**Group 389: Heart Disease Indication using Classification**

|  |  |  |
| --- | --- | --- |
| First Name | Last Name | Email address |
| Parth | Ghatpande | [pghatpande@hawk.iit.edu](mailto:pghatpande@hawk.iit.edu) |
| Jay | Gupta | [jgupta3@hawk.iit.edu](mailto:jgupta3@hawk.iit.edu) |

Table of Contents

[**1. Introduction** 2](#_Toc56029461)

[**2. Data** 2](#_Toc56029462)

[**3. Problems and Solutions** 2](#_Toc56029463)

[**4. KDD** 3](#_Toc56029464)

[4.1. Data Processing 3](#_Toc56029465)

[4.2. Data Mining Methods and Processes 3](#_Toc56029466)

[**5. Evaluations and Results** 4](#_Toc56029467)

[5.1. Evaluation Methods 5](#_Toc56029468)

[5.2. Results and Findings 5](#_Toc56029469)

[**6. Conclusions and Future Work** 5](#_Toc56029470)

[6.1. Conclusions 5](#_Toc56029471)

[6.2. Limitations 5](#_Toc56029472)

[6.3. Potential Improvements or Future Work 5](#_Toc56029473)

# **1. Introduction**

# During the year 2022, there have been 805K attacks in the United States. Based on the data available, few of the main concerns of the heart attack constitute of High Blood Pressure, Cholesterol and Diabetes. Thus, upon further research, we have decided to formulate different classification models which will tell us the causes of the heart attack. Our model can help us to predict the factors that are most common for a person who suffers from a heart attack.

# **2. Data**

We have obtained this data source from Kaggle. The URL for the following dataset is as listed below:

URL: <https://www.kaggle.com/datasets/alexteboul/heart-disease-health-indicators-dataset>

This dataset consists of approximately 250K (253,680) rows and 22 columns.

This data consists of the different parameters which can a person to cause a heart attack. The attributes/columns based in the table are listed below.

1. HeartDiseaseOrAttack 12. HeavyAlcoholConsmp
2. HighBP 13. AnyHealthCare
3. HighChol 14. NoDocBcCost
4. CholCheck 15. GenHealth
5. BMI 16. MentHealth
6. Smoker 17. PhyHealth
7. Stroke 18. DiffWalk
8. Diabetes 19. Sex
9. PhyActivity 20. Age
10. Fruits 21. Education
11. Veggies 22. Income

All the data in the dataset is numeric data. Few of the columns are continuous and few of them are discrete.

# **3. Problems and Solutions**

The problems that we find for the given data are listed below:

* From the above data, we want to make a prediction based on the entry of a user. If a user has some medical history or condition, Will they suffer from a heart attack?
* Given a particular set of habits, will the user have a heart disease or could suffer from a heart attack?
* From the above data, is the result of the given label correct or not?

From the above questions, we have decided to develop different classification algorithms. Initially we select the best features from the above listed features and develop a different data frame to check whether there is an increase in the accuracy of our models. After this, we decided to implement the following classification models.

1. KNN (K Nearest Neighbor)
2. Logistic Regression
3. Decision Tree
4. Naïve Bayes Classifier
5. SVM
6. Neural Networks.

# **4. KDD**

## 4.1. Data Processing

In this section, initially we checked if there are any missing values in the data. There were no missing values in the data.

Graphical user interface

Description automatically generated with low confidence

Thus, after we checked all the null values, we had to standardize the data. For the standardization of the data, we used to Min-Max scaler to standardize the data. This is also known as the Min-Max Standardization process.

Table

Description automatically generated

As most of our values are numeric, we did not need to change them from nominal to numeric.

Further ahead, we have selected HeartDiseaseOrAttack as our prediction Label. We have transformed this label using the One Hot Encoding process for the label. This way, we can convert this column into an array of the size of the data. For this, we have used the LabelEncoder function which is built in sklearn preprocessing library/package.

Graphical user interface, text, application, email

Description automatically generated

## 4.2. Data Mining Methods and Processes

Initially, we decided to map out the best features from the data. These features can be used to map the data, they can be used to increase the accuracy. Thus, we decided to implement feature selection based on ANOVA (Analysis of Variance).

Graphical user interface, text, application

Description automatically generated

Furthermore, we created a dataframe based on the top 5 features listed here.

Table

Description automatically generated with low confidence

For the data mining process, we choose the train-test split method to evaluate the data. We have split the data into training and testing data. For this, we have used the 70% - 30% ratio to split the data randomly.

Along with the ratio, we have used the TrainTestSplit method which is included in the scikitlearn package/libraries.

Graphical user interface, text, application

Description automatically generated

We have also implemented Hyperparameter Tuning. Hyperparameter Tuning can be defined as the process in which we alter the parameters of our model to reduce the loss of accuracy and to determine the best model for each of the method.

**Method 1 – KNN Classifier (K Nearest Neighbor)**

K Nearest Neighbor is used to find the nearest neighbors for a given label. The algorithm is used to make predictions based on distance in between the nearest label and the unseen label (Test label). For KNN, we have used both the distances (i.e., Manhattan Distance and Euclidean Distance). We have calculated the value of K approximately (i.e., 200) and have roughly assigned a random K value by the interval of 20.

Graphical user interface

Description automatically generated with low confidence

Thus, we have developed the other KNN models as follows (Hyperparameter Tuning):

Graphical user interface

Description automatically generated with medium confidence

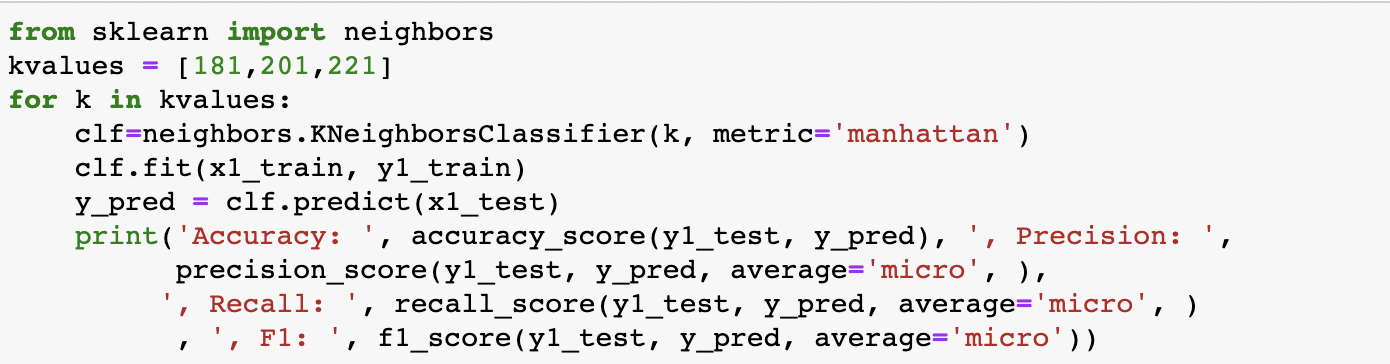
Chart, scatter chart

Description automatically generated

We have then implemented the same model for the best features as well.

Text

Description automatically generated



**Method 2: Decision Trees**

Decision Trees is a classification model which is based on the tree structure. The output of the tree is considered as the output label for the unseen data. In a Decision tree, the outcome of each layer is considered as the input label for the next level until the occurrence of a leaf node (terminal node). In short, we have used different methods to calculate and find out the leaf node. One of the main methods we would like to mention is the early stopping or pre-pruning method.

We have reduced the number of iterations for the trees so that the model avoids overfitting of the data.

Our initial model for the decision tree can be seen as follows:

A picture containing text

Description automatically generated

Hence, we have also implemented hyperparameter tuning for the Decision tree to get the best model with a good accuracy and a high F1 score.

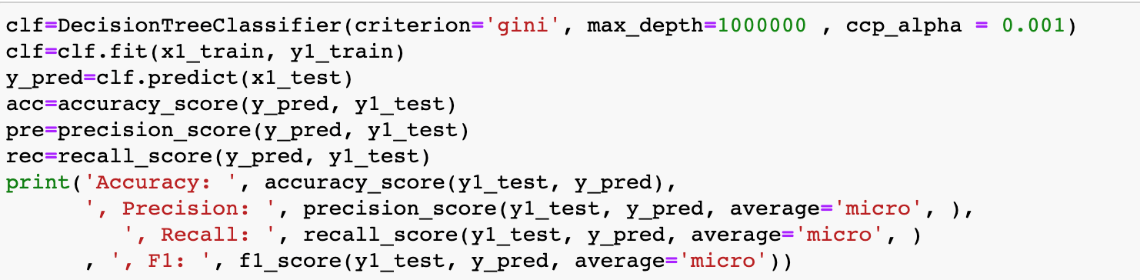
Text

Description automatically generated

Chart, scatter chart

Description automatically generated

We have also developed the different models for the best features to check if there is an improvement in the accuracy and the precision.



Text

Description automatically generated with medium confidence

Thus, we have developed models for the best features and have implemented it for different criteria’s as well.

**Method 3 – Logistic Regression**

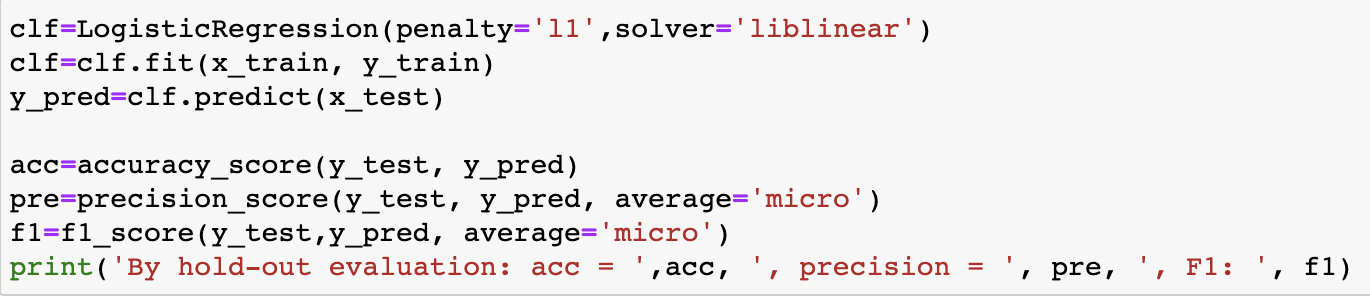
Logistic regression is a type of regression method that we can use to make predictions based on a particular penalty. In logistic regression, we have used the HeartDiseaseOrAttack label as the independent variable and have used the other variables as the dependent variables. Thus, the initial model that we have built is based on the l2 penalty which uses a logarithmic penalty to the base of 2.

For reducing the error and runtime, we have used different models for Logistic Regression.

**Diagram

Description automatically generated with low confidence**

Implementing Hyperparameter tuning for Logistic Regression to find out the best model and to check if there is an increase in the accuracy:

****

**Text

Description automatically generated with low confidence**

**A picture containing text

Description automatically generated**

We have also implemented logistic regression with the best selected features to check if there is a change in the accuracy:

**A picture containing text

Description automatically generated**

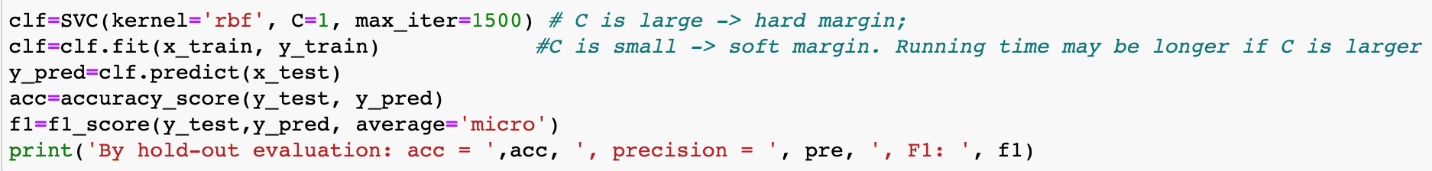
**Method 4 – SVM**

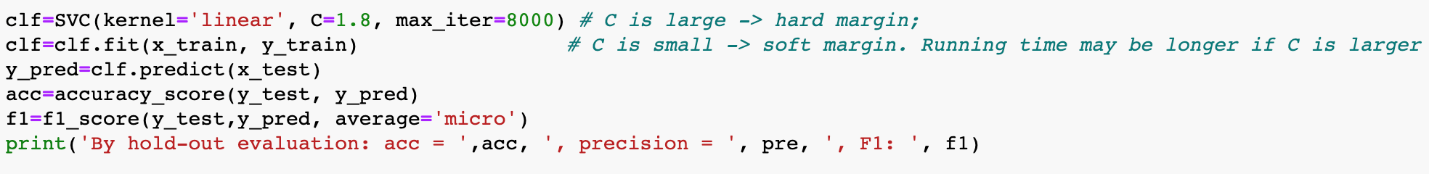
SVM stands for Support Vector Machine. It is a type of supervised learning algorithm which is used to determine and analyze the data. Thus, we have used different iterations for the SVM classifier to determine the best possible accuracy and F1 score. The advantage of using SVM is that there is an increase in the accuracy.

**A picture containing chart

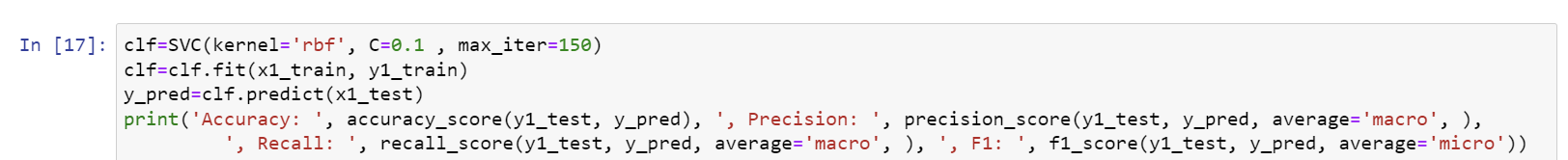
Description automatically generated**

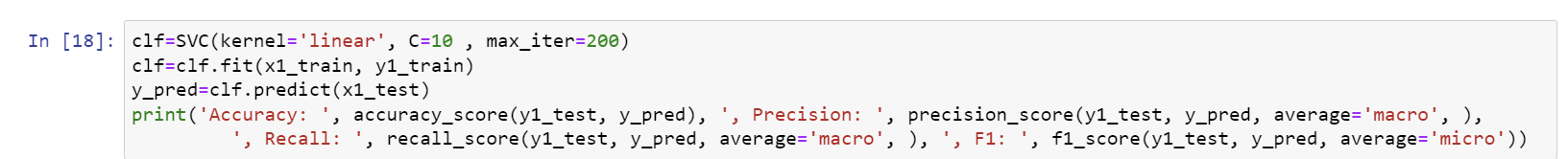
Performing Hyperparameter Tuning to determine if there is a change in the accuracy of the models. The main parameters

****

****

Implementing SVM for the best selected features





**Method 5 – Naïve Bayes**

Naïve Bayes classifier is a supervised learning method which is also known as probabilistic classifier which uses the bayes theorem to make predictions on the data. We have implemented different types of algorithms which use the Bayes theorem.

The different types of algorithms which we have used are:

1. Categorical Naïve Bayes
2. Bernoulli Naïve Bayes
3. Complement Naïve Bayes
4. Multinominal Naïve Bayes
5. Gaussian Naïve Bayes

***Text

Description automatically generated***

**Text

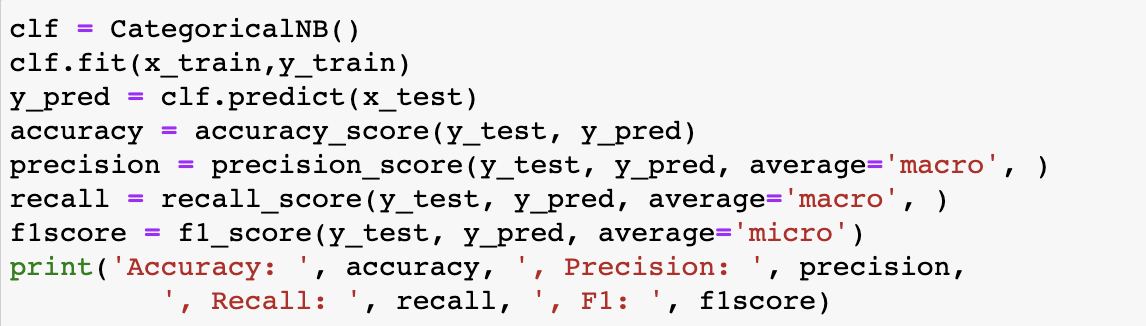
Description automatically generated**

**Text

Description automatically generated**

**Text, letter

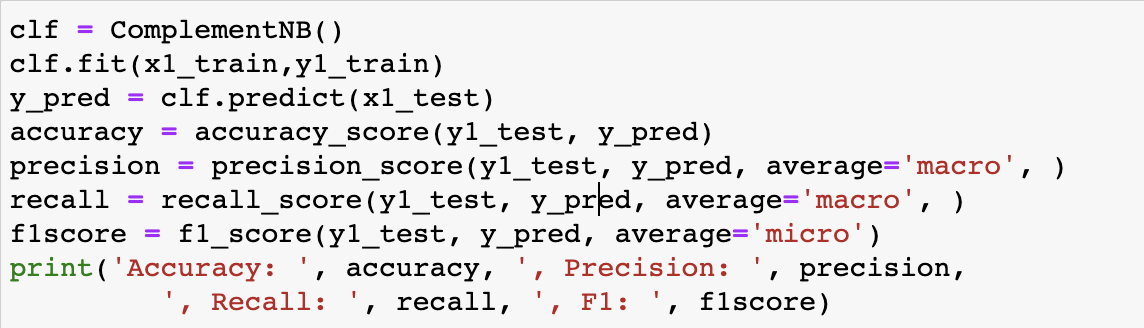
Description automatically generated**

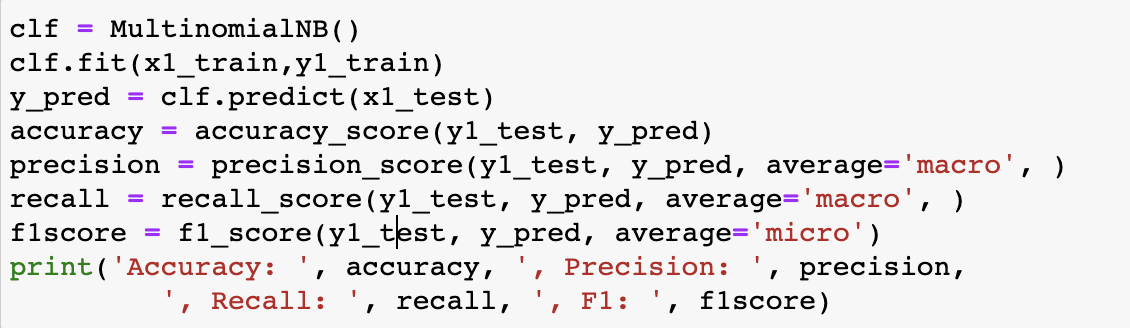
****

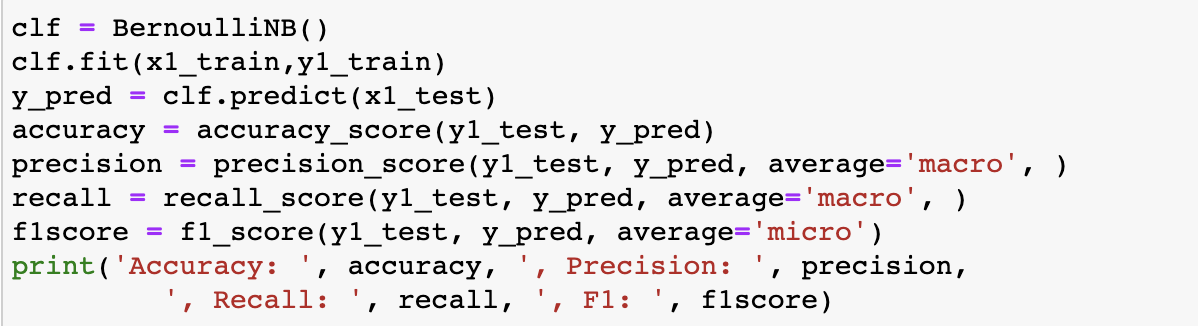
Now, checking for the best features to check if there is any change in accuracy for the Naïve Bayes classification.

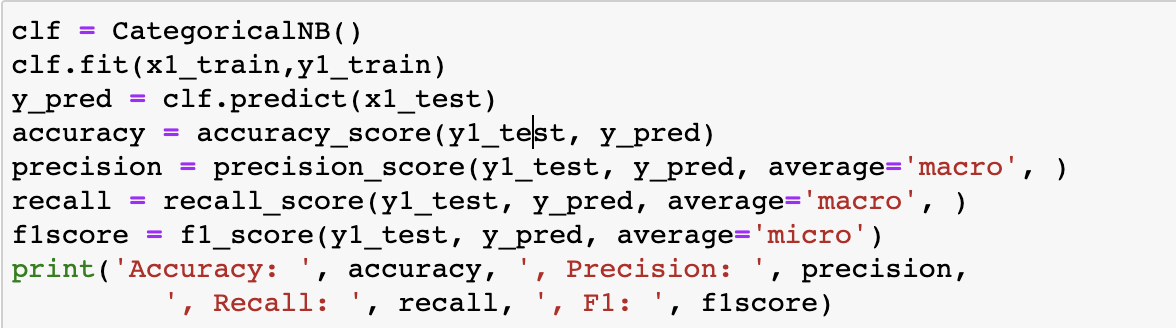
Text

Description automatically generated





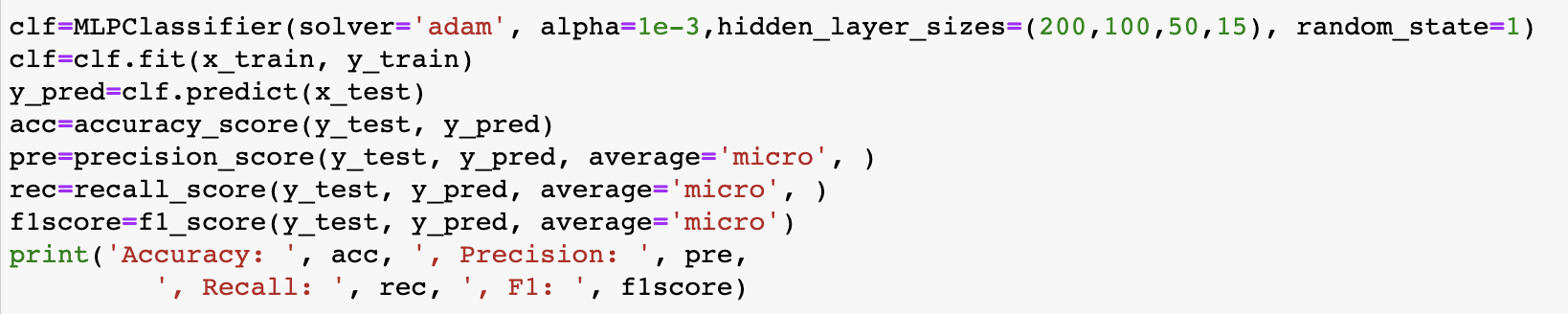




**Method 6 – Neural Network**

Neural Network is a supervised learning method which is used to make predictions on the data. Neural Network is based on the principle of neurons in the brain. Thus, while constructing a neural network one of the most important parameters are the number of hidden layers and the activation function. We have developed and designed different layers with different activation function.

****

****

We have also changed the learning rates and hidden layers size for the parameter Tuning process and added a few different parameters to check the change in the accuracy.

**Chart

Description automatically generated**

**A picture containing text

Description automatically generated**

Implementing Neural Network for Best Features.

**Chart, scatter chart

Description automatically generated with medium confidence**

**A picture containing text

Description automatically generated**

# **5. Evaluations and Results**

## 5.1. Evaluation Methods

For the evaluation methods, we have considered 4 different parameters to evaluate the correctness of the models. These 4 parameters are as follows:

1. Accuracy
2. Precision
3. Recall
4. F1 Score

## 5.2. Results and Findings

Below are the conclusions of the models which we have implemented with the best parameters.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S NO | Method | Accuracy | Precision | Recall | F1 Score |
| 1 | KNN | 90.8% | 79.6% | 51.5% | 90.2% |
| 2 | Decision Tree | 90.5% | 90.5% | 90.5% | 90.5% |
| 3 | Logistic Regression | 90.8% | 90.8% | 90.8% | 90.8% |
| 4 | SVM | 81.8% | 54.2% | 55.6% | 81.8% |
| 5 | Naïve Bayes | 90.5% | 67.3% | 56.8% | 90.5% |
| 6 | Neural Networks | 90.8% | 90.8% | 90.8% | 90.8% |

# **6. Conclusions and Future Work**

## 6.1. Conclusions

We are getting higher accuracy for each classification model after using the best selected features.

After comparison with all the classifiers Neural networks is giving the highest accuracy for predicting the label of the target variable.

## 6.2. Limitations

* With the presence of image data, we can construct a convolutional neural network which can yield a better accuracy.
* The accuracy can still be further improved with the help of outlier detection.

## 6.3. Potential Improvements or Future Work

* We can add image data to get a more precise outcome whether it is a heart disease or a heart attack
* We can also implement outlier detection to remove the outliers which can improve the accuracy of our models
* With a more accurate data set we can implement heart disease detection system which can inform the patient well in advance.