**Blog Submission**

**Insurance Claims-Fraud Detection**



**Submitted By:**

**Rekha Adak**

**Batch no:1832**

**INTRODUCTION**

**The auto insurance industry is complicated and involves millions of dollars changing hands every day. And whenever there is a large amount of money running through complex systems, there is opportunity for fraud. This fraud can be committed by professionals and company working in the industry. But it can also be committed against them. By reading this, the first thing that comes into your mind is, what is insurance fraud. Let’s understand this.**

**So, what is insurance fraud?**

**“Improper community committed by an individual in order to obtain favourable outcomes from the insurance company”.**

**The insurance fraud can be broadly classified into 2 types.**

1. **Soft Insurance Fraud**
2. **Hard Insurance Fraud**

**Soft insurance fraud: A common example for this is, if the accident has taken place, but the amount of damage what has happened to the vehicle is very less. In such cases, the individual claims to the insurance company that huge amount of damage have occurred to the vehicle with the goal of charging the insurance company a higher bill.**

**Hard insurance fraud: In this type of fraud, an individual intentionally plans and invest the loss so that he can claims for the insurance from the company. A common example for this type of fraud is staging a car wreck with the goal of benefitting from the resulting claim.**

**This project focuses on claim data of an Automobile insurance company. Because of fraudulent claims the insurance companies are losing huge amounts of money, which indirectly affects the public. Therefore, it is important to know which claims are genuine and which are fraud.**

**In this article we’ll walk through how to spot insurance fraud and the consequences of engaging in it by building machine learning models and getting prediction of which claims are likely to be fraudulent. This enables an insurer to detect more fraudulent claims**

**1.Problem Definition**

**Insurance fraud is a huge problem in the industry. It's difficult to identify fraud claims. Machine Learning is in a unique position to help the Auto Insurance industry with this problem.**

**In this project, we are provided with a dataset which has the details of the insurance policy along with the customer details. It also has the details of the accident on the basis of which the claims have been made.**

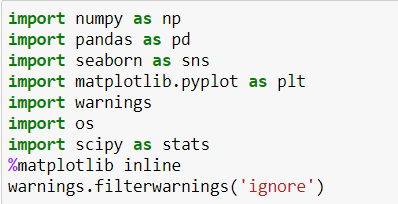
**In this example, we will be working with some auto insurance data to demonstrate how we can create a predictive model that predicts if an insurance claim is fraudulent or not**

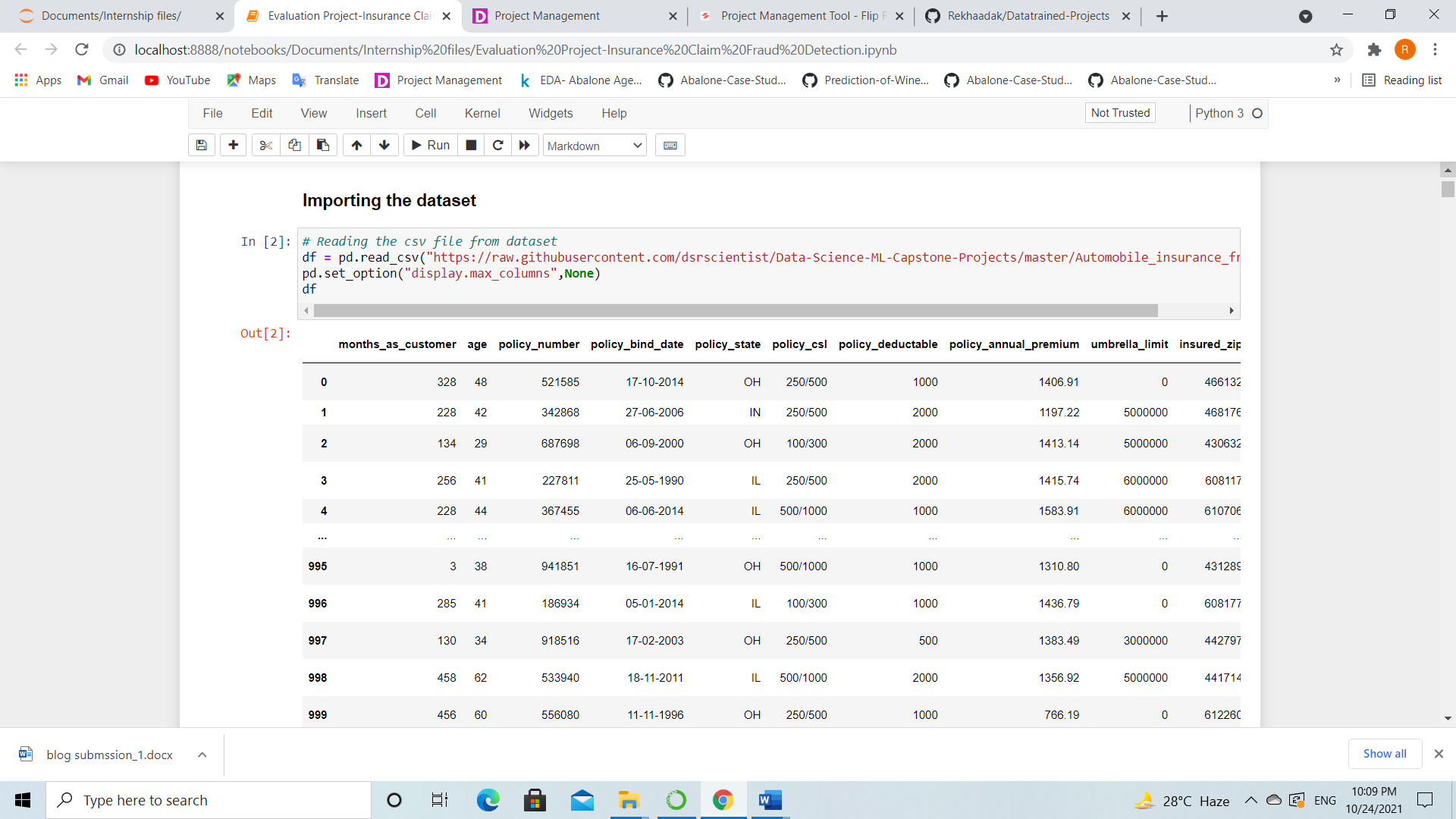
**The problem statement explains that the target variable contains the categories, so it is a “Classification Problem” where we need to predict whether an insurance claim is fraudulent or not.**

## 2.Data Analysis

## The process of cleaning, transforming and extracting data to discover the useful information for business decision making is called data analysis.

**Importing necessary libraries and dataset**

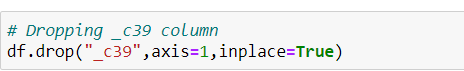
****

****

**I have imported the dataset which was in the csv file using pandas. The dataset contains 1000 rows and 40 columns having both numerical and categorical data. Here I can make use of PCA to reduce the columns, but this will give huge data loss. To avoid this, I am keeping the dataset as it is. We can also observe the null values, some “?” signs and some of the columns are in date format (dd/mm/yy), so we need to perform lots of preprocessing, cleaning to make our data usable to build the ML models.**

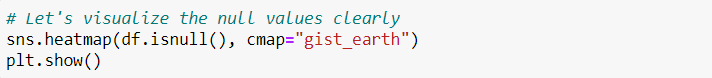
**Data Preparation and cleaning:**

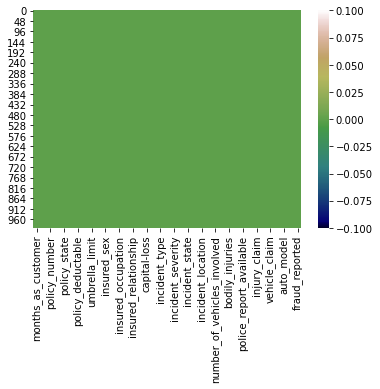
* **First, we need to check some statistical information about the dataset like checking shape, datatypes, nunique, value counts, info() etc.**
* **After checking the value counts, if we find any unwanted columns in the dataset then we need to do feature engineering on those columns based on the problem.**
* **While I ran df.info(), I found c\_39 column having one unique count as NAN throughout the dataset and it is of no use, so I dropped that column.**

****

**After dropping the above column, if we observe the info, the counts of all the columns are same hence there are no missing values present in the dataset.**

**Checking null values**

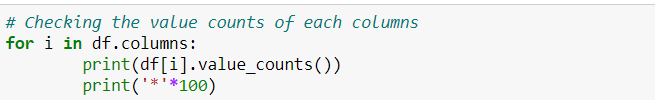
****

****

**We can observe there are missing values present in the data. So we can proceed further.**

**Feature Selection**

**Let's check the list of value counts in each column to find if there are any unexpected or corrupted entries present in the dataset.**

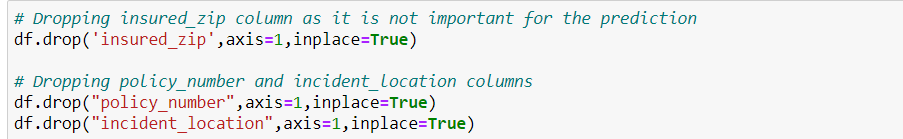
****

**By running the above for loop, you will get the value counts of all the columns present in the dataset.**

* **By looking into the value counts of each column I came to know that the column umbrella\_limit contains about 80% of zero values. It might create skewness in the data so better to drop this column here only.**

****

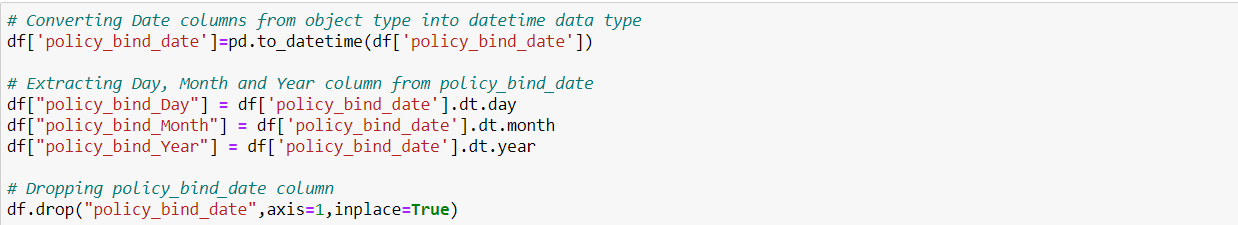
* **There is a column named insured\_zip, it is the zip code given to each person. If we take a look at the value count and unique values of the column insured\_zip, it contains 995 unique values that means the 5 entries are repeating.**
* **Since it is giving identier information about the person, it is not important for the processing so we can drop this column as well.**
* **Also, the columns policy\_number and incident\_location have only one unique count through out the data, it is not required for the prediction so drop it.**

****

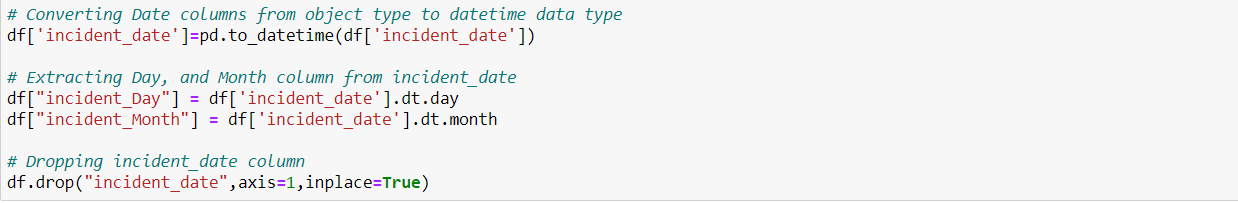
**Feature Extraction**

* **The columns policy\_bind\_date and incident\_date have data in the form of dd/mm/yy and showing object data type which should be in datetime data type that means the python is not able to understand the type of these columns and giving default data type, this could be because of some special characters present in the data.**
* **So, we will convert this object data type into datetime data type to use them properly for the prediction and we will extract the values from these columns.**

**policy\_bind\_date: Policy bind date means, on which date the policy was made. On this basis I have extracted day, month and year of policy made and dropped policy\_bind\_date as it is of no use.**

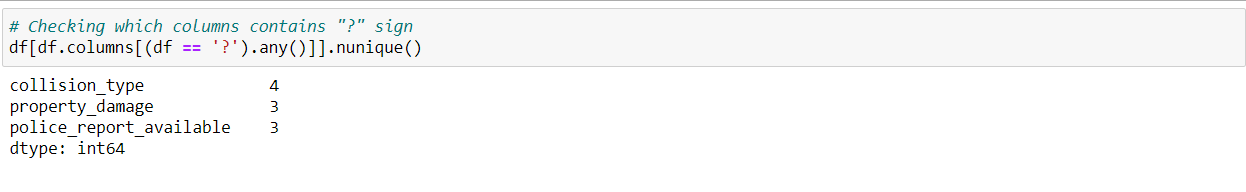
****

**Incident\_date: It gives us that, on which date the incident occurs. This column contains only 2015 year data so I have extracted only day and month from this column and after that dropped the column accordingly.**

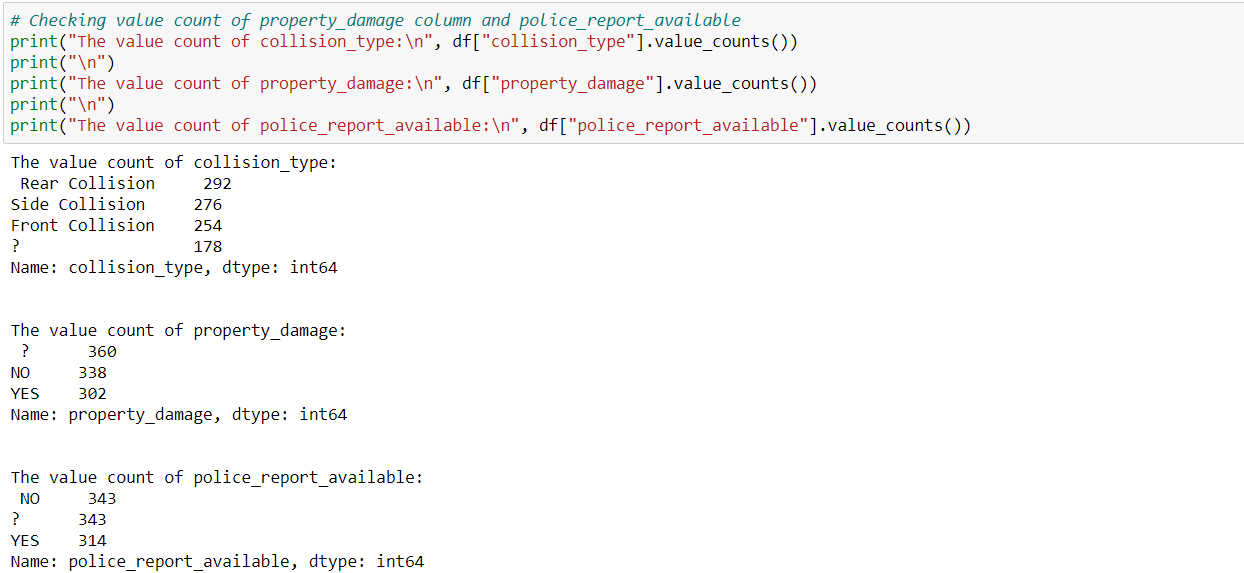
****

**Replacing “?” sign**

* **By looking at the dataset and value count function, we have found some columns having “?” sign. These are not to be considered as NAN values but we need to fill them. First, we will check which columns contains “?” sign.**

****

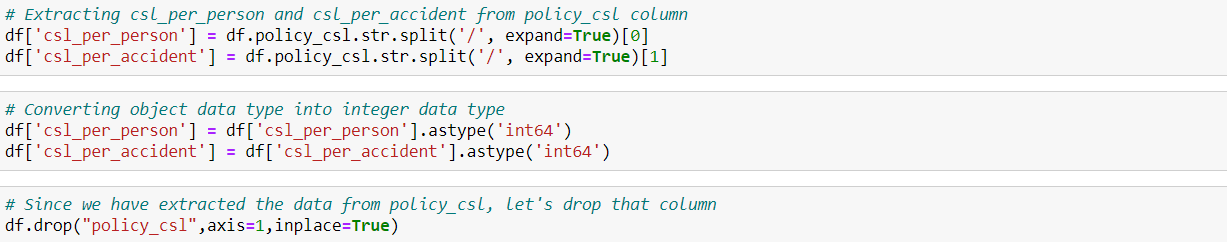
* **These are the columns which contains "?" sign.**
* **There are two ways to fill these values. Either you can fill them by using appropriate values or you can fill them by using their mode.**
* **Since these columns seems to be categorical so we will replace "?" values with most frequently occuring values of the respective columns that is their mode values. To do this let’s check the value count of these columns.**

****

* **The mode of property\_damage and police\_report\_available is again “?”, so we will use the second highest count in these columns that is “NO”.**
* **So, we will replace the “?” sign in collision\_type, property\_damage and police\_report\_available are Rear Collision and NO respectively.**

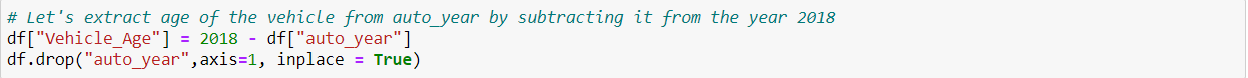
****

* **The policy\_csl column is also showing object data type but it is numerical in nature. May be, it is because of the presence of "/" sign in that column.**
* **This policy csl auto insurance is a combination of both bodily injury and property damage. On this basis we will extract two columns namely csl\_per\_person and csl\_per\_accident from policy\_csl colums and then will convert their object data type into integer data type.**

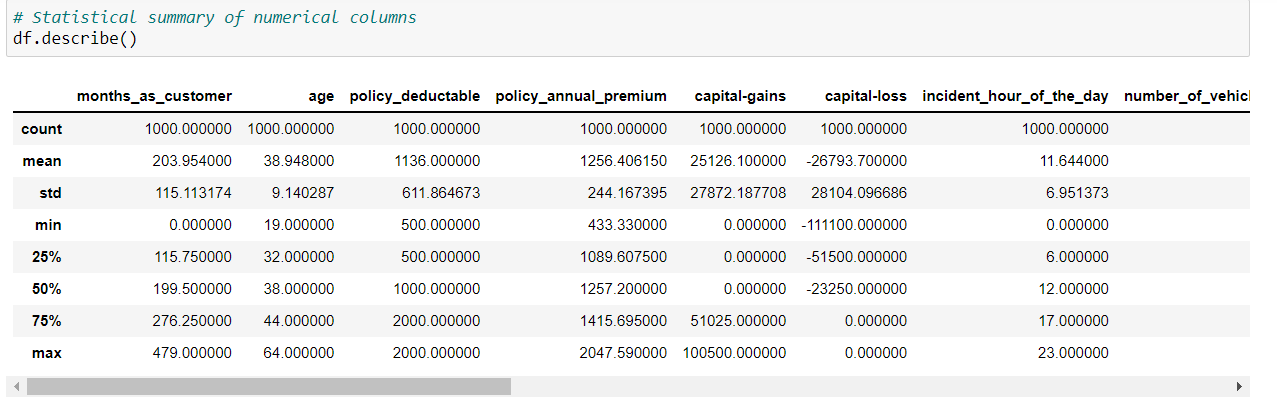
****

**Here I have extracted 2 columns from policy\_csl and converted them into integer data and dropped policy\_csl column after getting the required data from it.**

* **In the dataset there is one column named auto\_year which gives the age of the vehicle. So, we will extract age of the vehicle from auto\_year by subtracting it by 2018. Assuming the data is collected in the 2018.**

****

**We have successfully dealt with the feature engineering and now we will move further to know about the statistical summary of the dataset.**

****

**The describe method gives the statistical information of the dataset. The summary of this dataset looks perfect since there are no negative/ invalid values present. It gives the summary of numerical data.**

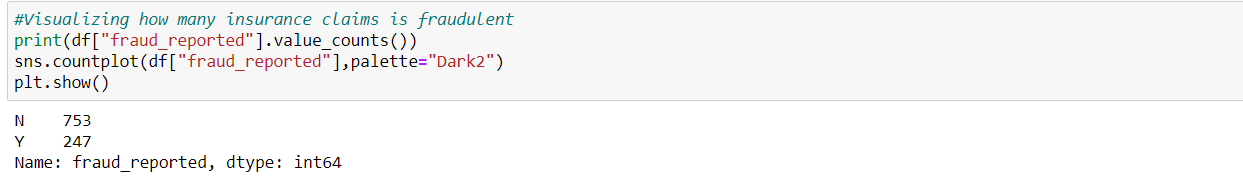
**From the above description we can observe the following things**

* **Here the counts of all the columns are equal which means there are no missing values in the dataset.**
* **In some of the columns like policy\_deductable, capital-gains, injury\_claim etc we can observe the mean value is greater than the median (50%) which means the data in those columns are skewed to right.**
* **And in some of the columns like total\_claim\_amount, vehicle\_claim...etc we can observe the median is greater than the mean which means the data in the columns are skewed to left.**
* **And some of the columns have equal mean and median that means the data symmetric and is normally distributed and no skewness present.**
* **There is a huge difference in 75% and max it shows that huge outliers present in the columns.**

**Data Visualization**

**Before going to visualize the data, first we need to separate numerical and categorical columns.**

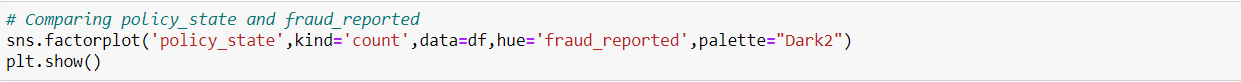
****

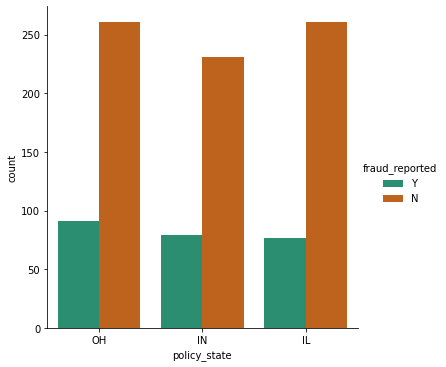
****

****

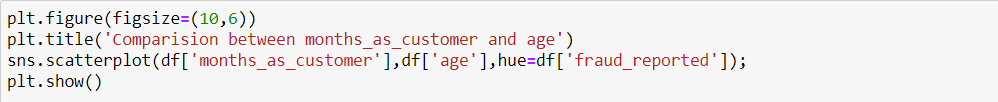
* **From the plot we can observe that the count of "N" is high compared to "Y". Here we can assume that "Y" stands for "Yes" that is the insurance claim is fraudulent and "N" stands for "No" means the insurance claim is not fraudulent. There are 247 frauds and 753 non-frauds.**
* **Since we are dealing the classification problem, the target fraud\_reported it indicates the class imbalance issue. We need to balance the data before proceed with our models.**

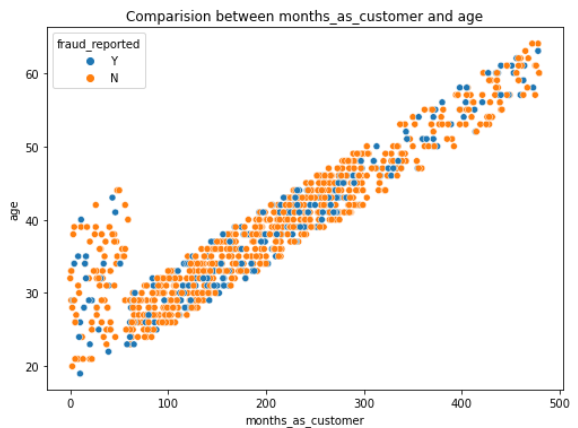
**Now we will compare the features and label by visualizing the data.**

****

****

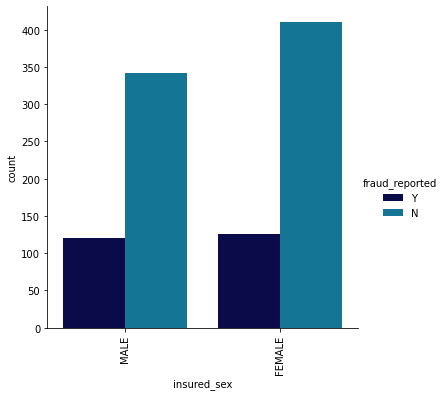
* **I have used count plot to compare policy state and fraud report. I can notice that the types of the policy claimed by the customers are almost same but still the fraud report is bit high in OH policy state compared to others and the types IL and IN have similar fraud reports.**

****

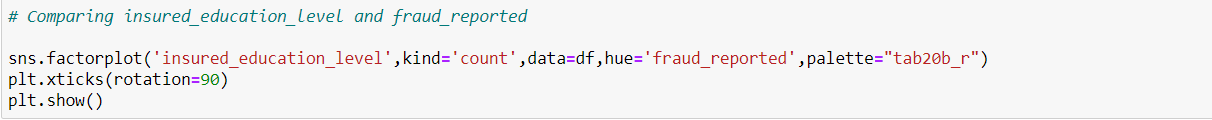
****

* **From the above scatter plot, we can observe the strong linear relationship between the age and month\_as\_customer. Which means as the month\_as\_customer increases, the age of the person also increases. Also, as the person getting older, the frequency of the both fraud report classes are vanishing slowly. That means, the people having young age are more likely to have high fraud reports.**

****

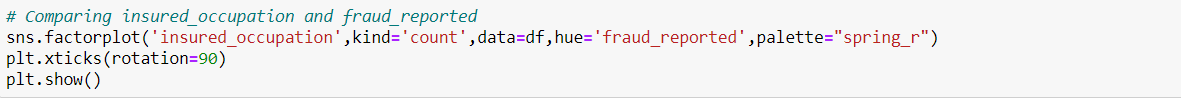
****

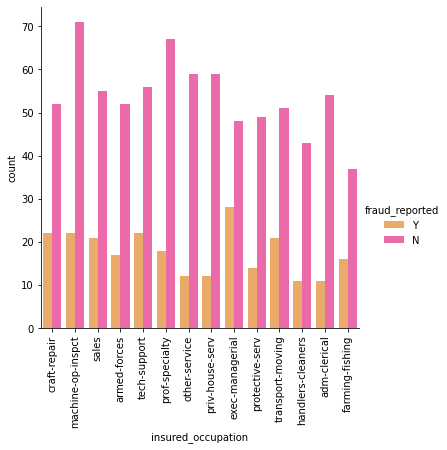
* **Above is the factor plot to compare insured sex and fraud reports. I can notice both male and female customers have insurance but the count for Female is bit higher than Male counts.**
* **The fraud reports are almost same in both the genders but the non-fraud reports are bit high in case of females that means the female customers are more trust worthy than male.**

****

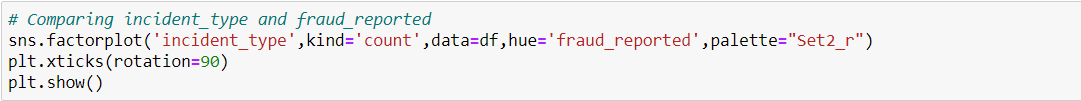
****

* **From the above factor plot we can observe the fraudulent level is very less for the people who have high school education and the people who have completed their "JD" education have high fraud report.**
* **The people who have high insured education are facing insurance fraudulent compared to the people with less insured education level. That means the people with less education level are more trustworthy.**

****

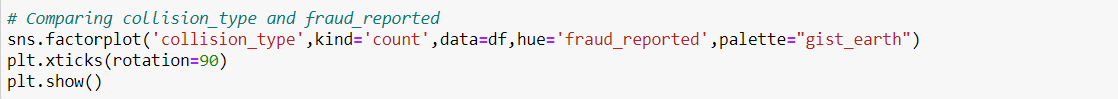
****

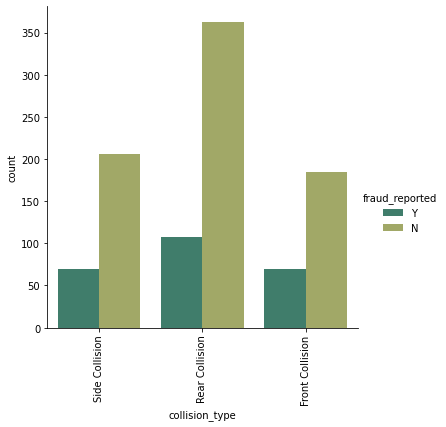
* **In the insured occupation we can observe most of the data is covered by machine operation inspector followed by professional speciality. Apart from this all the other insured occupations have almost same counts.**
* **The people who are in the position exec-managerial have high fraud reports compared to others and the non-fraud report is high for machine operation inspector that means the people having good occupations have more fraud reports.**

****

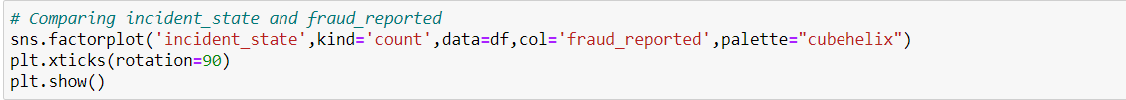
****

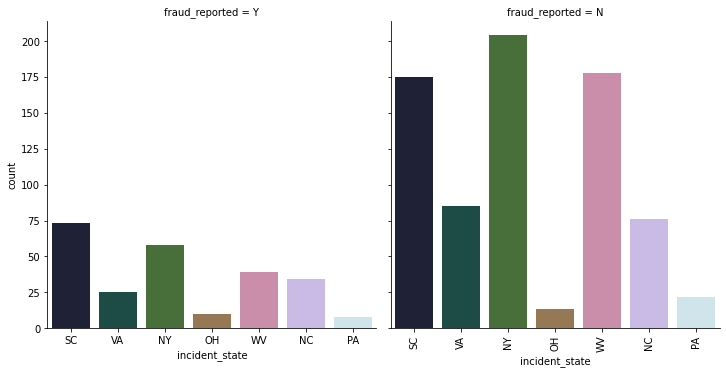
* **From the above factor plot we can notice that the Multi-vehicle collision and Single Vehicle Collision have pretty much similar counts. But the count is very less in Parked car and Vehicle Theft.**
* **In Multivehicle collision and single vehicle collision, the fraud report is very high compared to others and the nonfraud reports are also almost similar for these types.**

****

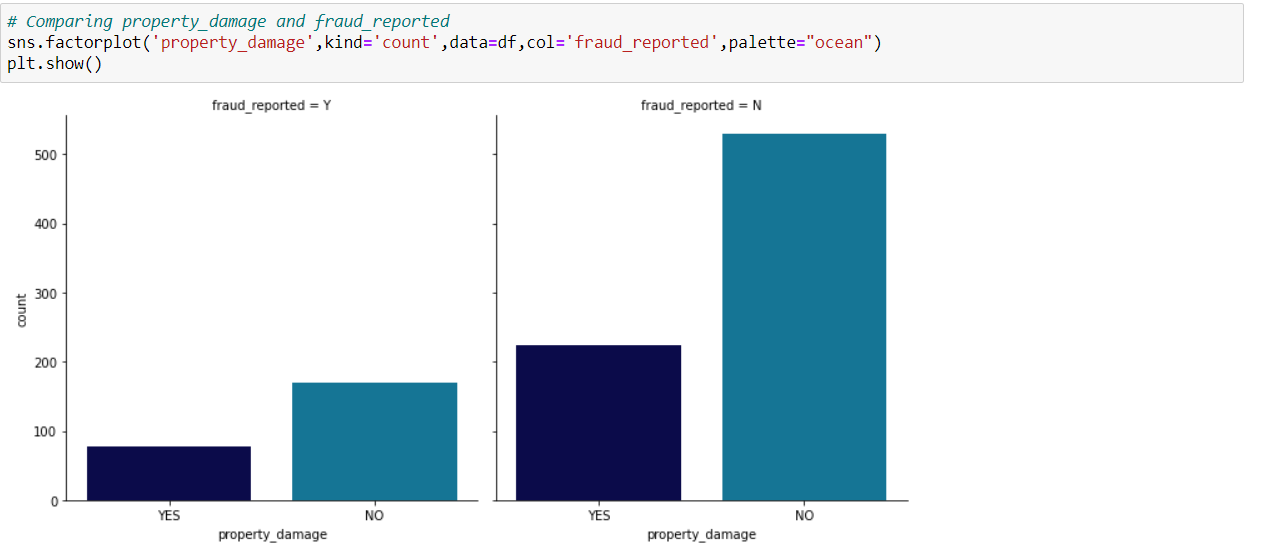
****

* **The collision type has 3 different types. The count is high in Rear collision and the other two types have almost equal counts.**
* **The fraud level is high in the collision type Rear Collision and other two collision type have average reports. Also the nonfraud reports also high in Rear collision type.**

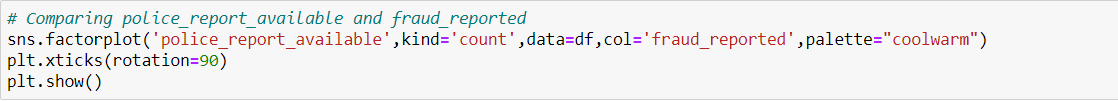
****

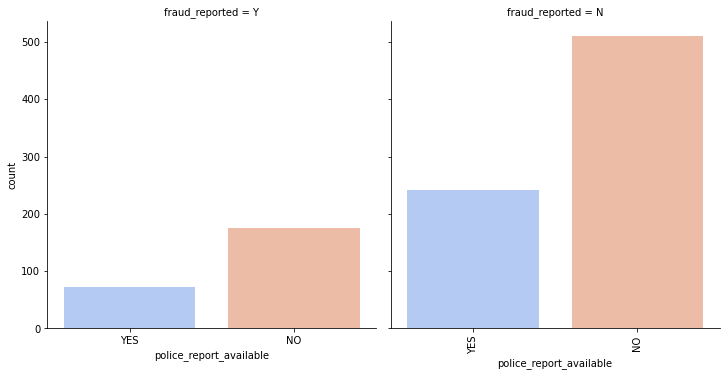
****

* **With respect to the incident state, New York, South Carolina and West Virginia states have highest counts.**
* **The state SC has high fraud reports followed by New York compared to other states.**

****

* **From the above count plot, we can notice that the customers who do not have any property damage case they have high fraud reports.**
* **Around 69% of the people did not face any property damage while 30% of the people faced the property damage**

****

****

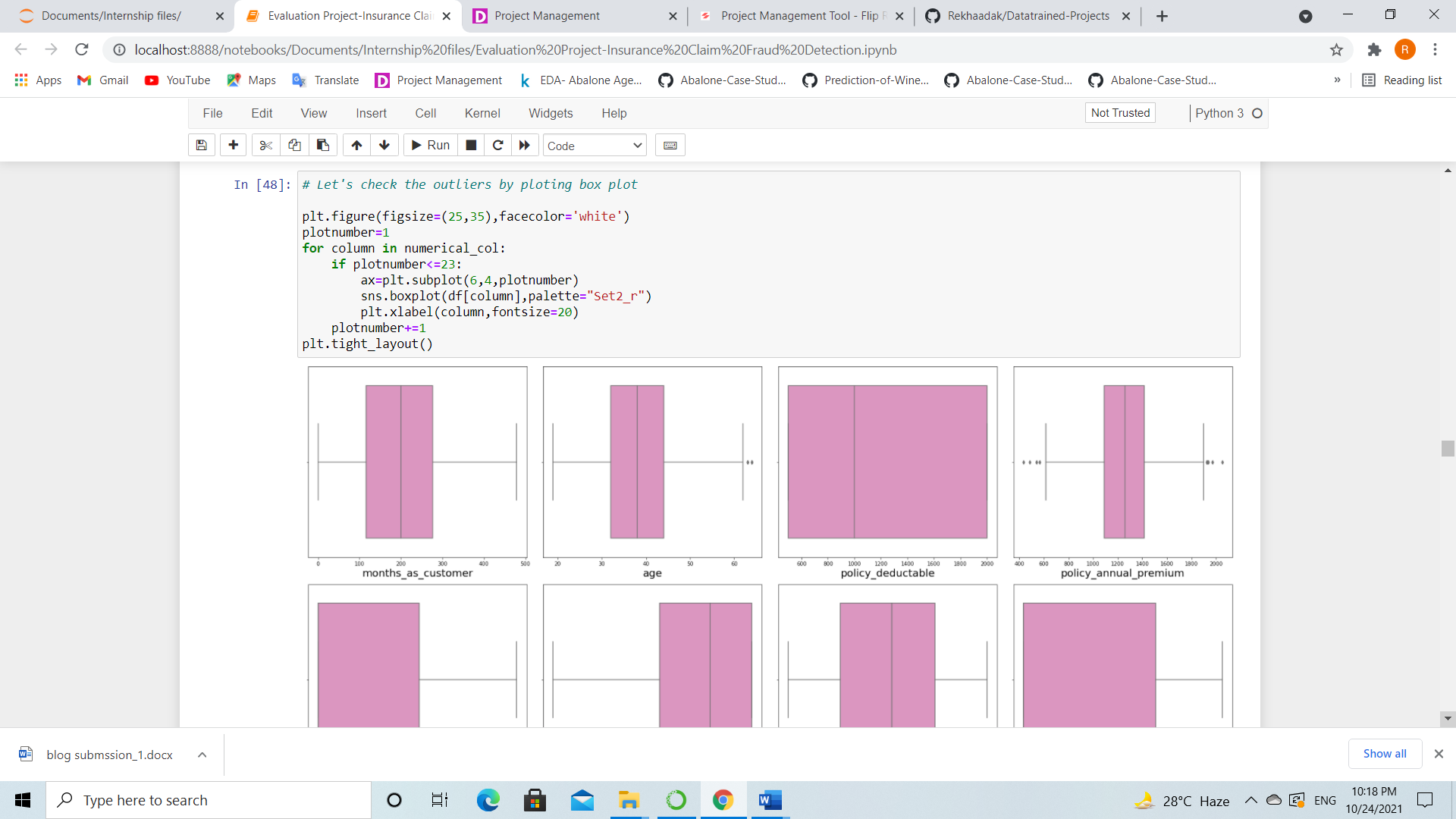
* **Over 68% of the people were produced the police reports while 31% of the people didn't submit any police reports.**
* **If there are no police report available then the fraud report is very high. So, while giving the insurance it’s very important to check the background of the customers.**

**In a similar manner we can plot the graphs for other columns too and we can make a good visualization by them.**

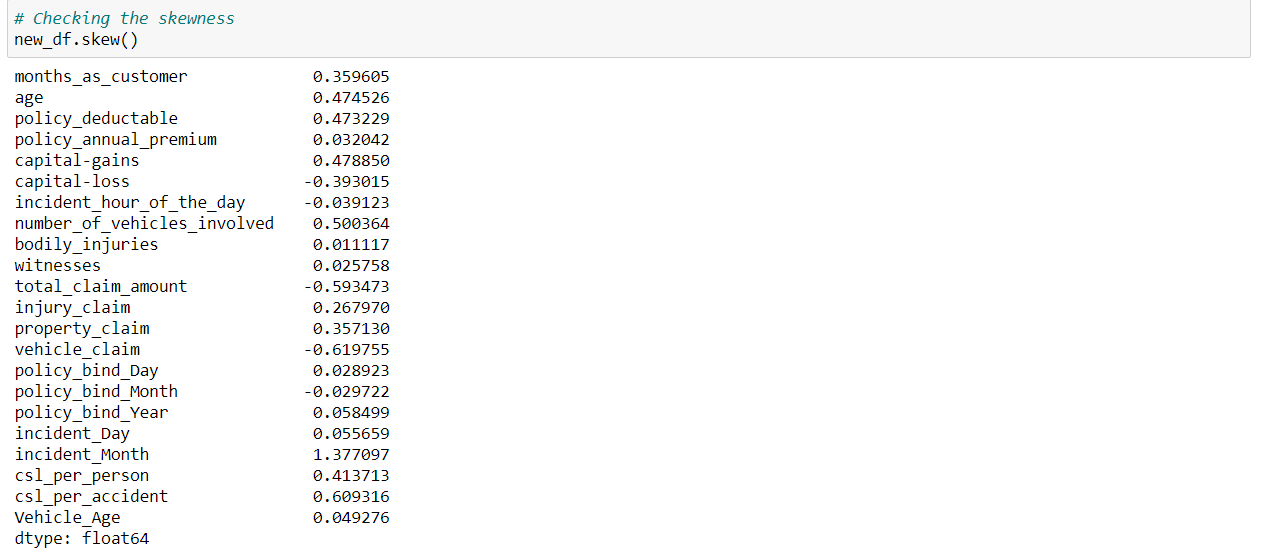
**3.EDA Concluding Remark**

* **I have checked the null values in the dataset and there were no missing values found.**
* **I have dropped some of the irrelevant columns to overcome with the multicollinearity problem.**
* **I have extracted some new features from the existing features to get better results without any hindrance. And dropped the old columns, if I keep them as it is they will act as duplicates and that leads to multicollinearity problem.**
* **Replaced the corrupted entries “?” in the columns with their respective mode values.**
* **Coming to the visualization part, we have found when and where the fraud reports are high in number.**
* **To get the better insights about the features, I have used violin plots, box plots, scatter plots and distribution plots.**

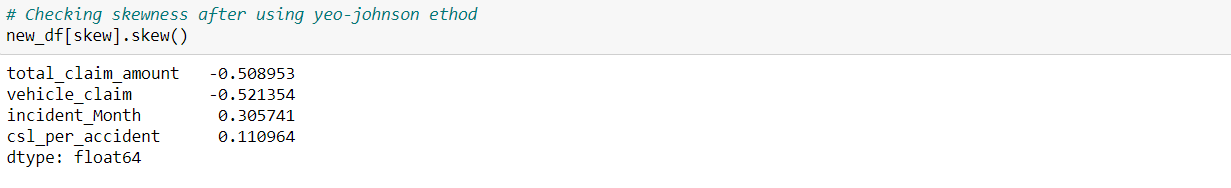
**Checking outliers and skewness in the data**

****

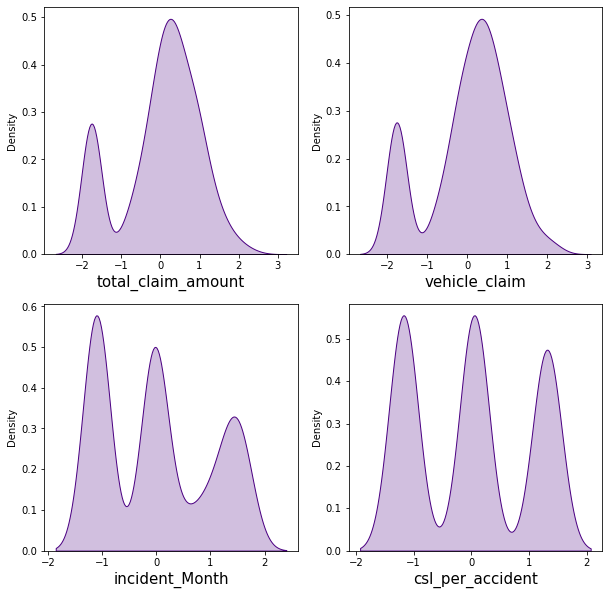
* **I have used boxplots to check the outliers in the dataset and found outliers in age, policy\_annual\_premium, total\_claim\_amount, property\_claim amd incident\_Month.**
* **To remove outliers, I used both Zscore and IQR methods. Using Zscore, I got 0.4% data loss and in IQR method I got about 5.8% of data loss. So, I decided to consider Zscore and process my model. After removing the outliers, I stored my dataframe in new\_df variable.**

****

* **I can notice the skewness in total\_claim\_amount, vehicle\_claim, incident\_Month and csl\_per\_accident.**
* **I used power transformation method (yeo-johnson method) to remove the skewness in the data. After using it, the skewness has almost been reduced.**

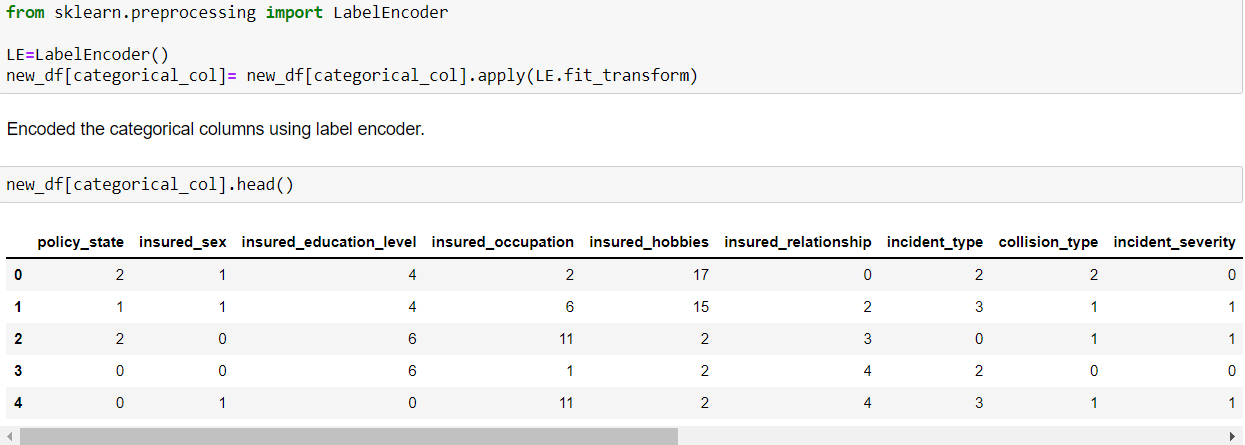
****

****

****

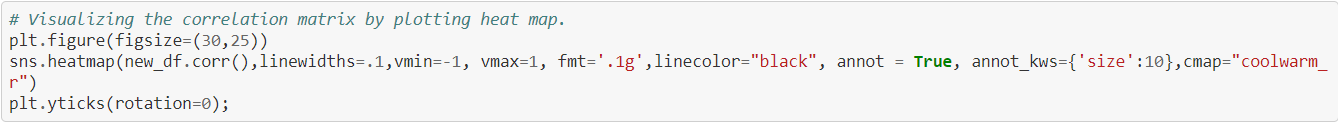
**We can notice our data is almost normal and skewness is also removed. Now we can proceed further.**

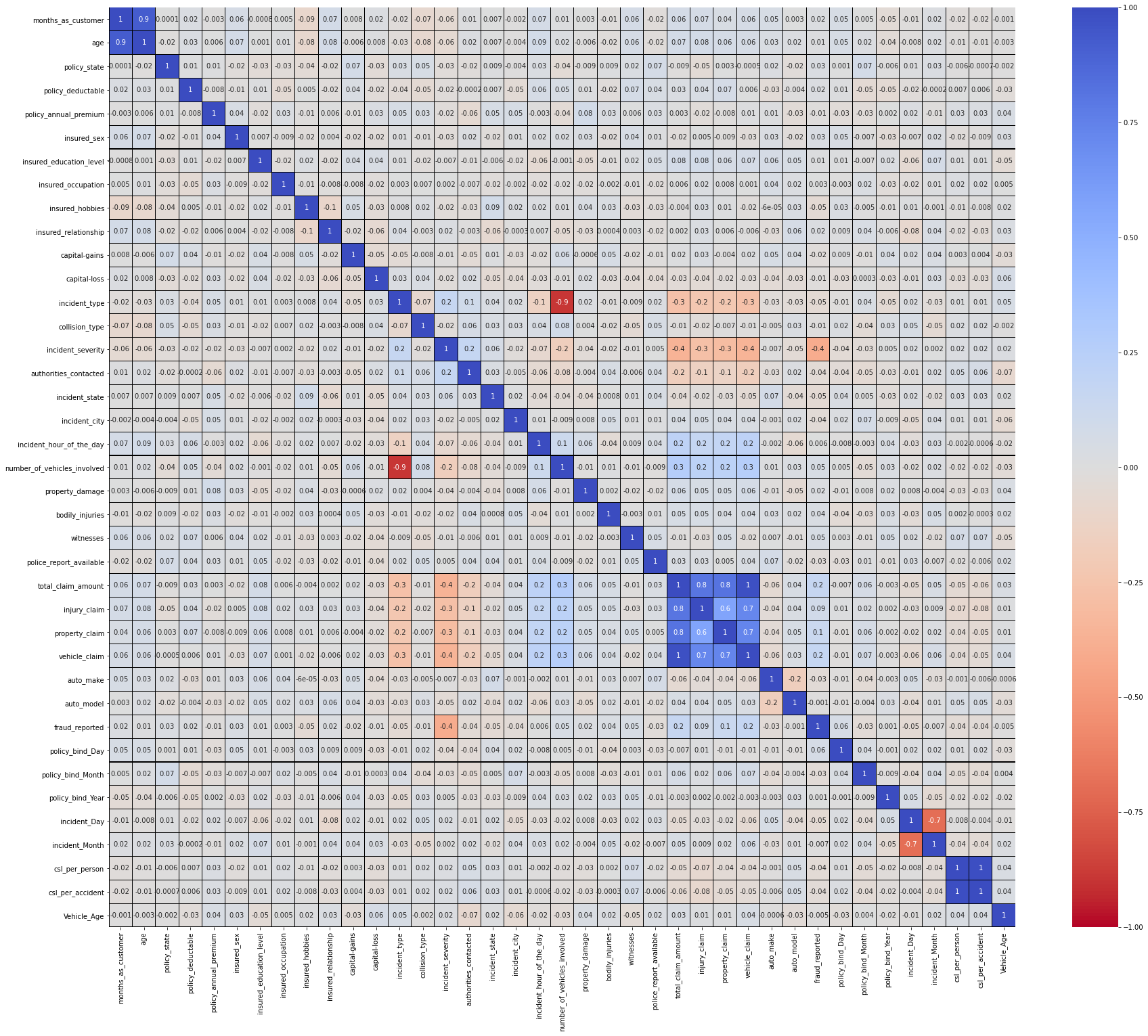
**Since our dataset contains object data, we need to encode them using any of the encoding methods. Here I have used label encoding method.**

****

**I have applied label encoding method to our cleaned dataframe new\_df and converted the categorical columns in to numerical.**

**Correlation between the label and features using Heat map**

****

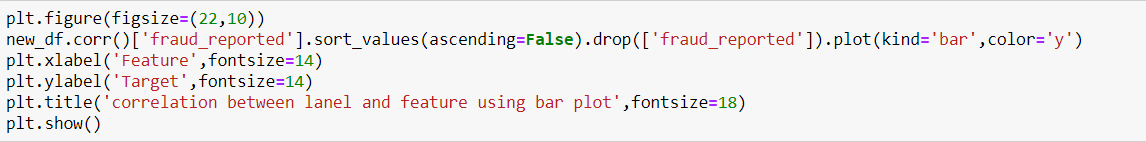
****

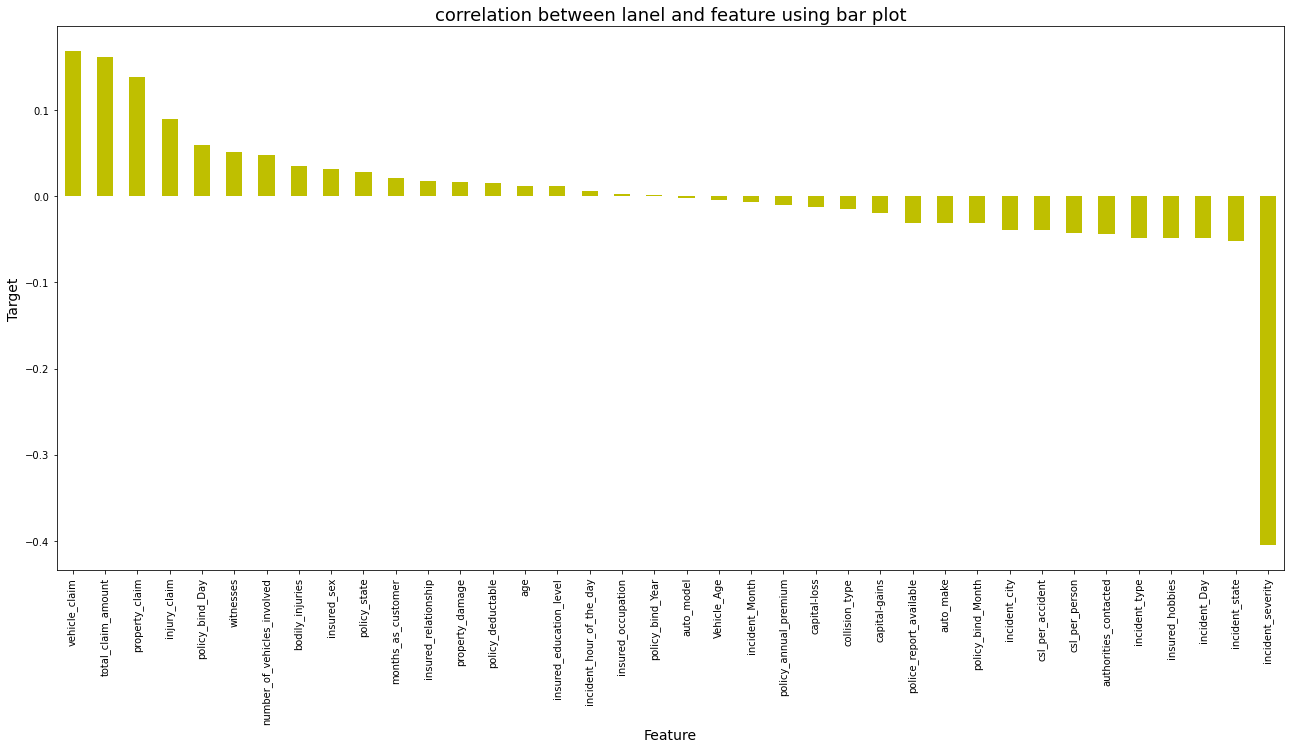
**This heatmap shows the correlation matrix by visualizing the data. we can observe the relation between one feature to other.**

**This heat map contains both positive and negative correlation.**

* **There is very less correlation between the target and the label.**
* **We can observe the most of the columns are highly correlated with each other which leads to the multicollinearity problem.**
* **We will check the VIF value to overcome with this multicollinearity problem.**

**To get the better insights from the heat map I have used bar plot to show the positive and negative correlation.**

****

****

* **From the bar plot we can observe that the columns policy\_bind\_Year, insured\_occupation and auto\_model age are very less correlated with the target. We can drop these columns if necessary.**
* **But for now, I am keeping these columns as it is, if I do not get the better accuracy then I can drop these columns in order to get better accuracy.**

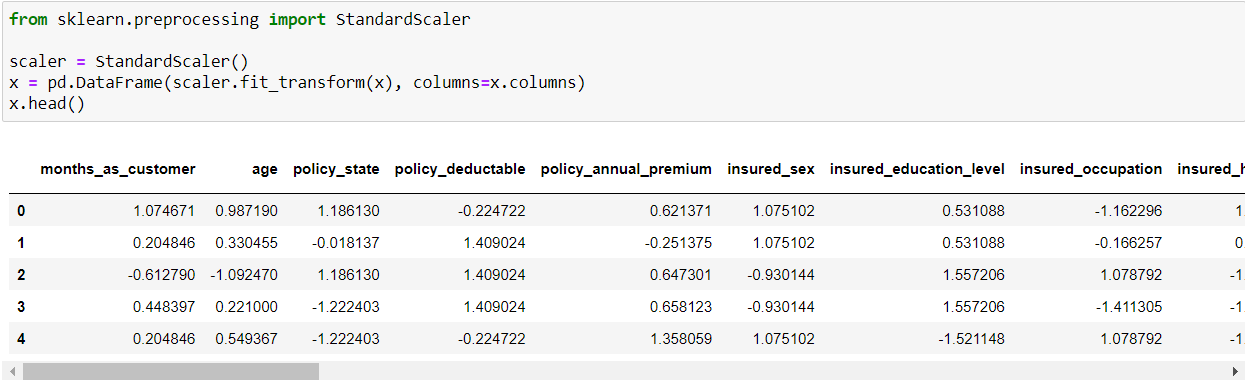
**4.Pre-Processing Pipeline**

**First, I have to separate the label and features to process my dataset.**

****

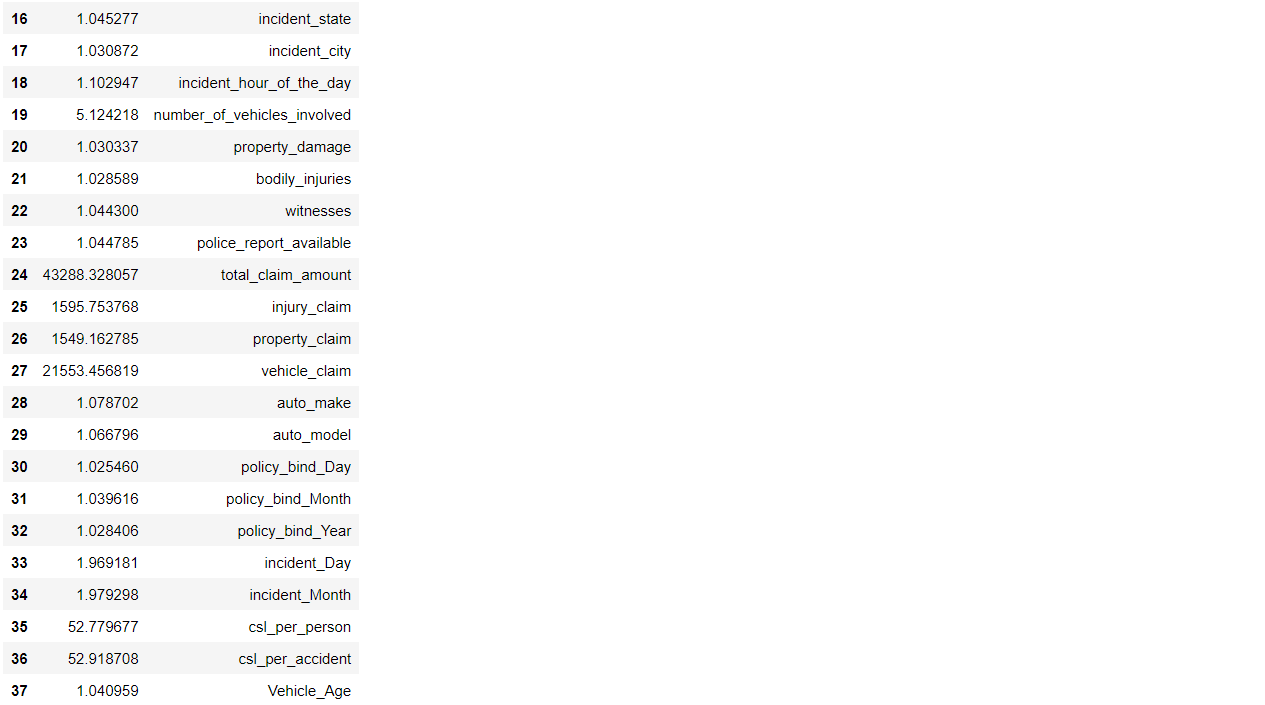
**I have separated independent and dependent features and stored them in x and y respectively.**

* **Now, I have to scale the data containing independent variables (x) in order to overcome with the data biasness.**
* **Since I have removed the skewness and outliers and my data is also normal so I can use Standard Scaler method to scale the data. If it is not the case then we could use Min Max Scaler.**

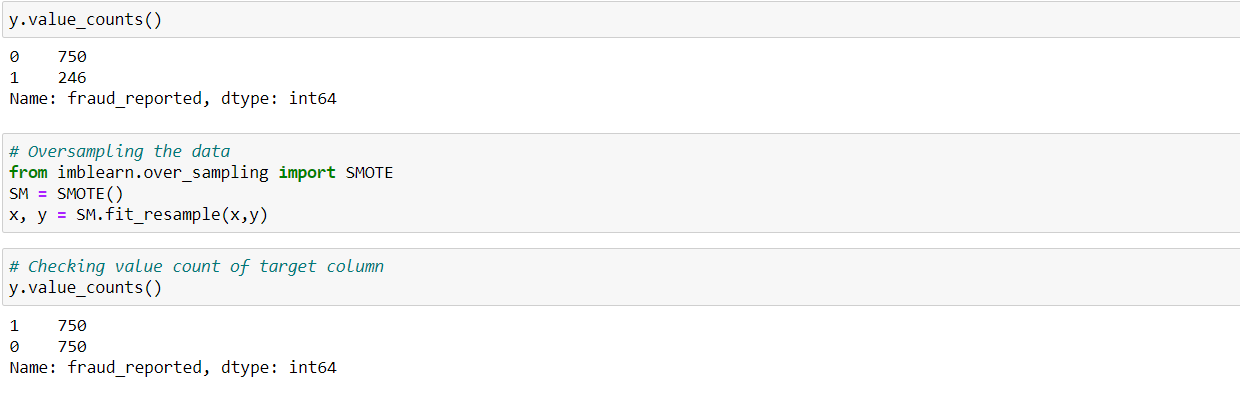
****

* **I have scaled the data using standard scaler method to overcome with the issue of data biasness.**

**In the heat map we have found some features having high correlation between each other which means multicollinearity problem so let's check the VIF value to solve multicollinearity problem.**

****

* **The acceptable range of VIF is below 10.**
* **I can observe high VIF in total\_claim\_amount, so I dropped this column first and again checked the VIF to confirm whether the multicollinearity issue solved or not.**
* **Again, I found high VIF in csl\_per\_accident column. So, I dropped that column too.**
* **After removing 2 columns my multicollinearity got solved by giving VIF below 10 in all the columns**
* **Since I have come across the data imbalance issue, I need to fix it either oversampling or under sampling the data.**
* **I always prefer to go with oversampling, because in under sampling there will be huge data loss.**

****

****

****

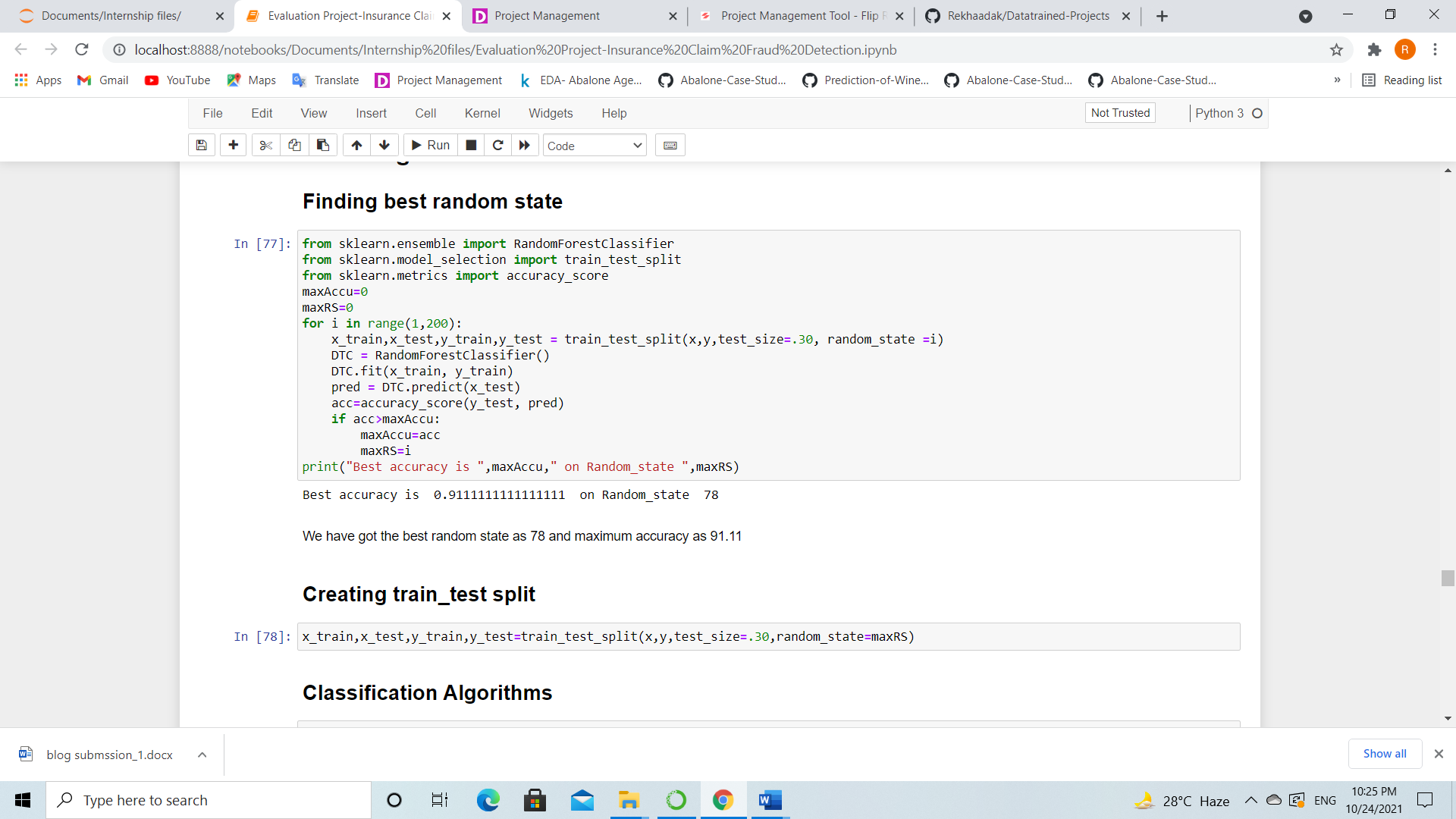
**The data is now balanced that we can observe in the count plot.**

**Since I have done all the pre processing and data cleaning, now my data is ready to build the model. Let’s get the predictions by creating some classification algorithms as it is a classification problem.**

**5.Building Machine Learning Models**

**Before building the models, first we need to find the best random state and accuracy using any one of the classification models.**

**Finding best random state and accuracy**

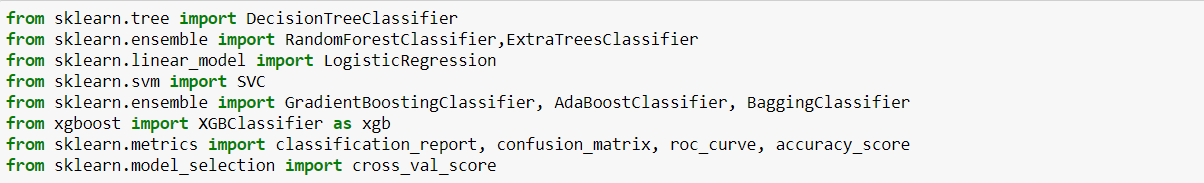
****

* **I have got best random state as 78 and best accuracy as 91.11% using Random Forest Classifier.**
* **Now let’s create new train and test and fit them into the models to find our ideal model.**

****

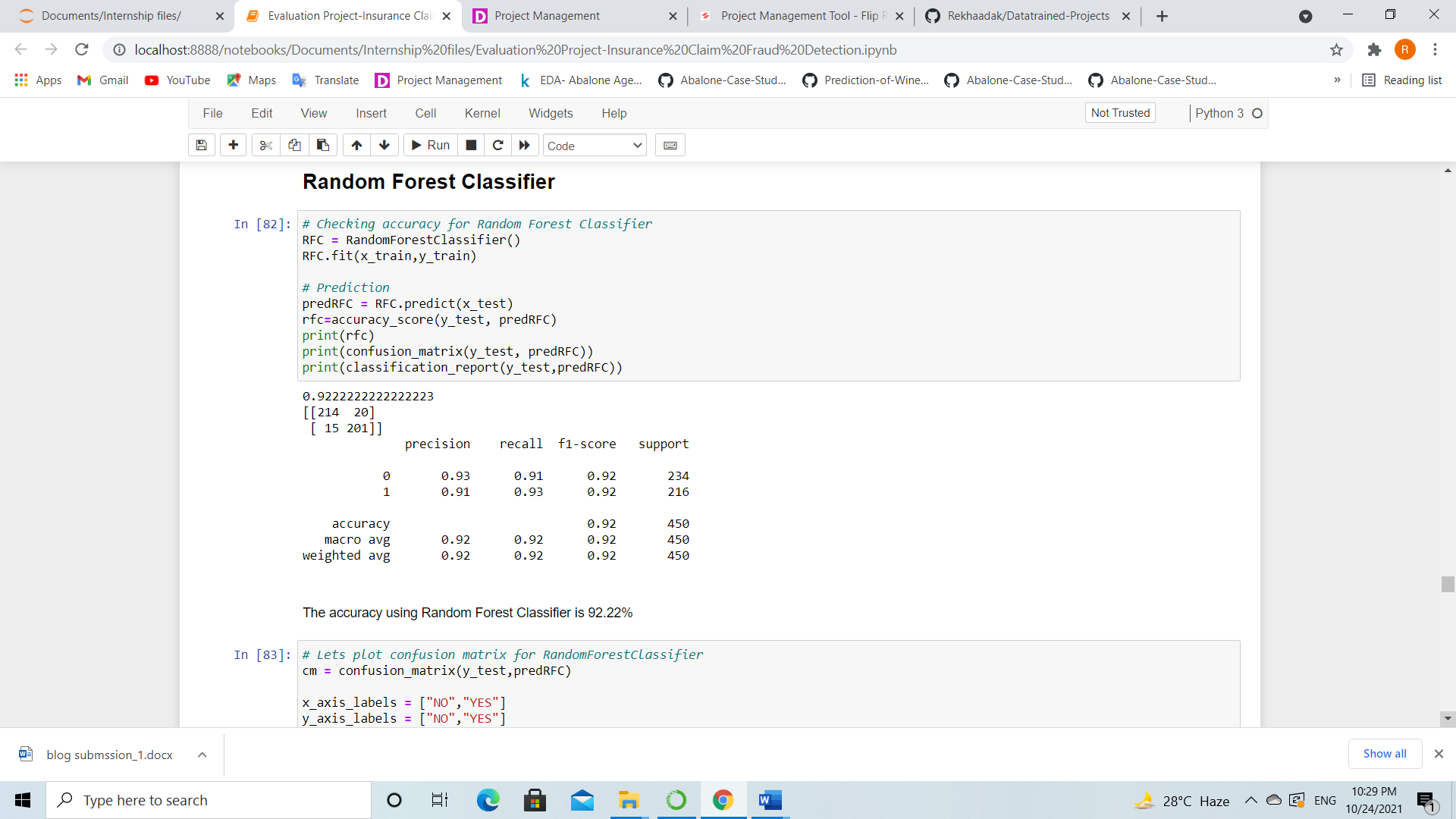
**I have created new train and test data as x\_train, x\_test and y\_train, y\_test.**

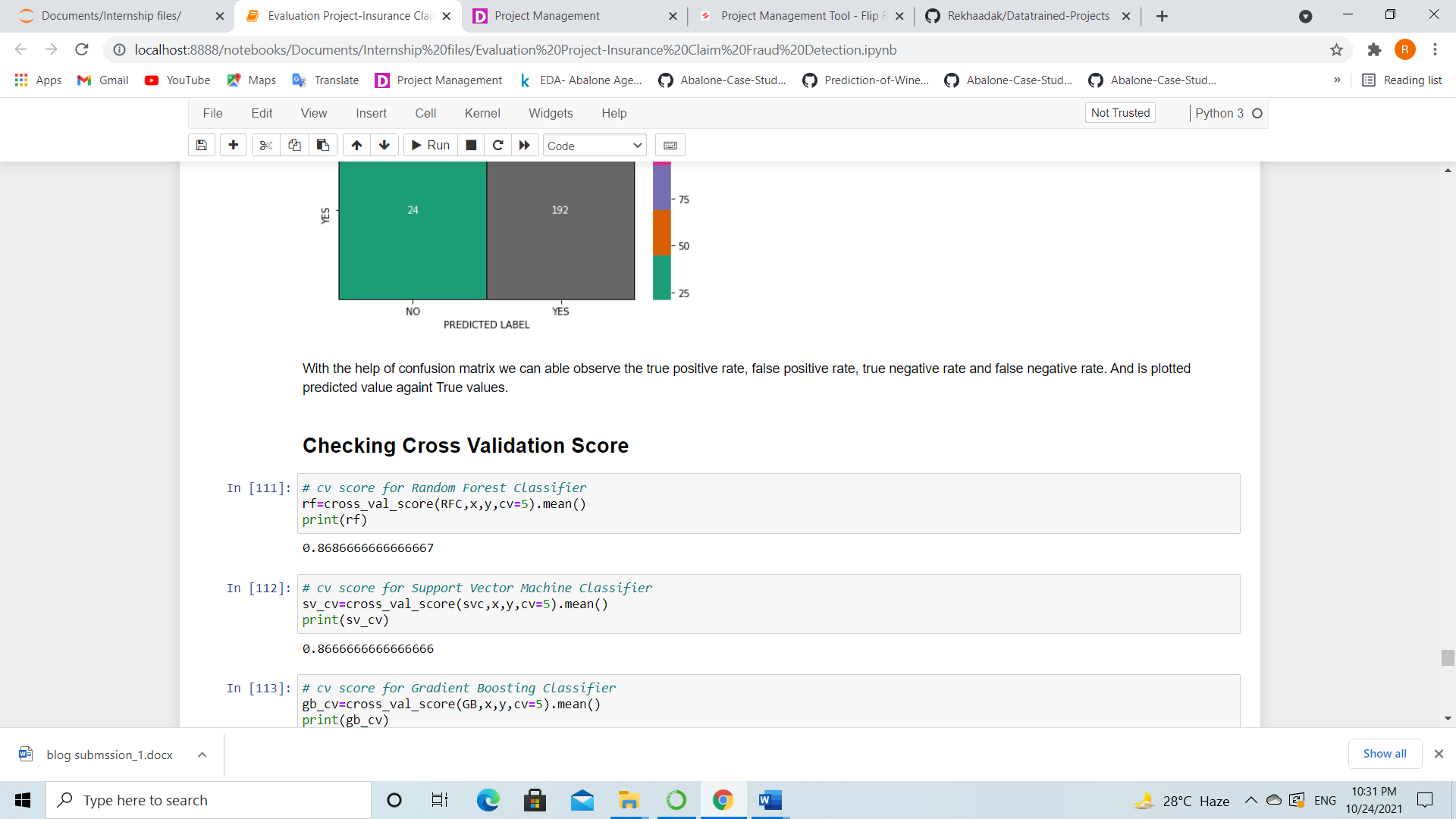
**Classification Algorithms**

****

* **I will be using the above classification algorithms to get the ideal one for prediction.**
* **I have used evaluation metrics like classification report, confusion matrix, roc score and accuracy score. Also used cross validation score to get the difference from the model accuracy.**

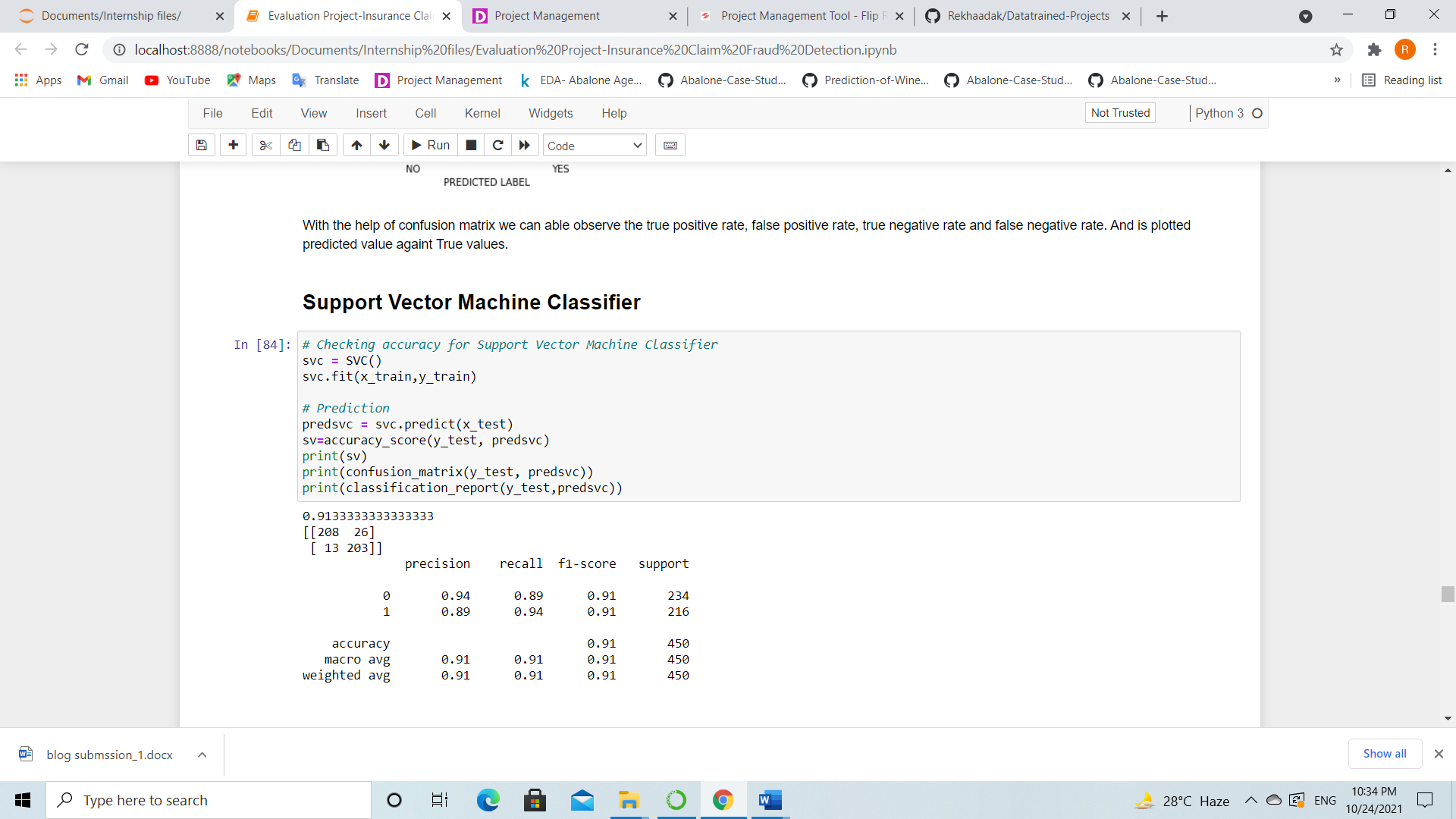
**Random Forest Classifier**

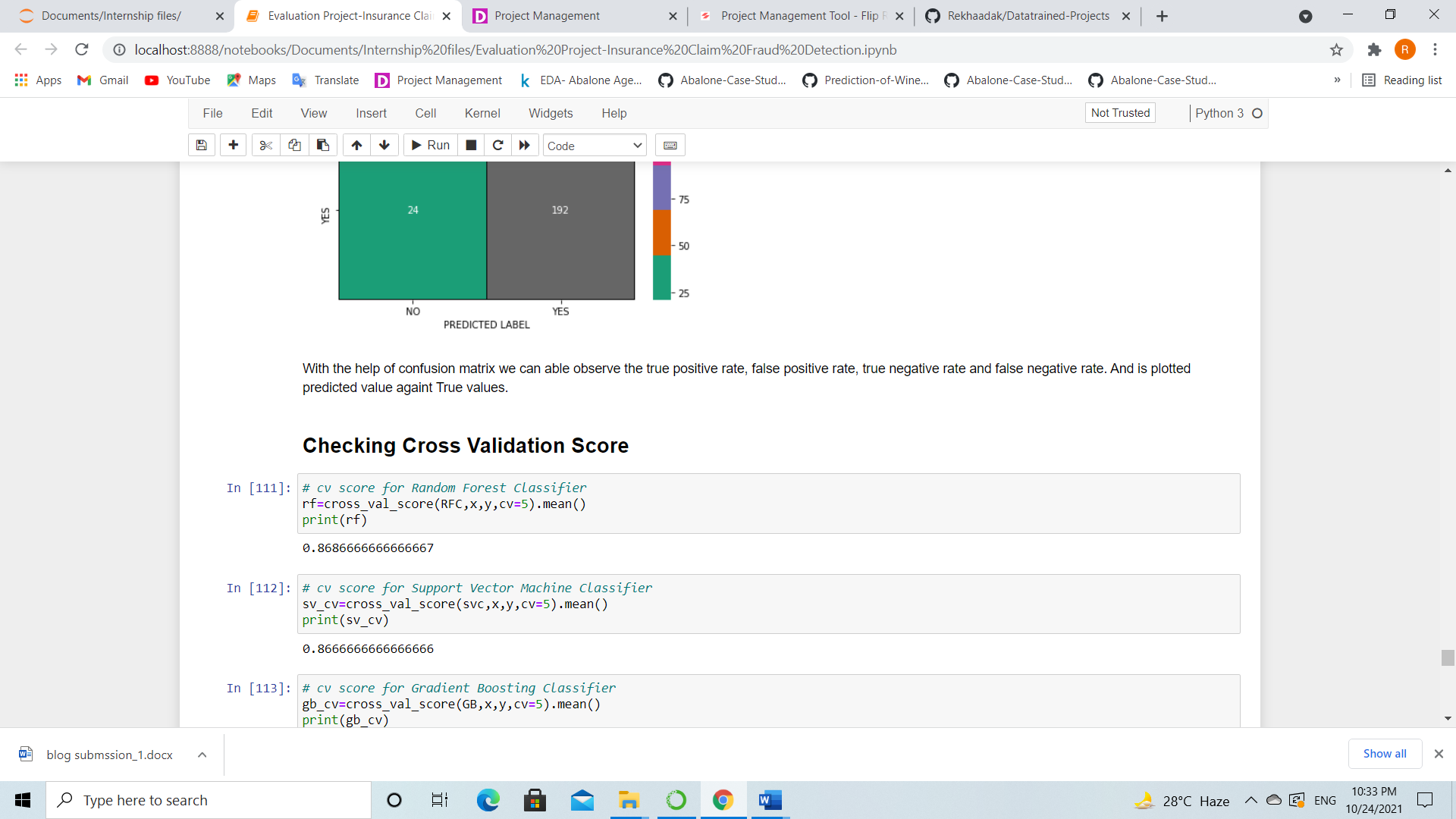
****

****

* **Created a Random Forest Classifier and it is giving good accuracy as 92.22% and cross validation score is 86.86%. It is giving best accuracy but let’s check other models.**

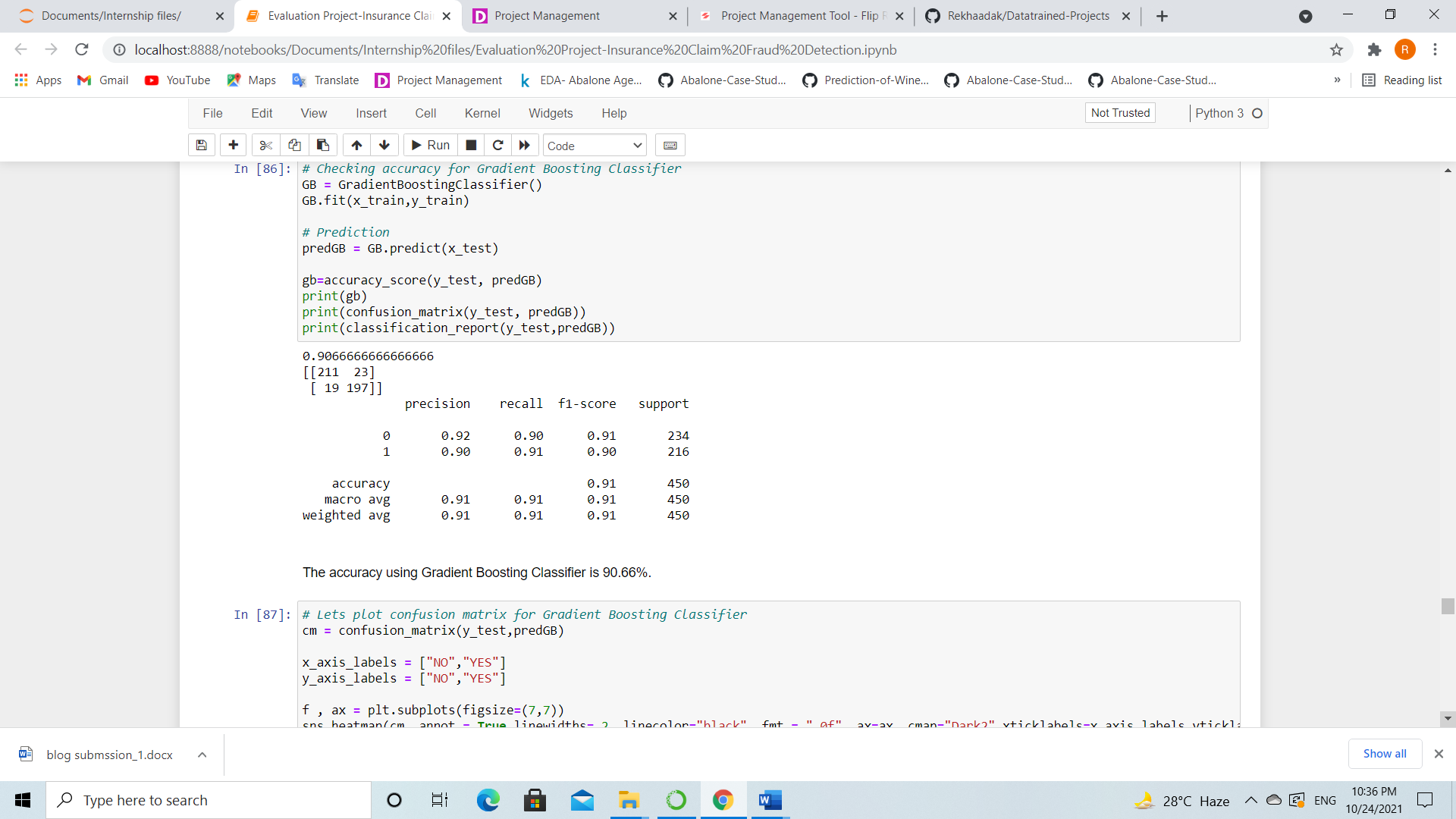
**Support Vector Machine Classifier**

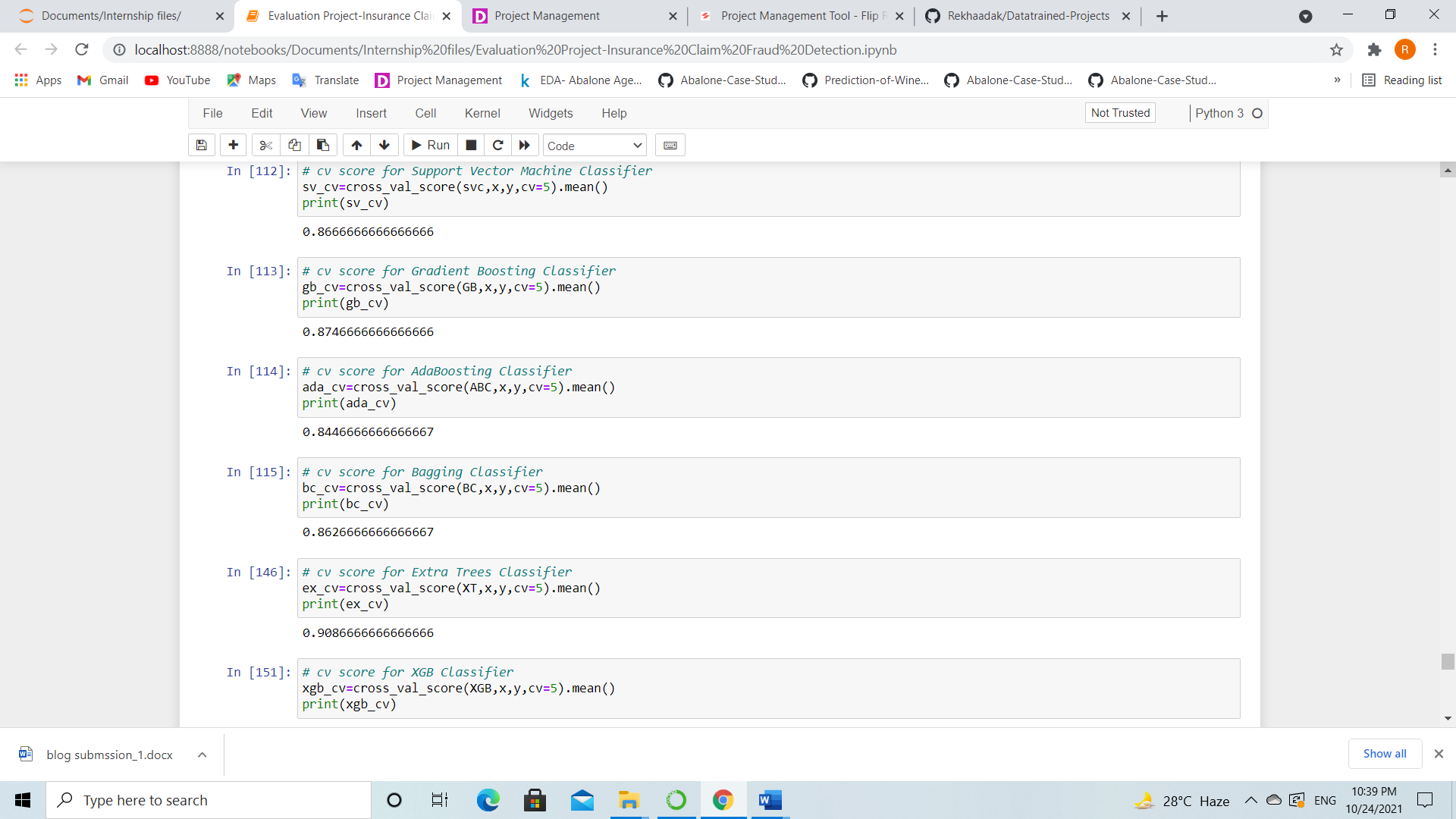
****

****

* **Support Vector Machine Classifier giving 91.33% accuracy and cross validation for this is 86.66%.**

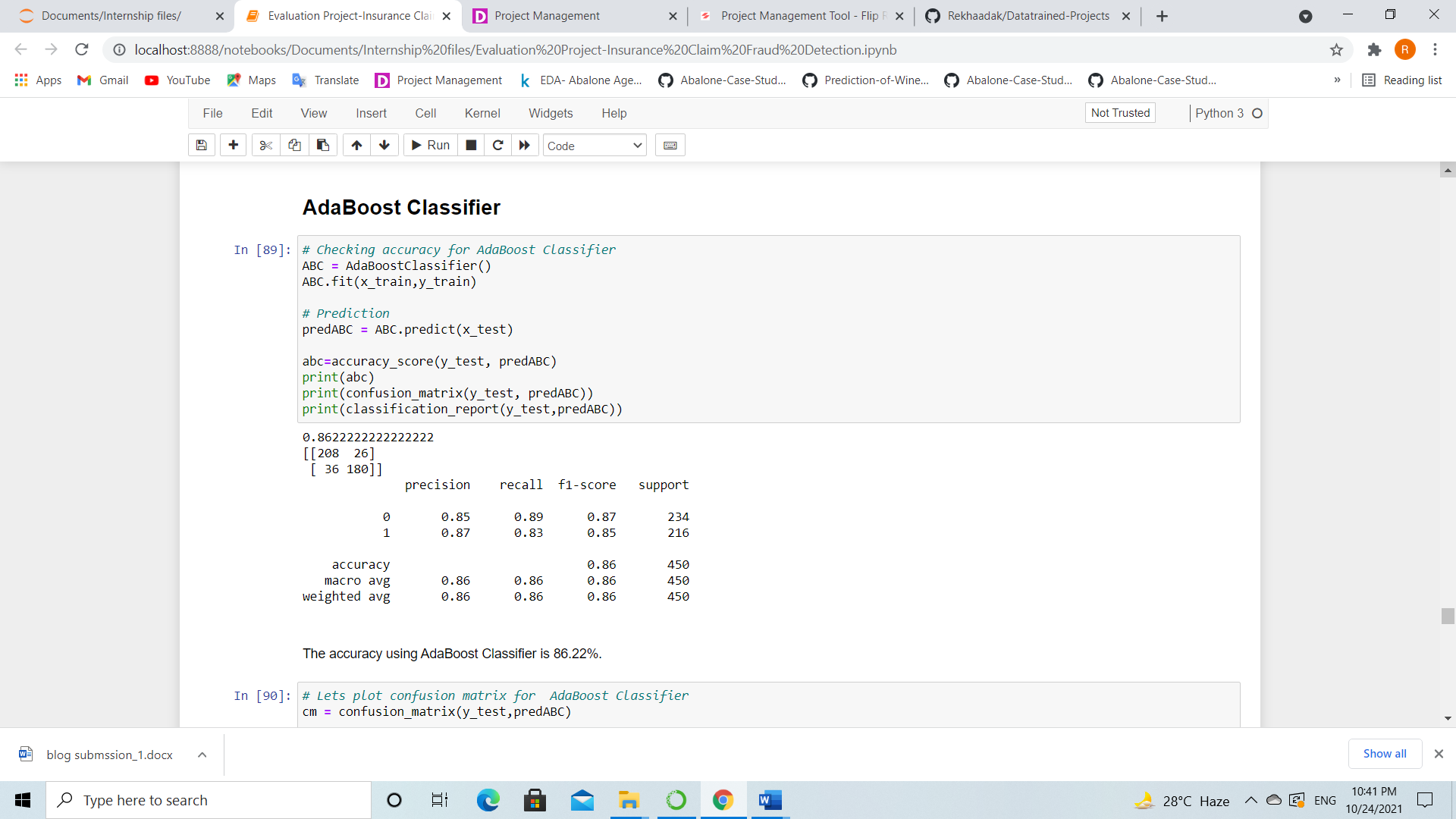
**Gradient Boosting Classifier**

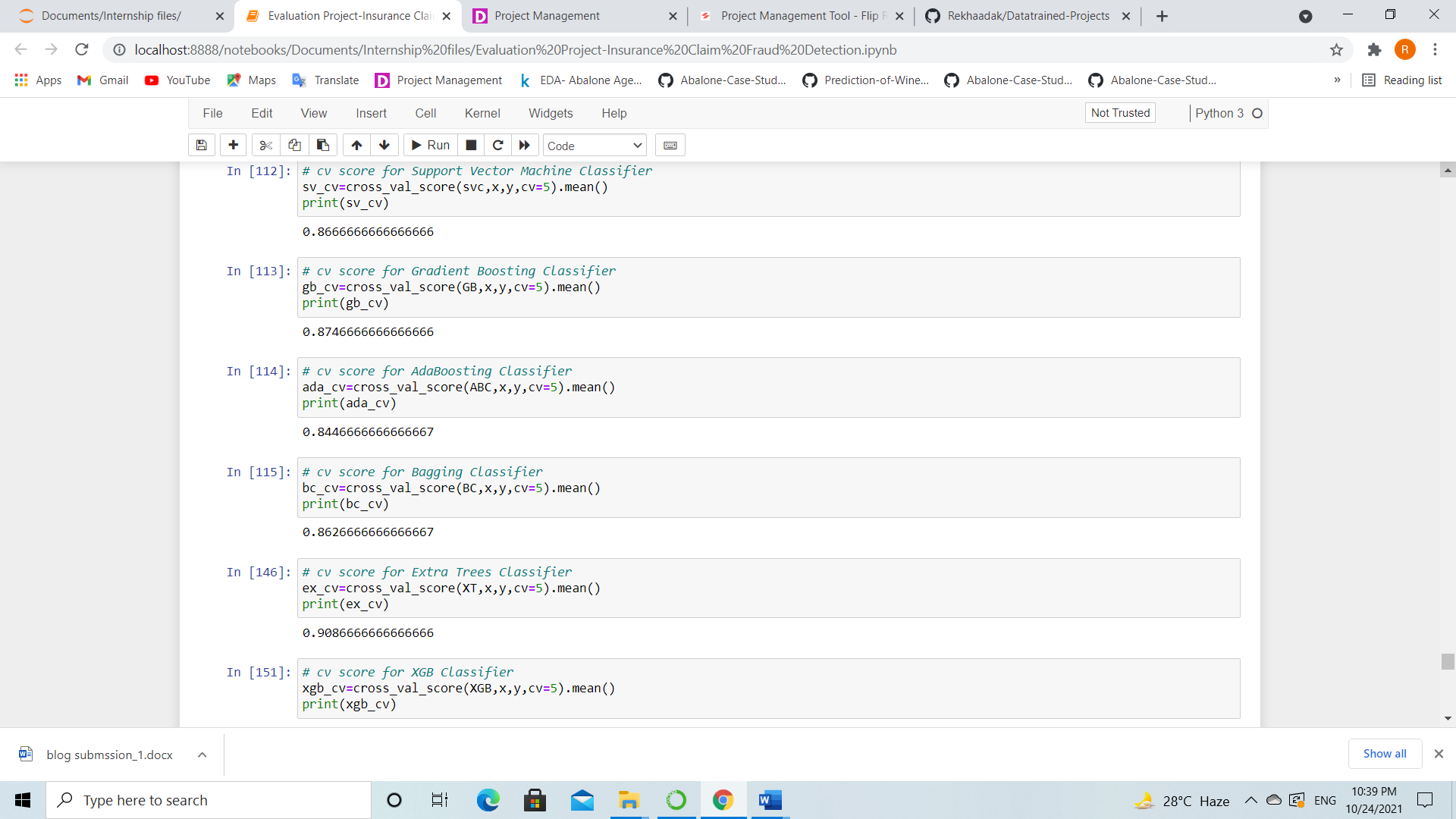
****

****

* **Gradient Boosting Classifier giving 90.66% accuracy and the CV score is 86.46%.**

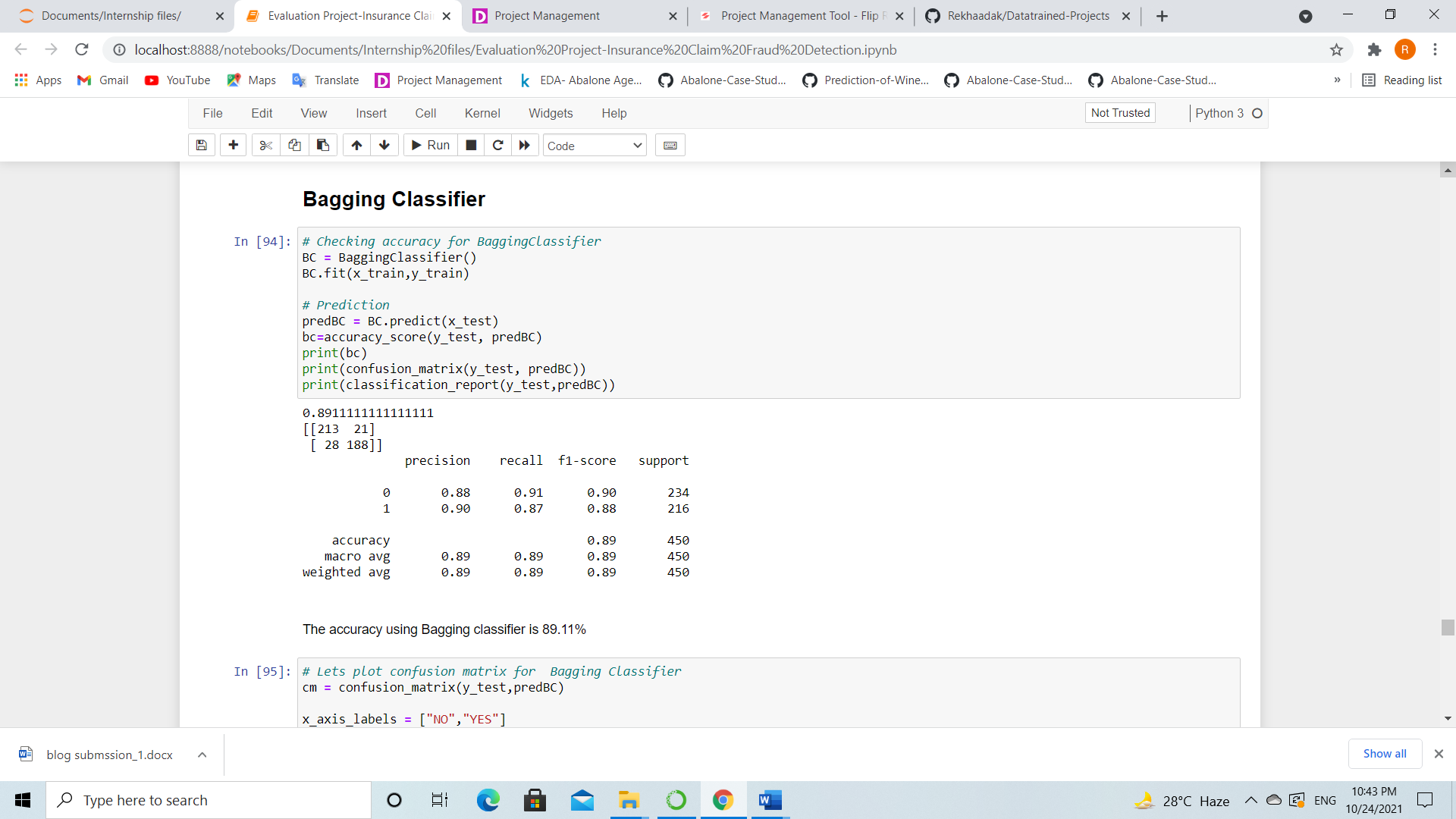
**AdaBoost Classifier**

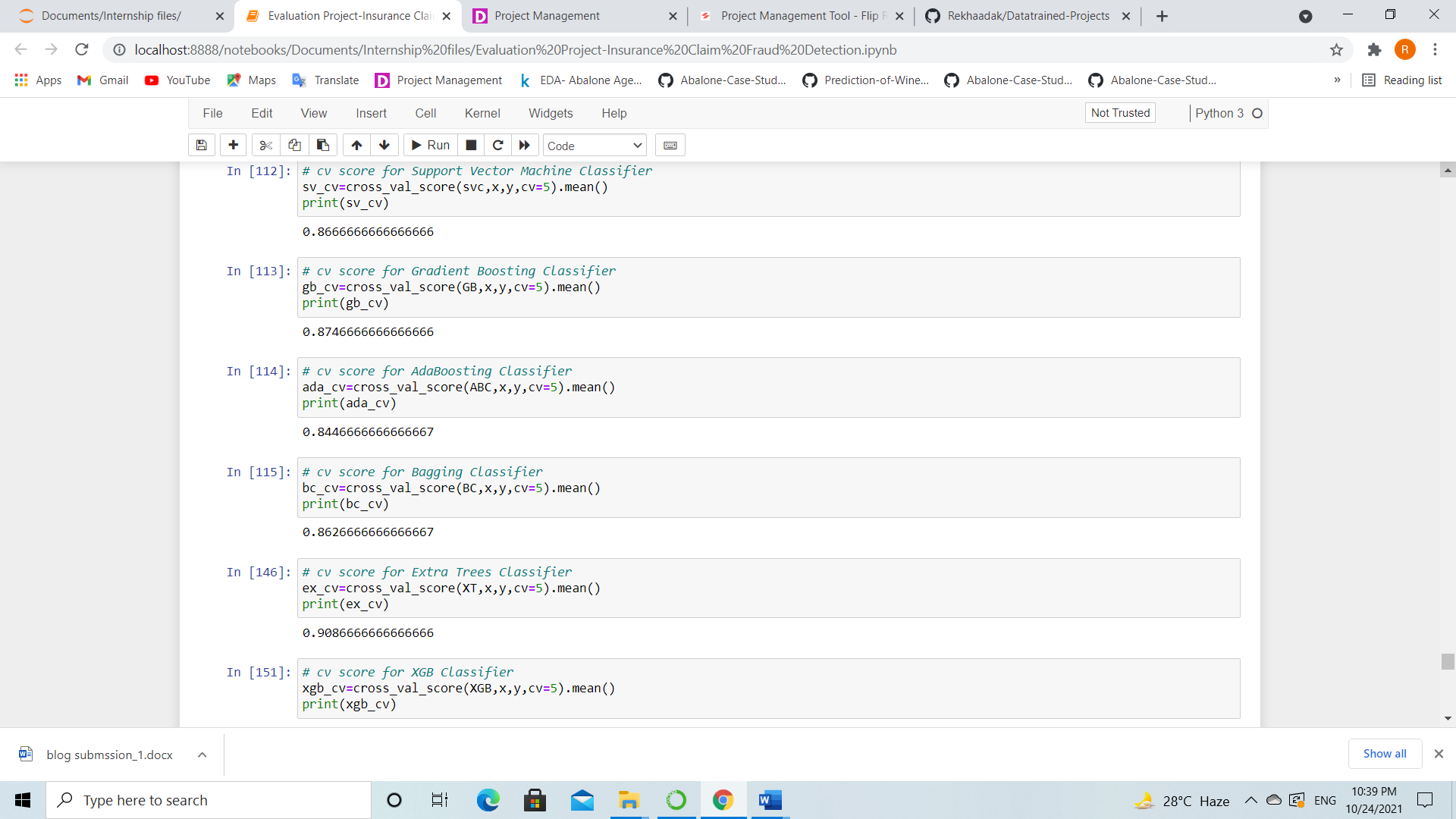
****

****

* **AdaBoost classifier giving 86.22% accuracy and the CV score is 84.66%.**

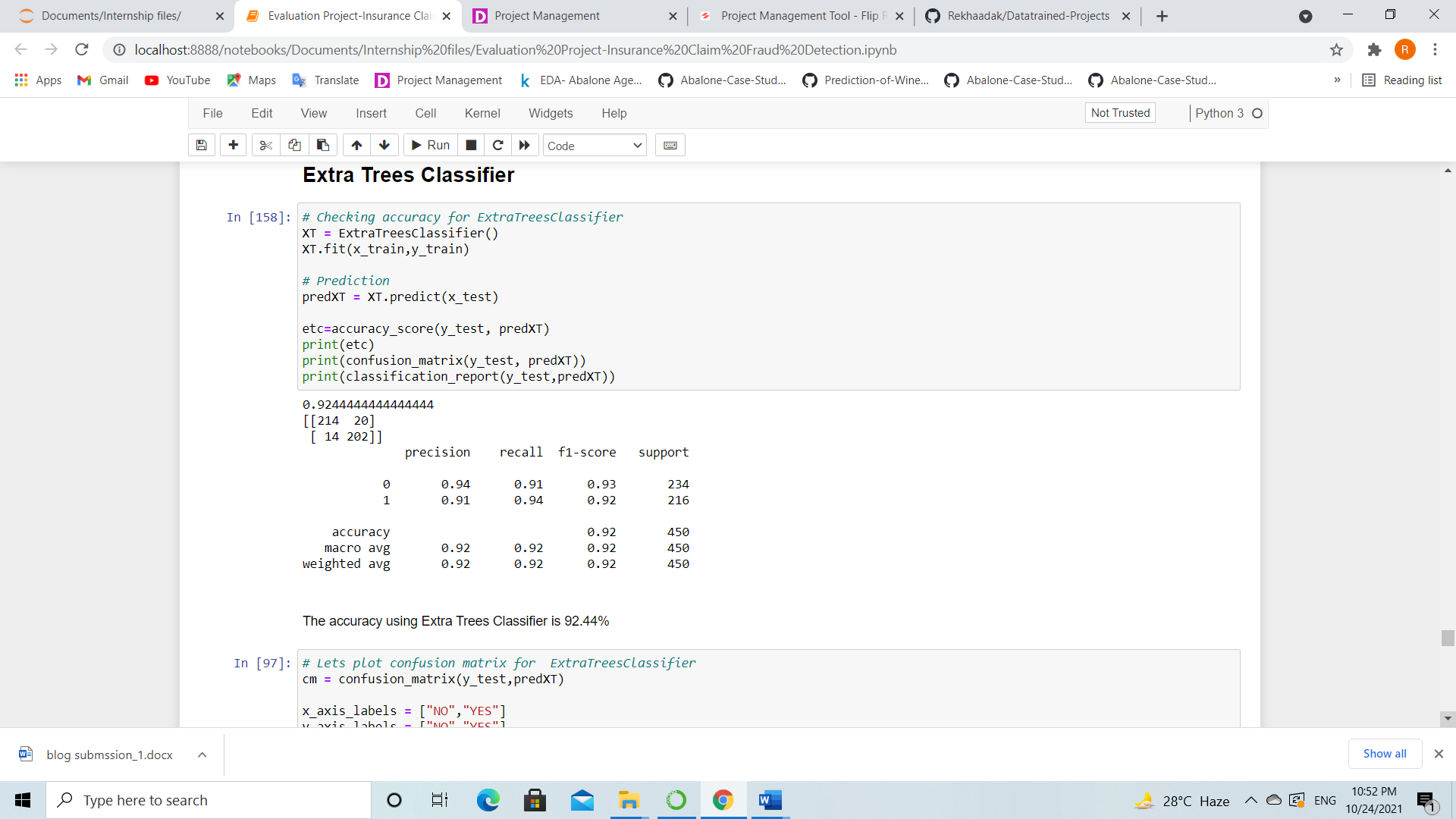
**Bagging Classifier**

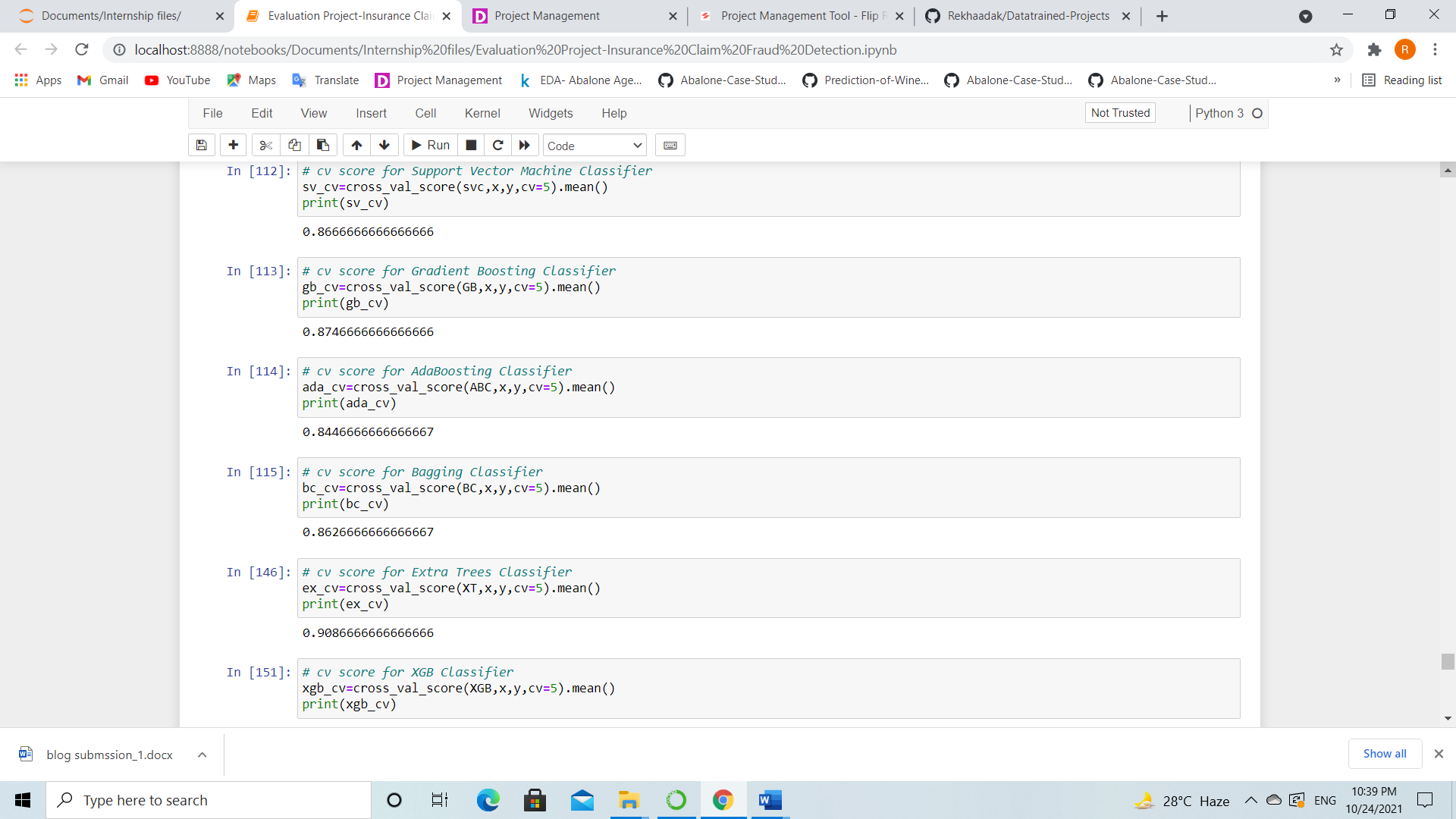
****

****

* **Bagging Classifier giving 89.11% accuracy and CV score is 86.26%.**

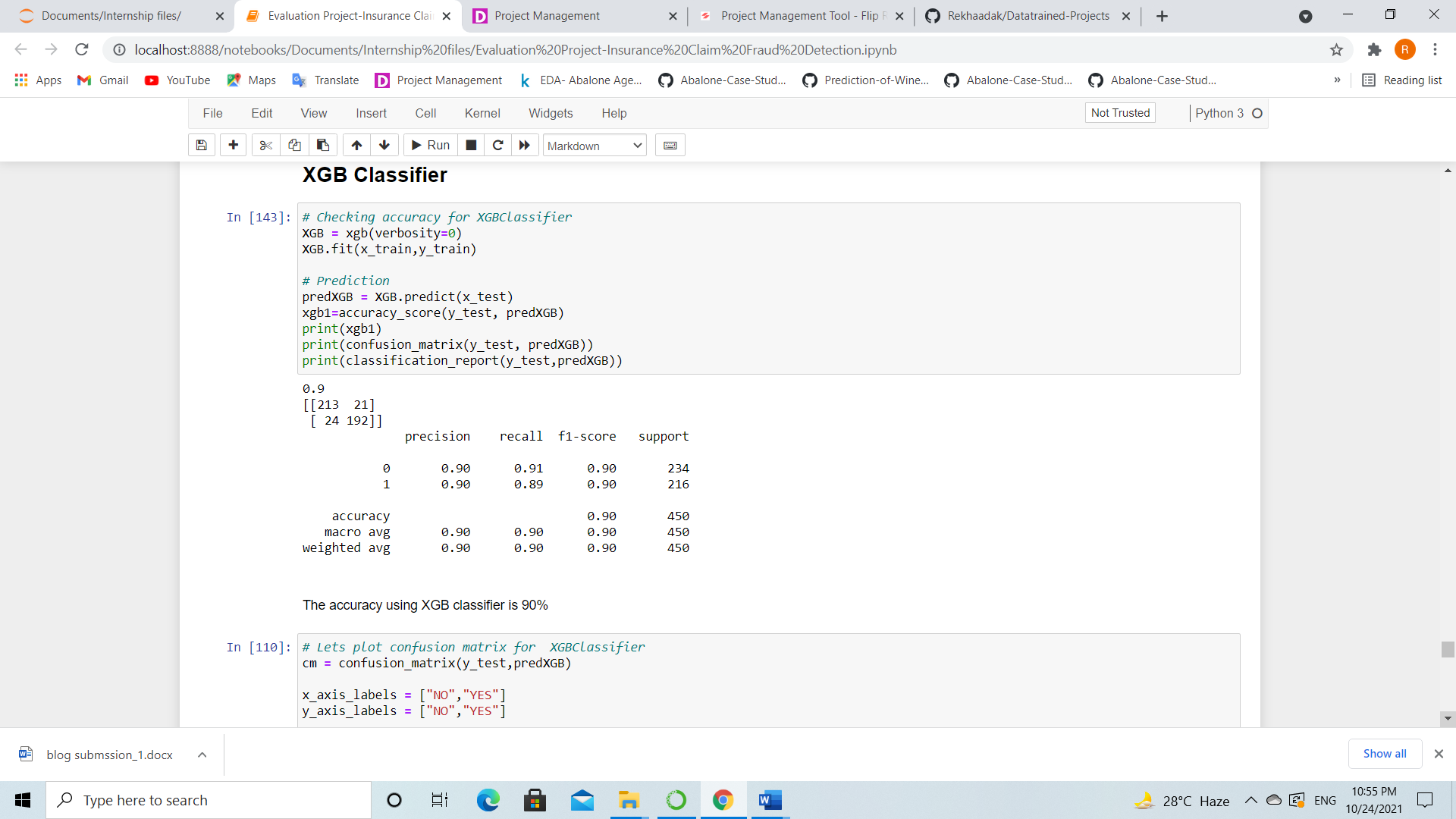
**Extra Trees Classifier**

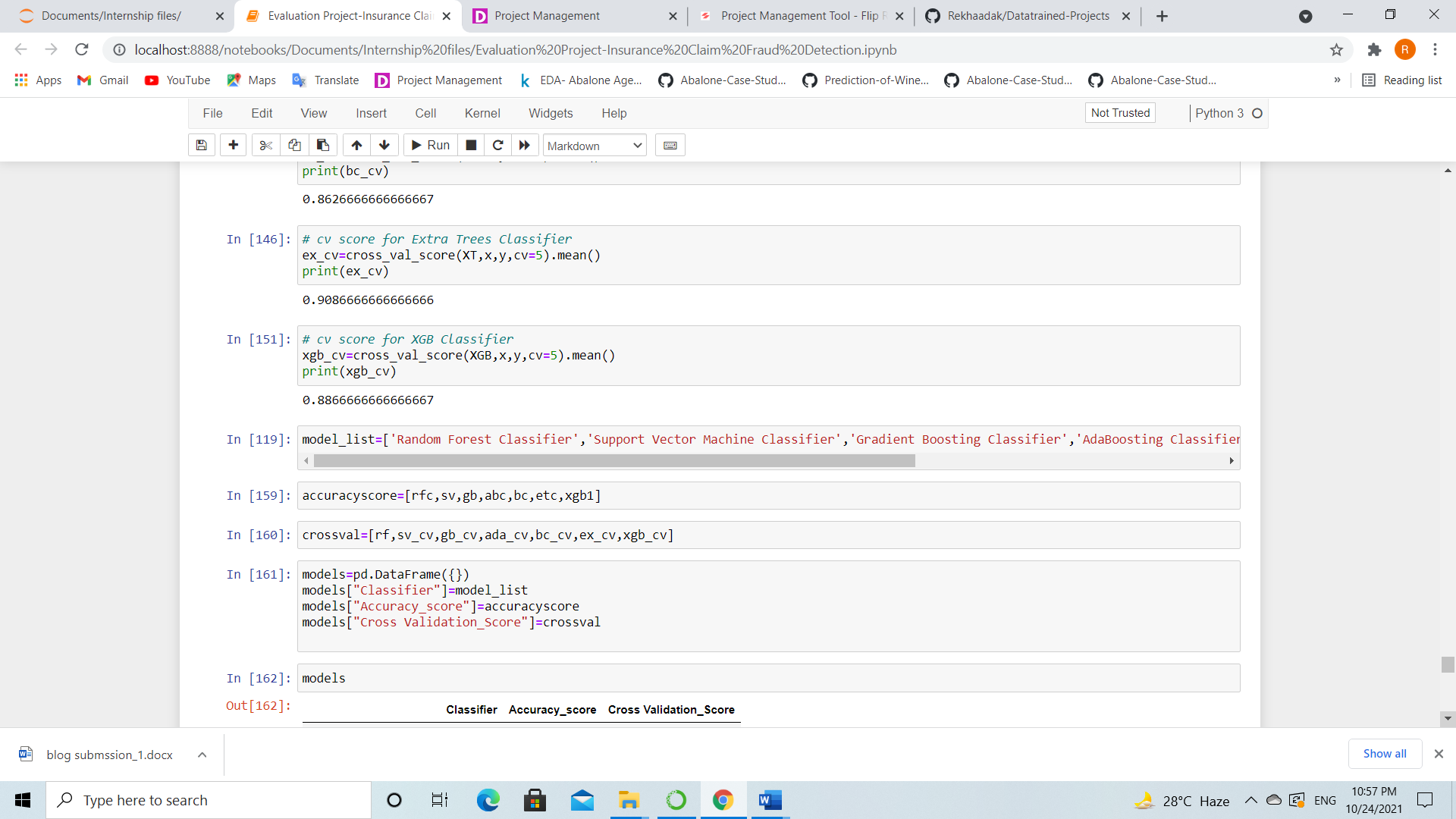
****

****

* **Extra Trees Classifier giving 92.44% accuracy and the CV score is 90.86%.**

**Extreme Gradient Boosting Classifier (XGB)**

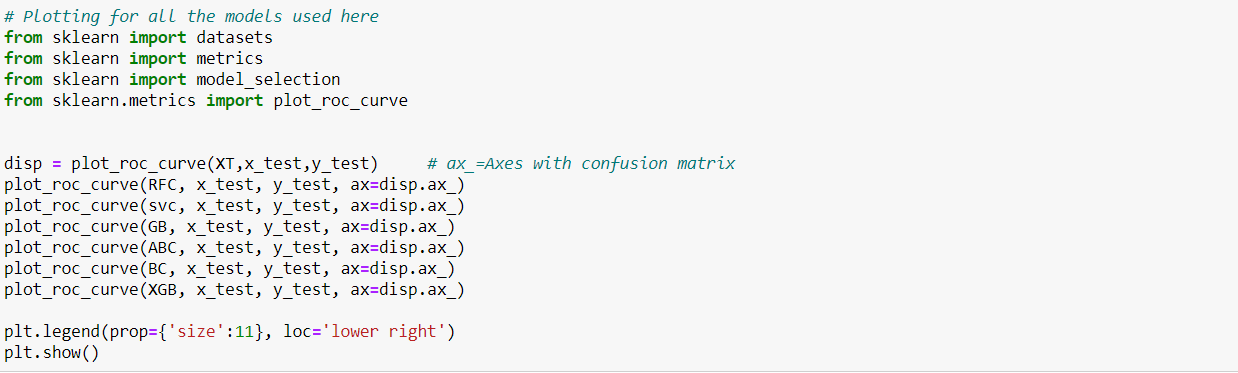
****

****

* **The XGB Classifier giving accuracy as 90% and the CV score as 88.66%.**

****

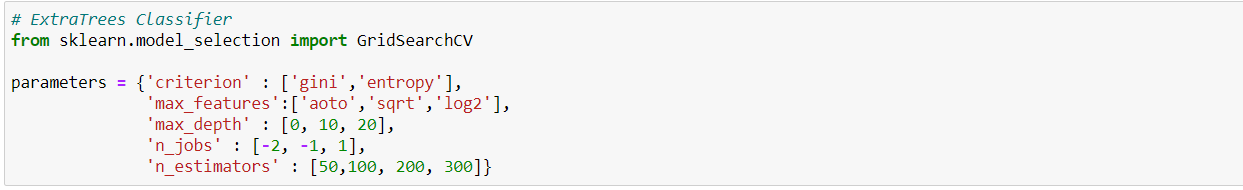
**ROC-AUC curve for all the models**

****

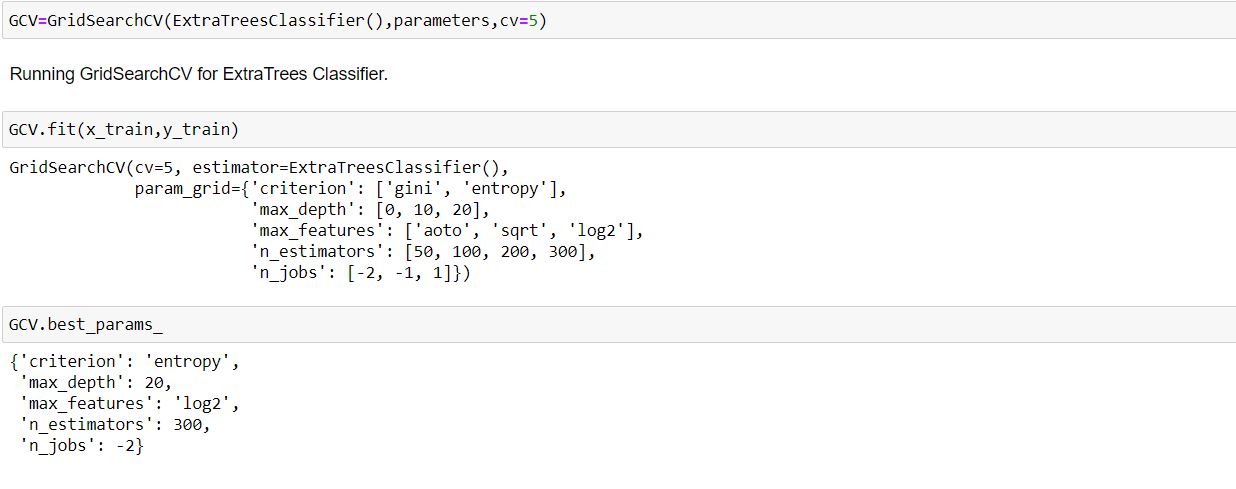
****

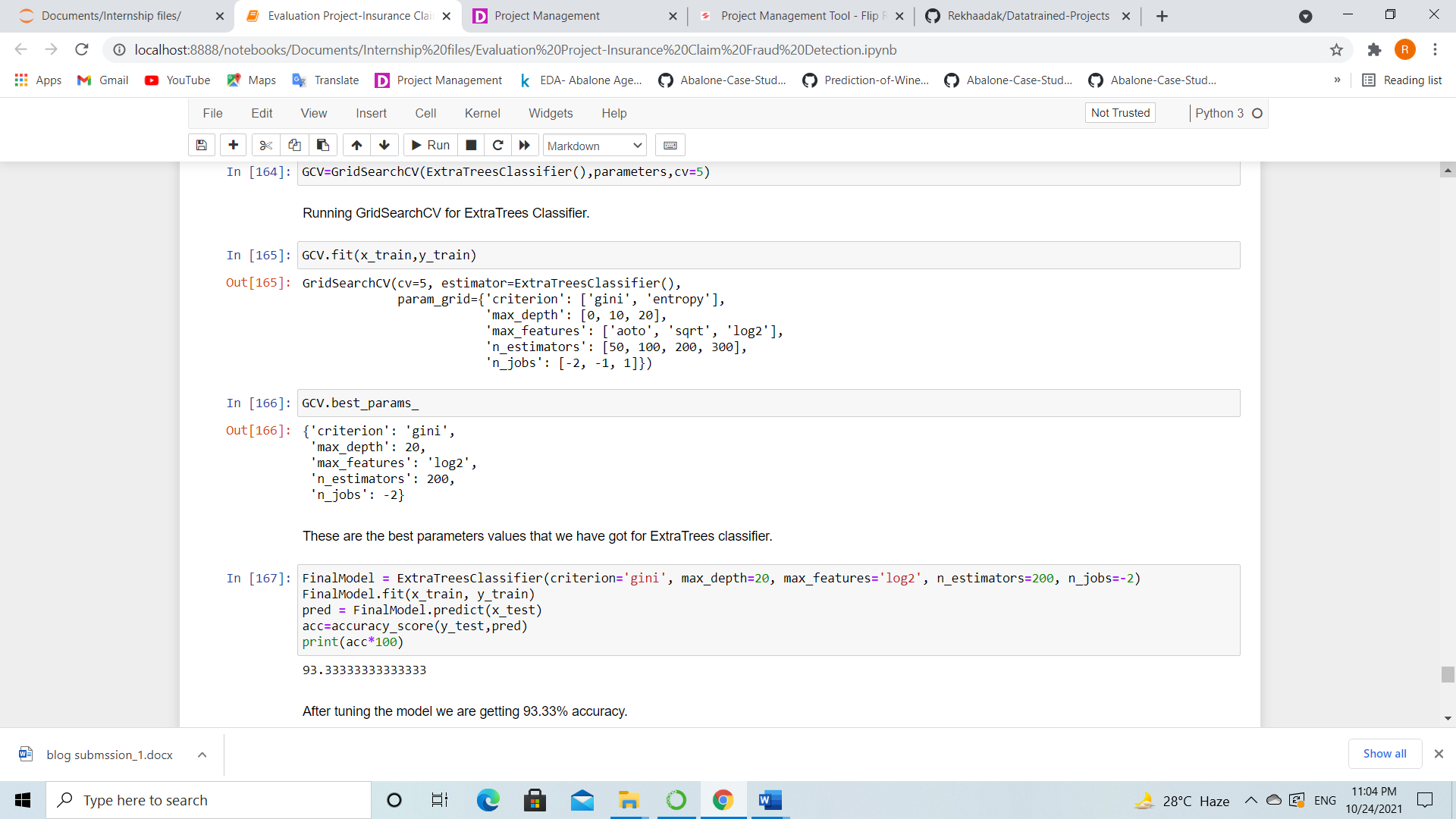
* **From the difference between the accuracy score and cross validation score, we can conclude that Extra Trees Classifier as our best fitting model for prediction which is giving very less difference of 1.58%.**
* **Also from the above ROC curve, I can notice that the Extra Trees Classifier has high AUC as 98%.**
* **Based on these results I am performing hyper parameter tuning to improve my accuracy.**

**Hyper Parameter Tuning**

****

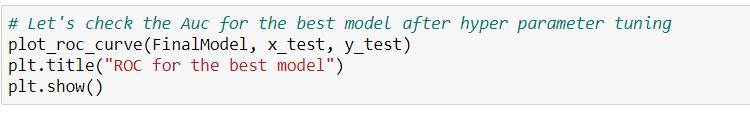
**By using above parameters, I am tuning my best model and after tuning I have to choose the best parameters from the above list. Let’s get the best parameters.**

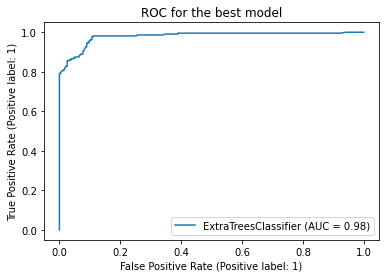
** I have used GridSearchCV and find the best parameters. Now I will use these parameters to get the accuracy of the best model.**

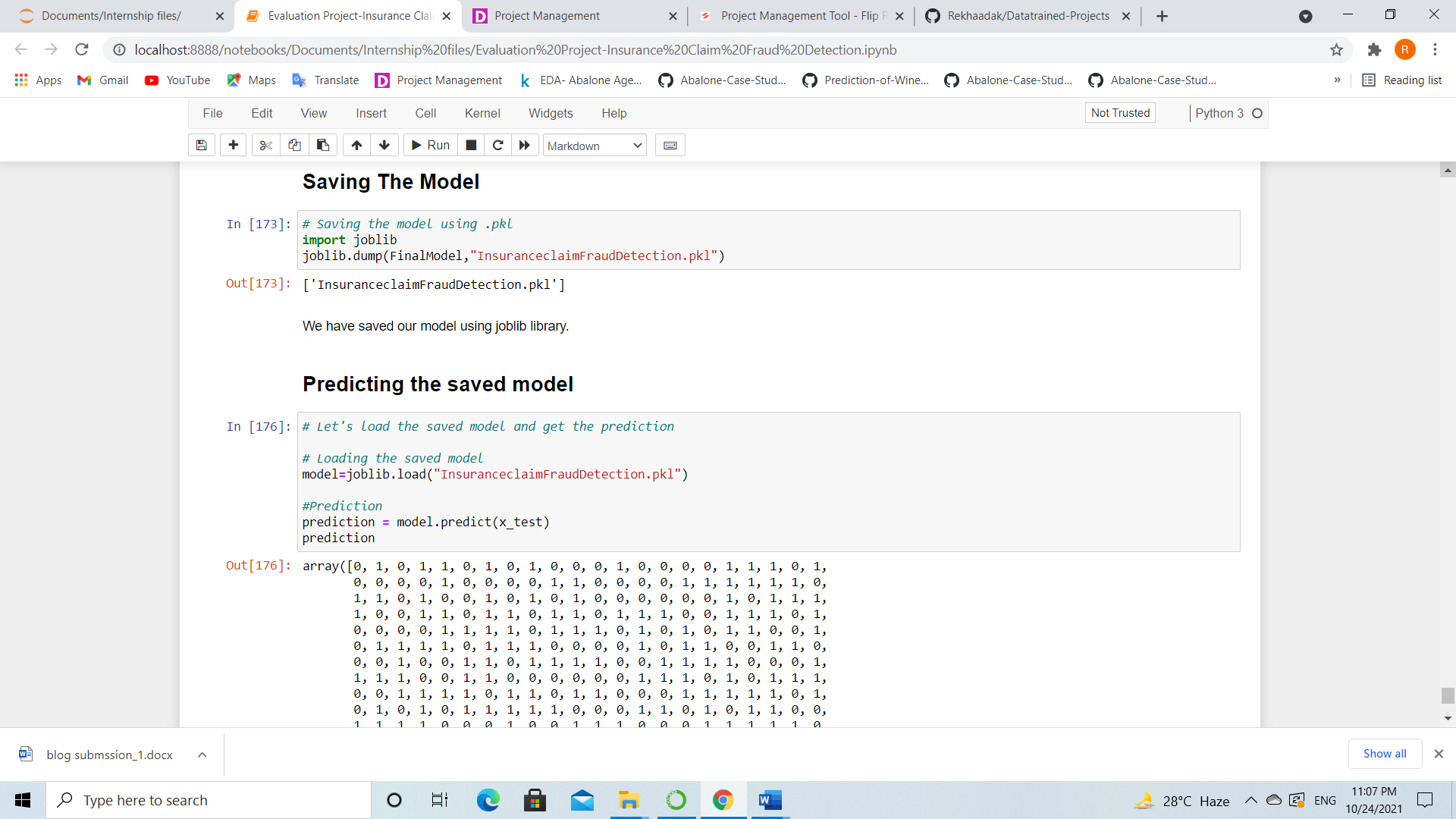
****

**After tuning the model, the accuracy increased to 93.33%**

**Now we will plot ROC curve and compare the AUC for the best model.**

****

****

**The AUC value is also remained same as before. Since I have done with the model building, it’s time to save the model. Here I have saved my model as below.**

****

**The actual and predicted values are almost same, that means our model worked well.**

**6.Concluding Remark**

In this project we have gone through the feature engineering which is the most crucial thing and removed the outliers and skewness. Also handled the categorical columns by encoding the data, scaled the data and at last, we built different classification models to predict whether the insurance claim is fraudulent or not and performed the hyper tuning to improve the model by using different parameters.

With the help of above techniques, our model is able to predict the fraudulent report with the accuracy of 92.44%. Also, we have seen the actual and predicted values are almost same that means our model worked correctly. Building machine learning models for such problems can help the insurance companies to choose the correct insurer. So, Machine learning techniques are very useful to solve this kind of problem.