**Name:** Fares Islam Mansour

**Id:** 211001846

Data Analysis and Machine Learning on [heart disease] Dataset Report.

**Introduction:** There are many machine-learning models. These models can perform diverse tasks. This report aims to compare different models to predict whether the patient has a heart disease or not. This will be done by using different models such as (Logistic Regression, Random Forest, Gradient Boosting, Naïve Bayes, and K-Nearest Neighbors). By evaluating them, the best model will be used to predict the status of the patient.

**Steps:** Developing the model was done by performing 5 major steps, that included many steps in each one.

**1- Data Exploration and Preprocessing:**

This step is done to understand the data very well. And to clean it if there are any issues with the data. In the preprocessing step, many things are done to make the model deal with the optimum data. This includes handling missing values, and outliers, and calculating z-scores. Then, to explore the data well, exploratory data analysis (EDA) is done to gain insights into the characteristics and relationships of the data. Univariate, Bivariate, and Multivariate analyses were done as these are the best EDA types. The last step in data exploration is visualizing the data by using appropriate plots.

**2- Feature Engineering:**

This data included categorical variables, which will affect the model's accuracy, and some models can’t work with categorical variables. So, by transforming the categorical variables into numerical variables. This enhanced the predictive power of the dataset. The data didn’t need any other types of feature engineering, but rather One-Hot encoding. The function of one-hot encoding was done. But without using the one-hot encoding line in the codes. Because I wanted to replace the categorical variables with numbers I wanted, and this was done by replacing the categorical column “Heart Disease”, which has 2 categorical values inside its rows (Absence, Presence) with numerical values which are (0,1) respectively.

**3- Machine Learning Model Development:**

Based on the task and the dataset nature, the most suitable machine learning task is the classification task. After splitting the data into training and testing sets, after scaling and imputing. These sets were used to develop different machine-learning models. 9 different models were compared and evaluated to determine which one was the most accurate. The models used in the comparison were Logistic Regression, Random Forest, Gradient Boosting, Naïve Bayes, K-Nearest Neighbors, Support Vector Machine, Adaboost, Xgboost, and Decision Tree. After training all the models and evaluating them, Logistic Regression, Naïve Bayes, and Support Vector Machine (SVM) were the most accurate and their evaluation metrics were fine. The evaluation metrics used were accuracy, precision, F1 score, confusion matrix, recall, and the mean of the cross-validation score. After comparing all the models, some showed good accuracy, and some weren’t good.

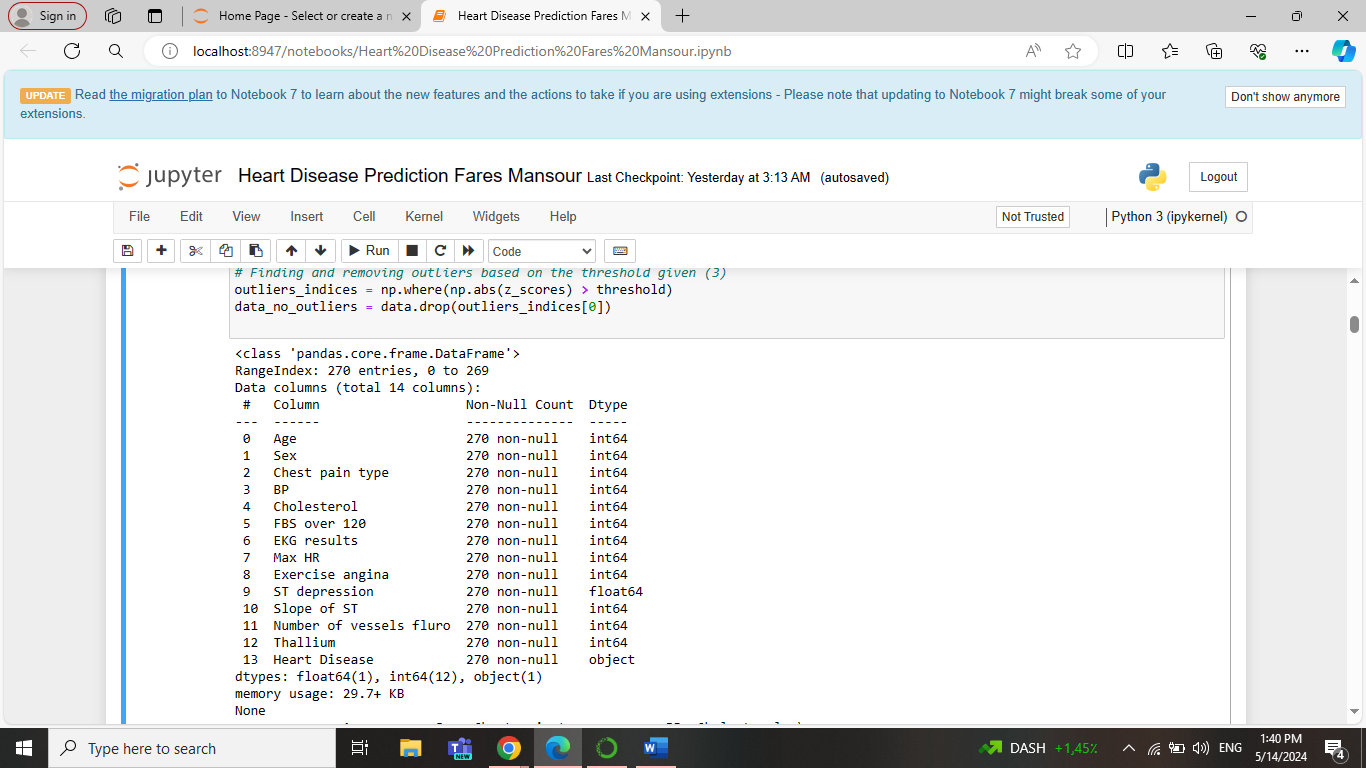
**4- Model Evaluation and Fine-tuning:**

After evaluating the models with the metrics discussed, use the hyperparameters as (n\_estimators, max\_depth, min\_samples\_split, min\_samples\_leaf). The grid search function was used to get the best hyperparameters for a model and his step is called Model fine-tuning. Fine-tuning settings in machine learning is important. It can make a big difference in how well a model works and its adaptability. But getting these settings right means testing them out carefully and understanding the problem you're trying to solve.

**5- Model Deployment:**

After finishing all the steps of the model development and interpreting the outputs, 3 models worked well. One of them was taken to build the deployment model. Logistic regression was used, and it yielded good predictions. The model deployment was done by using Streamlit web app utilities and functions. Streamlit provided a local host to try the model on and it worked well. The user needs to enter his/her age, sex, blood pressure, cholesterol, and many other features and then the model predicts whether there is a heart disease or not. Building the deployment model was easier as all the codes used to build the model were written before when putting them added to the codes that build the streamlit interface, the deployment model is done.

**Results:**

**1- Data Summary:**

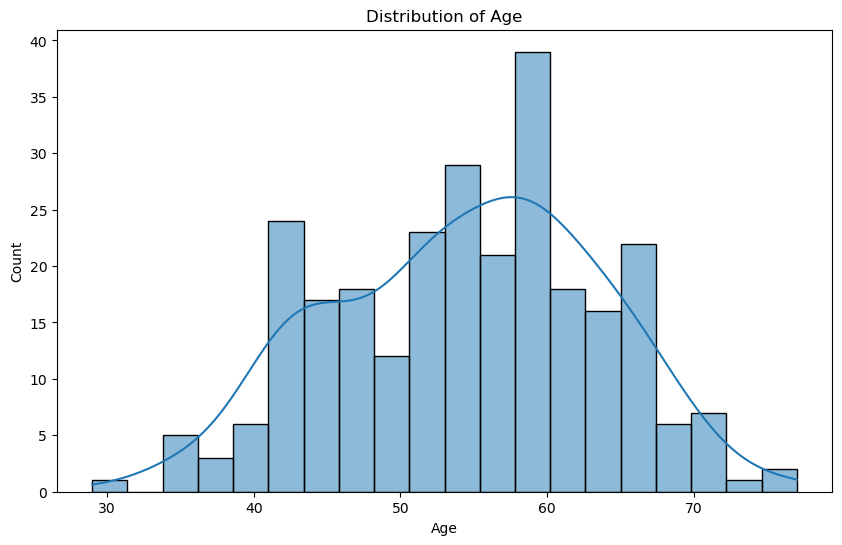
This is the first output which summarizes the data. There are 270 entries, a total of 14 columns with their indexes. No null values are present in the rows. Also, the data type of every column is defined. Lastly, the memory usage is defined.

**2- Descriptive Statistics:**

A screenshot of a computer

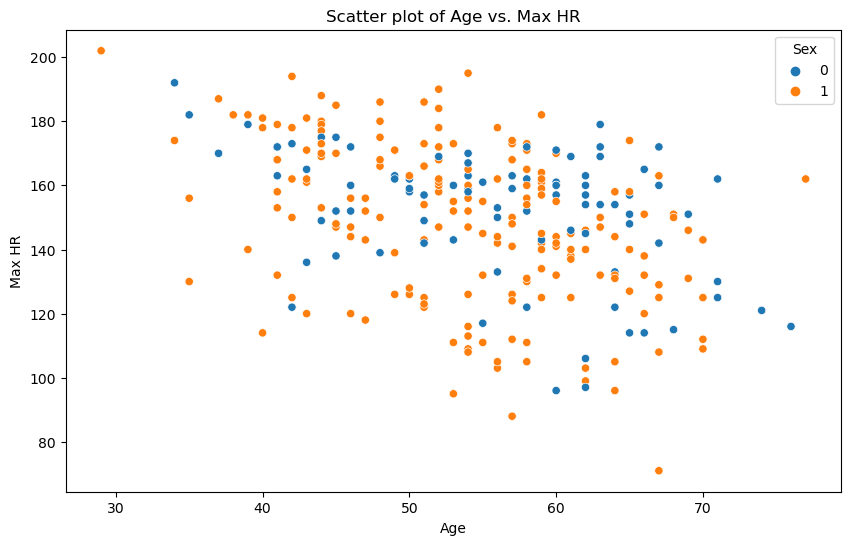
Description automatically generated

This is a sample of the output that displays the descriptive statistics of the dataset used. As the count, mean, standard deviation, the IQR ranges, and the minimum and maximum numbers.

**3- Univariate Analysis:**

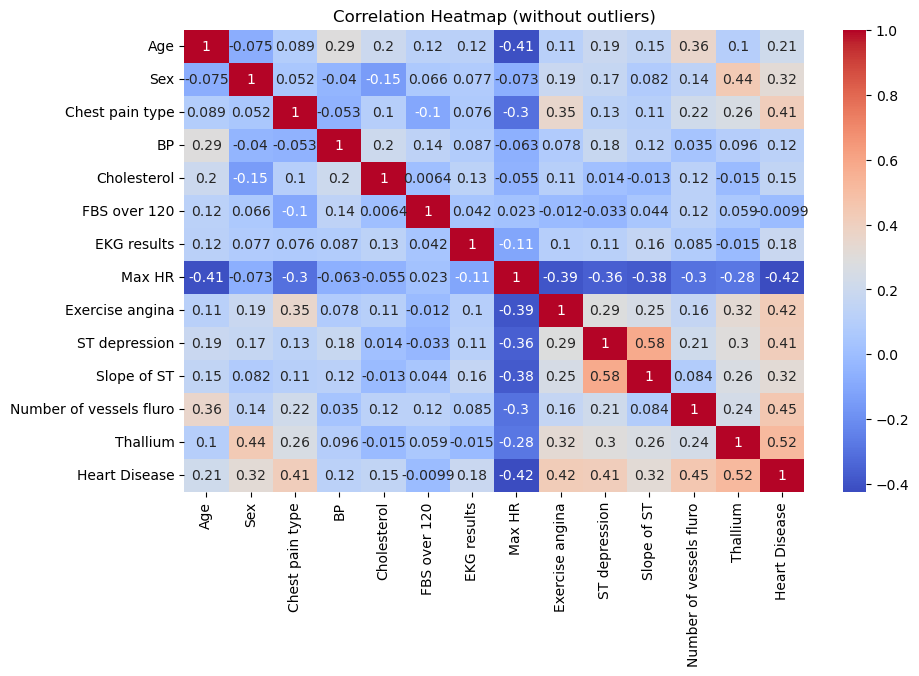
This is the first output plot from the EDA analysis, which is the univariate analysis. It shows the distribution of the age column in the dataset. This plot reveals that patients with ages between 40 and 60 are the highest. And the most common patient’s age is 60 as it has the highest counts.

**4- Bivariate Analysis:**

****

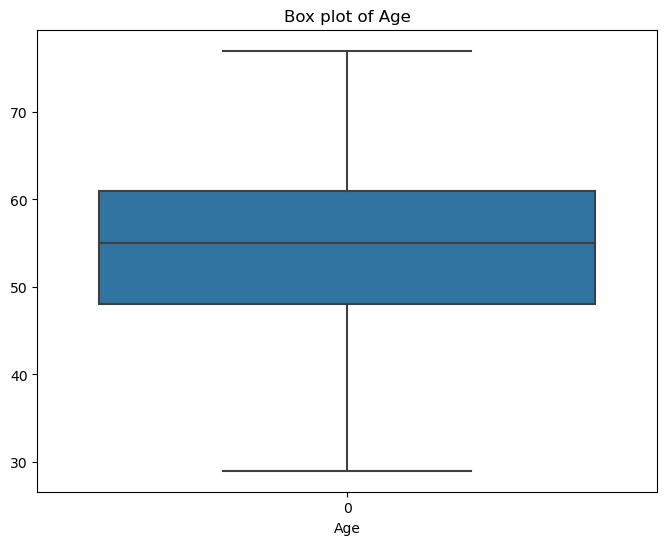
This is the second output plot from the EDA analysis, which is the bivariate analysis. It shows a scatter plot that contains the 2 columns Age and Maximum Heart Rate. They are plotted against each other, and the ball’s color is based on sex, the females are blue, and the males are orange. This plot shows that the average maximum heart rate is from 140 to 180. These numbers are found with patients aged between 40 to 60.

**5- Multivariate Analysis:**



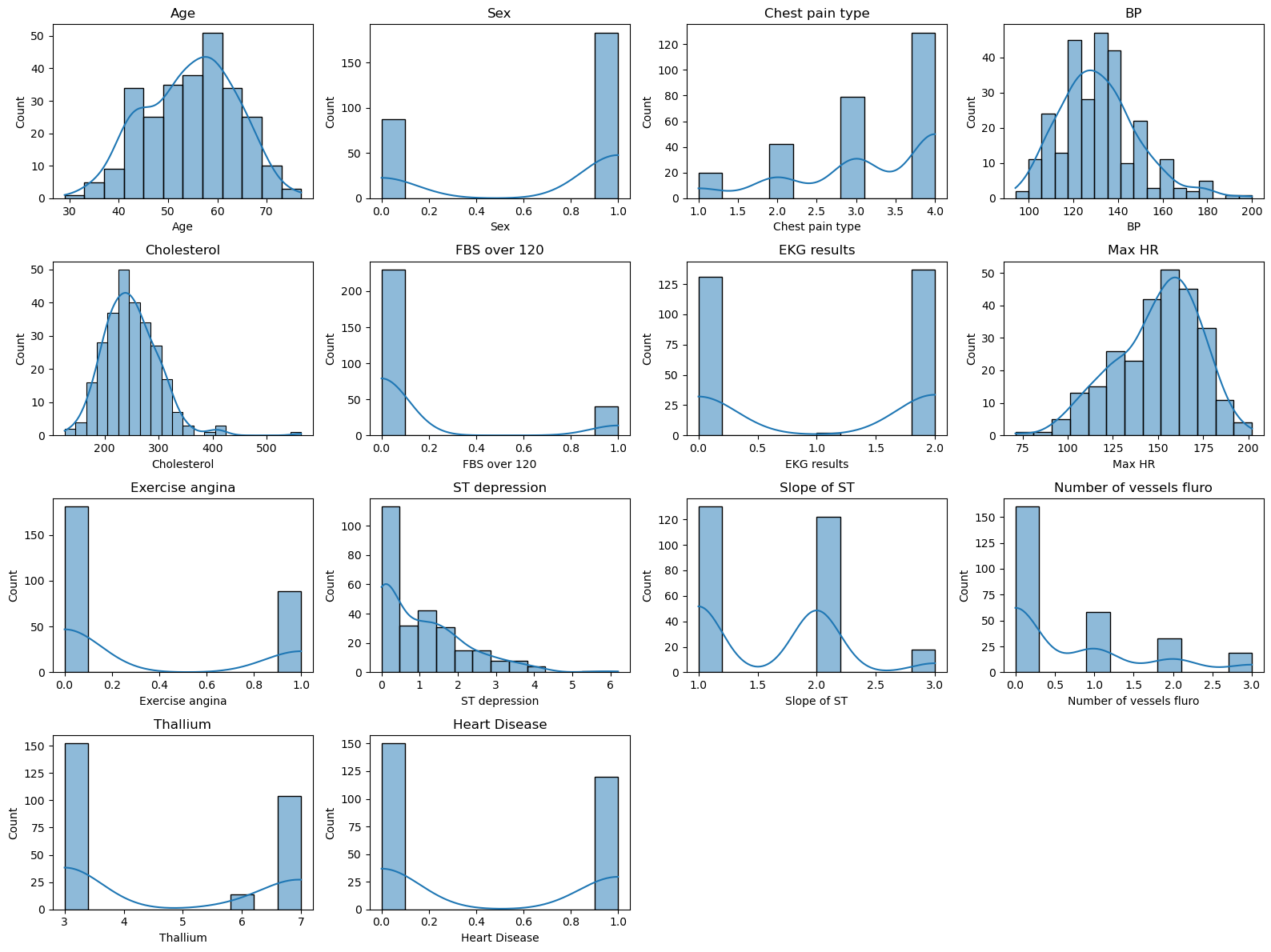
This is the last output plot from the EDA analysis, which is the multivariate analysis. It shows a heatmap plot, that plots all the data’s columns against each other. If the correlation between the variables increases, the color of the box becomes darker, the light boxes indicate low correlation.

**6- Box Plot for Single Numerical Feature:**



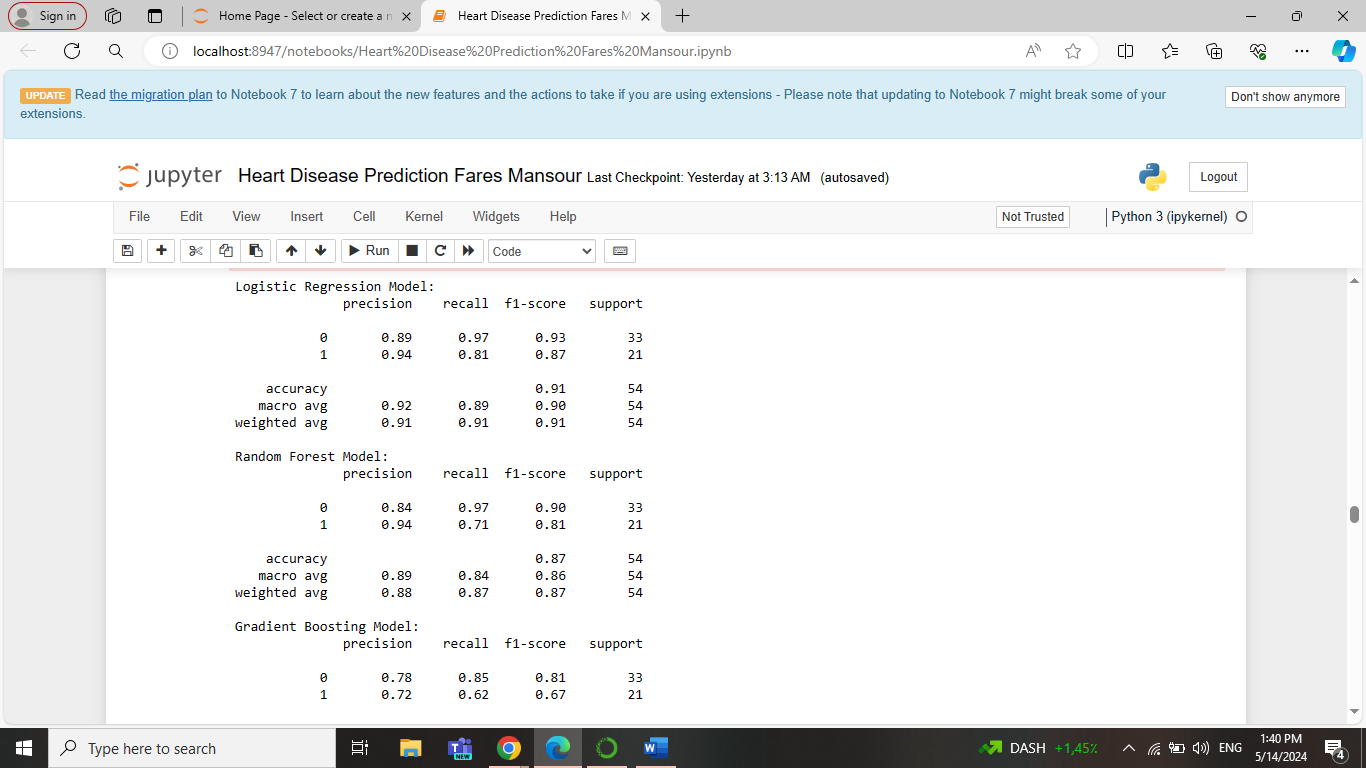
This is a box plot for a single numerical feature which is age, it shows the most ages are in the 50-60 region.

**7- Histograms for All the numerical features:**

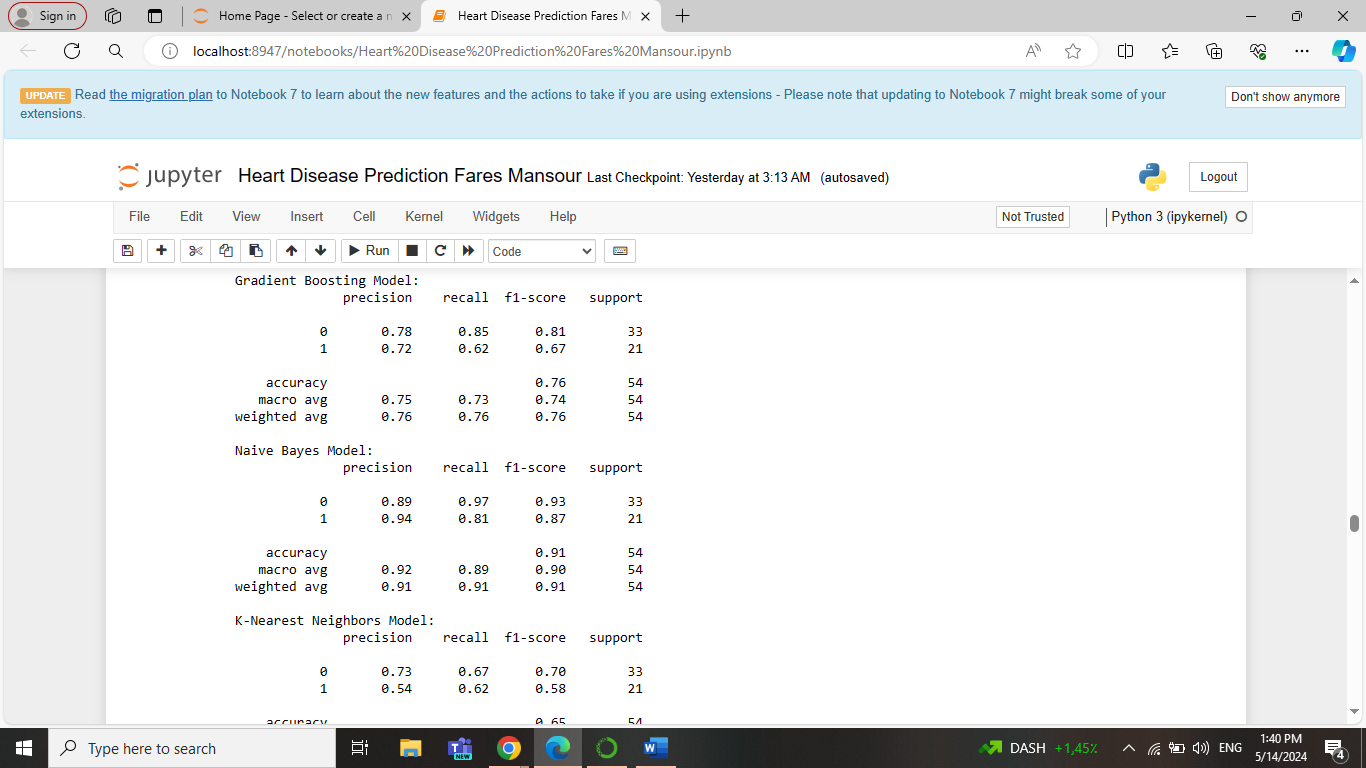


This histogram shows the distributions of all the numerical features of the data. Taking the chest pain type as an example, about 120 patients have chest pain type 4. And regarding the heart disease column, about 150 don’t have a heart disease.

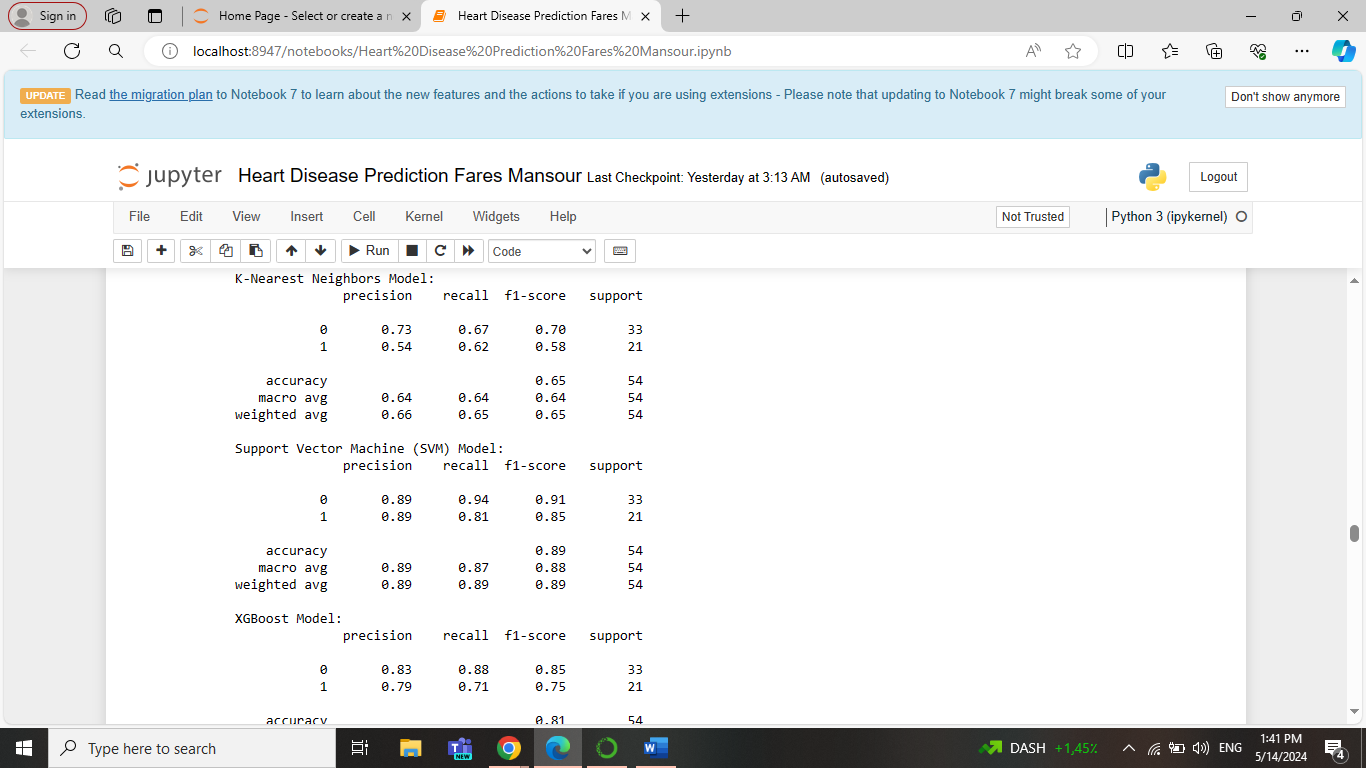
**8- Classification Reports for the models:**

****

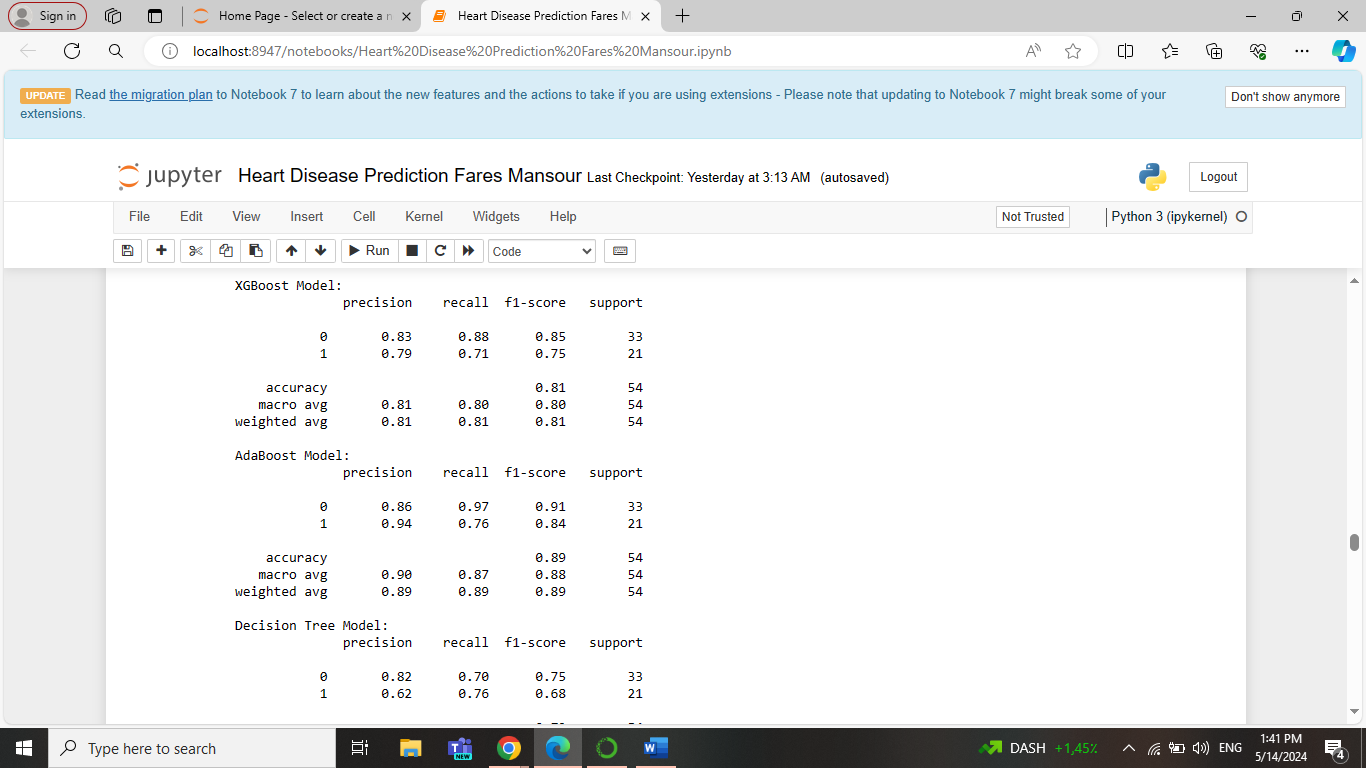
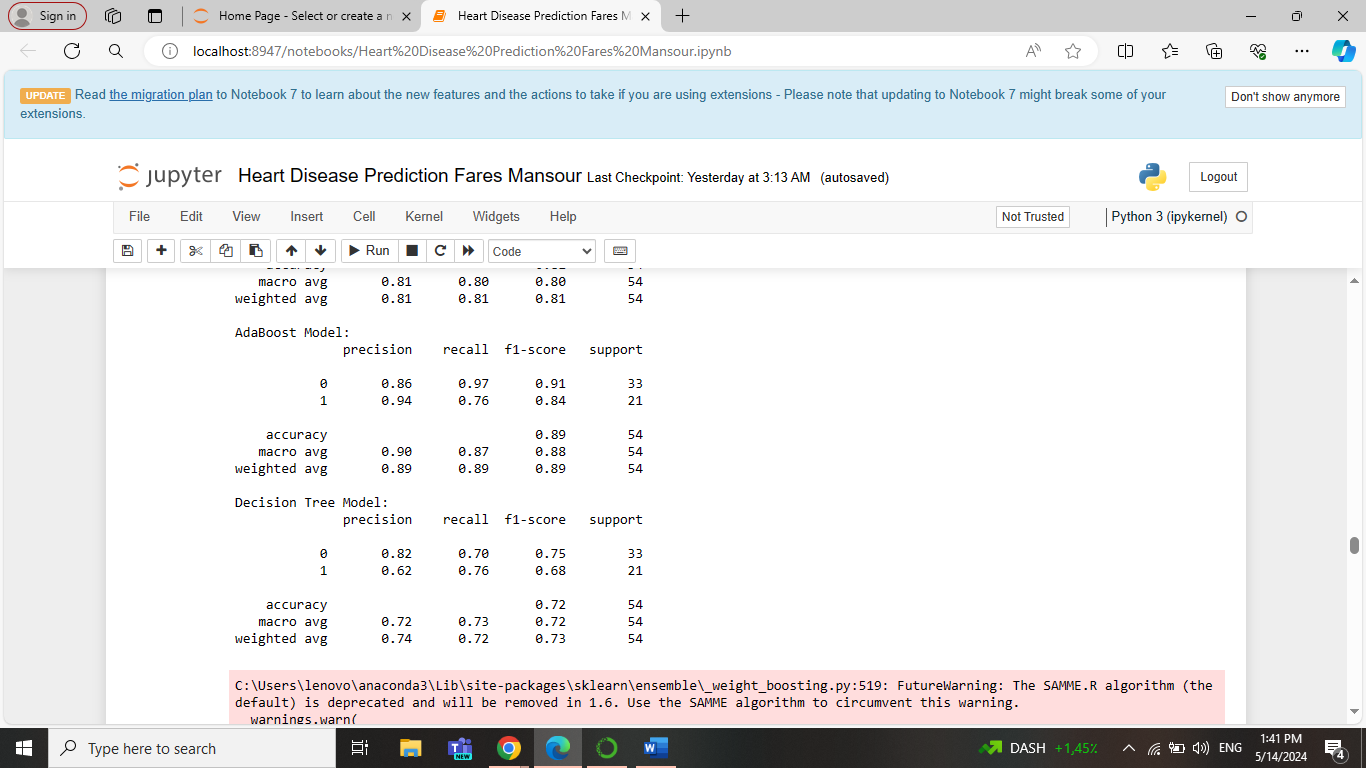
These are the outputs for the classification report for Logistic Regression and Random Forest models. The Precision, recall, and f1-score are better in Logistic Regression than Random Forest. Precision means that out of all the instances, your model predicts as positive, how many are correct. Recall is about capturing all the positive instances. It measures how many of the actual positive instances your model can identify. And the F1 score is a way to balance precision and recall. It's the harmonic mean of precision and recall. It gives you a single number that represents both precision and recall.



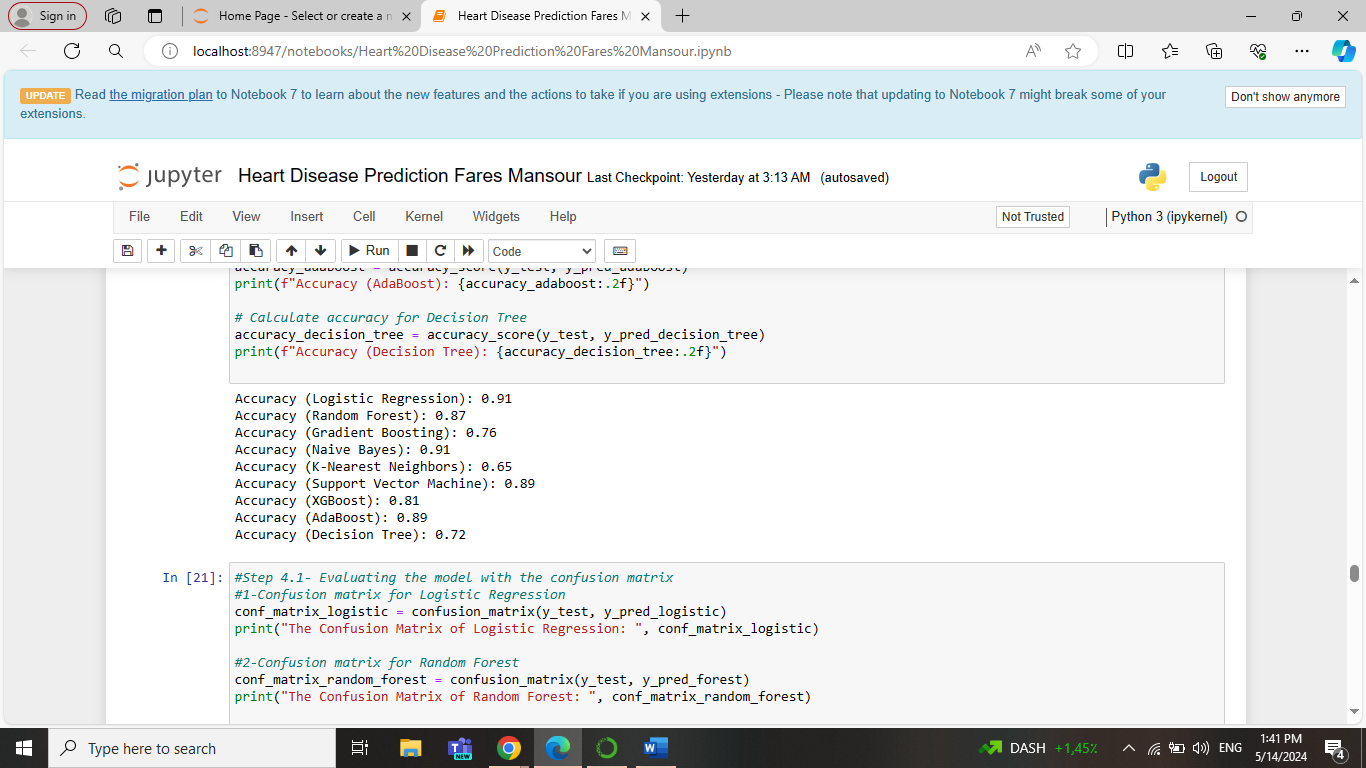
These are the outputs for the classification report for Gradient Boosting and Naïve Bayes models. The Precision, recall, and f1-score are better in Naïve Bayes than Gradient Boosting.



These are the outputs for the classification report for K-Nearest Neighbors and Support Vector Machine models. The Precision, recall, and f1-score are better in the Support Vector Machine than in K-Nearest Neighbors.

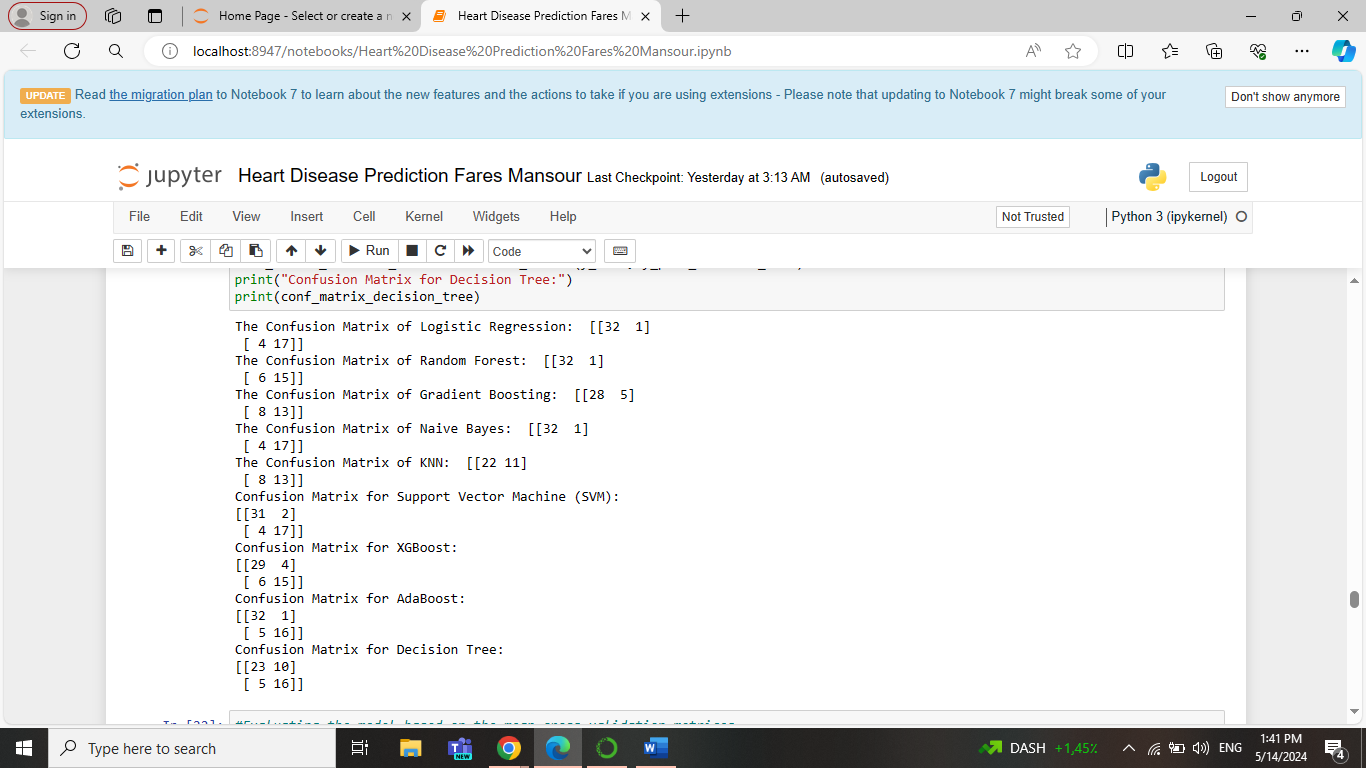


The best one among the last 3 models (XGBoost, AdaBoost, Decision Tree), is the AdaBoost model.

**9- Models Accuracies:**

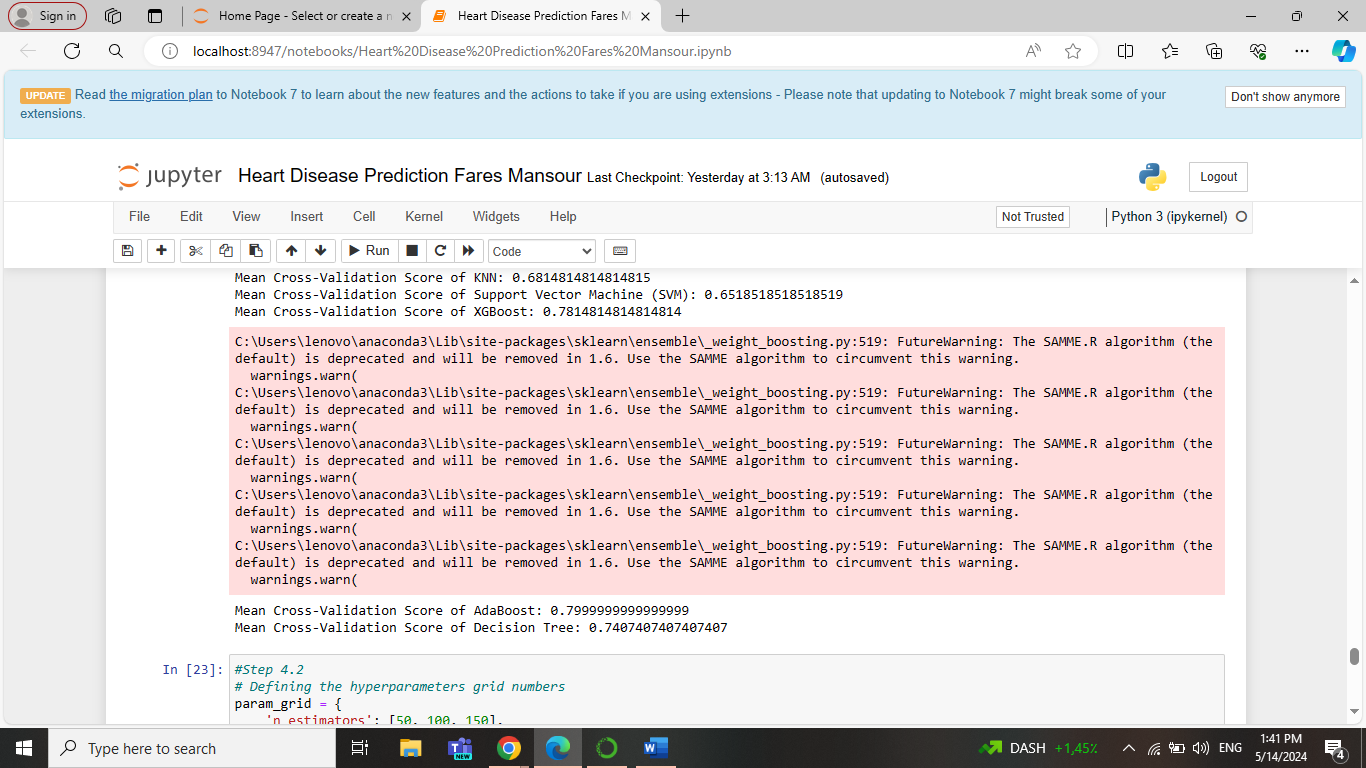
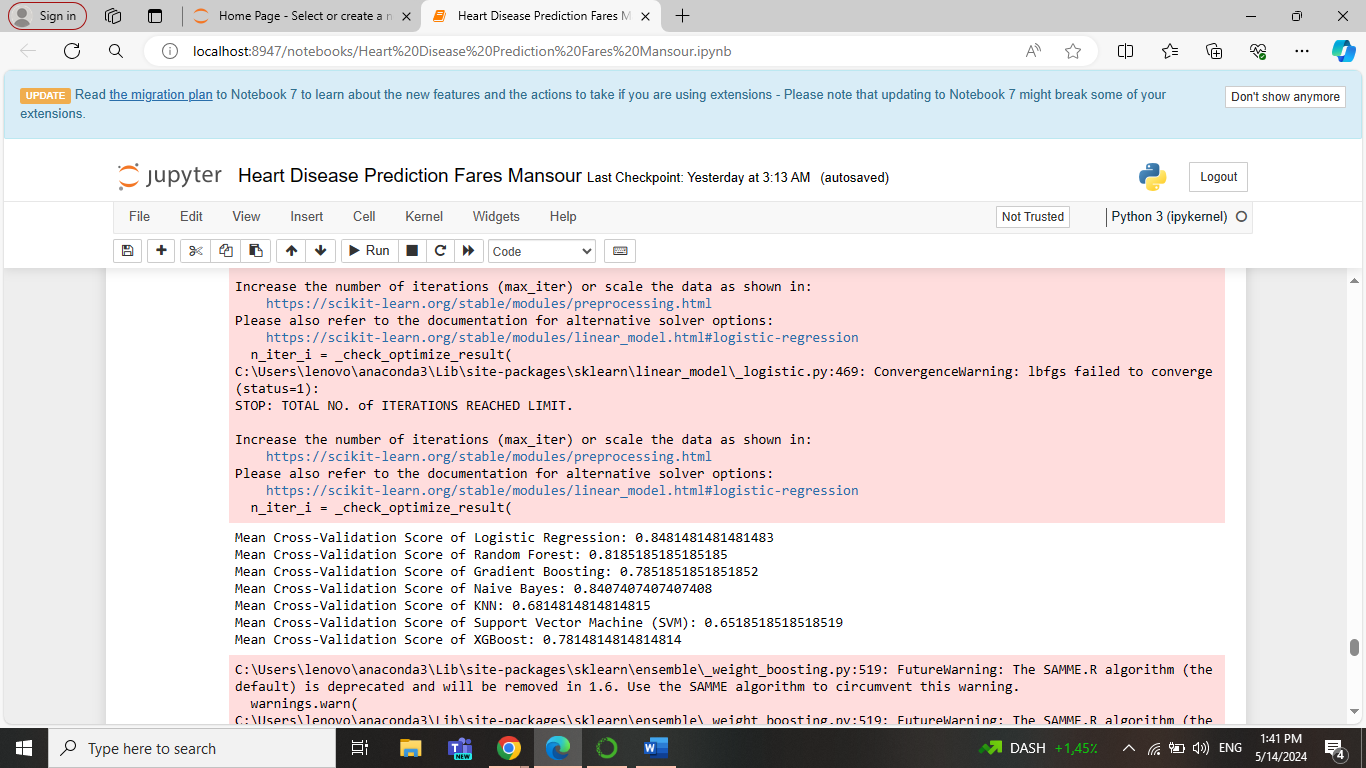
This output shows all the accuracies of the 9 models trained. Logistic Regression and Naïve Bayes were the highest among all with 91%. Followed by support vector machine and Adaboost with 89%. Then random forest with 87%. And the other’s accuracies weren’t as good as the others.

**10- Models Confusion matrices:**



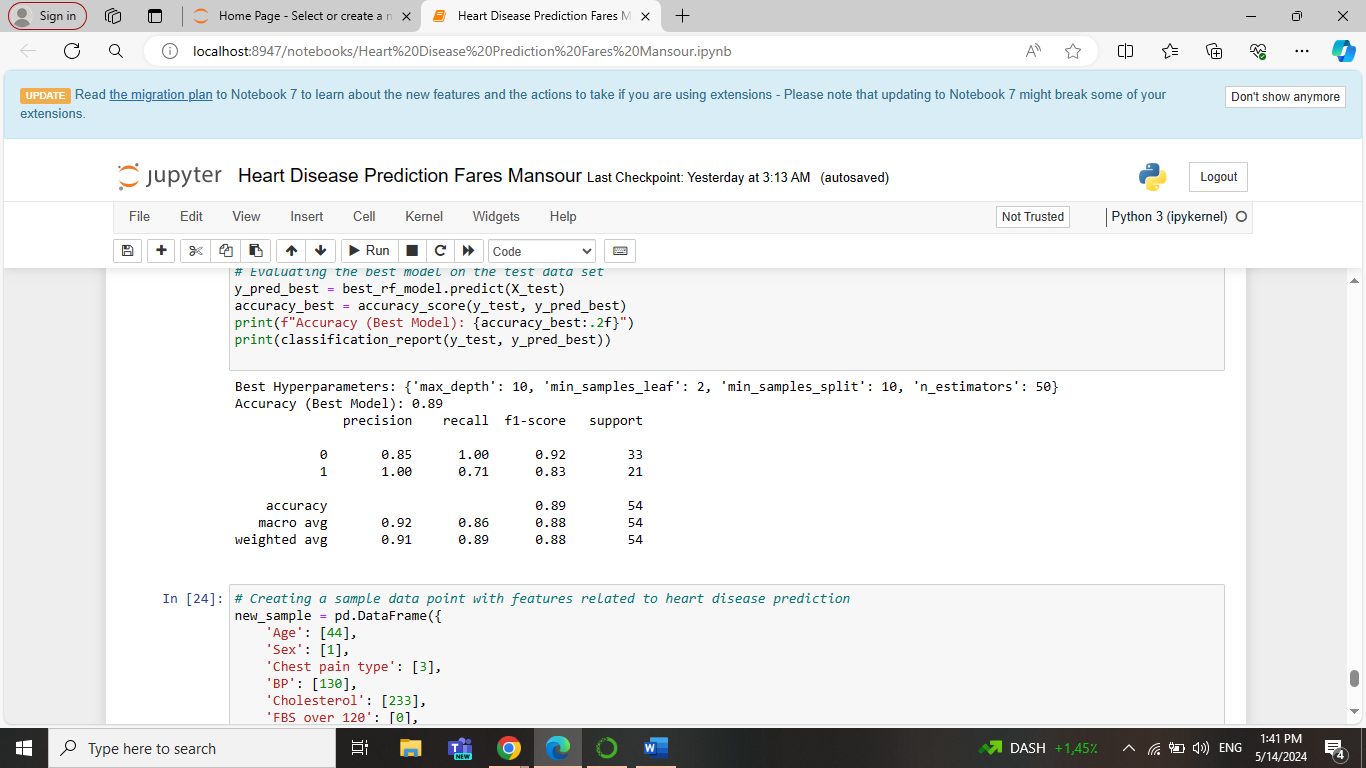
This output shows all the confusion matrices of the 9 models trained. Logistic Regression, Naïve Bayes, and support vector machine were the best among all. Followed by random forest and Adaboost. Then, Xgboost. And the other’s confusion matrices weren’t as good as the others.

**11-Models Cross-validation scores:**

****

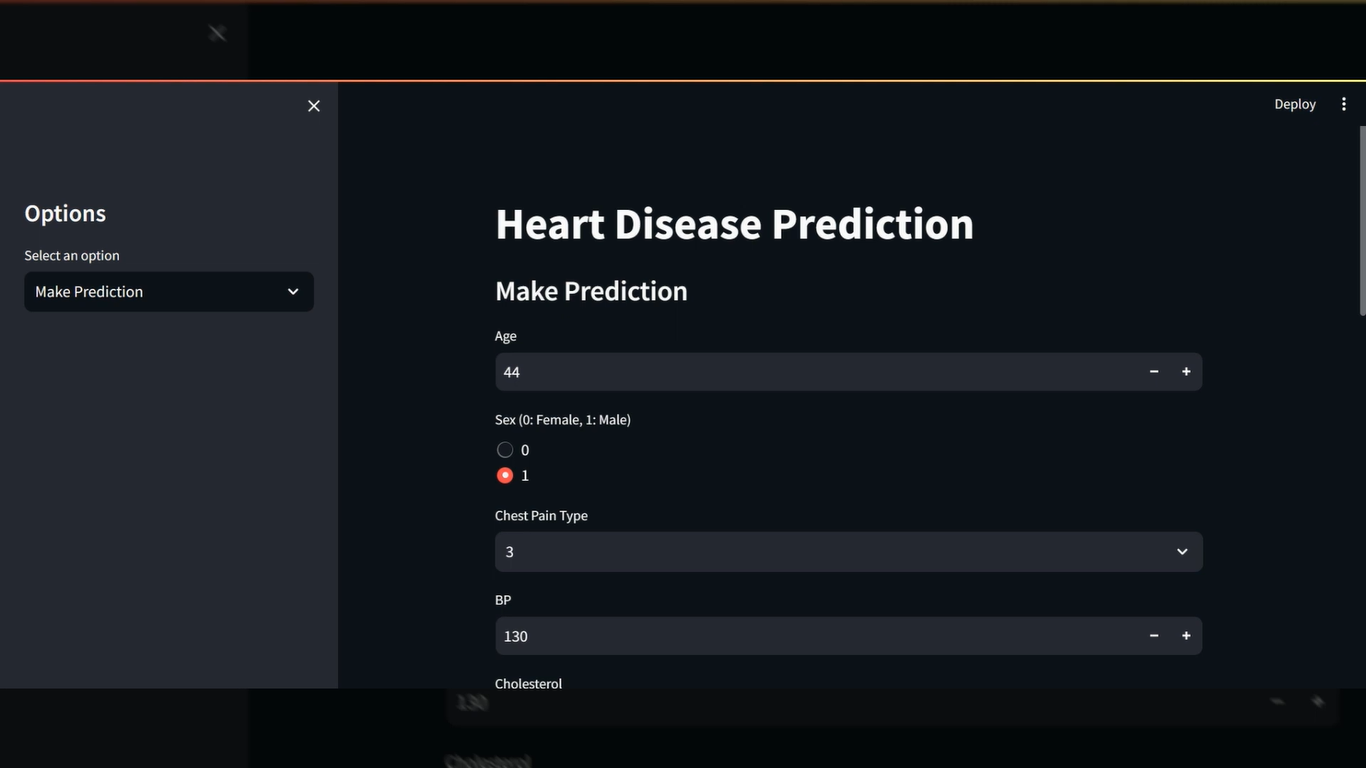
Like all the other outputs, Logistic Regression and Naïve Bayes are the highest. Followed by the others. Note that this output shows the mean cross validations not all of them.

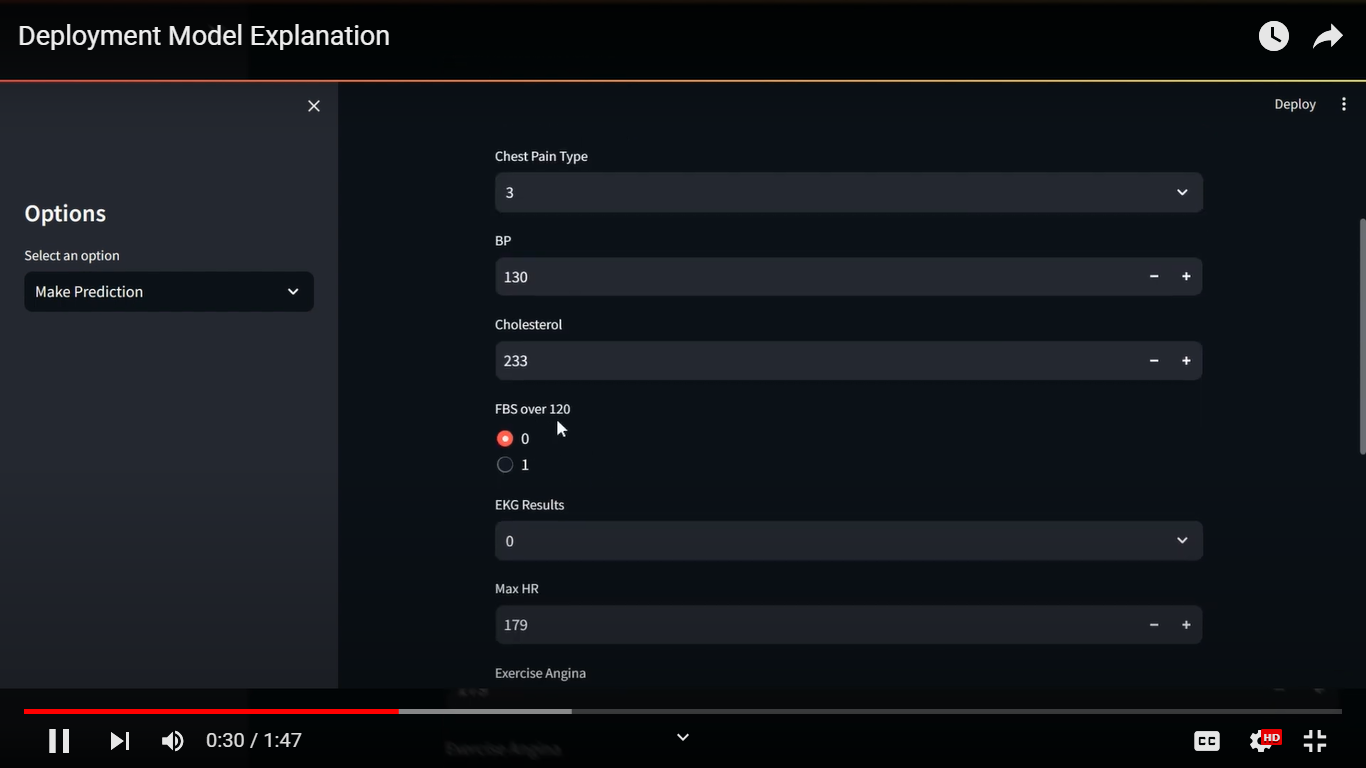
**12- Best Hyperparameters:**

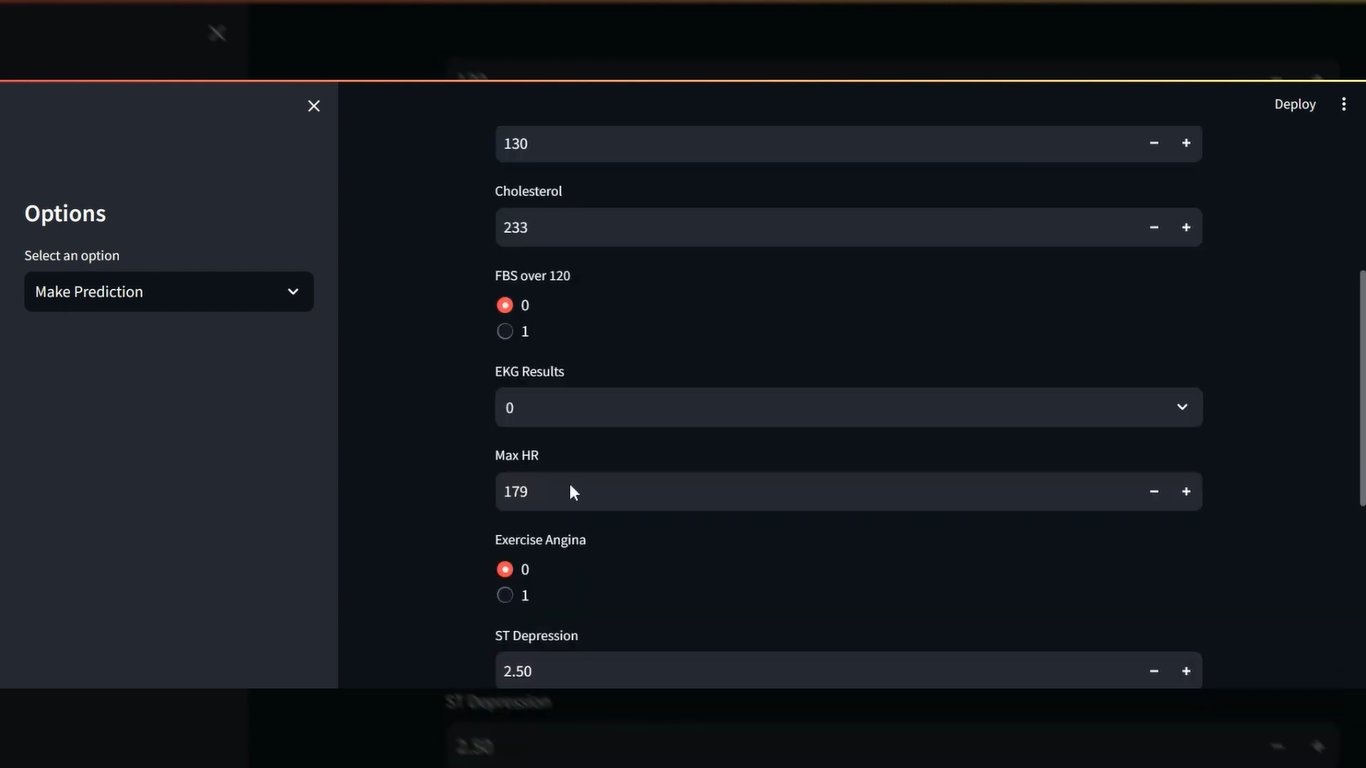


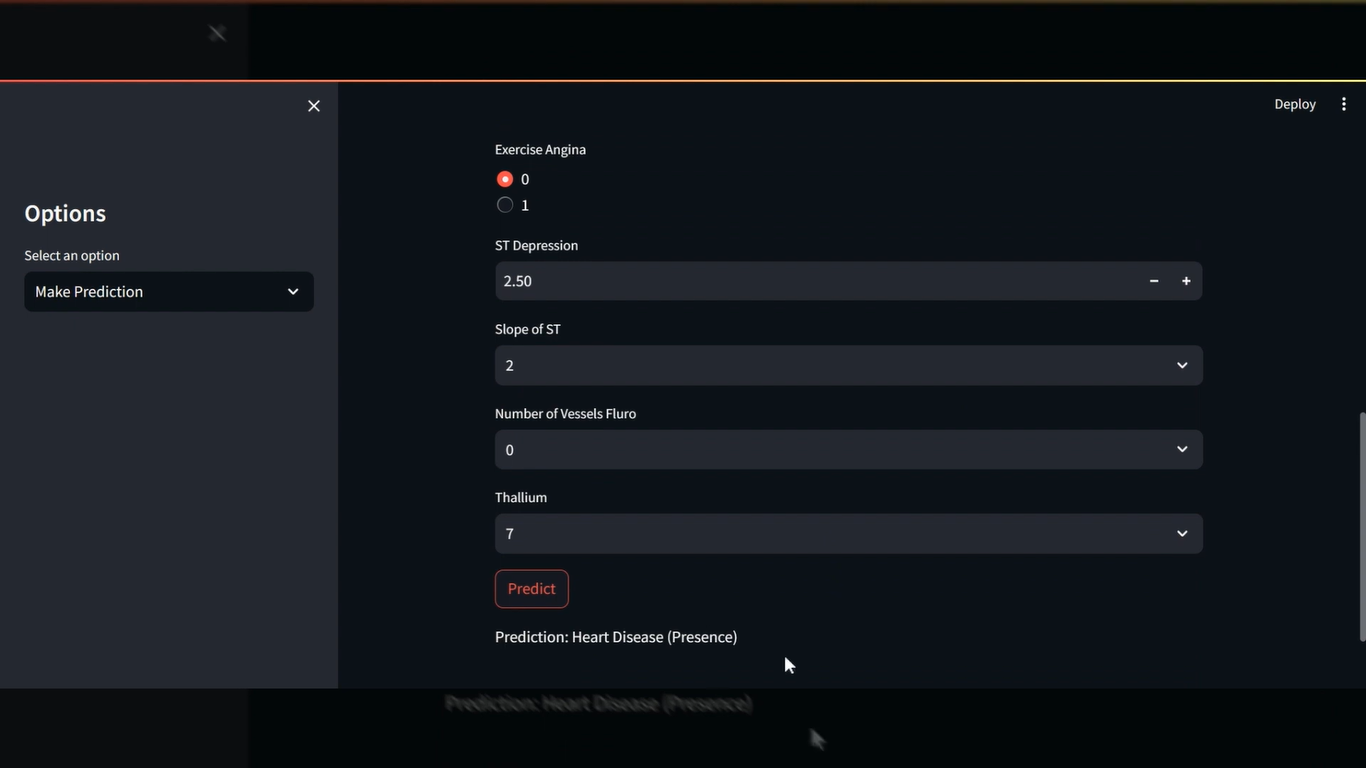
This output shows the best parameters for a model by using the grid search method. The best maximum depth is 10, the minimum sample leaf is 2, the minimum sample split is 10, and n estimators are 50. Regarding the best model, there were 3 good models. So, in every run the output changes to one of them.

**13- Model Deployment:**

****







This is the last step, the deployment model. These are pictures from the streamlit web app page. It makes the user enter his numbers and the model predicts whether the heart disease is present or not.

**Conclusion:** In conclusion, the project journeyed through understanding and predicting heart disease using data analysis and machine learning. It started by cleaning and preparing the data for analysis, then transformed categorical information into numbers to improve accuracy. Nine different models were tested. Overall, the analysis indicates that Logistic Regression, Naïve Bayes, and Support Vector Machine are effective models for predicting heart disease. Due to their high evaluation metrics and accuracies. Logistic Regression was chosen for deployment due to its good performance and simplicity. The deployment via the Streamlit web app provides a user-friendly interface for utilizing the model's predictions. The analysis on heart disease prediction via machine learning offers valuable insights but also has limitations. Diverse datasets could mitigate biases, advanced feature selection techniques and model interpretability methods may enhance transparency, and addressing class imbalance and conducting external validation could bolster generalizability.

Data analysis and machine learning play pivotal roles in real-world applications by extracting actionable insights from vast amounts of data and automating decision-making processes. From healthcare to finance, and transportation, these technologies enable organizations to optimize operations, improve efficiency, and drive innovation. Moreover, in fields like healthcare, machine learning models can assist in early disease detection, personalized treatment plans, and improving patient outcomes, ultimately saving lives, and transforming industries.

**Useful Links:**

1. The YouTube link explains the codes used in Jupyter Notebook to build the models and evaluate them: <https://youtu.be/xyocrYRZhq4>
2. The YouTube link explains the codes used in Visual Studio Code to build the Streamlit web app: <https://youtu.be/yMJfo_aWkKw>
3. The YouTube link explains how the user can interact with the deployment model: <https://youtu.be/fh_skk80wcY>
4. Github Link: <https://github.com/faresmansourr>