**Supervised Algorithms for DIABETIC**

**Importing all the basic libraries and the data.**

At the start I imported all the basic libraries. After that, I loaded the Diabetic dataset and checked the basic description and info of the dataset.

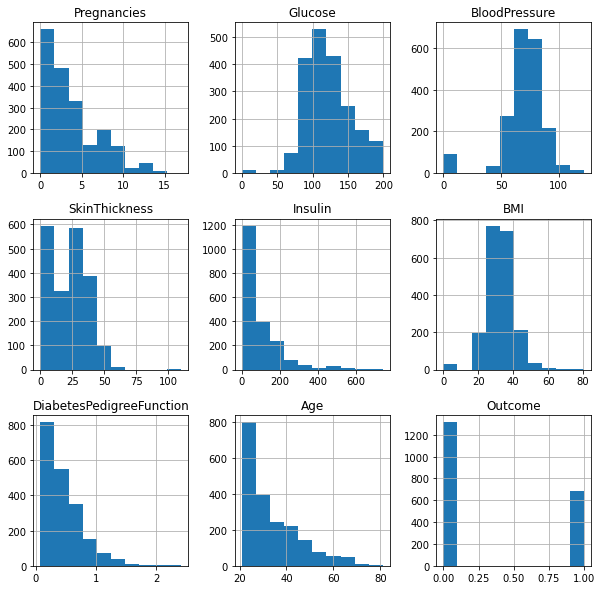
After loading the data I used df.head() to check the first 5 rows of data to get a basic insight of the data.

After importing the data, I moved to the next step

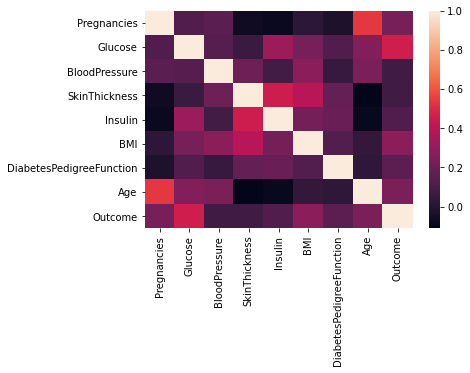
**Data Preprocessing.**

I checked if there were any missing values in any of the features. Since there were no null values, I moved on to the next step.

After that I plotted a Histogram to get a better understanding of the data.



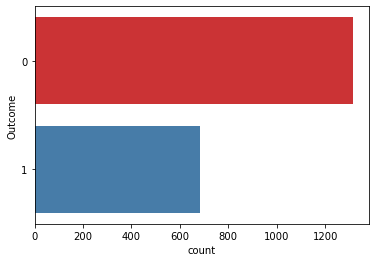
After that I plotted a Heatmap to check the correlation between the features.



Here I could see that,

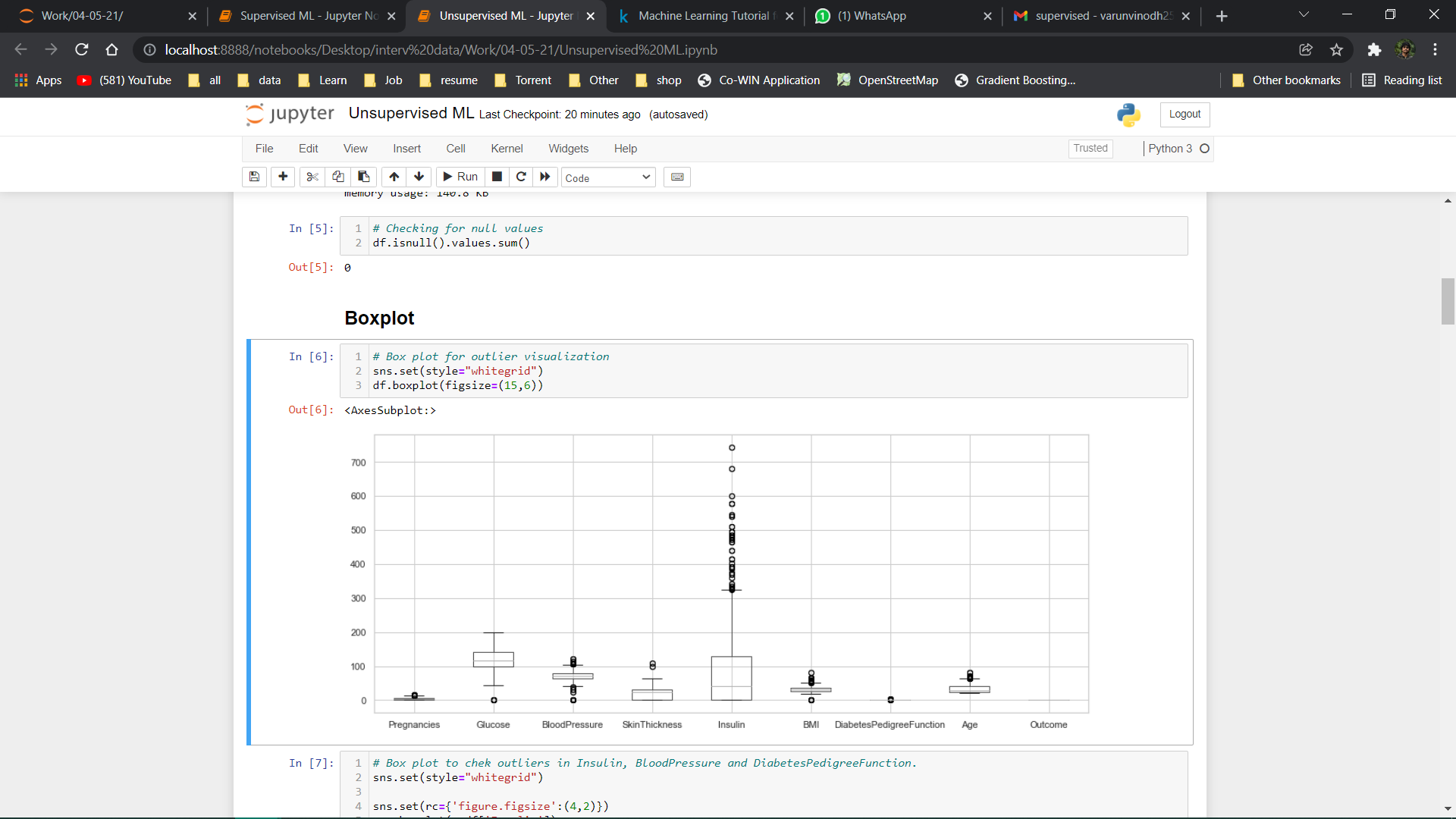
* *Skin thickness, insulin, pregnancies and age are full independent to each other.*
* *Age and pregnancies have negative correlation between them.*

After this I checked the number of people who were and were not diabetic.



*Here I could see that the number of diabetic people is less in this dataset.*

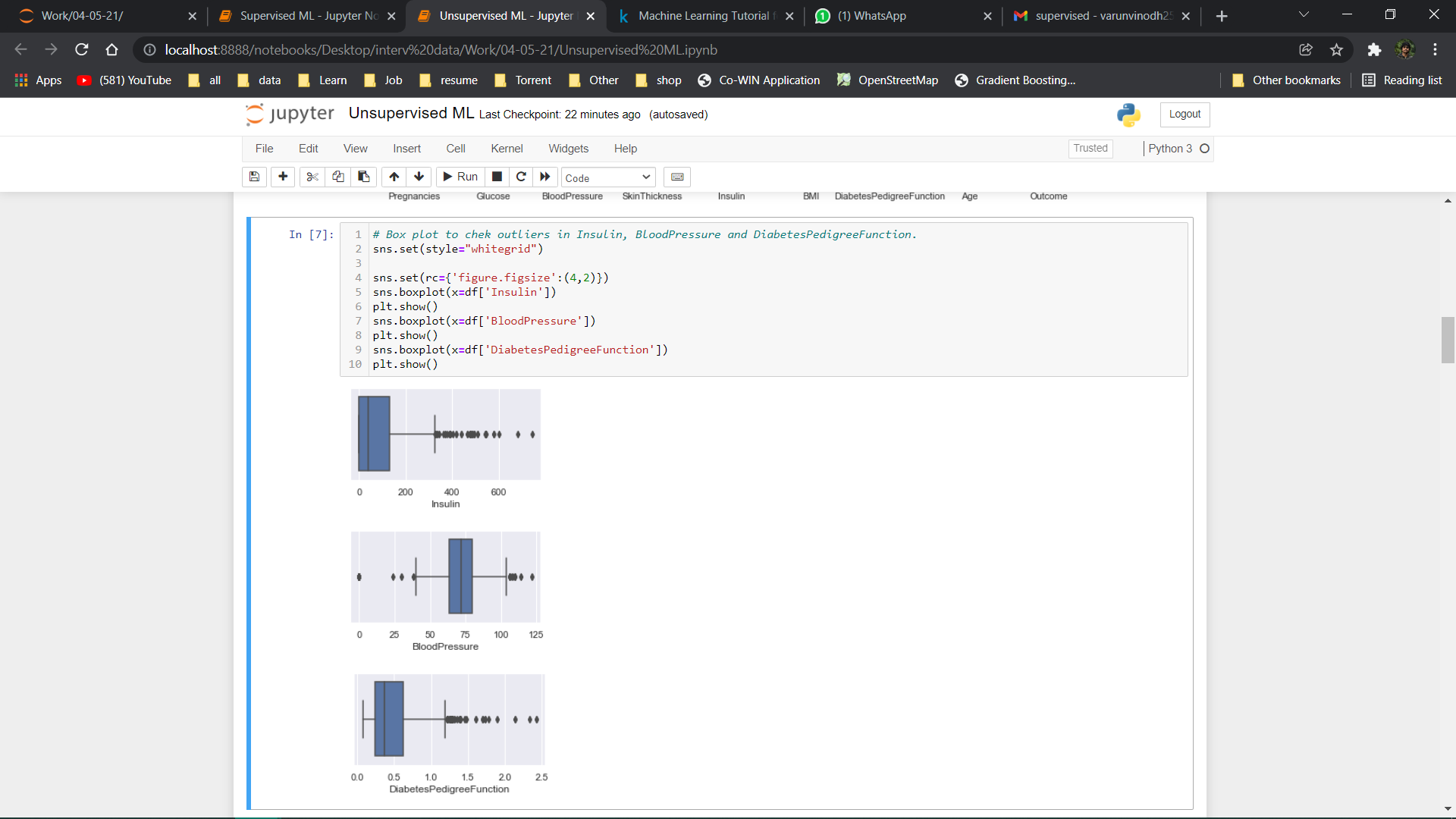
After this, I checked if there were any outliers in the data. For this I used Box plot.



Here I could see that there are some outliers in Insulin, BloodPressure and DiabetesPedigreeFunction.

So, to check even further I used the Box Plot on these 3 columns specifically.

Here I used seaborn library to get the box plot of the required columns.



Here I could see that there was a significant amount of data that was out of the Q1 and Q3 Range. So, I decided that here we have to do outlier treatment. For this I used Inter Quartile Range.

After that I moved on to the next step,

**Preparing for modeling.**

Here I split the data into test and train data using train\_test\_split with the test\_size at 30%.

After that I created Function for finding the True Positive, True Negative, False Positive and False Negative.

After this I moved on to the next step,

**Building the model.**

Here I used Logistic Regression model from sklearn. linear\_model. Then I used ‘*fit’* to train the model using the ‘*train\_X’* and ‘*train\_y’* data. Using this mode, I predicted the *‘test\_X’* data. I then used these predicted values(y\_pred) to check the accuracy score compared to the actual values(test\_y).

Using this model, I got an accuracy score of 78.02%.

For all the next models I did the same steps and got the following accuracy,

Support Vector Machine: 78.02%

K-Neighbors Classifier: 84.47%

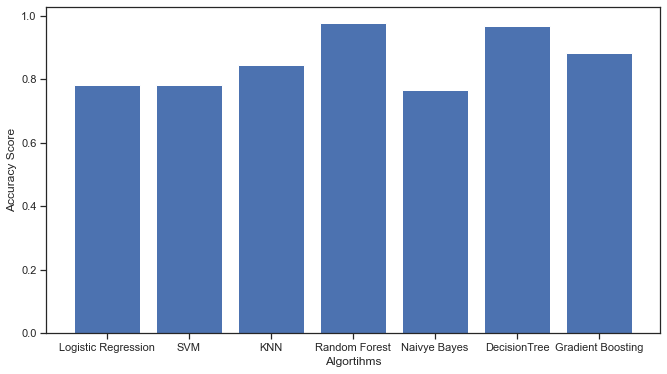
Random Forest: 97.78%

Gaussian Naive Bayes: 76.61%

Decision Tree Classifier: 96.57%

Gradient Boosting Classifier: 88.30%

Then I used Bar plot to compare these all these models,



Here we could see that the Random Forest model performed better compared to all other models.

The Main objective of this assignment was to understand and predict whether a person is diabetic or not using supervised ML algorithm, and we did this and the best model with better accuracy was the Random Forest algorithm with 98% accuracy.