Credit Card Fraud Project Documentation

Description

Credit Card Fraud data set is about number of many transactions which mostly are normal while a few number of fraud ones exists , we use this data set to try to reach an approach for classifying the behavior of the transaction from learning from this data , the original features are replaced with V1, V2, … V28 columns which are the result of (PCA transformation) applied to the original ones , The only features which have not been transformed with PCA are ‘Time’ and ‘Amount’. Feature ‘Class’ is the response variable and it takes value 1 in case of fraud and 0 otherwise.

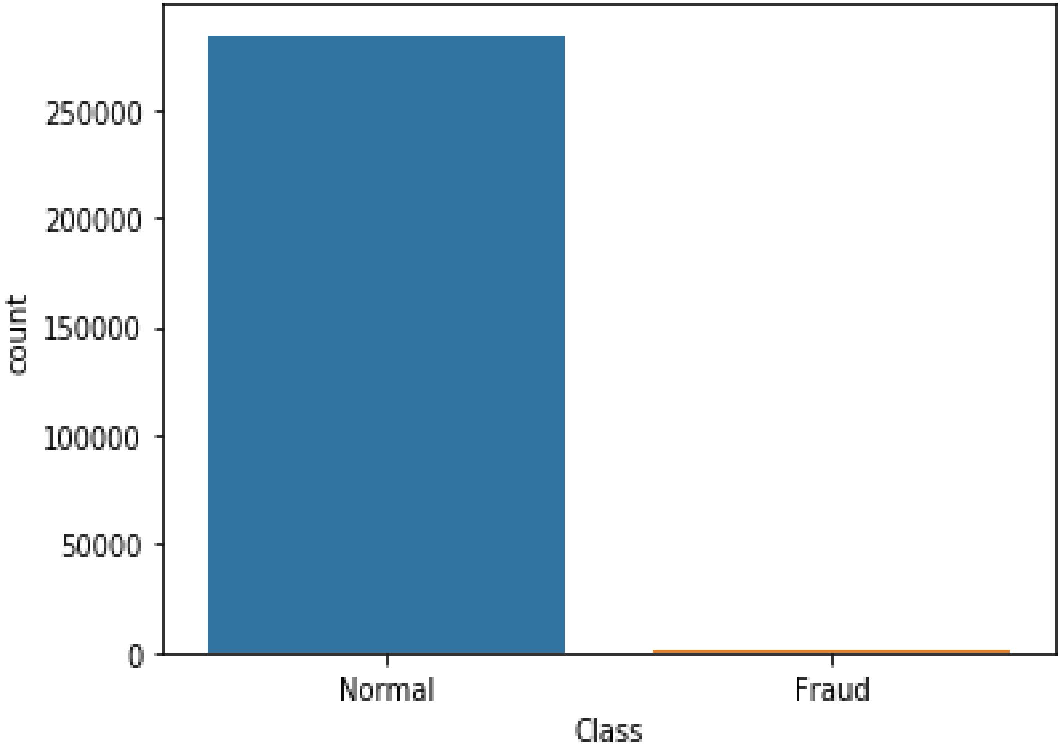
Pre-Processing

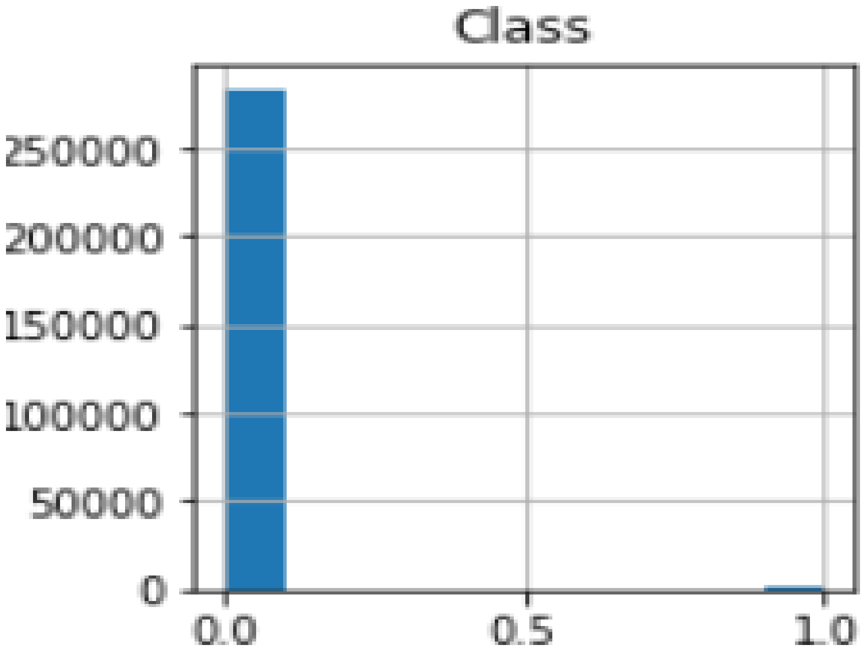
In order to use the data for analysis and modeling … The data set must be clean , sorted , has no missing data and preferred to be not biased , depending on this points our first job was to view the data and check it to be assured that the previous conditions do exist .

In our case after we checked the data it turns out to be clean and sorted and as in description a method was used (PCA Transformation) to keep the data of the clients safe as in this case our the data is sensitive .

However , we found out this data is very biased to the class 0 which represents the normal transactions and that inflects the real life transactions as nearly all the transactions are normal ones

Visualizing the biased class issue





As shown in the last two figures , the data is very biased towards the class of normal transactions , this kind of bias is troublesome for us when we try to train a model and highly affects the accuracy of the model and so the outcome of the model which is in our case is predicting the behavior of the transaction will not reflect the real world transactions and as the model needs unbiased data to learn effectively we decided to use an over sampling technique to make the data not-biased .

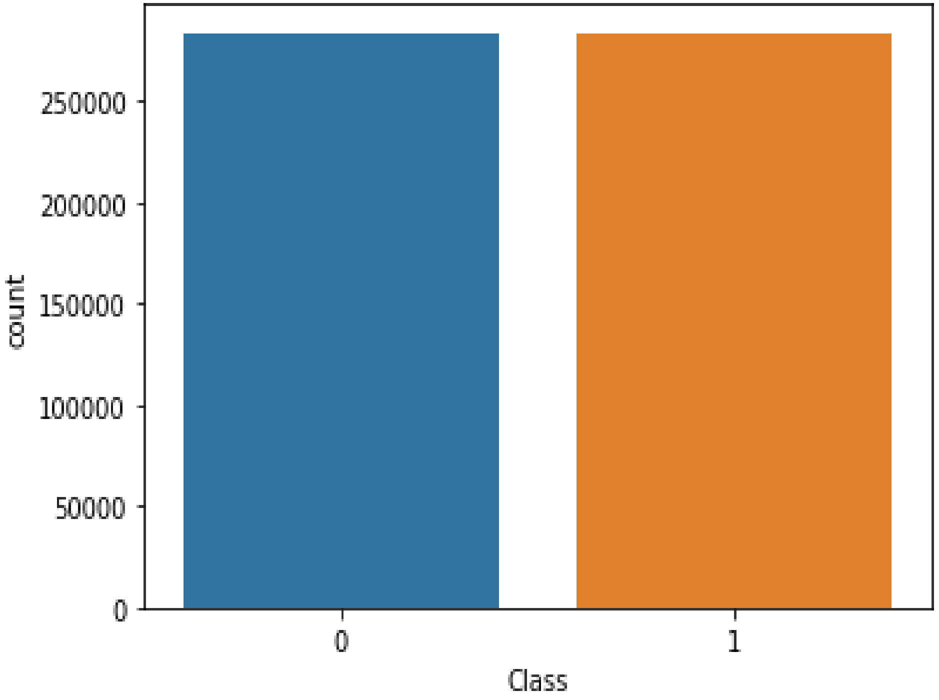
Oversampling by SmoteToken

In order to solve the issue of the biased data , we have many techniques of oversampling and undersampling but in our case we will use SmoteToken technique , since oversampling or undersampling are good ways to solve imbalanced data issues but every thing has a cost such as diminishing some data features relations or causing the model to have over fitting issues

We choose SmoteToken as this approach uses both Over and Under Sampling ways but generally it's considered to be oversampling , also the new dataset is very close in behavior to the original one and that's to make the model more closer to the real world .

New Dataset Classes count

This is the new dataset classes amount and It shows the 2 classes have same amount now thanks to the SmoteToken Technique.



Training and Testing

After Pre-Processing step is done and concluding important features that affects the classification , the dataset is divided into 2 datasets , The first is for training the Model and the other for testing the Model produced from the training , the test section is responsible for comparing the results of the Model and the true classes of this data and hence, Accuracy of the model is calculated.

Modeling

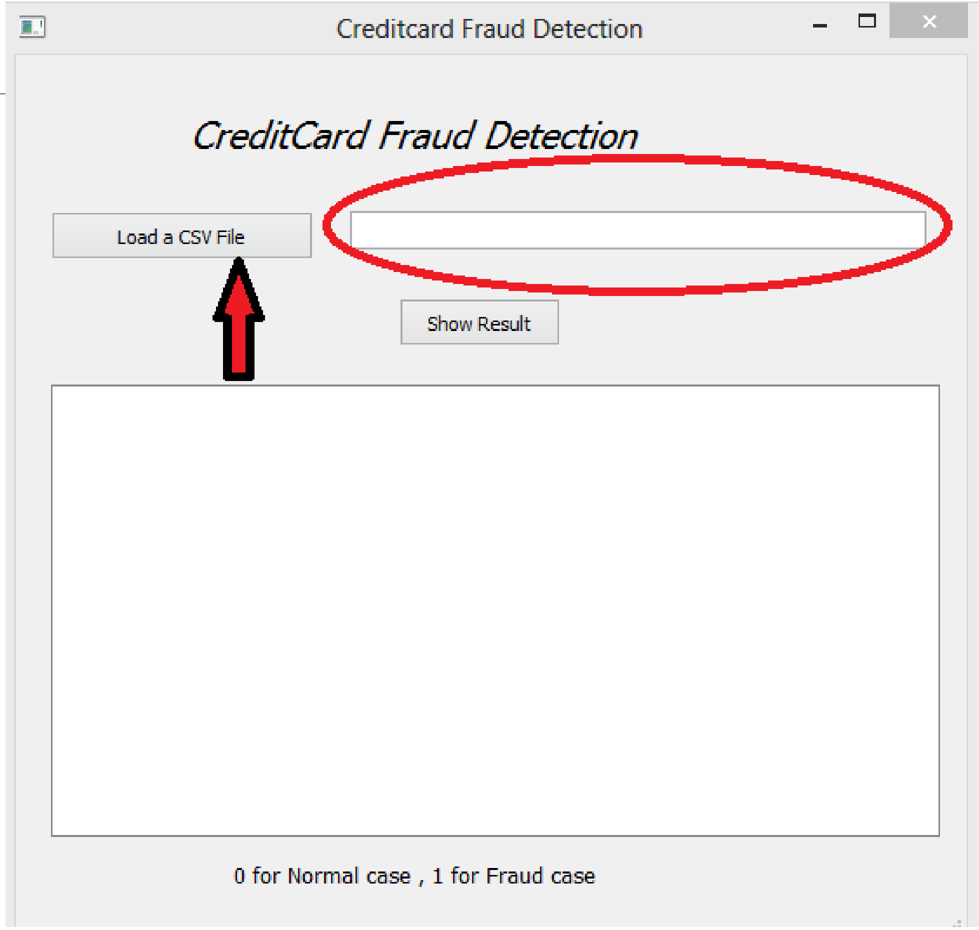
After data is divided for training and testing the modeling step comes , when it comes to modeling there are many algorithms used to predict the outcome of the data , After trying many Algorithms in our case as (SVM , Decision Tree , Random Forest , Logistic Regression) , The Random Forest Algorithm had the upper hand in data prediction with best Accuracy Score , Also since this data is related to a sensitive field the confusion matrix was important to calculate Precision and recall values depending on (False positives and False negatives) , we believe in our case even this Model is o avoid False Positive cases (A fraud Transaction but considered Normal) , It’s more important to avoid assuming many normal transactions as a Fraud as it’s also not good for the business and harms it , So the Random Forest was the best considering these factors and also for the accuracy score , to enhance the accuracy a grid search was used and that’s a way to try different combinations of parameters to determine best combination and reach maximum accuracy we could achieve.

The Desktop Application

After the Model is trained and tested and ready for applying on new datasets , The creation for an application so the normal user can use the model comes , Since the customer is not familiar with programming and codes it's our responsibility to create a user interface a normal customer can use which contains the model and shapes the final product that customer can use easily and understand it's functionality , in deep the Application contains 2 classes one for the functions the Application uses and the other for the Interface that appears in front of the customer , The Application is a very simple one we will show how to use it and apply prediction by the Model we created and saved within.

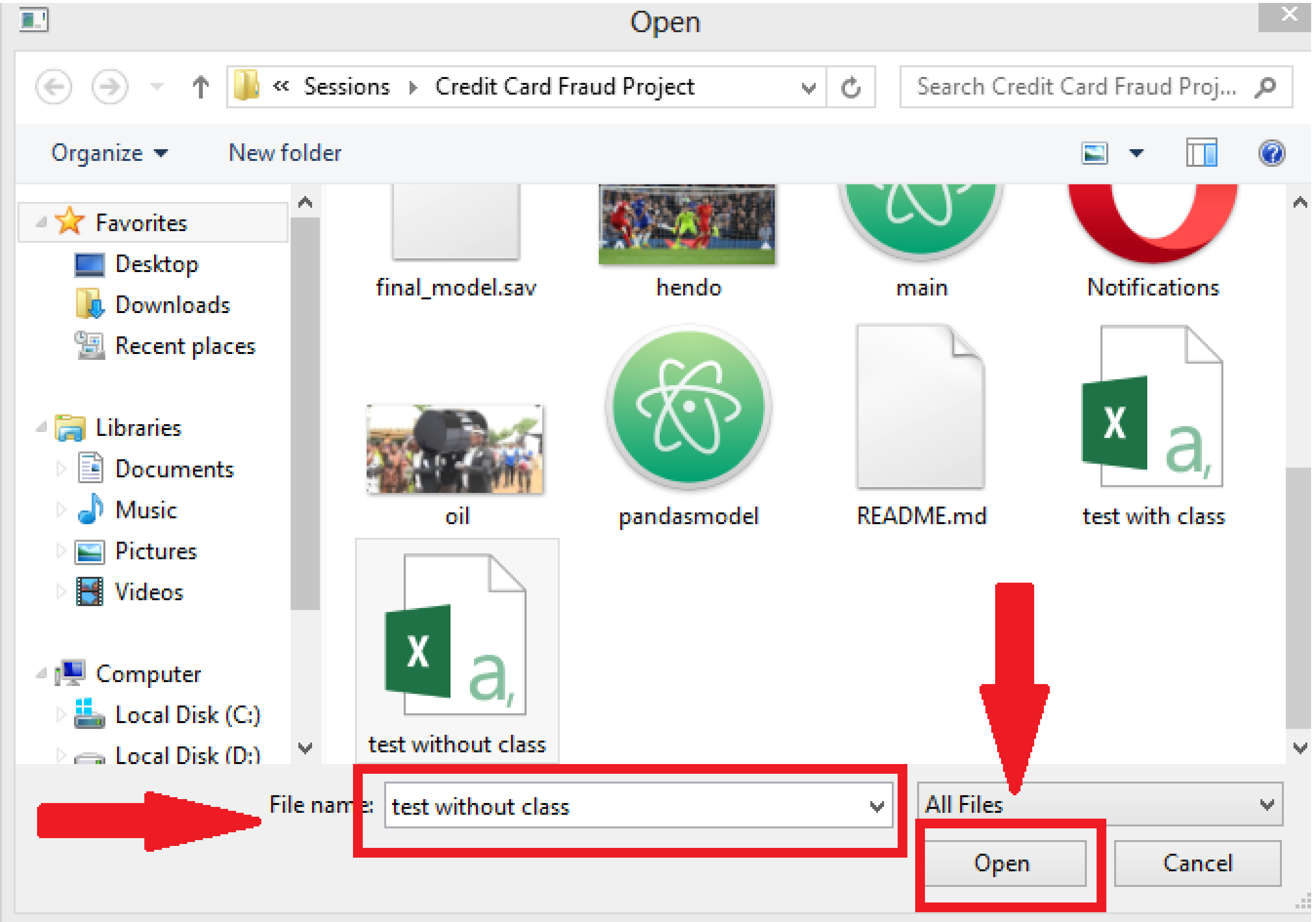
How to use

Firstly , After we open the application (main.py) a window appears as in next image , The Customer clicks “Load a CSV File “ to load the data which must be in a CSV format and that’s critical so the program works right and of course that data contains the 30 Features as the Model was trained to predict the class by them.


Creditcard Fraud Detection
CreditCard Fraud Detection
Load a CSV File
Show Result
0 for Normal case , 1 for Fraud case

How to use

After that, Choose the file with data you want to predict and classify , we remind you to choose a file with 30 features (Amount,Time,V1,V2,……………V8) so the program works correctly and click the Open button.



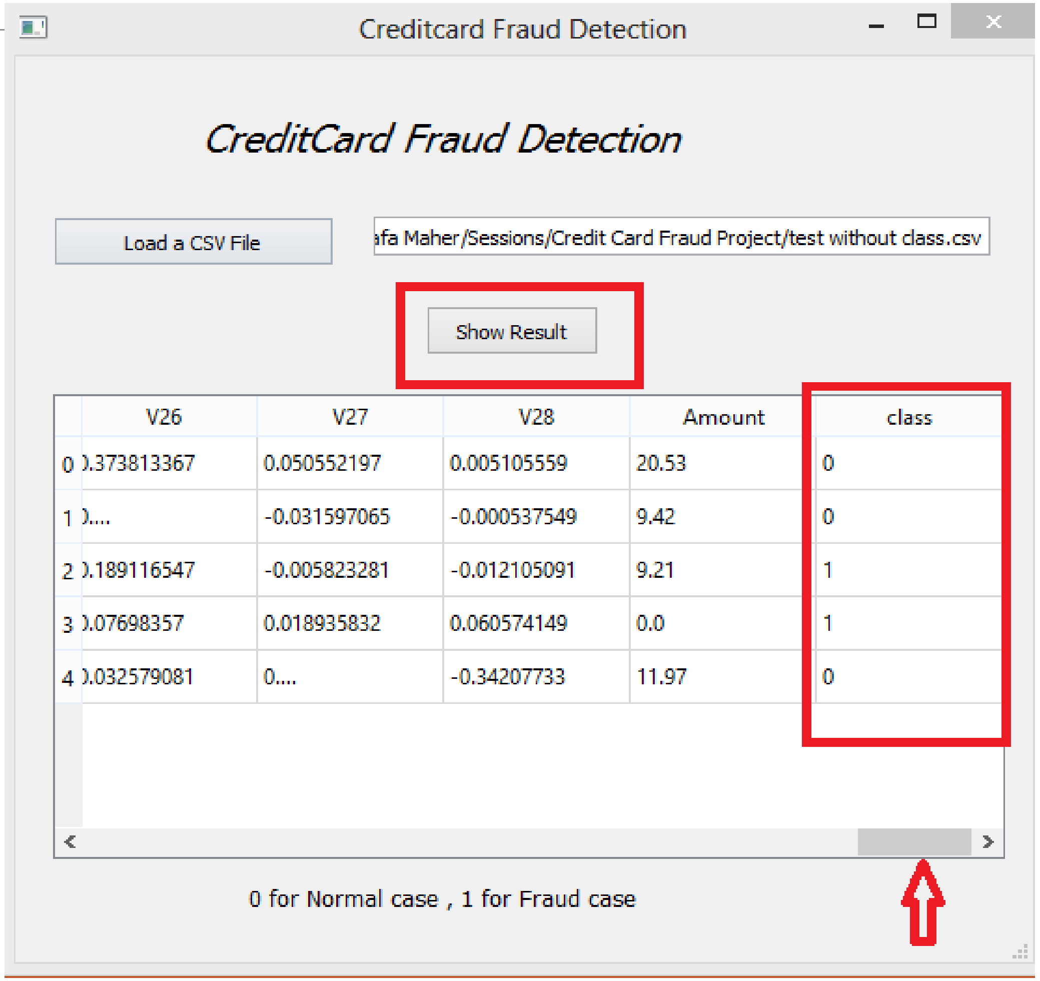
« Sessions › Credit Card Fraud Project

File name: test without class

All Files

How to use

The next step is to click on the “Show Result” button , The data will appear as in the figure and will be classified to Normal or Fraud as “0” for Normal and “1” for Fraud , you will find the Class column added and to view it scroll to the right , You repeat this process to predict another inputs or close the program by clicking on the “X” sign to the upper right corner.



CreditCard Fraud Detection

afa Maher/Sessions/Credit Card Fraud Project/test without class.csv

	V26	V27	V28	Amount	class
0	).373813367	0.050552197	0.005105559	20.53	0
1	)....	-0.031597065	-0.000537549	9.42	0
2	).189116547	-0.005823281	-0.012105091	9.21	1
3	).07698357	0.018935832	0.060574149	0.0	1
4	).032579081	0....	-0.34207733	11.97	0

0 for Normal case , 1 for Fraud case

Summary

Our project is to classify the transactions into normal and fraud , Pre-processing was applied to help in creating a good Model , The Random Forest Algorithm was used to create the Model , The desktop application was created to make it simple for the normal customer , even though in real life nothing is for sure but by learning from data and figuring relations between data features and determining the correlation and causation among them we can predict events that more likely to happen upon our conclusions , So we hope our project helps in the fraud transactions cases.