
             print('-', X.columns[i])
             #In this code, we first load the heart_statlog_cleveland_hungary_final.csv dataset using Pandas.
             # Then we separate the features and target variable into X and y, respectively.
             # Next, we create a logistic regression object and a RFE object with the n_features_to_select parameter set to 5.
             # We then fit the RFE object to the data using the fit() method.
             # Finally, we print the selected features using a for loop and the support_ attribute of the RFE object, which returns a Boolean mask indicating which features were selected.
```

Figure A.9: Reduced Features



```
✓  [99] import pandas as pd
0s      from imblearn.over_sampling import SMOTE

        # Load the dataset into a pandas DataFrame
        #df = pd.read_csv("heart_statlog_cleveland_hungary_final.csv")

        # Split the dataset into features and target variable
        X = df.drop('target', axis=1)
        y = df['target']

        # Apply the SMOTE algorithm to oversample the minority class
        smote = SMOTE()
        X_resampled, y_resampled = smote.fit_resample(X, y)

        # Print the number of samples in each class before and after oversampling
        print("Class distribution before oversampling:", y.value_counts())
        print("Class distribution after oversampling:", y_resampled.value_counts())
```

Figure A.10: SMOTE



```python
# Importing required libraries
import pandas as pd
import xgboost as xgb
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split



# Separating the features and target variable
X = df.drop('target', axis=1)
y = df['target']

# Splitting the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Defining the XGBoost model
xgb_model = xgb.XGBClassifier()

# Training the model
xgb_model.fit(X_train, y_train)

# Predicting on the testing set
y_pred = xgb_model.predict(X_test)

# Evaluating the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

Figure A.11: XGBoost Model