Firstly, I am starting about a project I did on “**Auto insurance fraud claim detection**” using an number of different classifiers and ensembles.

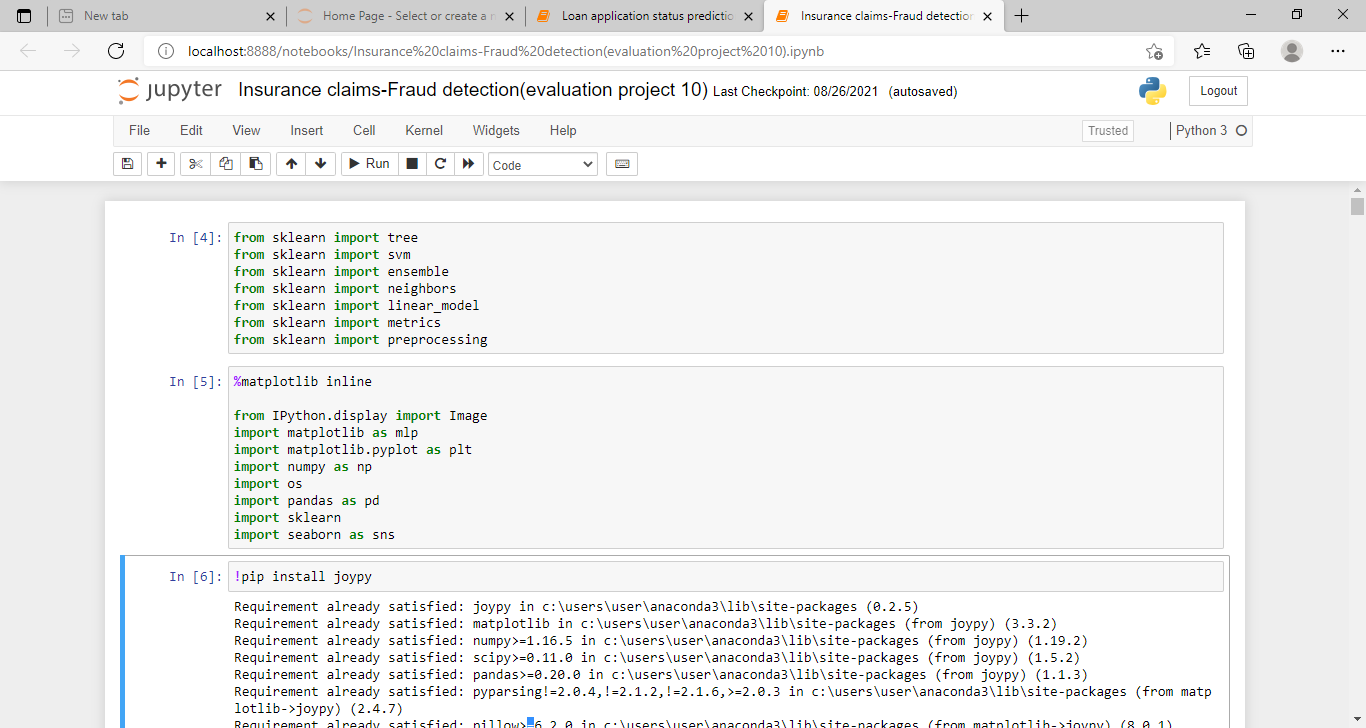
**Problem Statement :**

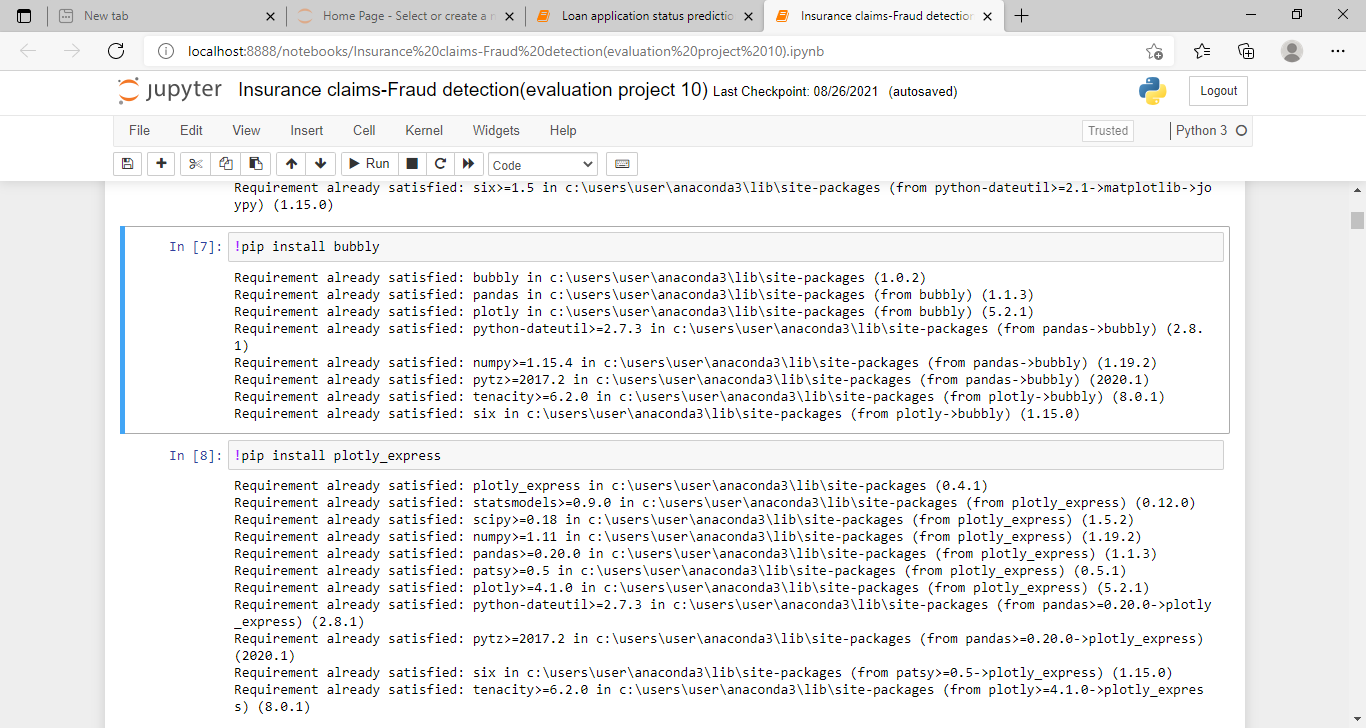
The main goal of this project is to build a model that can be detect auto insurance fraud. The main only one challenge behind the fraud detection in machine learning is that frauds are far less common as compared to legit insurance claims. This type of problems is also known as imbalanced class classification.

From the Criteria of success, The model should be able to classify if a claim is a fraud or not on a data set that has not seen accurately. This is measured with F1 score and compared against the base line.

By importing all the libraries first into the program to get exact result for the fraud detections.

**Importing all libraries:**

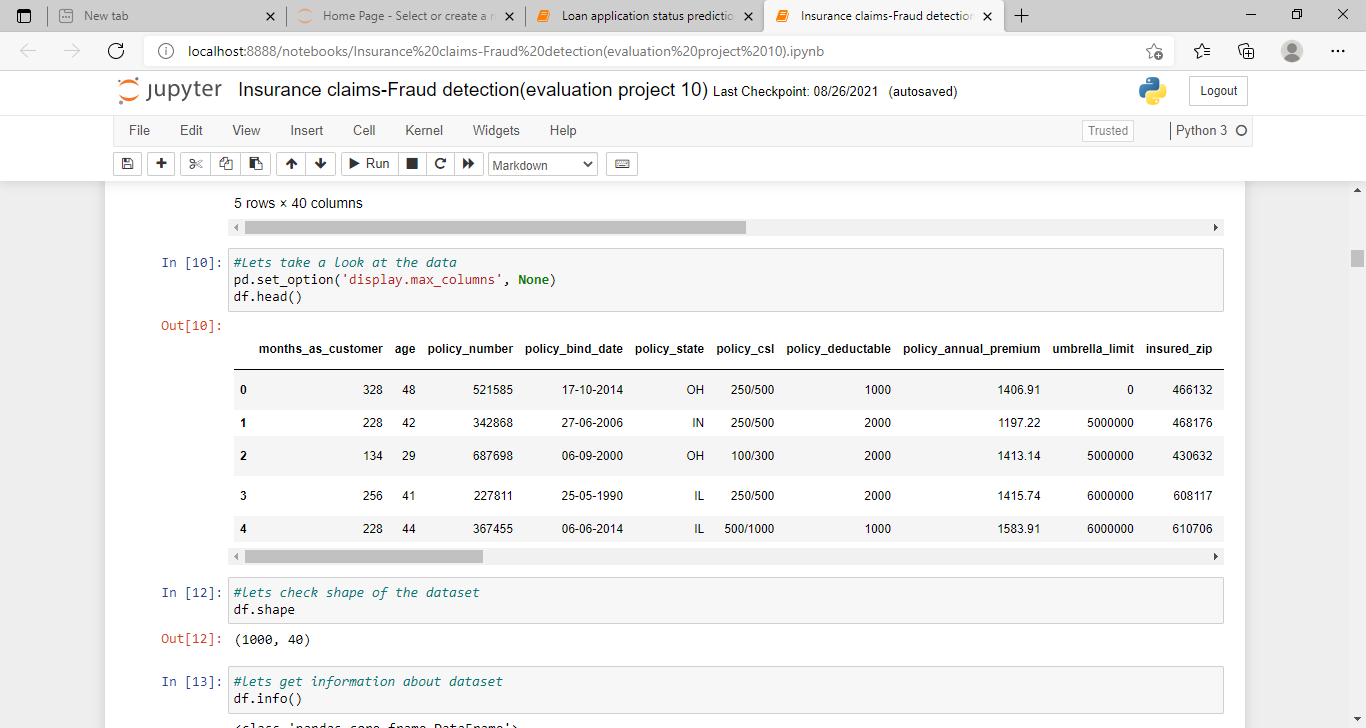
****

****

**From the Data Set:**

The most inspiration for this project is to perform classification on imbalance class data sets , in particular fraud. Fraud data sets are very hard to come by and often unable due to its sensitive nature.

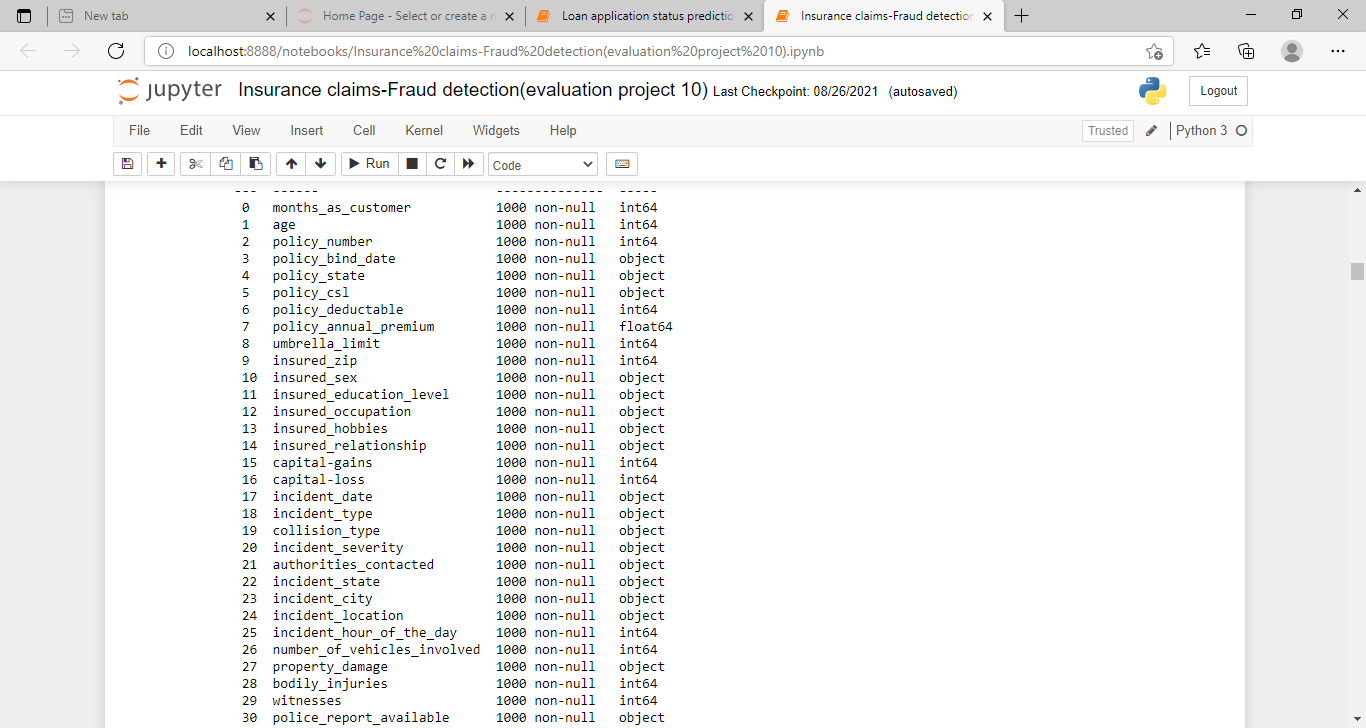
The current data set was labeled with n = 1000 samples.



From the above figure, we can list out the data set as variables. The data set can be defined as “df”. Then again we can overlap to the shape of the data and total info the data.

From Exploratory Data Analysis,

Firstly from the dependent variable, exploratory data analysis was conducted started with the dependant variable , fraud reported. They were 247 frauds and 753 non frauds. 24.7 % of the data were frauds while 75.3 % were non fraudulent claims.

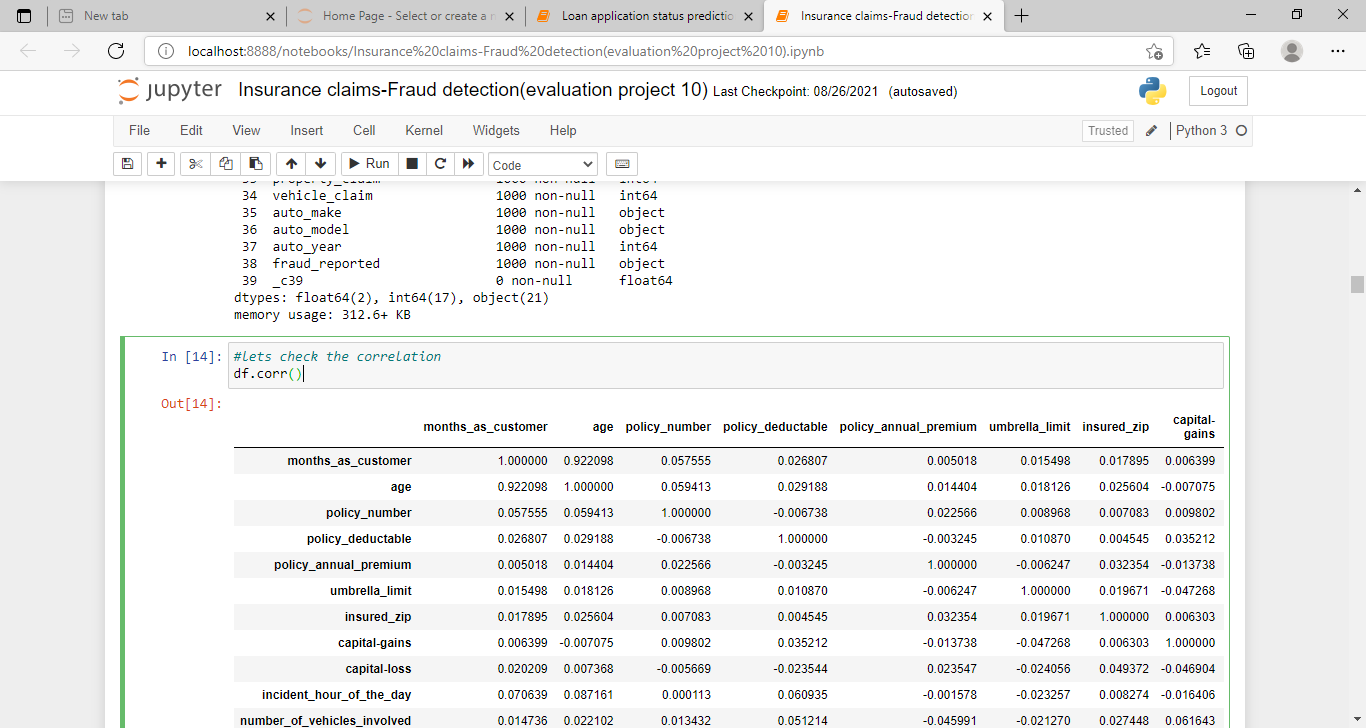


From the correlations among variables,

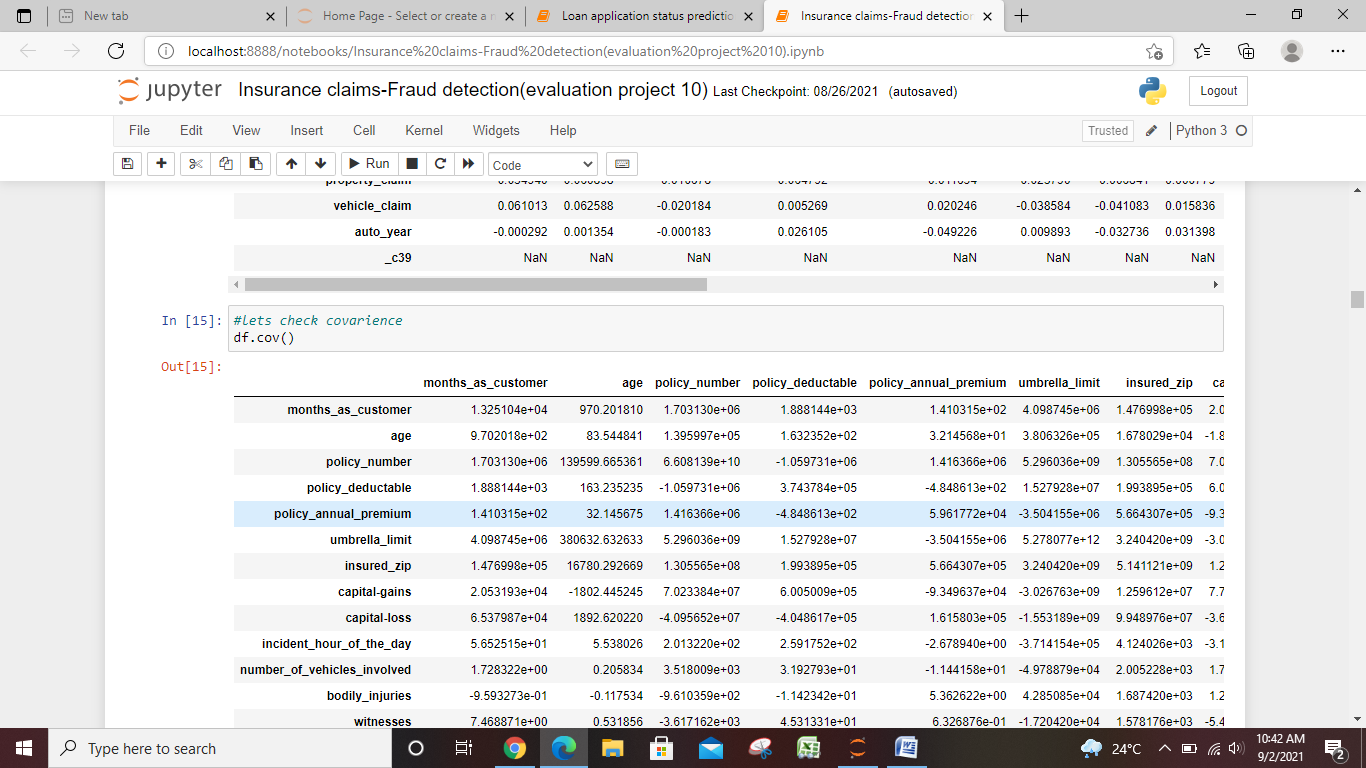
Correlations amongst continuous variables (ordinal, interval or radio variables) were inspected. Heatmap was plotted for variables with atleast 0.3 Pearson’s correlation coefficient , including the data variables.

Month as customer age had a correlation of 0.92. probably because drivers by auto insurance when they own a car and this time measure only increases only with the age. Incident severity and different types of claims have a clear correlation ($R$ = 0.36-0.50 ) apart from that , they do not seem to be much correlations in the data.

However, the claims provide some granularity that will not otherwise be captured by total claims. Thus these variables were kept.

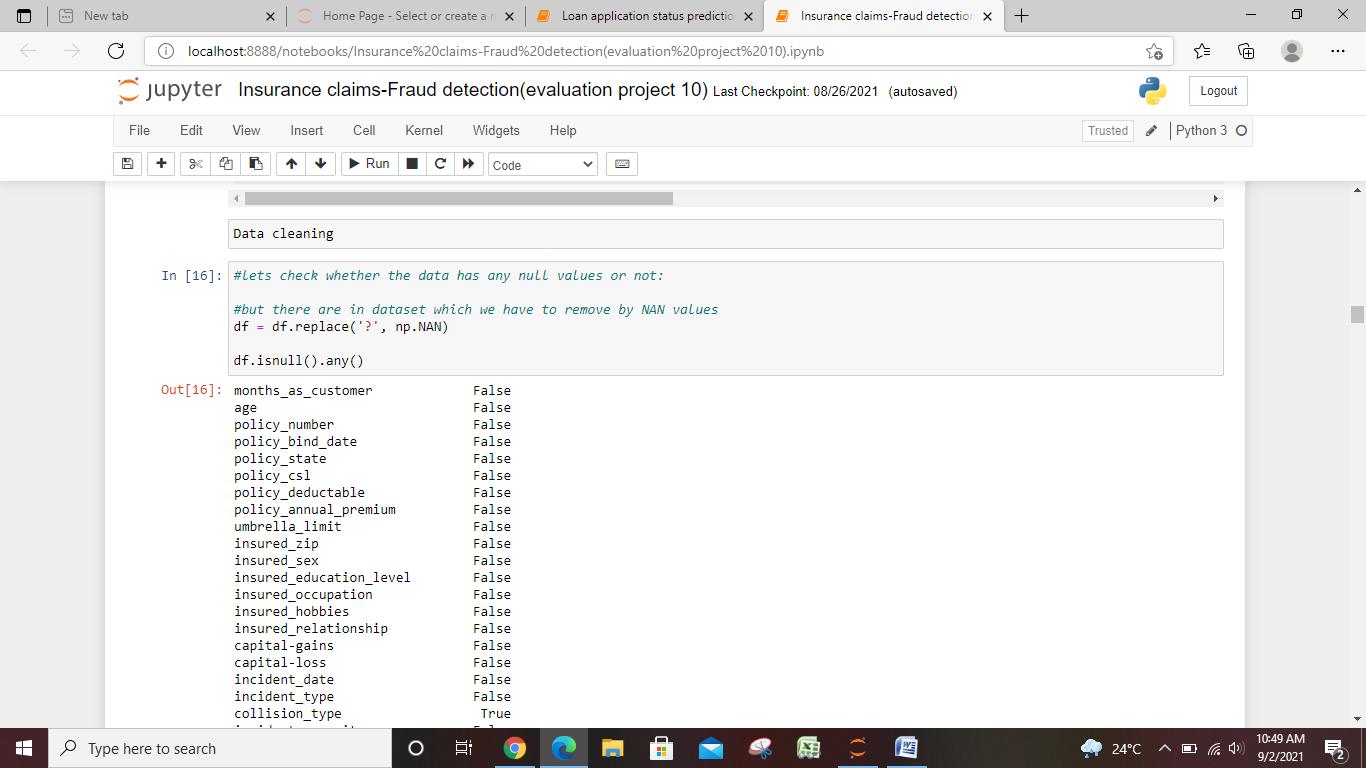


From the data set of the covariance, we can get some derived forms into data.



By Visualizing data into data variables.

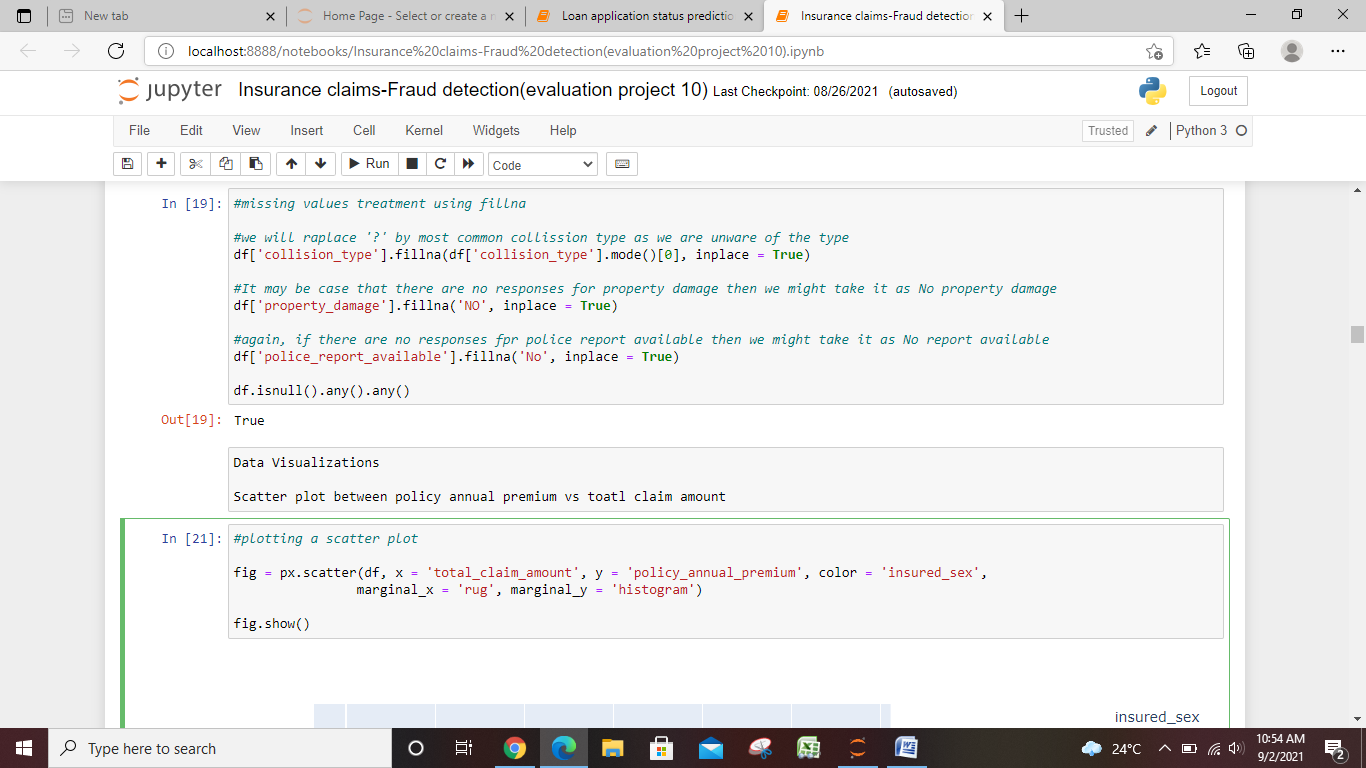
Counts of every variable split by the data variable was plotted further. Below are the some data sets for data cleaning process.

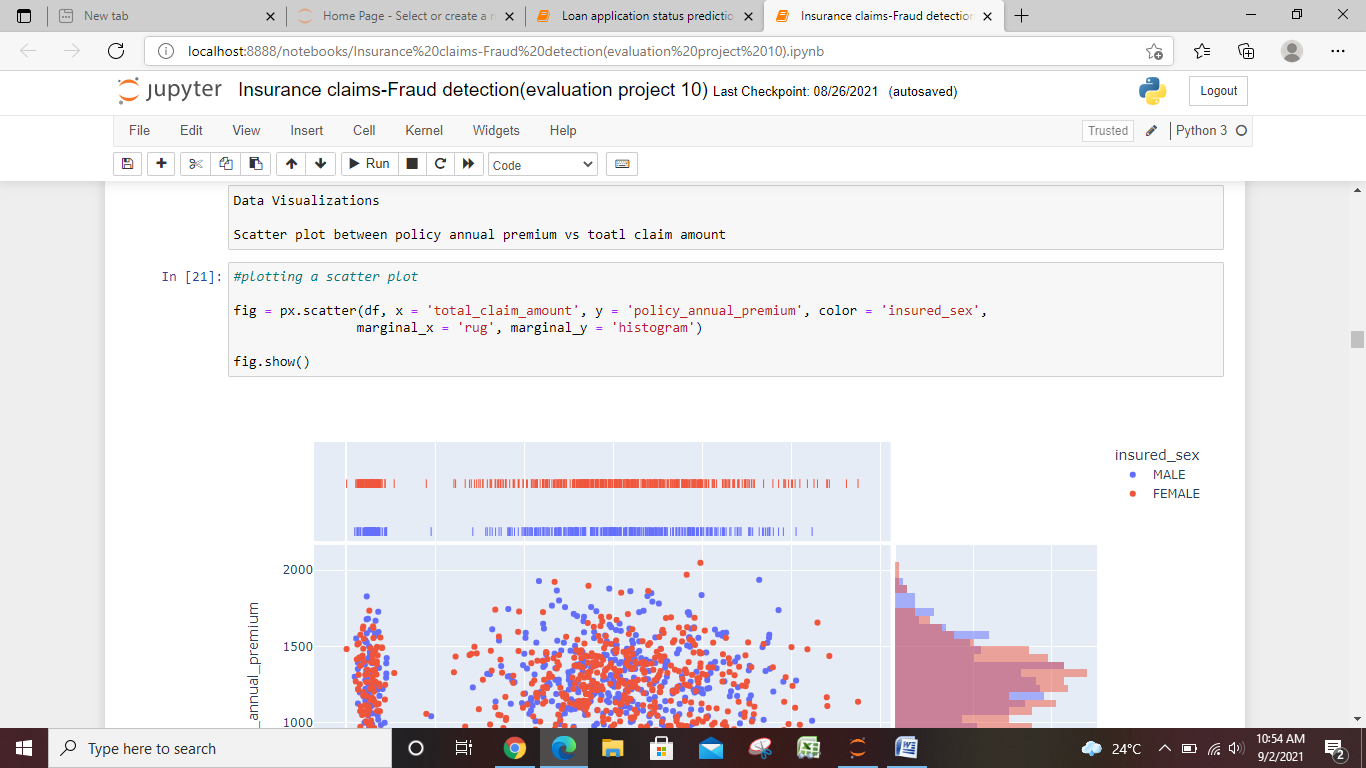


Losses for claims:

Here, I define as loss as simply money going out from the insurance company. Source of money coming in, on the other hand, are premiums.

Although we know claims and premiums are not the only source of money going in or out of an insurance company, these two variables are used since they are the only information, we have fro this data set. Typically , other source of money movement may be investments made by the insurance company, for instance.



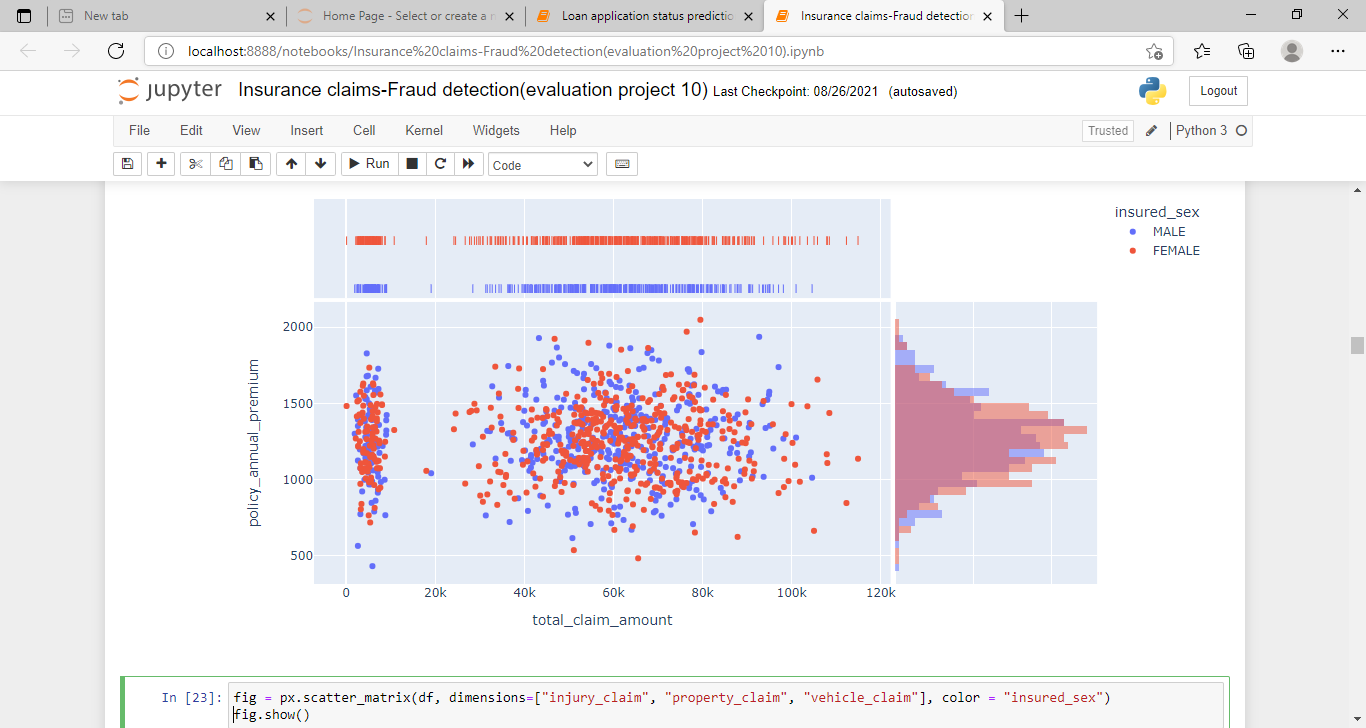


From the pre processing:

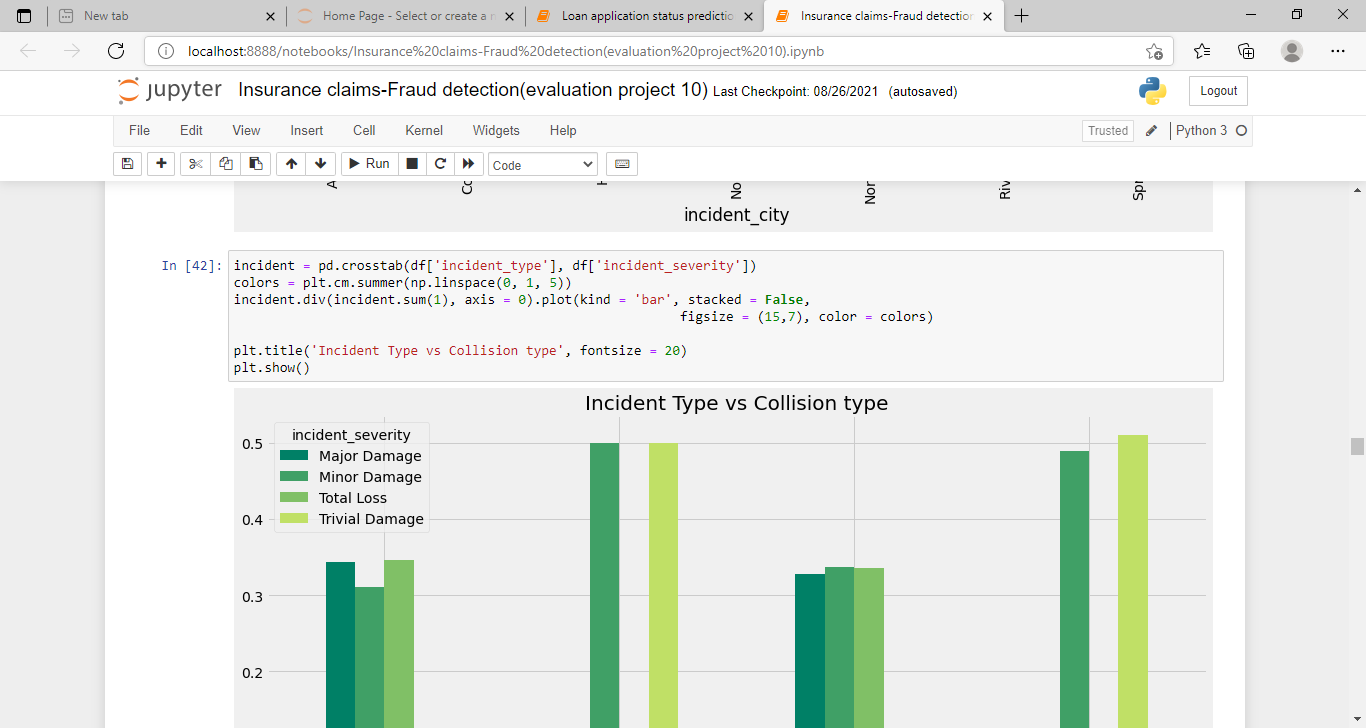
The data variable fraud reported was codes as 1 for fraud and 0 is for non fraud. Six interaction terms created. Interaction between property claim amount and incident severity. Nominal variables were one-hot encoded, and the data set was split into 75% train and 25 % test set, stratified on fraud reported.

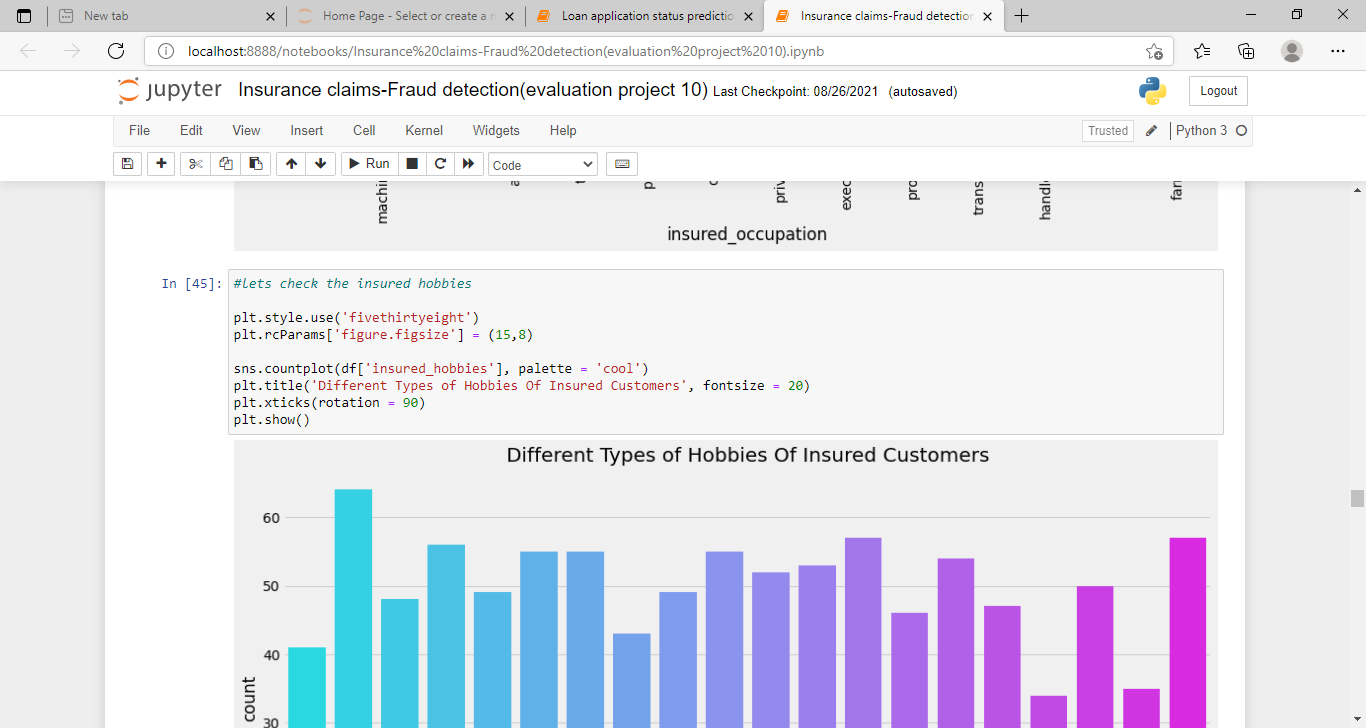
From the data Visualizations,

We can scatter the plot between policy annual premium and total claim amount.



From the base line score the cross tab incident of the data set is imbalance, accuracy is not a good measure of success. A high accuracy can be achieved by a poor model that only selects the majority class. Hence not detecting and measuring the accuracy of classifying the class of interest. In fact, the predicting only the majority class will give an accuracy of 75 % specificity of the 100 % but an sensitivity of 0%.





From the modeling, Five different classifiers were used in this project:

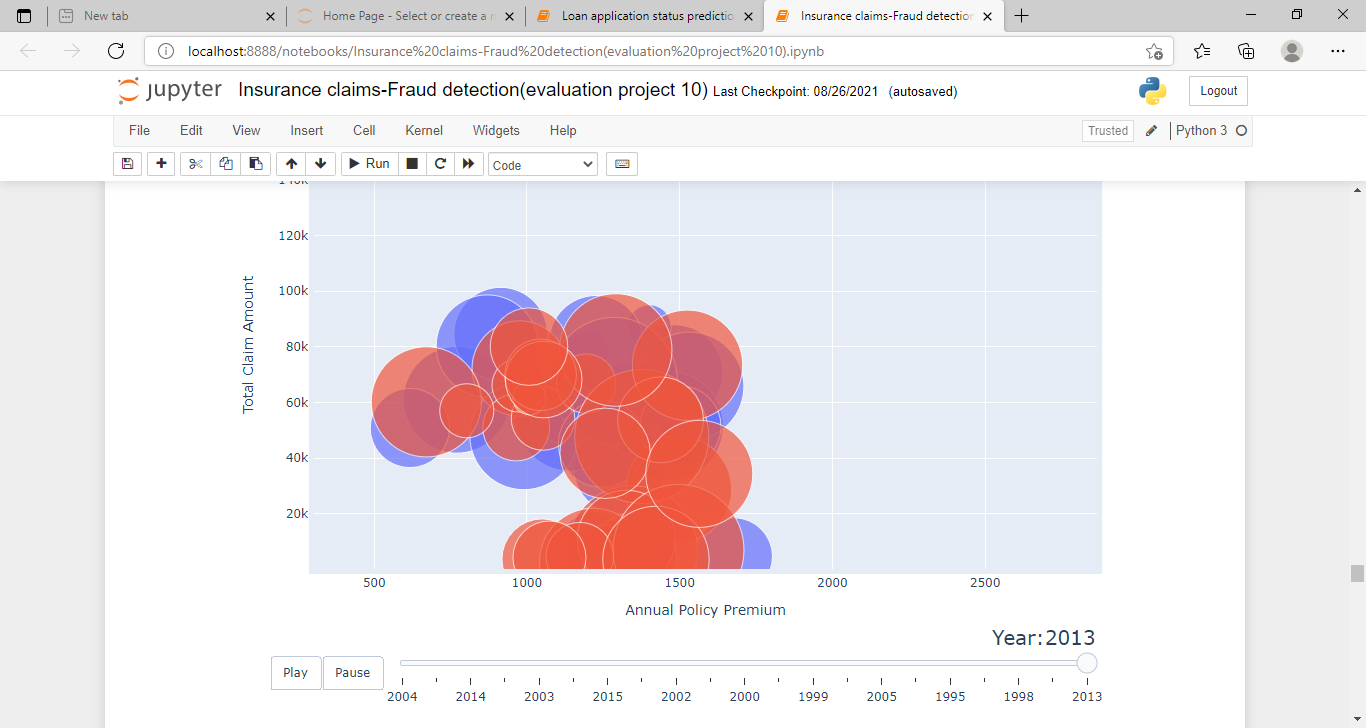
* Logistic Regression
* K-Nearest Neighbours
* Random Forest
* XG Boost
* Adaboost

Hyperparameter tuning and selection was done for all the models using RandomizedSearch.

Due to the no of parameters and models that were ran out from the RandomizedSearch is a faster more efficient choice as compared to gridsearch.

After a 10 – fold RandomizedSearchCv, the model with its select hyperparameters were fitted on the training set.

Mean accuracy scored for the best estimators of the RandomizedSearchCV, accuracy scores on the training set and accuracy scores on the test set was computed. Then the sensitivity, specificity, precision, F1 score and ROC AUC scores were computed.



From the section, we discuss how different blocks of models were ran. Evaluation of the models will be the evaluation section.

* Model with class weighting and hyperparameter tuning.
* Modeling with oversampling using smote.
* Modeling with oversampling using adasyn.
* Modeling with oversampling using bootstrapping.

From the Ensemble models :

Ensemble models in machine learning combine the decisions from multiple models to improve the overall performance and stability of the predictions.

Before, ensembling, correlations of the predictions were ran. XGB, Random forest and AdaBoost have high correlation , perhaps as they all are CARTs ( classification and regression tress ) . other than that models seem to capture different aspects of the feature space, as shown by the small to the average correlation ( Pearson’s heuristics ) of their predictions.

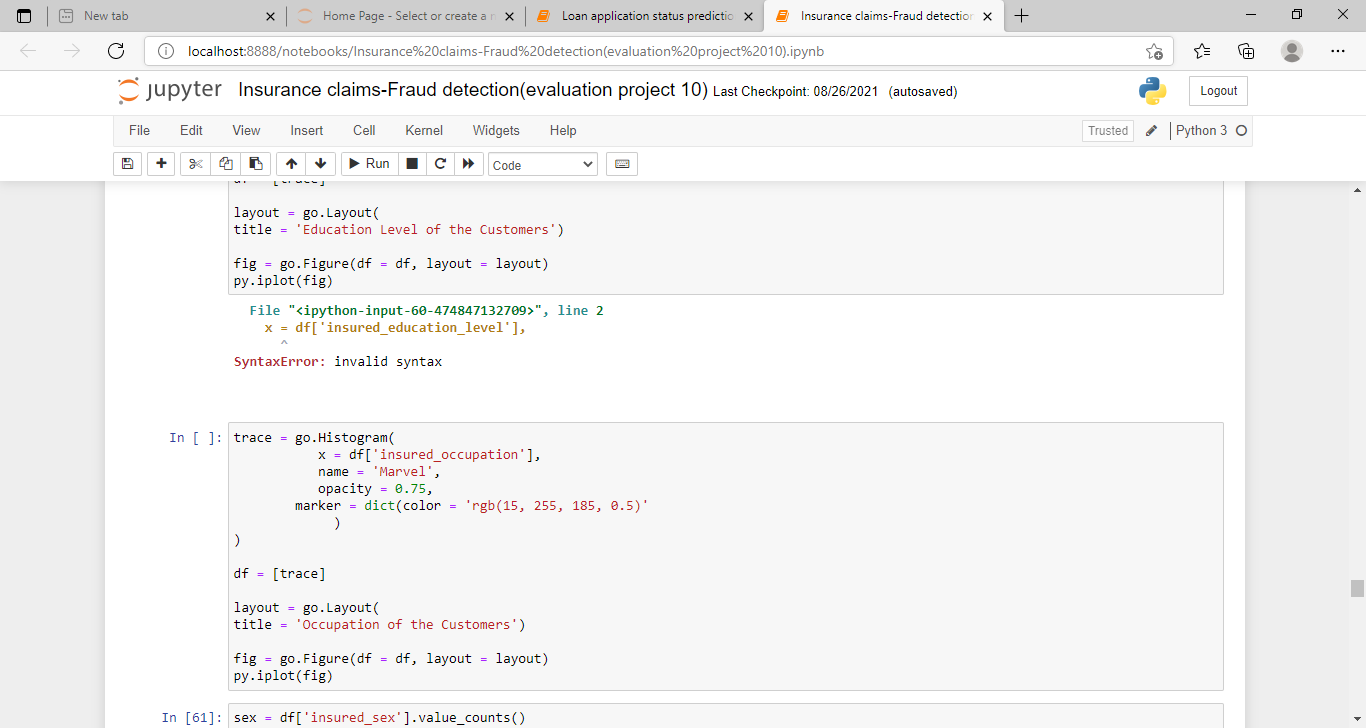
An ensemble may be able to out perform any single model by learning to combine their respective strength. However, the models we select for the ensemble should not be highly correlated. Else , our model will not be explaining unique variances and thus, unlikely to improve. The ensemble model will use the best logistic regression, KNN and the best of XGB, Random forest and adaboost (ensemble model 3 ) , based on F1 scores , from the models with class weighting, models with oversampling by smote , adasyn, and bootstrapping.

The tree models selected are:

Logistic Regression with AMOTE

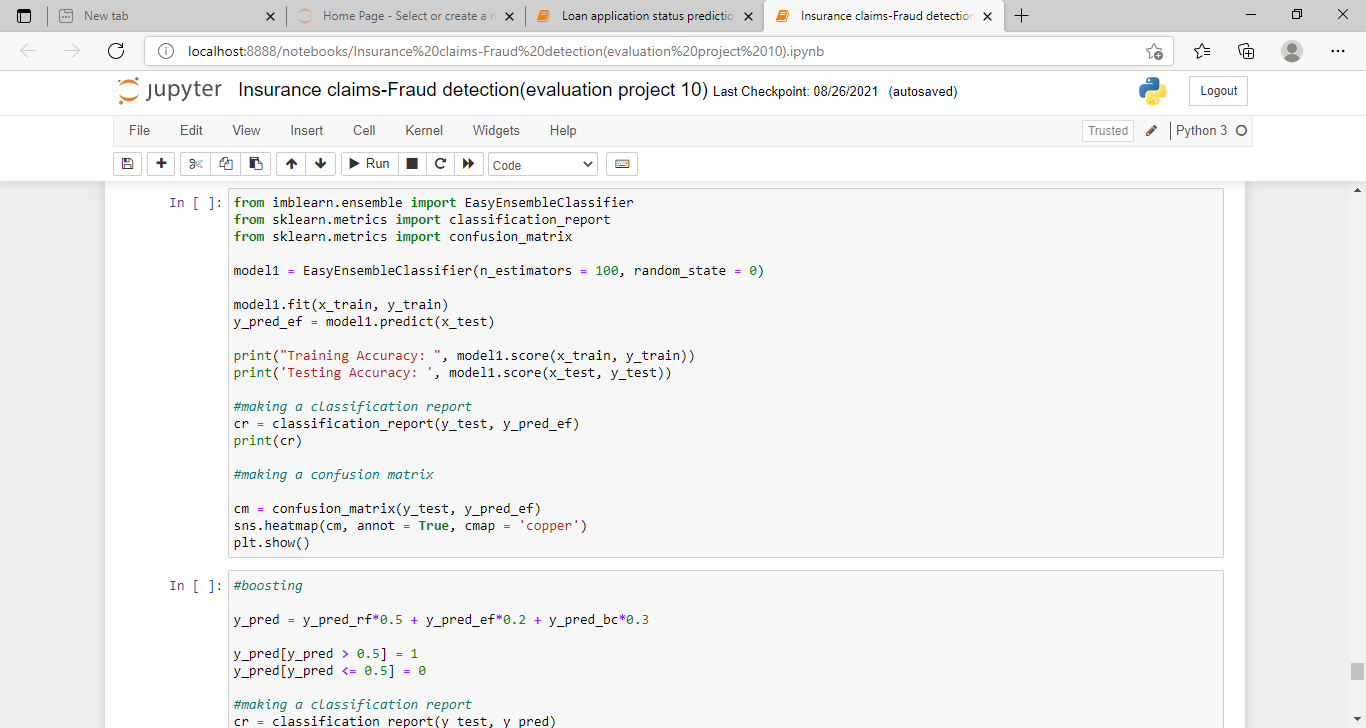
KNN with bootstrapping

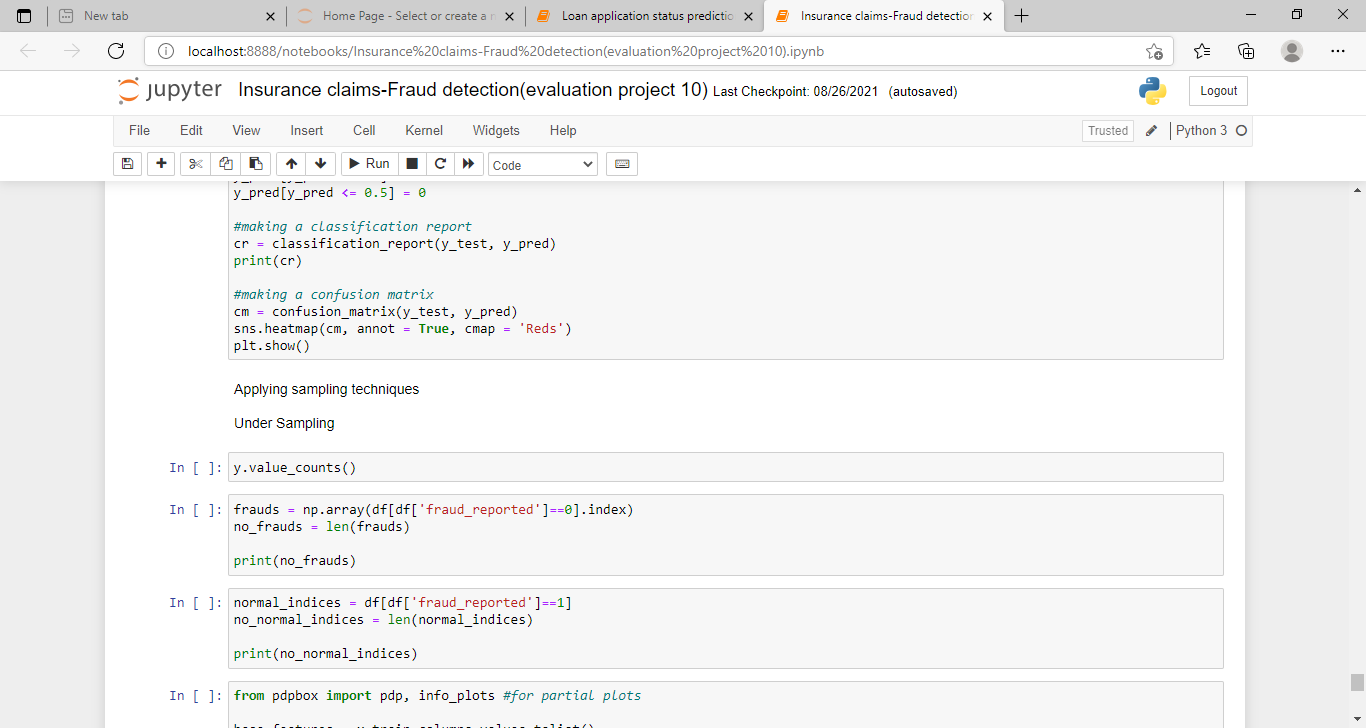
Weighted XGBoost



By applying the sampling techniques. Here we go as :

Under sampling and over sampling





* **Max voting with out over sampling:**

Max voting method is generally used for classification problems. In this technique, multiple models are used to make predictions for each data point. The predictions by each model are considered as vote.

10-fold cross validation was performed with its mean scores printed, followed by computing of train and test accuracy scores. Then, the sensitivity, specificity, precision, F1 Score and ROC AUC in Scores were computed.

* **Max voting with over sampling**

As bootstrapping in general produced the best F1 scores out of the other over sampling technique in this project. It was used here for the max voting ensemble. After the training set was bootstrapped, the process is the same as the max voting without over sampling.

* **Blending without over sampling**

In the first method of blending method, the training set is broken down into 2 parts, new training and a hold out set called validation set by the ratio respectively.

* **Blending with over sampling**

As bootstrapping in general produced the best F1 scores out of the other over sampling technique in this project , it was used here for the blending ensemble. After the training set was bootstrapped , the process is the same as the blending ensemble without over sampling.

From the evaluation and conclusion,

Cross validation accuracy scores, accuracy scores on training set, accuracy scores on test set, sensitivity, specificity, precision, F1 Score and ROC AUC was computed and printed.