Credit Card Fraud Detection

– Maryam Amjad

I am a Data Scientist, I turn boring info into total AWESOMENESS.

01

Problem Statement

Problem Statement

- Due to rapid advancement of technology fraudsters find out new strategies to breach the security and get their hands on others asserts. The best solution is to monitor the transactions, especially credit card transactions.

- Our objective is to analyse the past transactions and identify the features which acts as an important indicators for a fraudulent transactions. To build and train several classification model and find the best suitable model for credit card fraud detection.

02

Data Cleaning

Data Cleaning

Here the work starts with importing the dataset. The dataset contains 61080 rows and 21 columns. The data cleaning part includes the following steps :

- Data types of each column are checked and converted appropriately.
- Then null check is done and removed if any.
- The columns are renamed appropriately.
- Unwanted columns are dropped.
- The cleaned data set is then exported to carry on EDA.

03

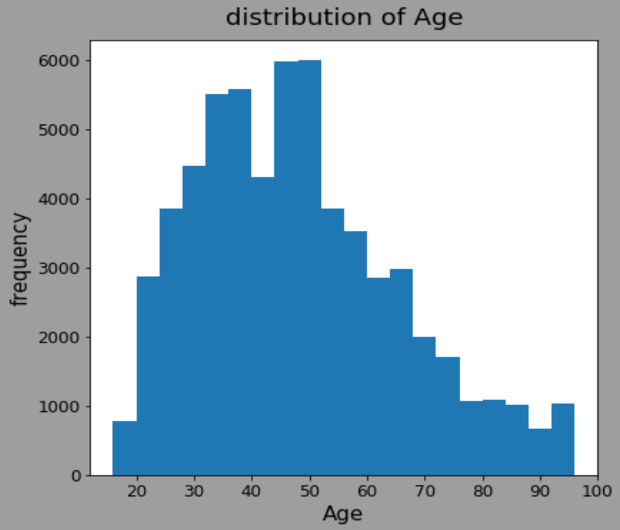
Exploratory data Analysis

Exploratory Data Analysis

- The **Distribution** of Age is **bimodal**
- Most of the fraudsters **Age** falls between **25** and **60**
- The **Street** with most no. of fraudulent activities - **43235 Mckenzie Views Apt. 837**
- The **City** with most no. of fraudulent activities - **Houston**
- The **State** with most no. of fraudulent activities - **New York**
- The **Merchant** with more fraudulent transactions - **fraud_Rau and Sons**
- The **Merchant Category** with more fraudulent transactions - **grocery_pos**
- First **name** of most of the card holders with fraudulent transactions - **Christopher, Robert**
- The highest **amount** in fraudulent transactions - **1376.04**
- The **Job** of most of the **male** card holders with fraudulent transactions - **Exhibition designer**
- The **Job** of most of the **female** card holders with fraudulent transactions - **Prison officer**
- The **Months** with higher fraudulent activities - **March** and **May**
- The Fraudulent activities are higher at **Weekends** (Fri, Sat, Sun)
- The Fraudulent activities are higher at around **10**th, **20**th and **30**th days of the month

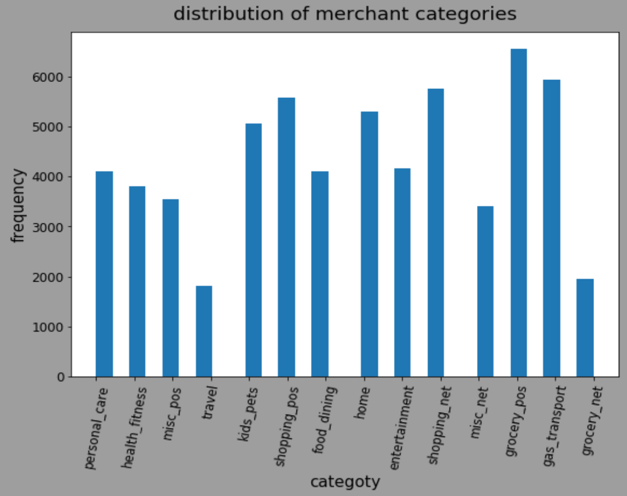
04

Visualization

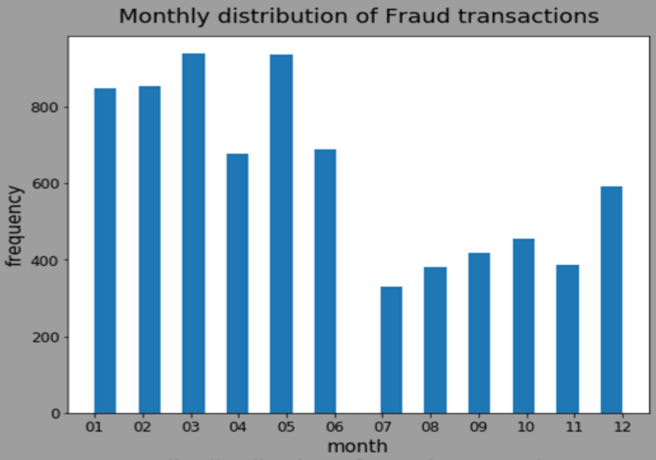


Data **Visualization**

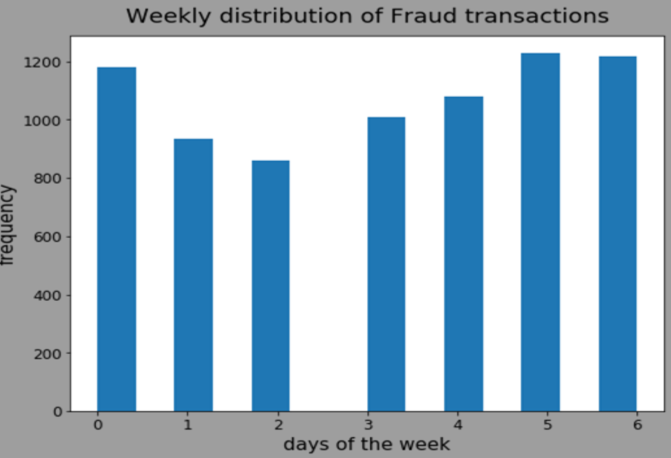
The graph shows the distribution of age. From the plot it is observed that the distribution is **bimodal** and right skewed. Most of the people falls between the age of **25** - **55**



The graph shows the distribution of **Merchant category**. From the plot it is observed that the category **Grocery_pos** has higher frequency and it has higher fraudulent transactions.



This graph shows the Monthly distribution of fraudulent transactions. From the plot it is observed that the distribution is **Bimodal**. Most of fraudulent transactions happens during the month of **March** and **May**.



The graph shows the day-wise distribution of fraudulent transactions. From the plot it is observed that the Most of fraudulent transactions happens during the **weekends**.

05

Modeling

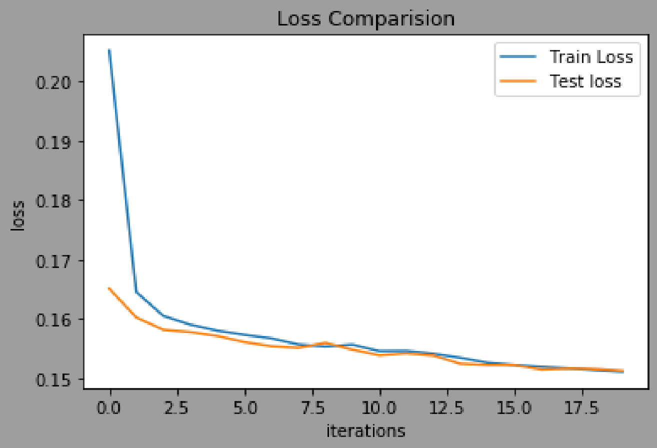
Preprocessing & Modeling

- The features with Object type are encoded using **LabelEncoder()**.
- Then the entire feature set is scaled uniformly with **StandarScaler()**.
- The scaled data is then splitted into train and test sets (70% and 30% ratios).
- Several machine learning models including Logistic Regression, Random Forest Classifier, CNN were built, trained and tested.

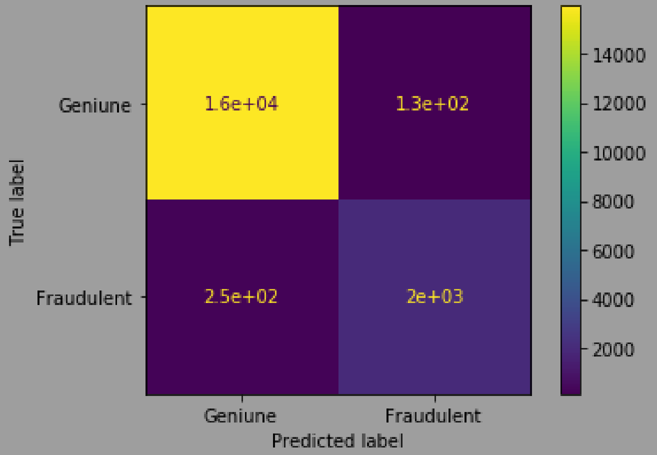
Performance **Summary**

The **F1 score** (the higher the better) is considered for grading, which is derived from the precision and recall. Various model results are given below

Model	F1 Training	F1 Testing
Logistic Regression	0.6448	0.6448
K-Neighbors Classifier	0.8928	0.8552
Decision Tree Classifier	1.0	0.8886
Random Forest Classifier	1.0	0.9128
AdaBoost Classifier	0.7836	0.7771
Support Vector classifer	0.8265	0.8206
Bagging Classifier	0.992	0.901
CNN	0.8374	0.8335



The graph show the **loss comparison** of train and test data with CNN model. It is observed that the losses are **decreasing** at every epochs.



The confusion matrix is derived from the predictions from **Random Forest Classifier**. From this we can infer the performance of the model, as it shows True Positive, True Negative, False Positive and False Negatives.

Gradio is an open-source python library that permits you to rapidly make simple to utilize, adjustable UI parts for your **ML model.** Our best performing model is saved using **pickle.dump()**, which is then used to create the app.

06

Conclusion

Conclusion And Recommendation

- The **Random Forest Classifier** was the best model with the highest score of around 91% for test data. It has **higher F1 score** than all other models comparatively. The Best model is saved using pickle library, which can be used later for classification. The saved model is then used to develop an App using **Gradio**, which takes user inputs and gives the corresponding prediction.

- The performance of the model can be further improved by identifying more features which plays a significant role in fraudulent transactions. One of the major obstacle faced here is that most of the credit card dataset's feature name and values are encoded to ensure **user privacy** and **safety**, which can't be used here. By Overcoming that, our model can be improved.

Thank You