Credit Card Fraud Detection

A Deep Dive into Advanced Anomaly Detection Techniques

By Hajar Kaddouri

Structure

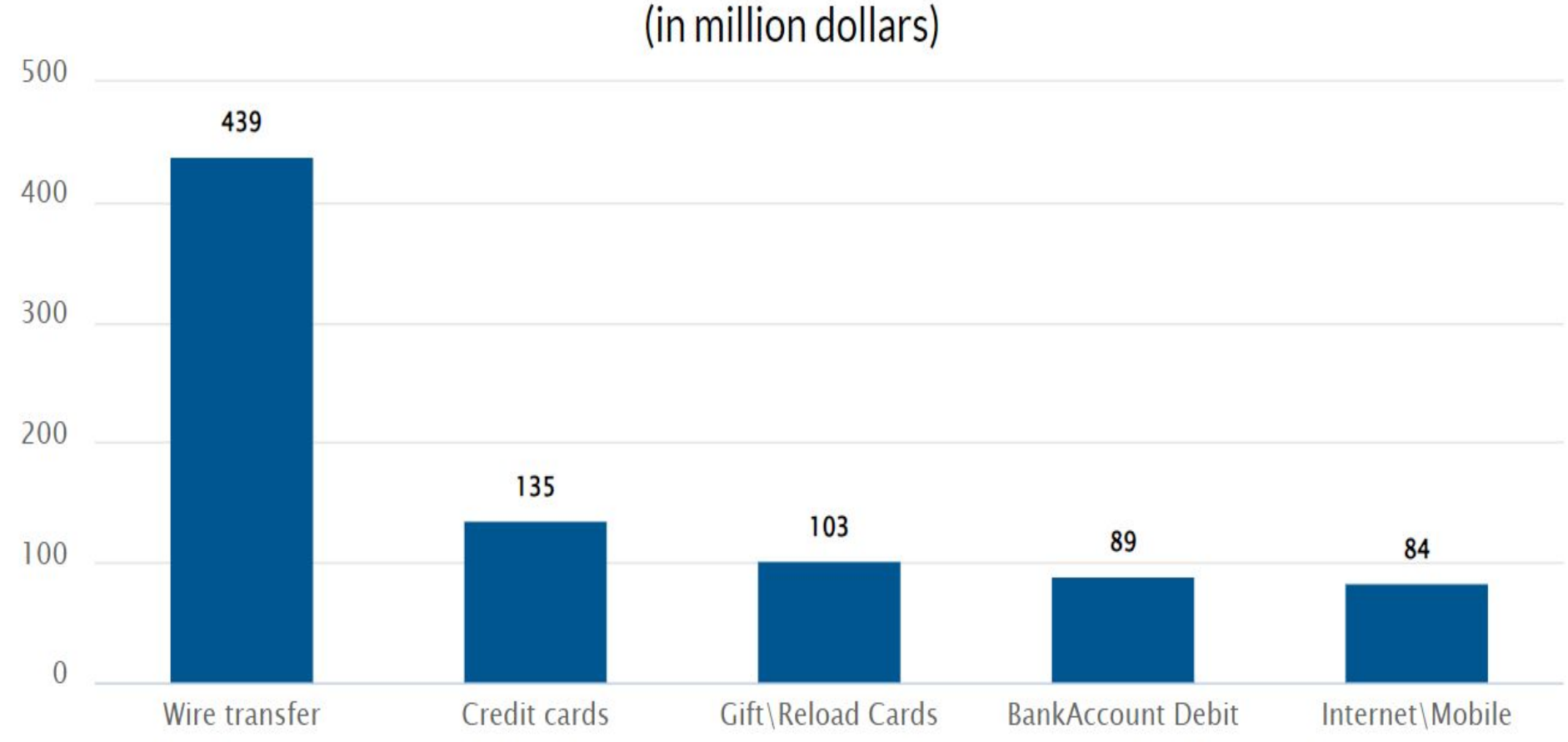
1. Introduction
2. EDA
3. Preprocessing
4. Comprehensive Approach
5. Conclusion

Introduction

The Imperative of Detecting Fraudulent Transactions:

1. **Financial Loss: Billions of dollars are lost annually due to undetected fraudulent transactions.**
2. **Customer Trust: Maintaining the trust of customers is vital. If a customer falls victim to a fraudulent transaction, the trust they have in the banking institution may be lost, which is often difficult to rebuild.**

Average Amount Lost to Fraud, by Payment Method



Problem statement

Identify fraudulent credit card transactions using machine learning. Given the imbalance between fraudulent and non-fraudulent transactions. By using a comprehensive technique that include unsupervised and then supervised models.

DATASETS

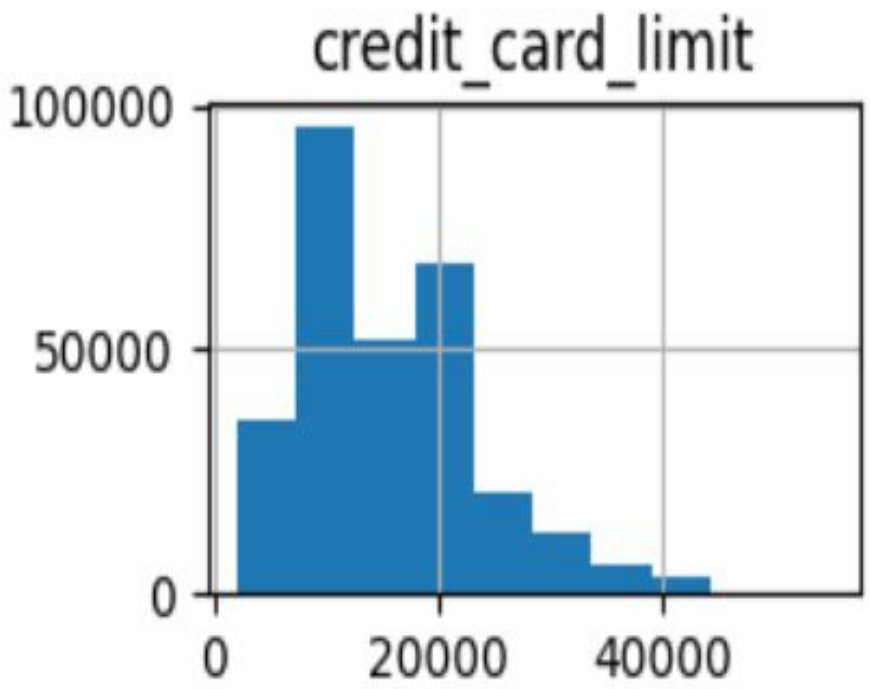
credit_card
city
state
zipcode
credit_card_limit

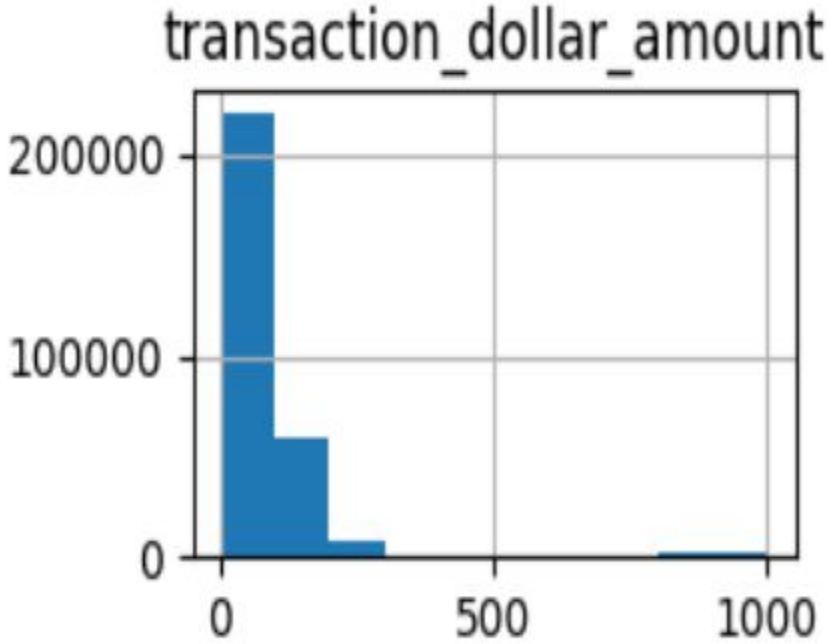
NFORMATIONS ABOUT DATASETS:
https://www.kaggle.com/code/iabhishekofficial/credit-card-fraud-detection/input

- he data provided consists of two tables: "cc_info," and "transactions," .
- containing details of credit card transactions that occurred between August 1st and October 30th.
- It’s clean data (no null - no nan - no duplicate.
- The both files merged to one dataframe (294588, 9)

credit_card
date
transaction_dollar_amount
Long
Lat
Lat

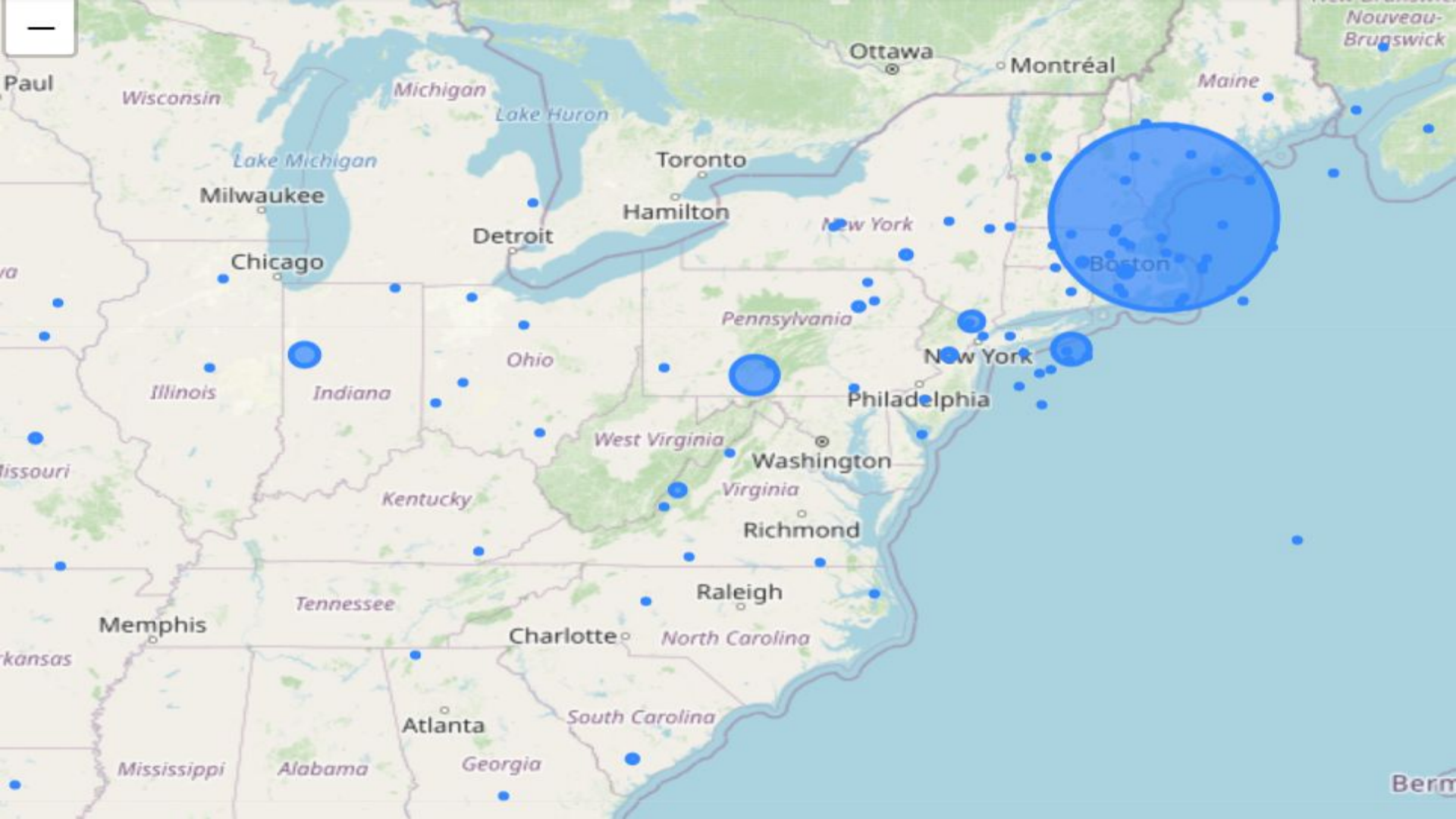
Part I - EDA

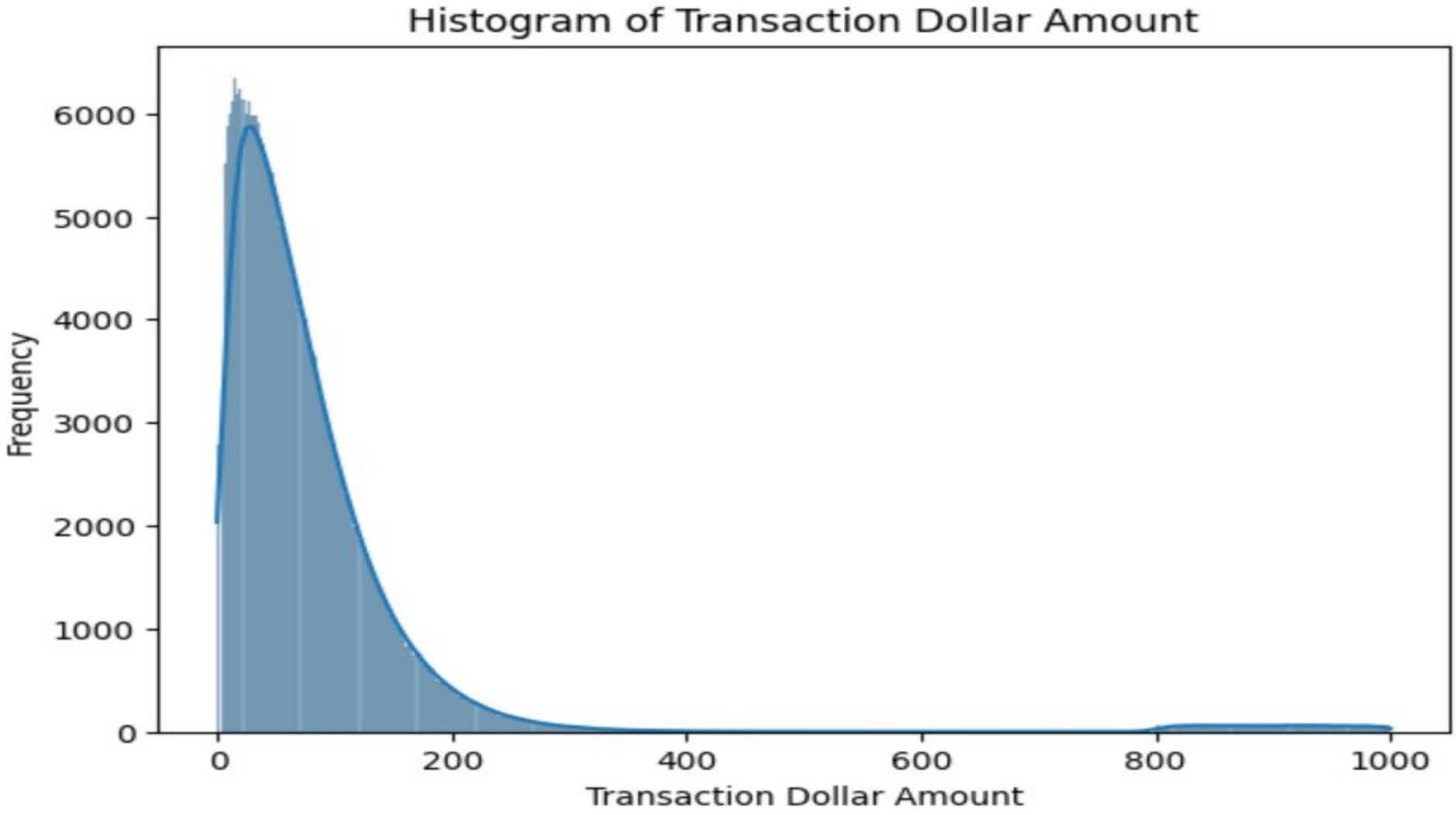




From this distribution we can expect to perform :

- Normalizing (scale the data)
- Handling Imbalanced Data.


Ottawa
Montréal
Nouveau-Brunswick
Maine
Paul
Wisconsin
Michigan
Lake Huron
Lake Michigan
Toronto
Milwaukee
Hamilton
Detroit
New York
Chicago
Boston
Pennsylvania
Ohio
New York
Illinois
Indiana
Philadelphia
West Virginia
Washington
Kentucky
Virginia
Richmond
Raleigh
Tennessee
Memphis
Charlotte
North Carolina
Atlanta
South Carolina
Mississippi
Alabama
Georgia


Histogram of Transaction Dollar Amount
Frequency
6000
5000
4000
3000
2000
1000
0
0
200
400
600
800
1000
Transaction Dollar Amount

Part II - Preprocessing

Data Preprocessing:

- One-hot encoding for credit card numbers
- Normalization using standard scaler.

Feature Engineering:

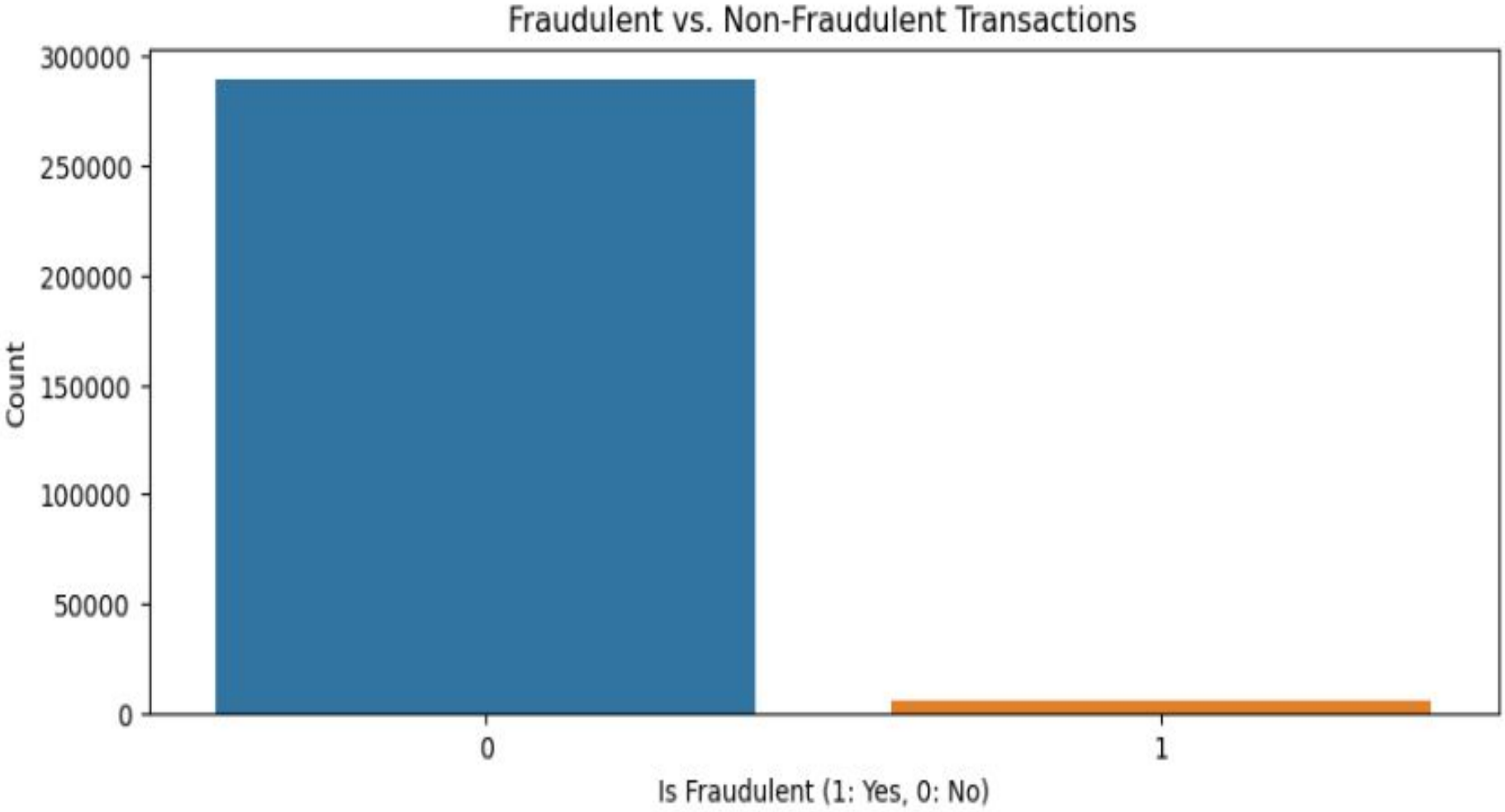
- Calculate the rolling count of transactions for each card
- Calculate the mean and standard deviation for the rolling count of transactions

Part II - K-Means Clustering

In the k-means clustering process, the optimal number of clusters was 2

Number of anomalies detected by KMeans: 14730

	cluster	distance_to_center	kmeans_anomaly
277391	0	2.633927	0
201068	0	0.577556	0
278181	0	3.113150	1
29066	0	0.850952	0
230996	0	1.524767	0


Fraudulent vs. Non-Fraudulent Transactions
300000
250000
200000
150000
100000
50000
0
Count
0
1
Is Fraudulent (1: Yes, 0: No)

Combining the result from the rolling mean technique and k-means :

```
Total anomalies after combining: 20364
```

	kmeans_anomaly	count_anomaly	combined_anomaly
277391	0	1	1
201068	0	1	1
278181	1	1	1
29066	0	1	1
230996	0	1	1

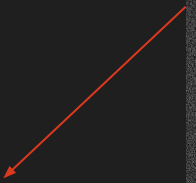
Part III - Supervised LM

- Splitting the dataset
  - x= columns (local and transactions)
  - y=['combined_anomaly']

- Applying SMOTE oversampling on training data
  - 290588 Vs 20364
- Hyperparameter grids
- Model training using GridSearchCV

The best model is Gradient Boosting

```
[INFO] Ensemble Model Results:
Random Forest Score: 0.95
Random Forest Precision: 0.65
Random Forest Recall: 0.73
Random Forest F1 Score: 0.69
Gradient Boosting Score: 0.97
Gradient Boosting Precision: 0.84
Gradient Boosting Recall: 0.73
Gradient Boosting F1 Score: 0.78
```



CONCLUSION

(CADF-2)

rolling statistics

K-Means clustering

Framework

Combining the results in new column

Random Forest

Gradient Boostin

97%

THANKS

Challenges & Questions

RESOURCES

- https://github.com/hajar-kaddouri/Final-Project2023
- https://shiftprocessing.com/credit-card-fraud-statistics/