Vector Institute - Caregivers and Machine Learning Program

Capstone Project - Credit Card Fraud Detection

Sadhana Wadhwa

May 6, 2023

Acknowledgements

I would like to express my deepest appreciation to Vector Institute for giving us caregivers, this opportunity to take the first step towards the field of Data Science. I would also like the extend a special thanks to our instructor Juan, the TAs - Kristina, Mason, Roeland, Andrew, Mohamed, and last but not the least - Flora and Sedef. This course would not have been possible for me without the amazing support extended by each of you at every step.

0.1 Executive Summary

Wikipedia mentions **Credit Card Fraud** as an inclusive term for fraud committed using a payment card such as a debit or credit card. It can be an authorized or an unauthorized access to proceed with a financial transaction.

We have been seeing a rapid increase in these type of fraudulent transactions on a global level, which all the more necessitates the requirement for identifying and preventing such transactions to be completed successfully.

The purpose of this project is to create a machine learning model that can detect the number of transactions that are fraudulent in nature and assess the various factors involved, e.g. the amounts, time, etc.

0.2 Introduction

In 2021, the Federal Trade Commission (FTC) fielded nearly 390,000 reports of credit card fraud, making it one of the most common kinds of frauds in the U.S.

In December 2022, the Nilson Report, which monitors the payment industry, released a forecast that predicts a loss of approx. $165.1 billion over the next 10 years, in the U.S. alone, plaguing every age group in each of the states.

Just one type of credit card fraud, dubbed the **card-not-present** fraud, that involves online, over-the-phone and mail-order transactions, accounted for a whopping $5.72 billion (estimated) in U.S. losses in 2022, as per the Insider Intelligence.

These statistics piqued my curiosity in learning how to mitigate these fraudulent transactions by using machine learning algorithms to predict the nature of the transactions made.

In this project, I have used two machine learning models to depict the accuracy in finding the illegitimate transactions, and examine the scores presented in both.

0.3 Problem Definition

0.3.1 Brief on Dataset

The dataset presented contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions.

It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, ... V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are **Time** and **Amount**. Feature **Time** contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature **Amount** is the transaction Amount, this feature can be used for example-dependant cost-sensitive learning. Feature **Class** is the response variable and it takes value 1 in case of fraud and 0 otherwise.

0.3.2 Problems Presented to be Solved

- The dataset provided is highly imbalanced.
- We do not have visibility on data in all the columns (V1 - V28).

- We would have to first balance the dataset to create an appropriate model to be tested.

0.3.3 Proposed Model and Approach

For classifying a transaction as either a fraudulent or non-fraudulent, we would be using the **Class** column to segregate the transactions into their respective bins.

Proposed Model: Binary Classification.

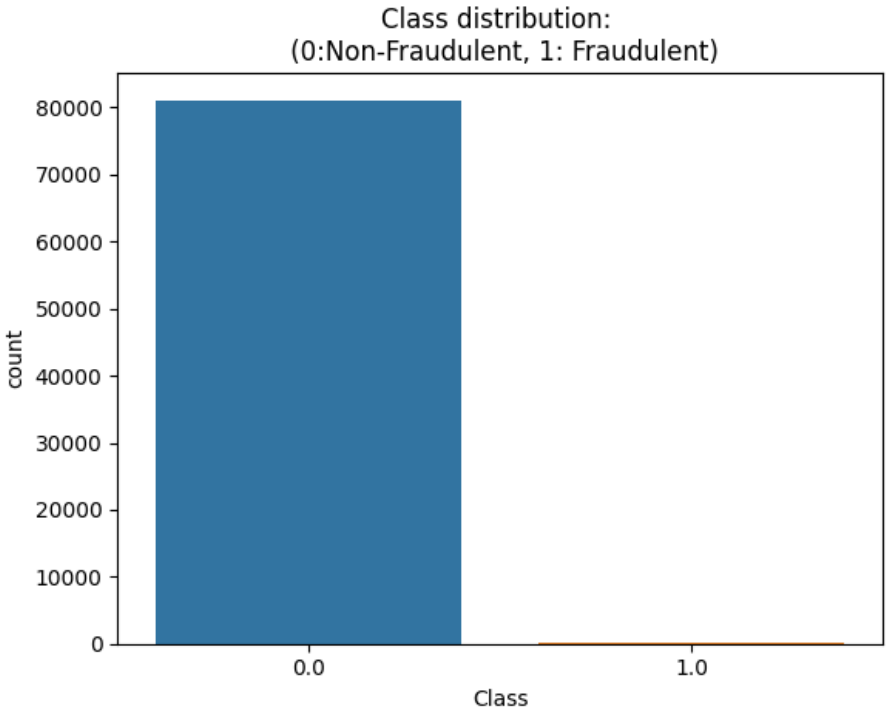
Reason: They are the most commonly used model for tasks involving classification of input data into one of two categories or classes.

There are a number of binary classification models to choose from, out of which I worked on the following models -

- Logistic Regression
- Decision Tree Classifier

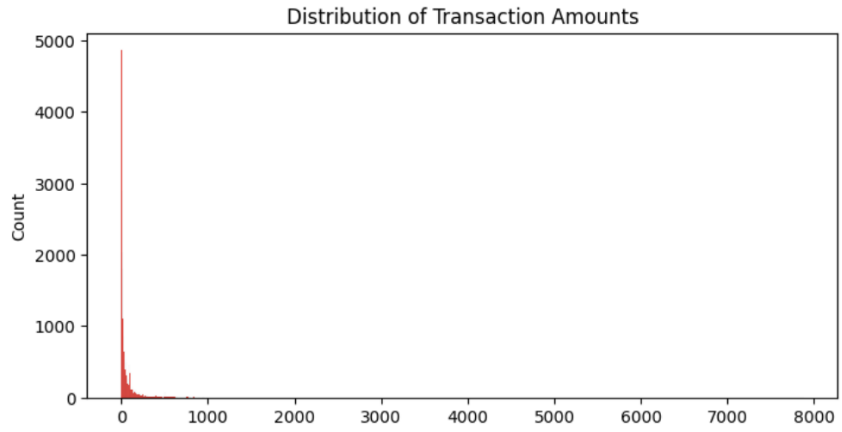
0.4 Data Exploration and Description

We know the data is highly skewed, rendering the dataset highly imbalanced. The percentage of fraudulent to non-fraudulent transactions is just 28.7%. It would be highly undesirable to use the data as is, since it would not provide us with accurate predictions.

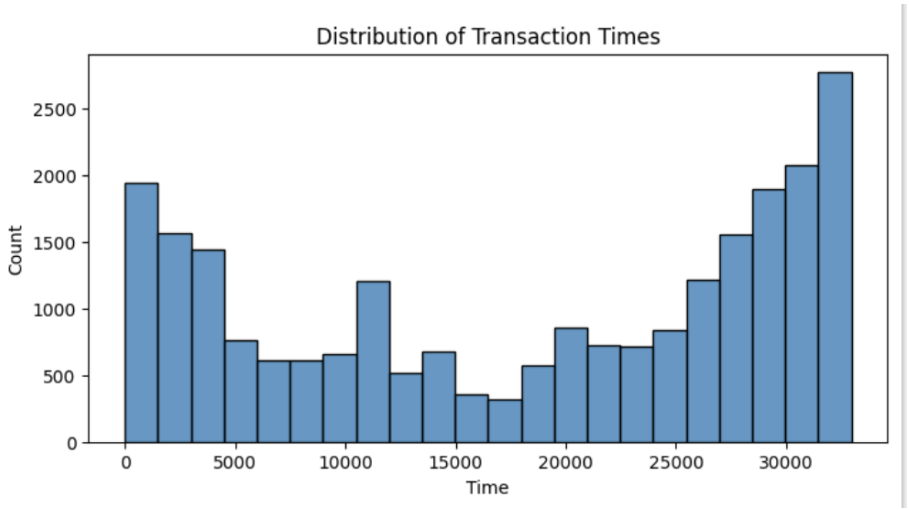


Next, we will look at the **Amount** and **Time** data to see how these affect the fraudulent transactions.

- Amount Distribution



- Time Distribution



Observations: From the graphs, it is clear that the count of fraudulent transactions are inversely proportional to the amount. In other words, the number of fraudulent transactions is more for lower amounts.

0.5 Model

Data Redistribution:

Undersampling: Undersampling is a technique used to address the class imbalance problem in machine learning, where the number of samples in one class is much larger than the values in the other class. It involves reducing the number of samples in the majority class to make the dataset more balanced. This can be done using techniques like random under-sampling, cluster centroids, or Tomek links.

Steps Performed to Train and Model the Data:

1. **Data Splitting**
    - Create individual subset dataframes for legitimate and fraudulent transactions respectively.
    - Combine both the dataframes to create a balanced dataset to work on.
    - Train the data - split the dataset into **Training** and **Test** data, ratio (80:20).

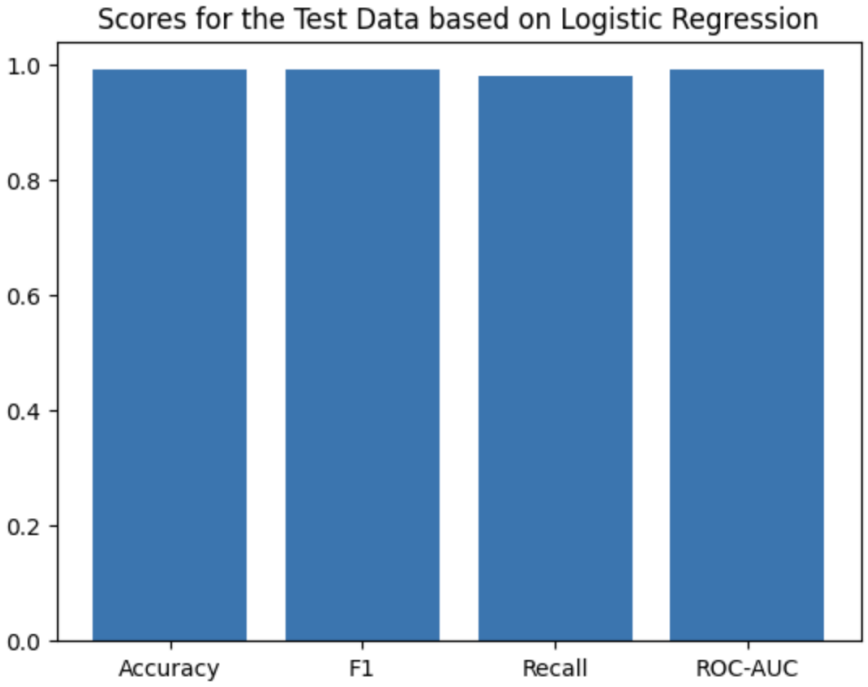
2. **Data Training**
   - Models selected to be worked on:
     (a) *Logistic Regression*
     (b) *Decision Tree Classifier*
   - Steps performed:
     (a) Fit the training and test data respectively into the models.
     (b) Train the models to get the following scores for both training and test data:
       – *Accuracy Score*
       – *F1 Score*
       – *Recall Score*
       – *ROC-AUC Score*

0.6 Results & Findings

For the **Test Data**, these were the results of the classifier report:

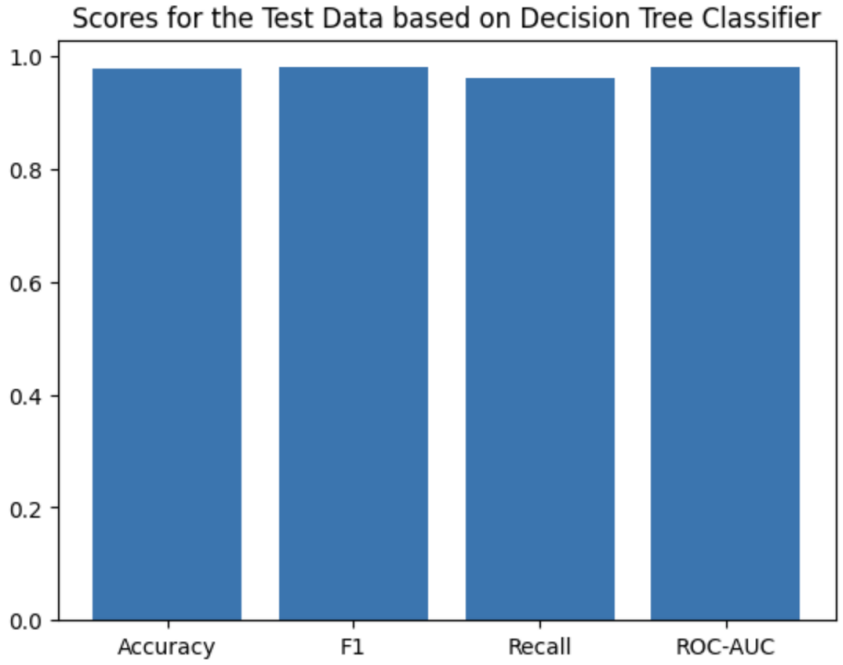
1. ***Logistic Regression:***

```
Based on Logistic Regression Model, the test data scores are:
The accuracy score is:  0.9898477157360406
The F1 score is:  0.98989898989899
The recall score is:  0.98
The ROC_AUC score is: 0.99

 The classification report is:
                precision    recall  f1-score     support
0.0             1.000000  0.979798  0.989796   99.000000
1.0             0.980000  1.000000  0.989899   98.000000
accuracy        0.989848  0.989848  0.989848    0.989848
macro avg       0.990000  0.989899  0.989847  197.000000
weighted avg    0.990051  0.989848  0.989847  197.000000
```



2. ***Decision Tree Classifier:***

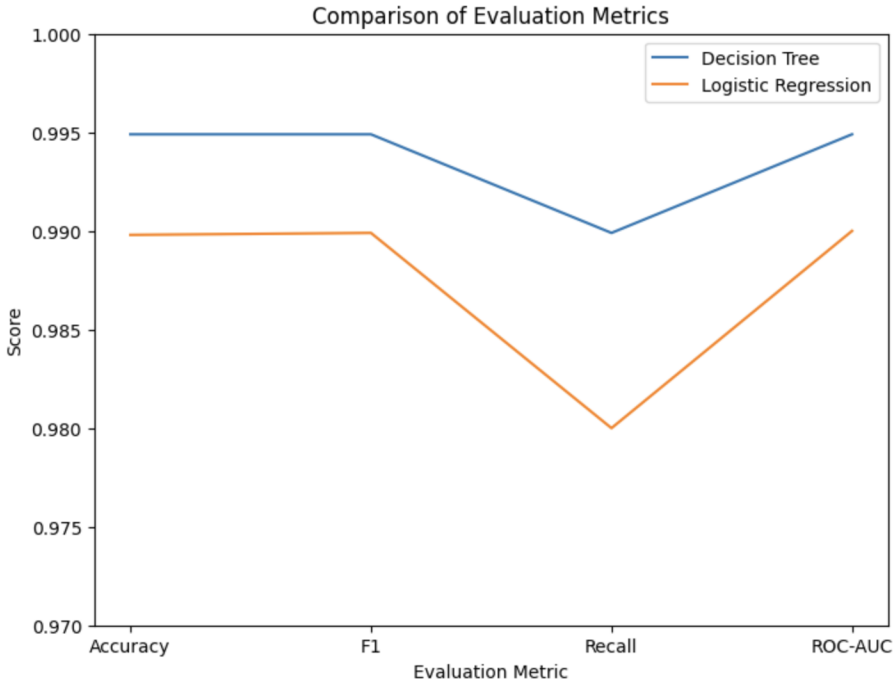
```
Based on Decision Tree Classifier, the test data scores are:
The accuracy score is:  0.9949238578680203
The F1 score is:  0.9949238578680203
The recall score is:  0.98989898989899
The ROC_AUC score is: 0.994949494949495

 The classification report is:
              precision    recall  f1-score     support
0.0           1.000000  0.989899  0.994924   99.000000
1.0           0.989899  1.000000  0.994924   98.000000
accuracy      0.994924  0.994924  0.994924    0.994924
macro avg     0.994949  0.994949  0.994924  197.000000
weighted avg  0.994975  0.994924  0.994924  197.000000
```



Conclusions

Comparison of the Evaluation Metrics for both models:

Accuracy score: Decision tree (0.9949) > Logistic regression (0.9898)
F1 score: Decision tree (0.9949) = Logistic regression (0.9899)
Recall score: Decision tree (0.9899) > Logistic regression (0.9800)
ROC-AUC score: Decision tree (0.9949) > Logistic regression (0.9900)



Based on the above comparison, we can see that ***Decision Tree Classifier*** produces more **accurate** results than the ***Logistic Regression*** model.

Note: This does not take into account the computational time and interpretability factors.

Bibliography

[1] Alvarez, J. (2023). Course Notes: Caregivers and Machine Learning, Vector Institute.

[2] Alvarez, J. (2023). Assignments: Caregivers and Machine Learning, Vector Institute.

[3] Kaggle. (n.d.). Credit Card Fraud Detection. Retrieved May 5, 2023, from https://www.kaggle.com/mlg-ulb/creditcardfraud

[4] Bankrate (2023). Credit Card Fraud Statistics. Retrieved May 4, 2023, from https://www.bankrate.com/finance/credit-cards/credit-card-fraud-statistics/

[5] OpenAI. (2021). GPT-3.5. Retrieved May 5, 2023, from https://openai.com/blog/gpt-3-5b-parameters/

[6] Google Search. (n.d.). Retrieved May 5, 2023, from https://www.google.com/