Group Members

- Valerius Owen (2301868371)
- Kelvin Aryanto (2301868415)
- Yesika (2301869784)

Heart Disease Classification based on Health Data

Introduction

Nowadays, the lifestyle of people, mainly in cities, are faster where people do not have time to prepare food and prefer to get fast food. And as we all know fast food leads to a poor diet which is one of the main causes of heart disease. From this, our project aims to be able to classify from the health data of a person to see if the person is likely or not likely to get heart disease. This would be very useful for doctors and hospitals to be able to advise their patients even further to be able to prevent or reduce the effects of heart disease. We believe that by using our program, we can raise awareness among people to maintain a healthy lifestyle to prevent heart disease.

In this project, we will be creating a program to check if a person is likely or not to have heart disease. We will be able to do this by using methods for classifying such as Logistic Regression, Single Layer Perceptron, Support Vector Machine (SVM), Multinomial Naive Bayes, Random Forest Tree, and K-Nearest Neighbor (K-NN). These methods are used for binary classification which outputs a result of 0 or 1 [1] which is also said not likely or likely to have heart disease. Training these models would include the adjusting of weights and biases for multiple epochs with an optimal learning rate to achieve accurate classification results. Because each classifier is best in only a select domain based upon the number of observations, the dimensionality of the feature vector, the noise in the data, and many other factors [2], thus we will train and test out all of the classifiers mentioned above and compare the accuracy of those models so that we can propose the best model for classifying whether or not a person is likely to have heart disease.

Data

Dataset retrieved from: https://www.kaggle.com/ronitf/heart-disease-uci. The dataset contains 303 data consisting of several factors that affect the heart health of a person. The table below shows the features and label description of the dataset used to classify if a person is likely or not likely to have heart disease.

Category	Column	Description	Possible Value
Feature	Age	Age	Number
	Sex	Gender male or female	Number (0 = female, 1 = male)

	CP	Chest pain type	Number (0, 1, 2, 3)
	TrestBPS	Resting blood pressure (in mm Hg on admission to the hospital)	Number
	Chol	Serum cholesterol in mg/dl	Number
	FBS	Fasting blood sugar > 120 mg/dl	Number (0 = false, 1 = true)
	RestECG	Resting electrocardiographic results	Number (0, 1, 2)
	Thalach	Maximum heart rate achieved	Number
	ExAng	Exercise induced angina	Number (0 = no, 1 = yes)
	OldPeak	ST depression induced by exercise relative to rest	Number
	Slope	The slope of the peak exercise ST segment	Number (0, 1, 2)
	CA	Number of major vessels colored by fluoroscopy	Number (0, 1, 2, 3, 4)
	Thal	Thalassemia	Number (1, 2, 3)
Label	Target	Heart disease status	Number (0 = not likely to have heart disease, 1 = likely to have heart disease)

Method

We will be classifying the dataset by testing the results of 6 models which are Logistic Regression, Single Layer Perceptron, Support Vector Machine (SVM), Multinomial Naive Bayes, Random Forest Tree, and K-Nearest Neighbor (K-NN). Logistic regression is an appropriate predictive regression analysis for binary classification. Furthermore, it is used to describe data and to explain the relationship between a dependent variable and its independent variables. [4]. Single-layer perceptron (SLP) is the simplest type of artificial neural network which is a feed-forward network that is based on a threshold transfer function that can only classify linearly separable cases with a binary target (1, 0). [5]. In addition, during the feed-forward network, it would also apply an activation function that uses a hard limit function. Support Vector Machine (SVM) is an algorithm that is used to find a hyperplane in an N-dimensional space that distinctly classified the data points. SVM can be used for both regression and classification tasks, but it's more broadly used to solve classification problems [6]. Multinomial Naive Bayes implements the naive Bayes algorithm for multinomial distributed data and is widely used for text classification tasks. But this algorithm also works well for binary classification tasks [7]. Random forest classifier consists of a large number of individual

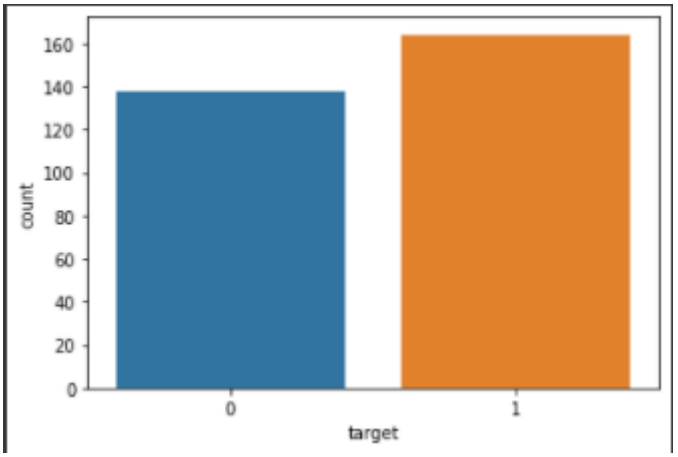
decision trees that operate as an ensemble. Each tree will output a class prediction and the class with the most votes wins [8]. The K-Nearest Neighbor algorithm assumes that similar data points are close to each other. This algorithm classifies a data point by calculating the distance between that data point and the K number of data points closest to it [9]. The data point will be classified as the class with the most data points close to it.

1. Pandas Profiling Report

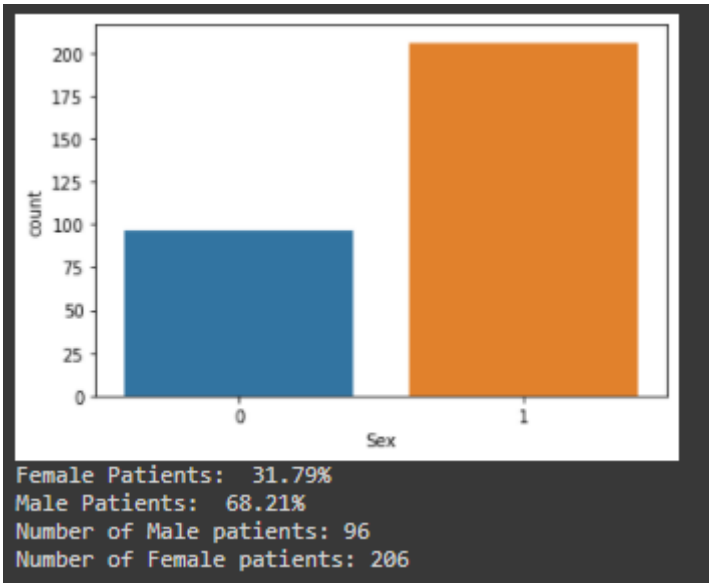
Based on the report generated by the pandas profiling library, it shows that the variables of age, sex, cp, restecg, thalach, exang, oldpeak, slope, thal, and target has high correlation however it also shows that the variables trestbps, chol, fbs and ca has low correlation. In addition to that, it also shows that our dataset has a duplicate row, so we will have to remove the duplicate data. Furthermore, it also shows that we do not have any missing values in our dataset.

2. Exploratory Data Analysis (EDA)

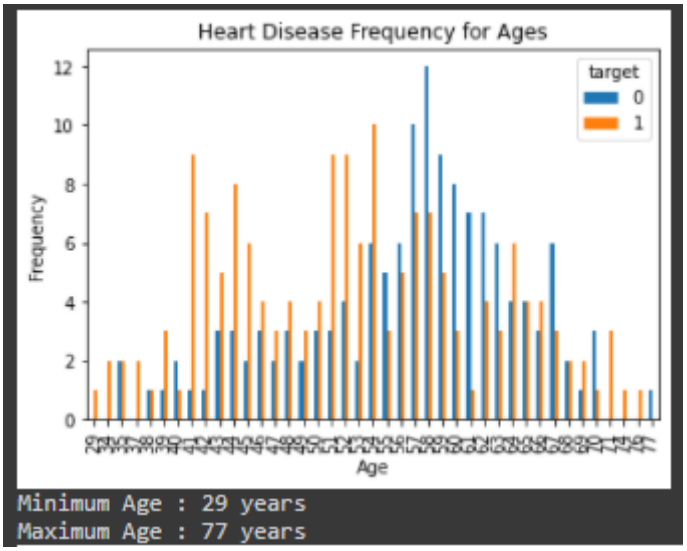
First, we will do EDA to analyze the dataset and the correlation between all of the features to omit certain features that have a low correlation to the target of our experiment. First, we will check how many rows have the target of 0 and 1. The analysis is shown below:



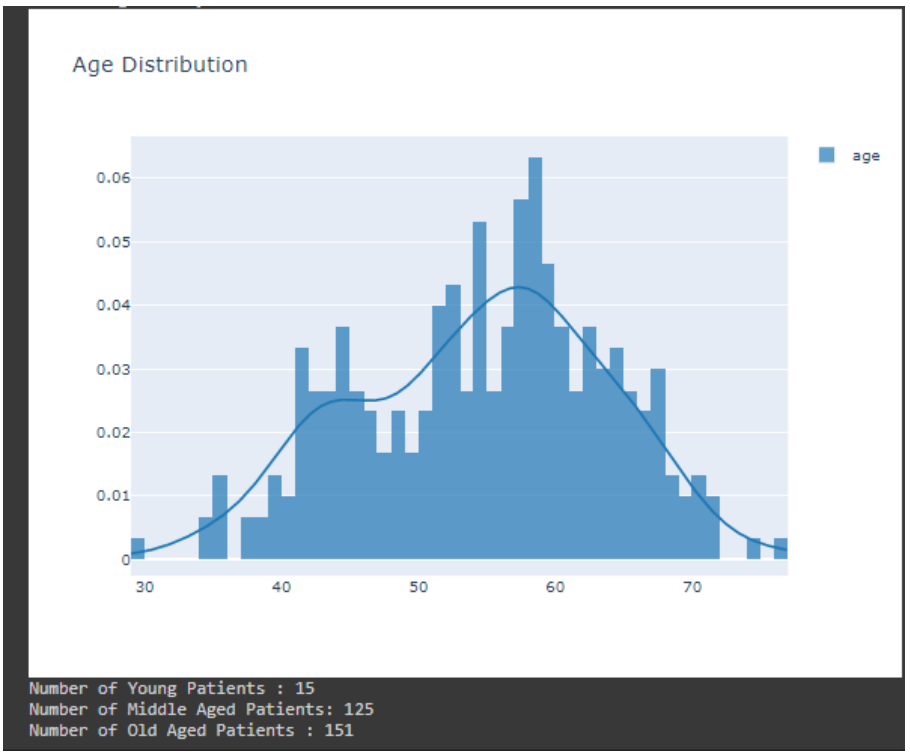
Then we will count the number of female and male patient and the result is shown below:



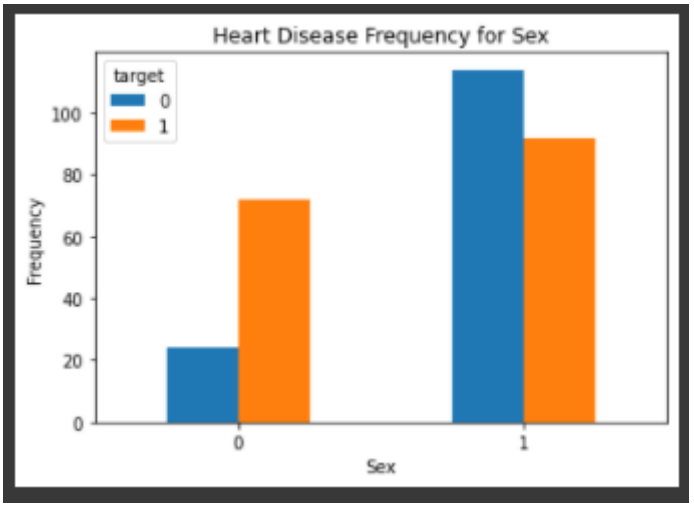
Then we will check the frequencies of the heart disease for all ages and the result is shown below:



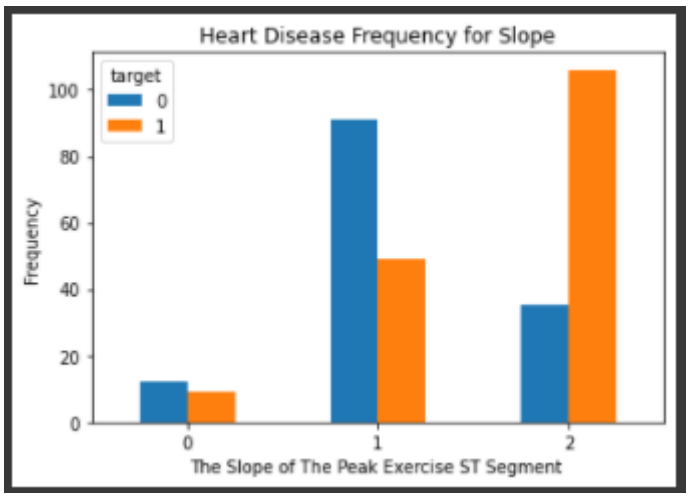
And the age distribution is shown below:



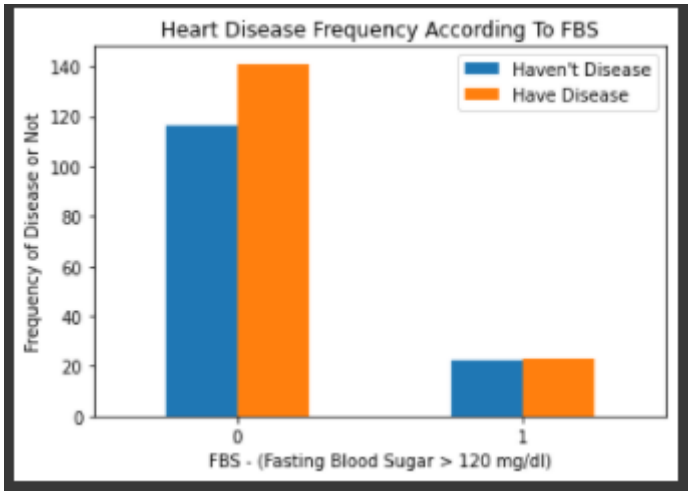
Then we check the frequencies of the heart disease on each gender and the result is shown below:



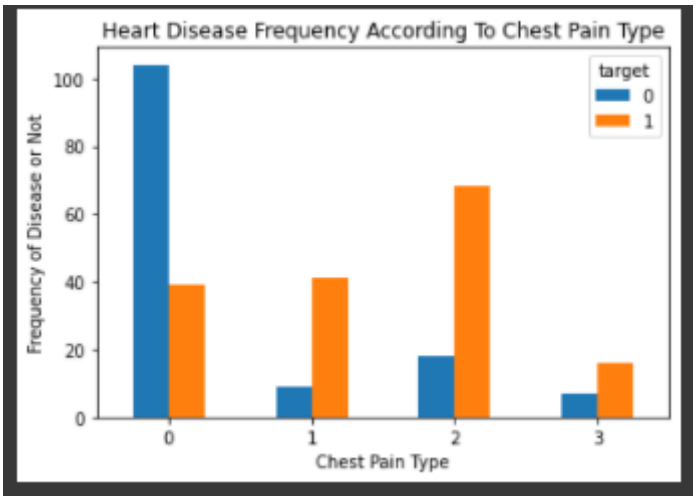
Then we check the frequencies of heart disease for each slope and the result is shown below:



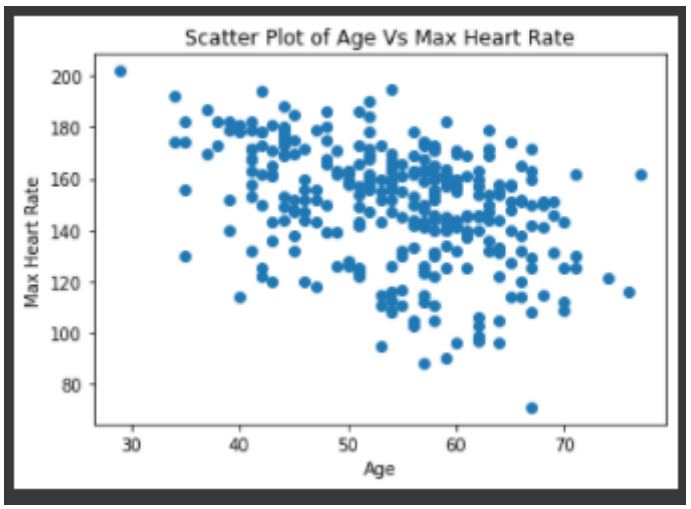
Then we check the frequencies of heart disease according to the Fasting Blood Sugar (FBS) and the result is shown below:



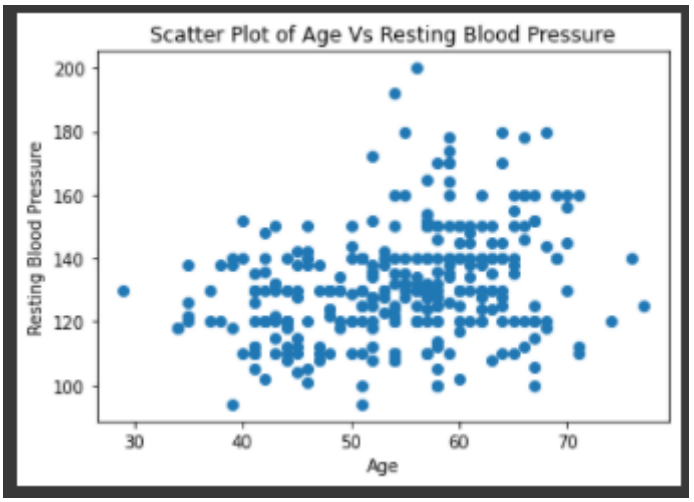
Then we check the frequencies of heart disease according to the Chest Pain Type and the result is shown below:



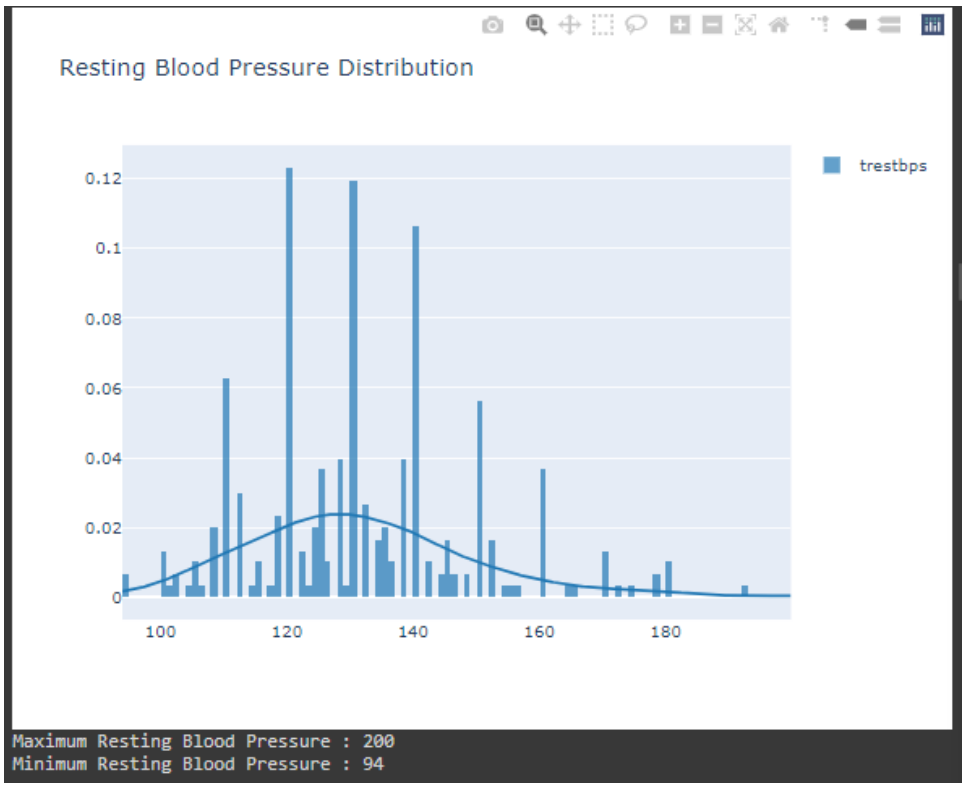
Then we check the correlation of age to maximum heart rate and the result is shown below:



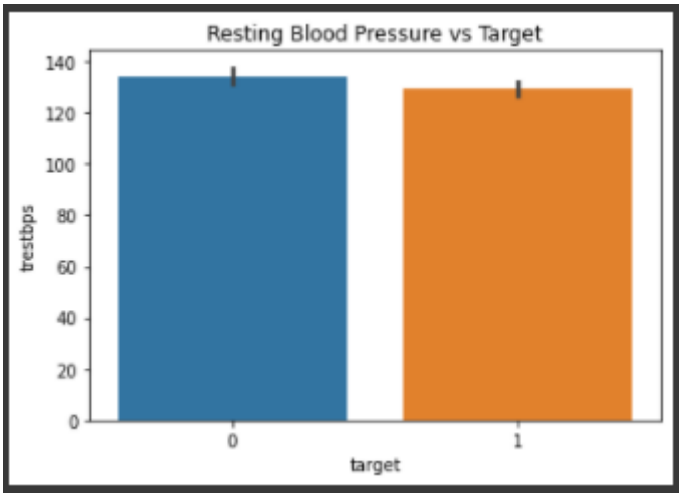
Then we check the correlation of age to Resting Blood Pressure and the result is shown below:



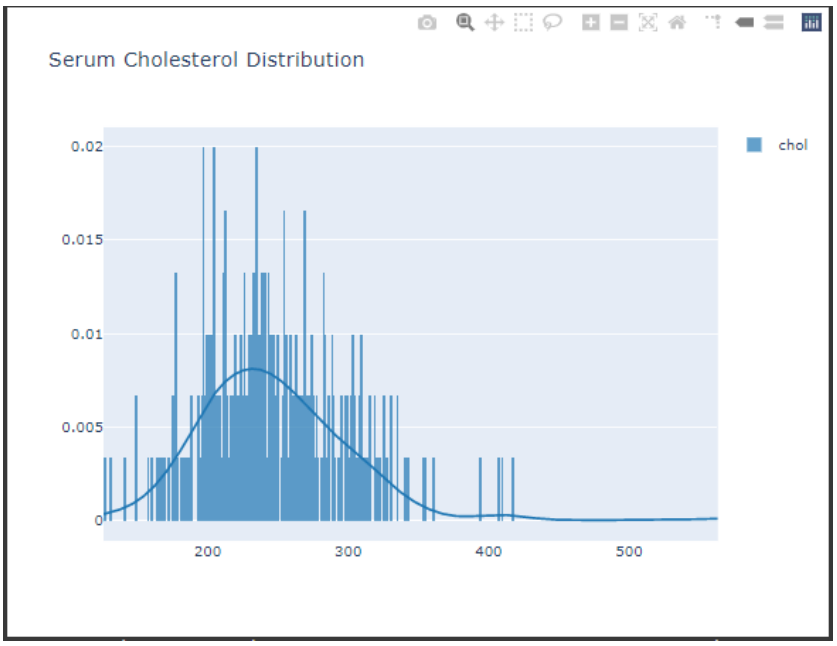
The Resting Blood Pressure distribution is shown below:



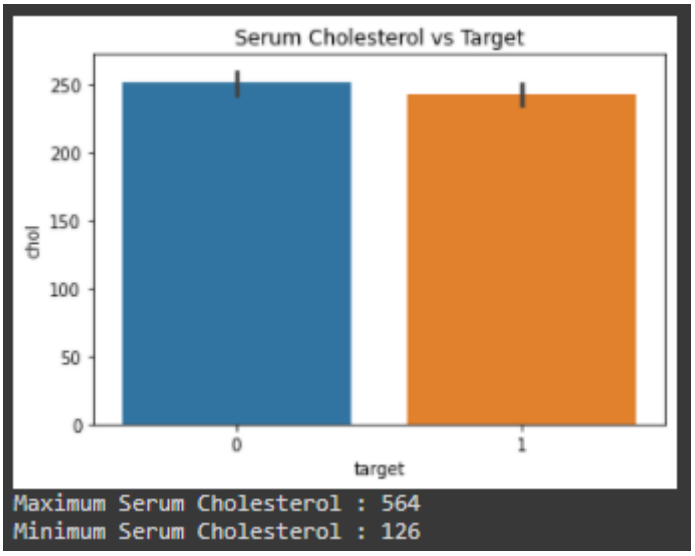
Then we will check the average value of Resting Blood Pressure on each Target and the result is shown below:



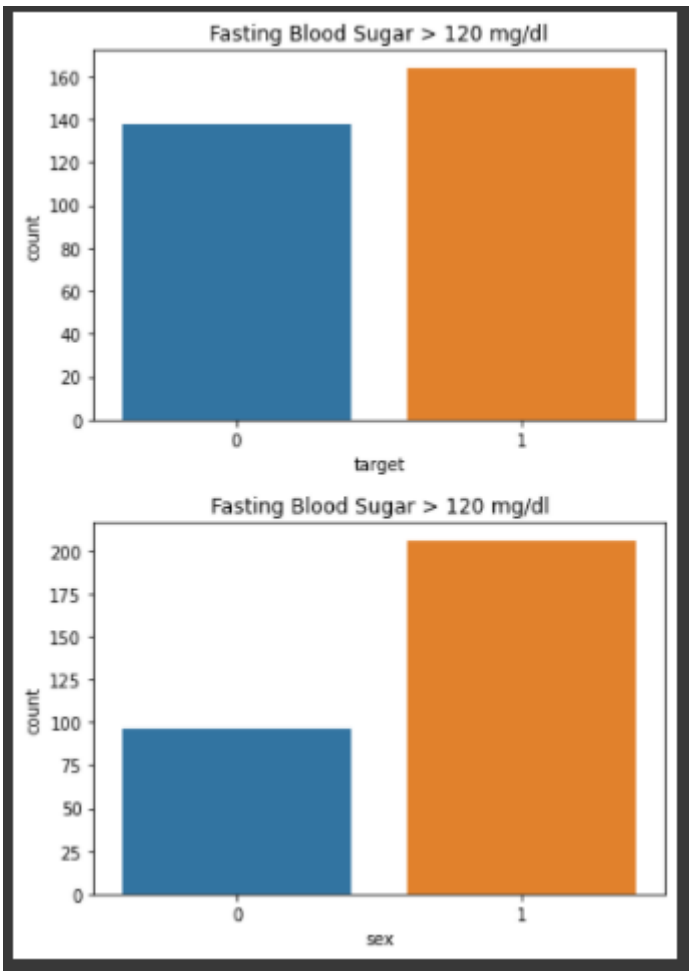
The Serum Cholesterol Distribution is shown below:



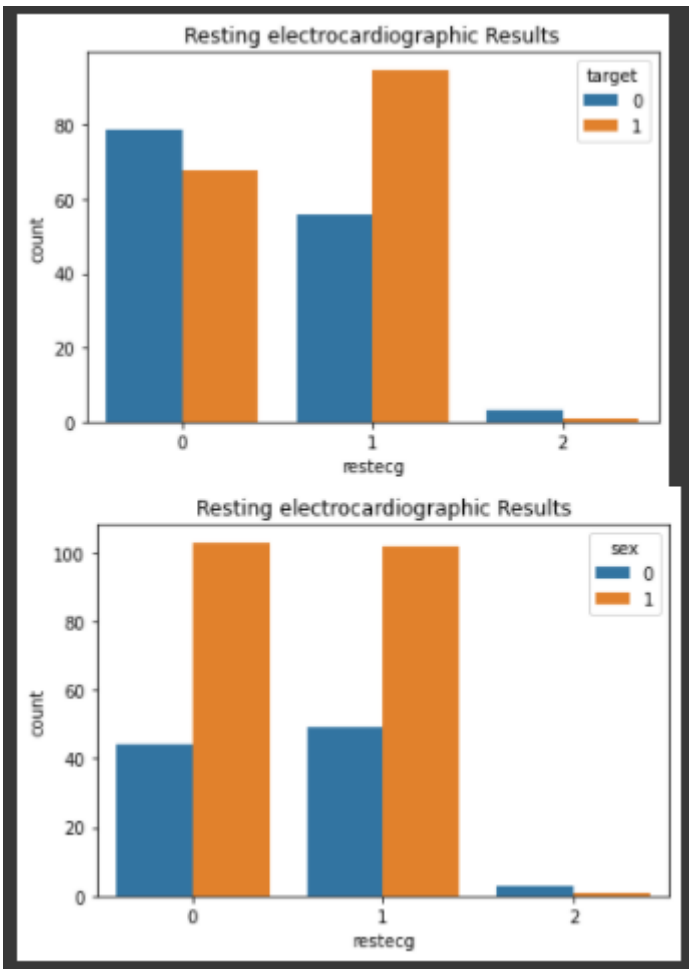
Then we will check the average value of Serum Cholesterol for each Target and the result is shown below:



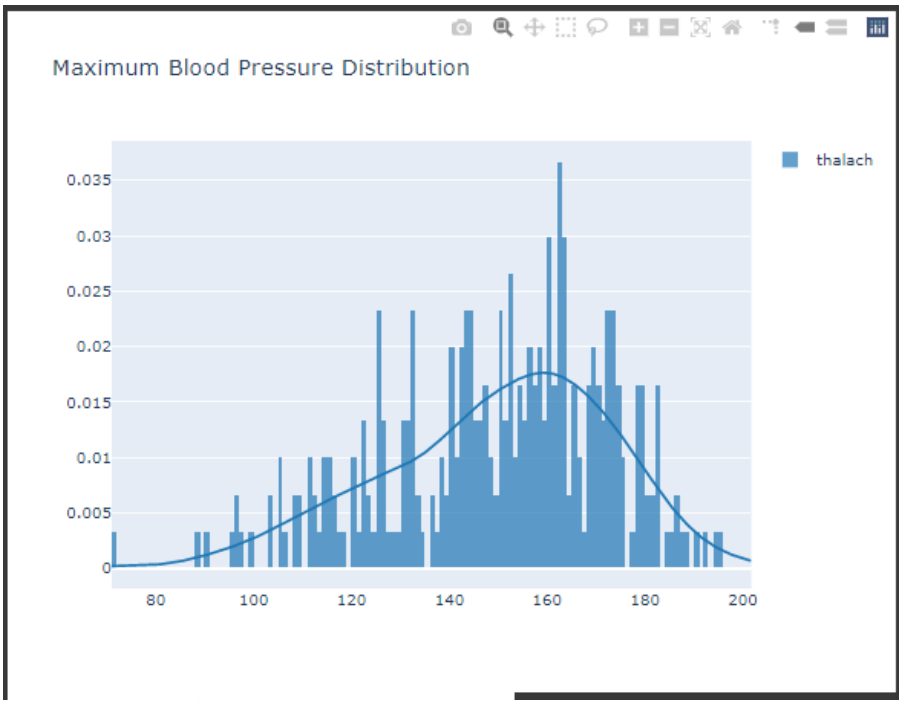
Then we check the average value of the Fasting Blood Sugar on each Target and for each Gender and the results are shown below:



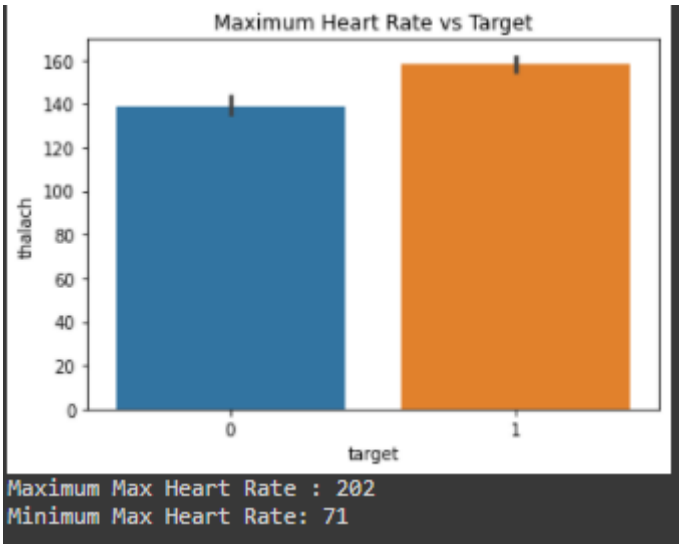
Then we will check the frequencies of each Target and each Gender for every Resting Electrocardiographic Results and results are shown below:



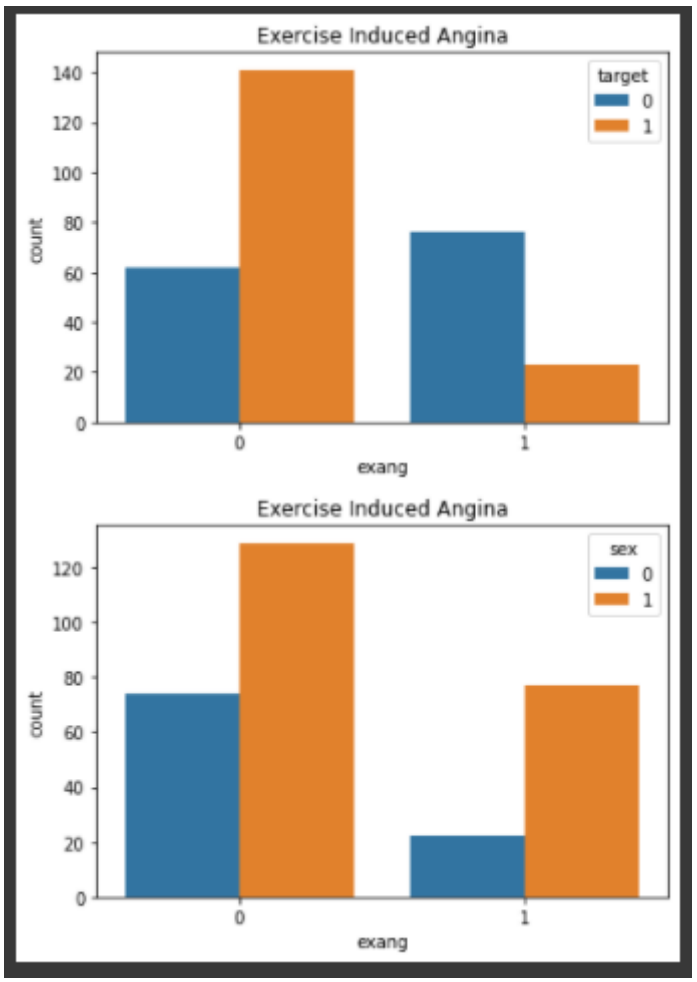
The Maximum Blood Pressure distribution is shown below:



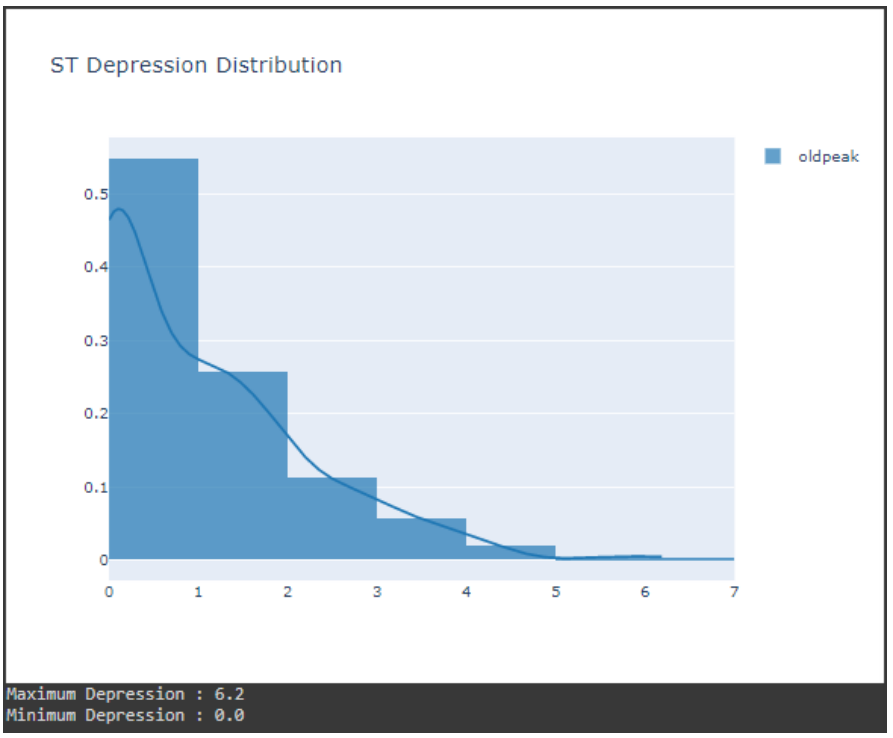
Then we will check the average value of Maximum Heart Rate for each target and the result is shown below:



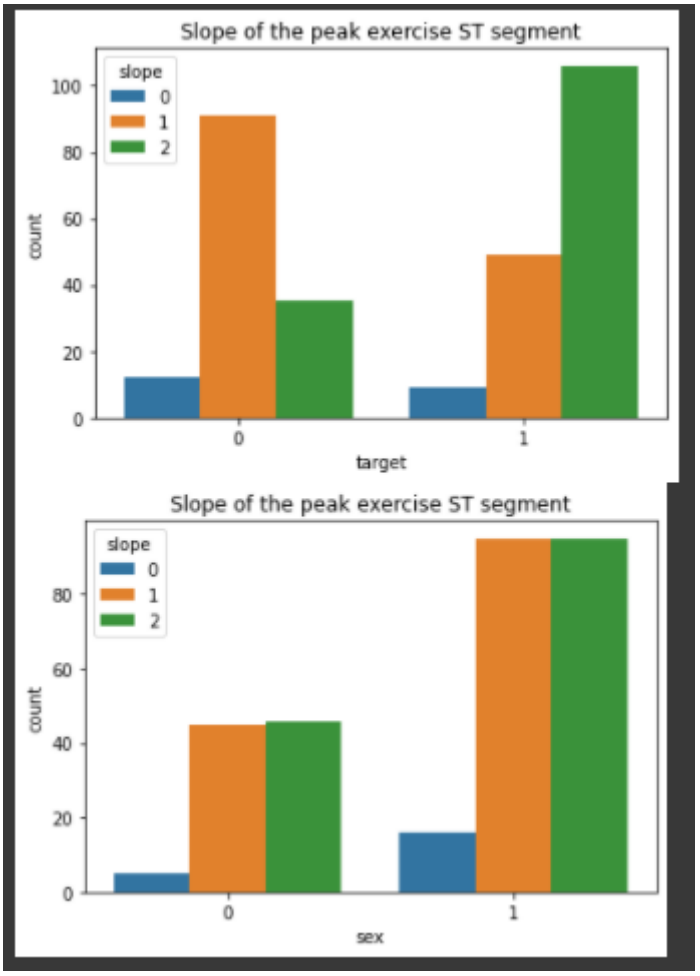
Then we will check the frequencies of each Target and each Gender for each value of Exercise-Induced Angina and the results are shown below:



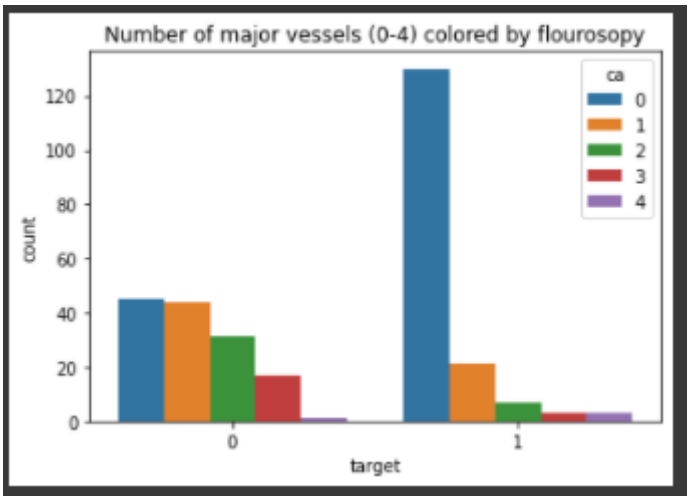
The ST Depression distribution is shown below:



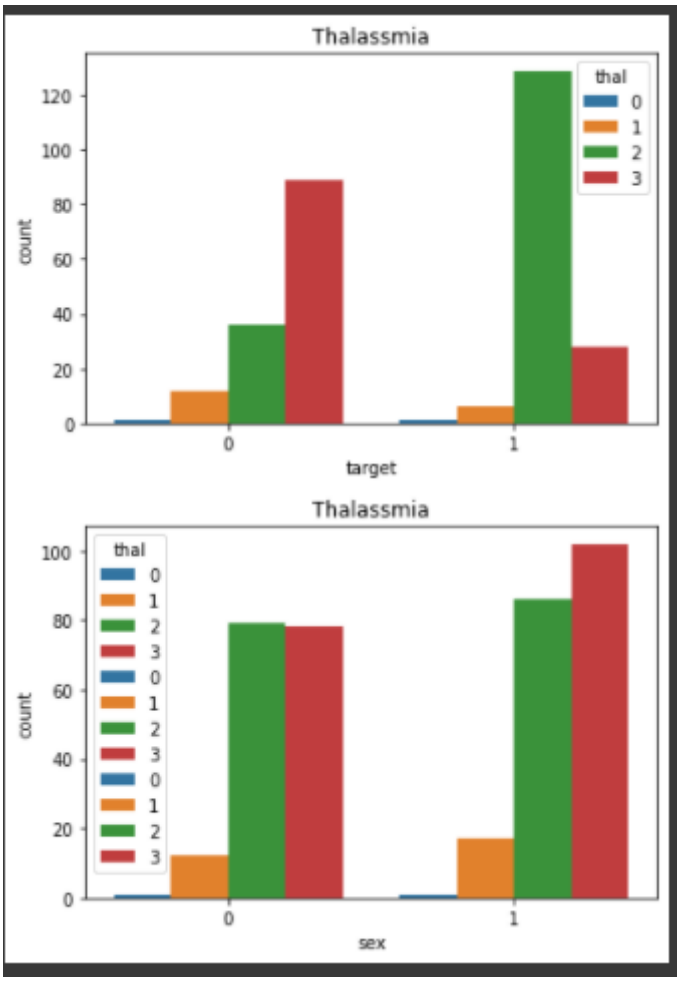
The frequencies of each value of the Slope of the peak exercise ST-segment for each Target and for each Gender is shown below:



The frequencies of each number of major vessels colored by fluoroscopy for each Target is shown below:



The frequencies of each value of Thalassemia for each Target and for each Gender is shown below:



3. Data Preprocessing

3.1. Drop Duplicates

From Pandas Profiling, we found out that there is duplicate data in our dataset. Therefore we drop the duplicate data by using pandas 'drop_duplicates()' method. The result is shown below:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
298	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
299	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
300	68	1	0	144	193	1	1	141	0	3.4	1	2	3	0
301	57	1	0	130	131	0	1	115	1	1.2	1	1	3	0
302	57	0	1	130	236	0	0	174	0	0.0	1	1	2	0

302 rows × 14 columns

3.2. Feature Selection

From our EDA, we can see that a few variables do not have a high correlation, so we can now perform feature extraction. We will be omitting the trestbps, chol, fbs, and ca feature and we will be using the age, sex, cp, restecg, thalach, exang, oldpeak, slope, and thal feature and the target as our label.

3.3. Splitting Dataset

We use the 'train test split' method from sklearn.model_selection to split our data for training and testing purposes. We split 80% of our data for training and 20% for testing.

3.4. Normalizing Dataset

After splitting the dataset, we took the feature data then normalized it using the MinMaxScaler method from sklearn.preprocessing.The training data after being normalized is shown below:

	age	sex	cp	restecg	thalach	exang	oldpeak	slope	thal
0	0.674419	0.0	0.000000	0.5	0.790323	1.0	0.290323	0.5	0.666667
1	0.697674	1.0	0.000000	0.5	0.274194	1.0	0.032258	0.5	1.000000
2	0.511628	0.0	0.333333	0.0	0.661290	0.0	0.209677	0.5	0.666667
3	0.604651	1.0	0.000000	0.5	0.717742	1.0	0.225806	1.0	1.000000
4	0.767442	0.0	0.666667	0.0	0.717742	0.0	0.258065	0.5	1.000000
...	...	...	...	...	...	...	...	...	...
236	0.534884	1.0	0.000000	0.5	0.782258	1.0	0.000000	1.0	1.000000
237	0.627907	1.0	0.000000	0.0	0.435484	1.0	0.580645	0.5	0.666667
238	0.534884	1.0	0.000000	0.5	0.620968	0.0	0.064516	0.5	0.333333
239	0.860465	0.0	0.000000	0.5	0.435484	0.0	0.258065	0.5	0.666667
240	0.860465	0.0	0.333333	0.5	0.733871	0.0	0.064516	1.0	0.666667

241 rows × 9 columns

And the testing data after being normalized is shown below:

	age	sex	cp	restecg	thalach	exang	oldpeak	slope	thal
0	0.558140	0.0	0.666667	0.5	0.814516	0.0	0.000000	1.0	0.666667
1	0.162791	1.0	0.333333	0.5	0.491935	0.0	0.000000	0.5	0.333333
2	0.255814	1.0	0.000000	0.0	0.612903	1.0	0.000000	0.5	1.000000
3	0.255814	1.0	0.333333	0.0	0.798387	0.0	0.000000	1.0	0.666667
4	0.511628	1.0	0.000000	0.0	0.258065	1.0	0.258065	0.0	1.000000
...	...	...	...	...	...	...	...	...	...
56	0.534884	1.0	0.000000	0.5	0.580645	1.0	0.483871	0.5	1.000000
57	0.558140	0.0	0.000000	0.0	0.604839	1.0	0.451613	0.5	0.333333
58	0.209302	1.0	0.666667	0.5	0.733871	0.0	0.306452	1.0	0.666667
59	0.023256	1.0	0.333333	0.5	0.830645	0.0	0.000000	1.0	0.666667
60	0.186047	1.0	0.333333	0.5	0.733871	0.0	0.000000	1.0	0.666667

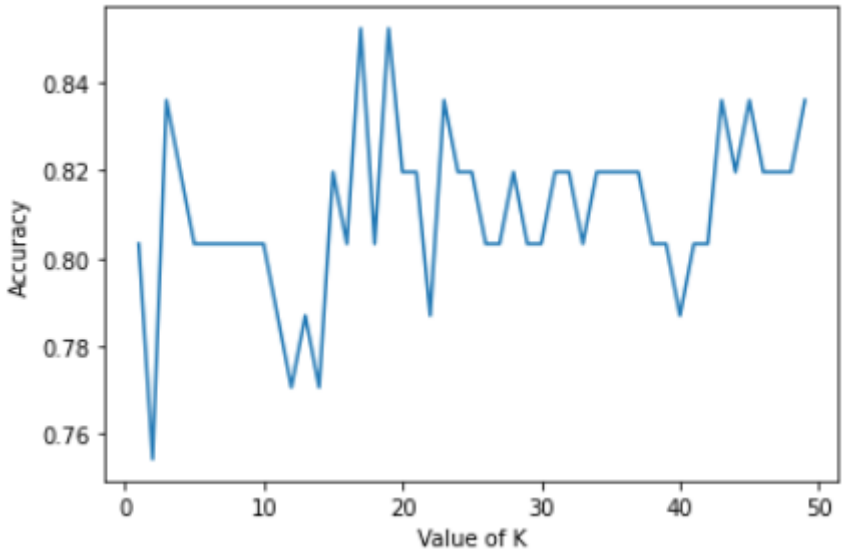
61 rows × 9 columns

4. Training and Testing the Models

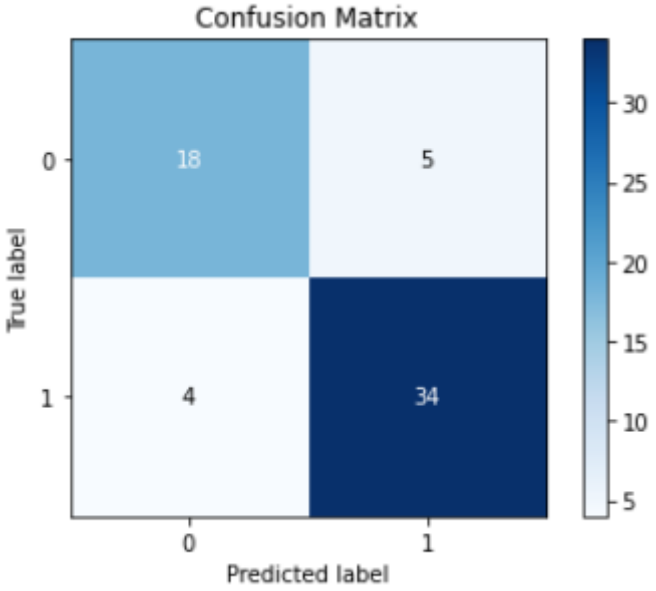
Here we train all of our models (Logistic Regression, Single Layer Perceptron, Support Vector Machine (SVM), Multinomial Naive Bayes, Random Forest Tree, and K-Nearest Neighbor (K-NN)) using the training data (x_train). Then we validate our prediction by calculating the accuracy of the prediction against the testing data (y_test). The results of these experiments can be seen in the next section.

Results

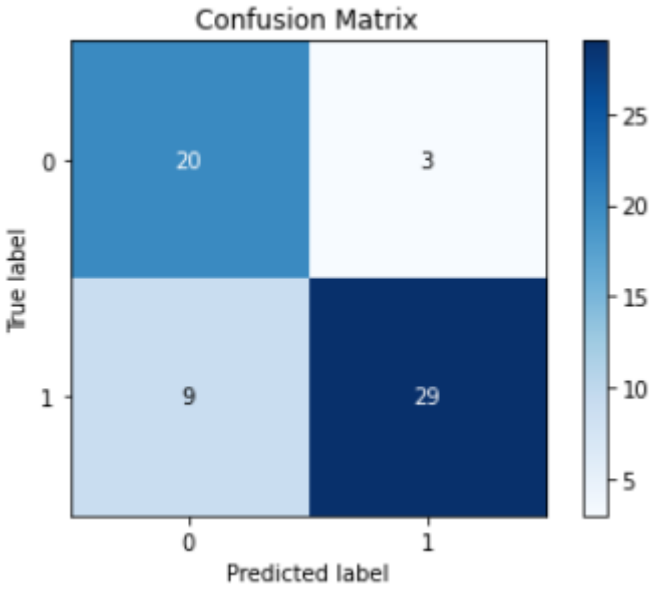
When testing the K-Nearest Neighbors (K-NN) model, first of all, we need the best k value to build the K-NN model that has the best accuracy. We tried to find the best k value in the range 1-50. The graphic below shows the accuracy of each k value in the range 1-50. We got the best k value at 17.



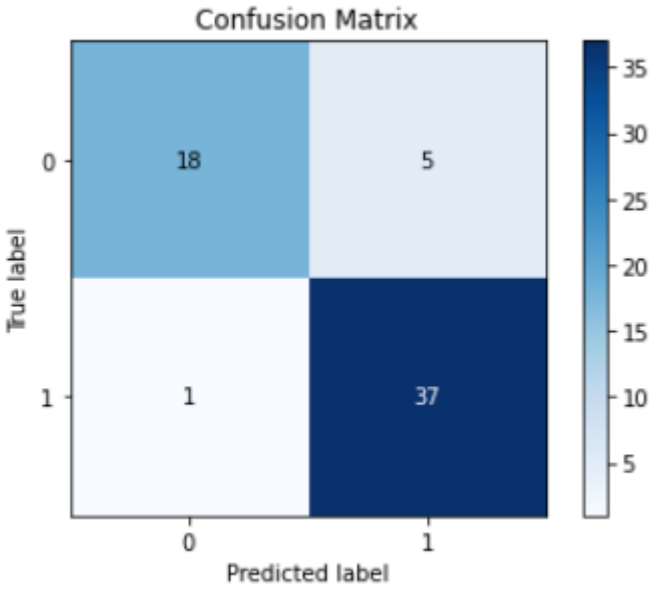
We built the K-NN model using the best k value obtained. The model is able to get an accuracy of 85.25%. Below is the confusion matrix of the K-NN model built:



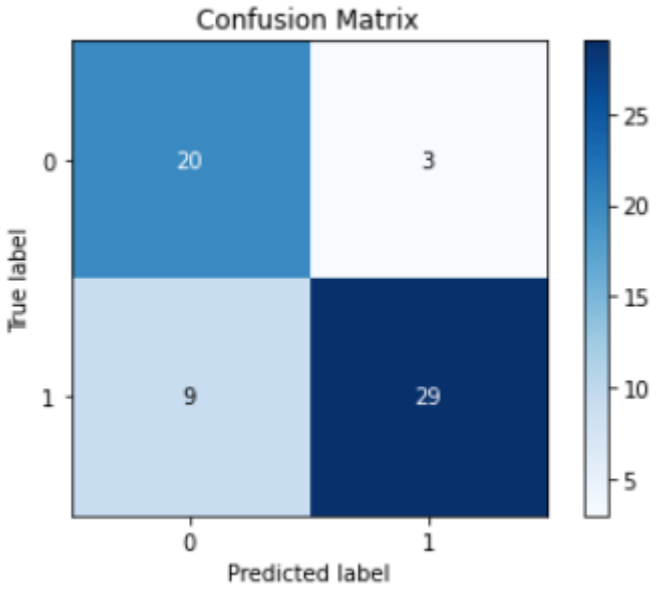
In the Logistic Regression trial, the model is able to get an accuracy of 80.33%. Below is the confusion matrix of the Logistic Regression model built:



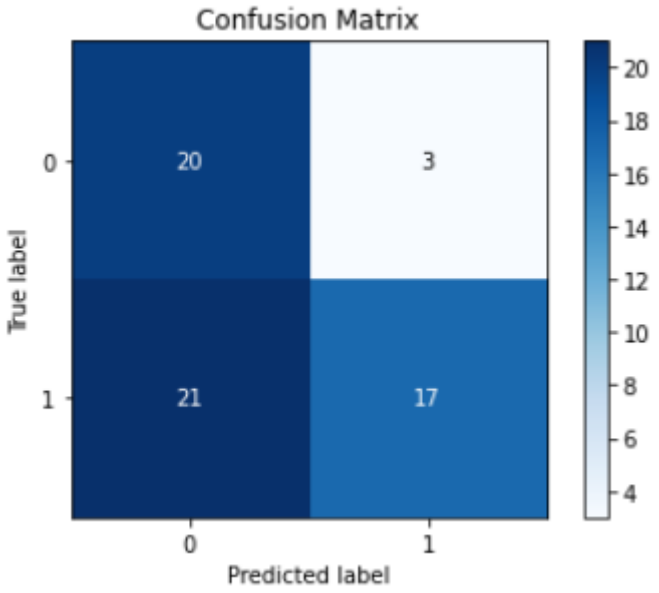
In the Multinomial Naive Bayes trial, the model is able to get an accuracy of 90.16%. Below is the confusion matrix of the Multinomial Naive Bayes model built:



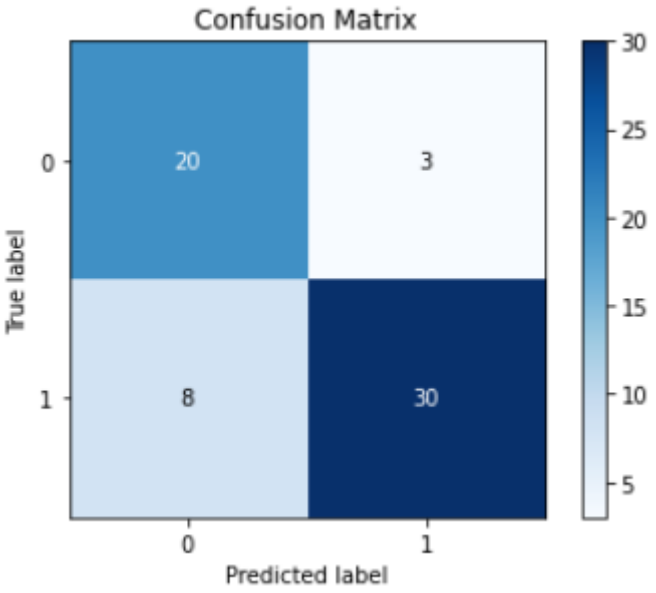
In the Random Forest Classifier trial, the model is able to get an accuracy of 80.33%. Below is the confusion matrix of the Random Forest Classifier model built:



In the Single Layer Perceptron trial, the model is able to get an accuracy of 60.66%. Below is the confusion matrix of the Single Layer Perceptron model built:



In the Support Vector Machine (SVM) trial, the model is able to get an accuracy of 81.97%. Below is the confusion matrix of the SVM model built:



From these experiments using python, we found that each model has its own accuracy results in classifying if a person is likely or not likely to have heart disease. Below is the evaluation of the testing process of each mode:

Models	Accuracy
2 Multinomial NB	90.163934
0 K-Nearest Neighbors	85.245902
4 SVM Classifier	81.967213
1 Logistic Regression	80.327869
3 Random Forest Classifier	80.327869
5 Single Layer Perceptron	60.655738

Conclusion

In conclusion, through this project, we can classify if a person is likely or not likely to have heart disease. And after that, we made 6 different models which are Logistic Regression, Single Layer Perceptron, Support Vector Machine, Multinomial Naive Bayes, Random Forest Tree, and K-Nearest Neighbor (K-NN) to classify the presence of heart disease. Overall, the best accuracy is obtained from the Multinomial Naive Bayes model with an accuracy rate of 90.16%.

From this experiment, we will be able to propose the best model which can show the most accurate prediction of classifying the presence of heart disease on a patient, which then allows doctors and hospitals to be able to further advise their patients on what they should expect. For future works, we would like to test a larger dataset from local hospitals to perform further training and testing to improve the accuracy and usefulness of our program to local hospitals.

References

1. Brownlee, J. (2020, August 19). 4 Types of Classification Tasks in Machine Learning. Retrieved June 17, 2021, from https://machinelearningmastery.com/types-of-classification-in-machine-learning/

2. Binary classification. (2021, May 09). Retrieved June 17, 2021, from https://en.wikipedia.org/wiki/Binary_classification

3. Ronit. (2018, June 25). Heart Disease UCI. Retrieved June 16, 2021, from https://www.kaggle.com/ronitf/heart-disease-uci

4. What is Logistic Regression? (2021, May 04). Retrieved June 16, 2021, from https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/what-is-logistic-regression/

5. Sayad, S. (n.d.). Artificial Neural Network - Perceptron. Retrieved June 16, 2021, from https://www.saedsayad.com/artificial_neural_network_bkp.htm

6. Gandhi, R. (2018, July 05). Support Vector Machine - Introduction to Machine Learning Algorithms. Retrieved June 17, 2021, from https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

7. 1.9. Naive Bayes. (n.d.). Retrieved June 17, 2021, from https://scikit-learn.org/stable/modules/naive_bayes.html

8. Yiu, T. (2019, August 14). Understanding random forest. Retrieved June 17, 2021, from https://towardsdatascience.com/understanding-random-forest-58381e0602d2

9. Harrison, O. (2019, July 14). Machine learning basics with the k-nearest Neighbors ALGORITHM. Retrieved June 17, 2021, from https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761