Real-time Fraud Detection for Secure Financial Transactions

Milestone: Project Report

Group 32

Anagha Veena Sanjeev

Gargi Gokhale

857-398-5307

617-785-4941

sanjeev.an@northeastern.edu

gokhale.g@northeastern.edu

Percentage of Effort Contributed by Anagha: 50%

Percentage of Effort Contributed by Gargi: 50%

Signature of Anagha:

Signature of Gargi:

Submission Date: 12th April 2024

Problem setting

The context of this problem setting lies within the domain of financial transactions, where the goal is to develop a robust fraud detection system. Financial institutions, including banks and credit card companies, face the constant challenge of identifying and preventing fraudulent activities within their transactions. The rise of digital transactions has amplified the need for sophisticated fraud detection systems to safeguard against evolving fraudulent techniques. The application involves analyzing large volumes of transactional data to identify patterns and anomalies that may indicate fraudulent behavior. Challenges in this domain include the dynamic nature of fraud techniques, the vast amount of data generated, and the necessity for near-real-time detection to prevent financial losses.

Problem definition

Developing an effective fraud detection system to identify and prevent fraudulent financial transactions. The focus is on creating models that can accurately distinguish between legitimate and fraudulent activities.

Data Sources

The data source for the project is a Kaggle dataset available at the following link

https://www.kaggle.com/datasets/kartik2112/fraud-detection?resource=download

Data Description

- This is a simulated credit card transaction dataset containing legitimate and fraudulent transactions from the duration of 1st Jan 2019 - 31st Dec 2020. It covers the credit cards of 1000 customers doing transactions with a pool of 800 merchants.
- The dataset includes information such as transaction date and time, credit card number, merchant details, transaction category, amount, customer details (first name, last name, gender, street address), and a label indicating whether the transaction is fraudulent or not.

- There are two files: fraudTrain.csv and fraudTest.csv, representing the training set and test set, respectively.
- The dataset is large, with 14 columns and 1296677 rows providing diverse information about each transaction.
- Labels for fraud and non-fraud transactions are available.

Data Exploration

- **Data Loading and Initial Inspection:**

We loaded the dataset, fraudTrain.csv, to initiate our analysis. Verifying the correct loading of data is crucial in identifying any potential issues or inconsistencies at an early stage.

- **Handling Missing Values:**

After loading the dataset, we conducted a check for missing values. Missing values can compromise the integrity and reliability of analyses and predictions. To ensure data quality and consistency, we opted to drop rows with missing values using the df.dropna() method.

After analyzing the data, it was found that each column contained only one missing value. Therefore, that row was removed.

- **Outlier Identification:**

Utilizing the Z-score method, we proceeded with outlier detection. The Z-score approach helps identify potential anomalies in the dataset. Outliers, if present, may signify data errors or unusual patterns deserving further scrutiny, thus ensuring the reliability of our dataset for subsequent analysis.

- **Column Removal:**

During our initial inspection, we noticed a redundant column labeled 'Unnamed: 0'. Recognizing its non-essential nature, we removed this column to streamline the DataFrame and alleviate computational overhead, promoting efficiency in subsequent analyses.

- **Summary Statistics:**

We computed and displayed summary statistics to gain insights into the numerical features of the dataset. Summary statistics, encompassing metrics such as mean, standard deviation, and quartile values, offer a comprehensive view of the data distribution, aiding in better comprehension and subsequent analysis.

Few insights are as follows:

Transaction Variability: The dataset contains a wide range of transaction amounts, from as low as $1 to as high as $11,872.21, indicating a significant variability which can be useful for detecting outliers or unusual transactions that might indicate fraud.

Rarity of Fraud: Fraudulent transactions are rare, constituting less than 1% of all transactions. This low incidence rate presents a challenge for predictive modeling due to class imbalance.

City Population Diversity: The city population associated with transactions varies dramatically, from just 23 to 2.91 million. This suggests that the data spans both highly urban and rural areas, potentially affecting transactions and fraud risks.

- **Data Type Conversion**

We converted the data type of the 'zip' column to string for consistency in handling ZIP code information. Additionally, the 'dob' column, representing dates of birth, was converted to datetime type to facilitate easier manipulation and analysis of date-related information.

- **PCA**

To decide on the number of components for PCA, we plot the cumulative explained variance ratio as a function of the number of components. This curve shows the proportion of variance explained by the first k principal components.

We chose to perform PCA with 5 components based on the analysis of the above plot. We observed that the explained variance ratio starts to level off around 5 components. This suggests that adding more components beyond this point does not significantly increase the amount of variance explained by the model. Therefore, selecting 5 components captures a satisfactory proportion of the variance in the data while avoiding overfitting. Additionally, using 5 components strikes a balance between dimensionality reduction and preserving valuable information in the data, making it a reasonable choice for further analysis or modeling.

Variance captured by each component:

Component 1: 32.11%

Component 2: 22.77%

Component 3: 14.37%

Component 4: 11.11%

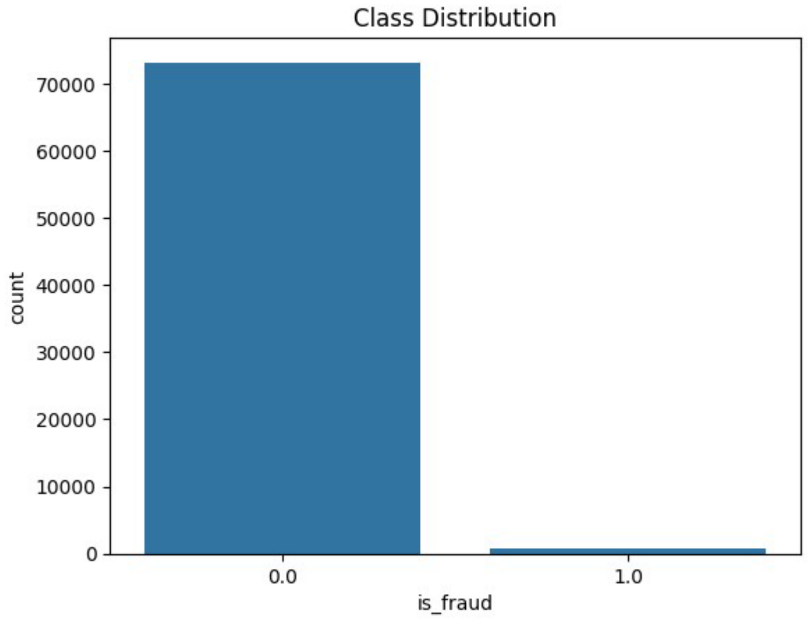
Component 5: 10.48%

Total Variance Captured: 90.84%

Since the variance captured by PCA is low, there is a significant loss of information. Therefore, we will not use PCA for dimensionality reduction.

- **Data Visualization**

a) Fraud Distribution



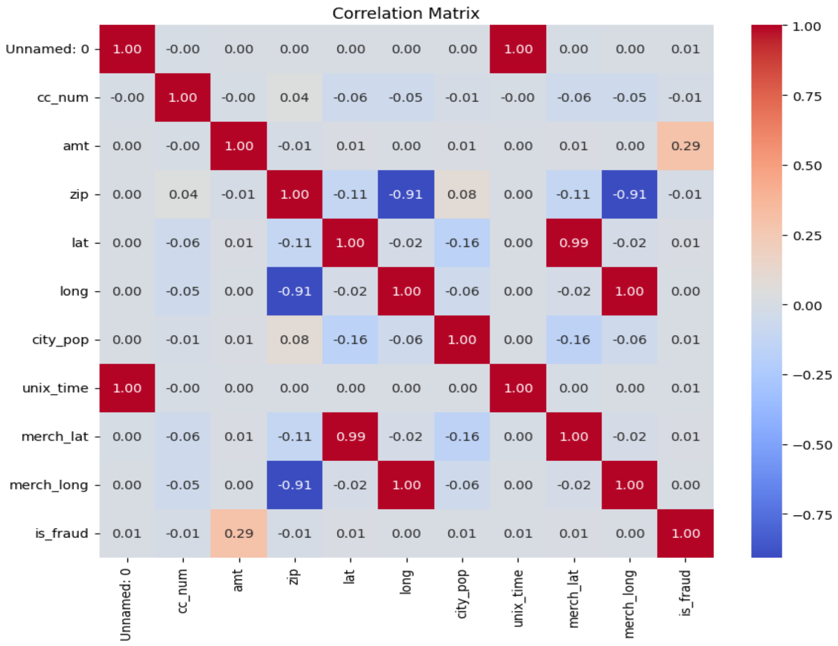
* The plot visually presents the distribution of transactions into two categories - fraud (1) and non-fraud (0). This categorical breakdown allows us to understand the composition of the dataset based on the occurrence of fraudulent and non-fraudulent transactions.

* On the plot, the x-axis corresponds to the two categories (fraud or non-fraud), while the y-axis quantifies the count of transactions within each category. The height of each bar represents the

number of transactions belonging to each category, providing a clear visual representation of the distribution.

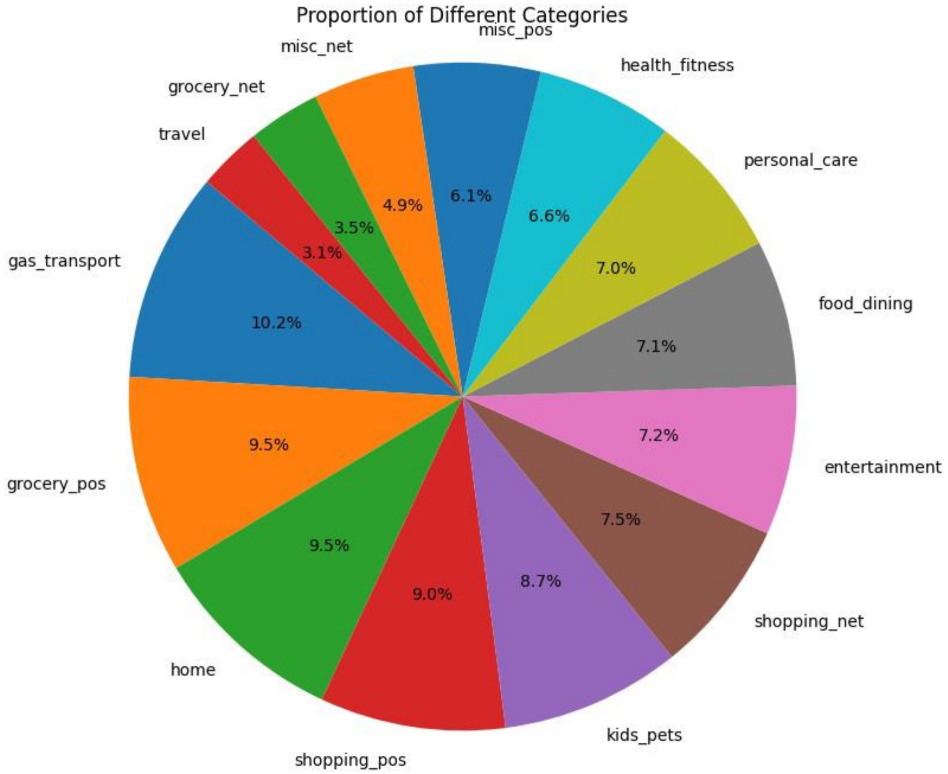
* The plot clearly highlights a significant class imbalance between fraudulent (Class 1) and non-fraudulent (Class 0) transactions. This awareness is crucial for developing an effective fraud detection model, as it indicates that the dataset contains far fewer instances of fraudulent transactions compared to non-fraudulent ones.

b) Correlation Matrix



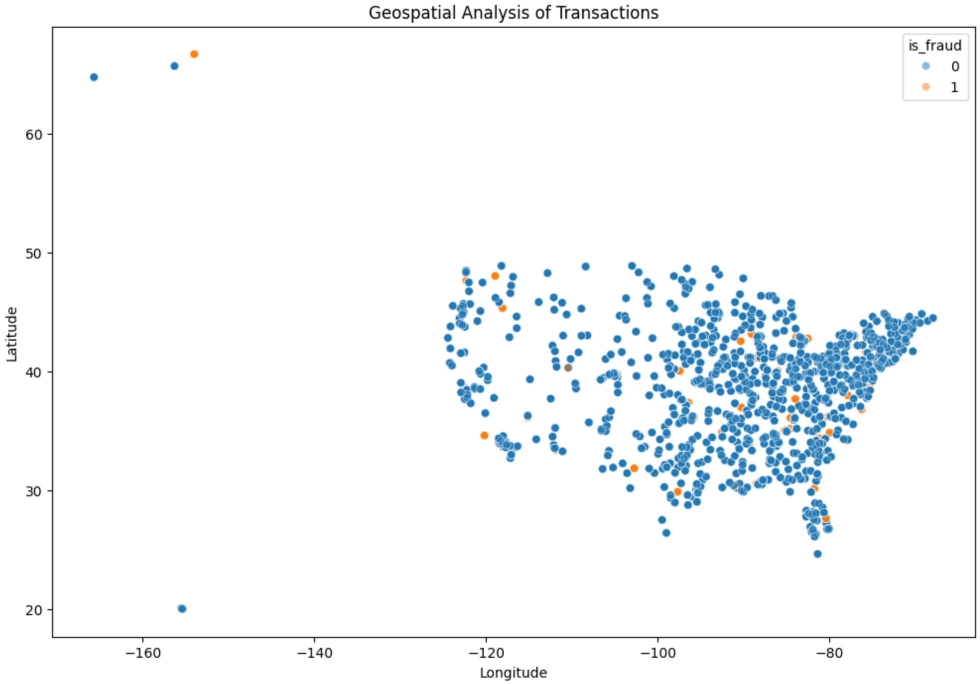
The heatmap uses colors to represent the strength and direction of the linear relationship between pairs of variables. The correlation coefficient ranges from -1 to 1, where -1 indicates a perfect negative correlation, 1 indicates a perfect positive correlation, and 0 indicates no correlation. We can see that merch_long and long have positive correlation and is_fraud and cc_num have no correlation between them.

c) Proportion of different categories



The pie chart visually represents the distribution of transactions across different categories. Each slice of the pie corresponds to a specific category, and the size of each slice represents the proportion of transactions associated with that category relative to the total number of transactions. From the pie chart we can observe that gas_transport accounts for the largest proportion of transactions i.e, 10.2%.

d) Geospatial Analysis



Each point on the scatter plot represents a transaction. The x-axis represents the longitude of the transaction location, while the y-axis represents the latitude. The hue of each point is determined by the 'is_fraud' column. This column contains binary values indicating whether each transaction is fraudulent or not. Fraudulent transactions occur close to legitimate transactions, potentially due to factors such as shared locations or similar transaction patterns.

Data Mining Tasks

Handling the Imbalance in data using Random Under - Sampling

First, we apply the model without handling the imbalance. We will use Logistic Regression and Decision Tree classifier for our exploration of the method to apply to handle imbalance in data.

Advantages

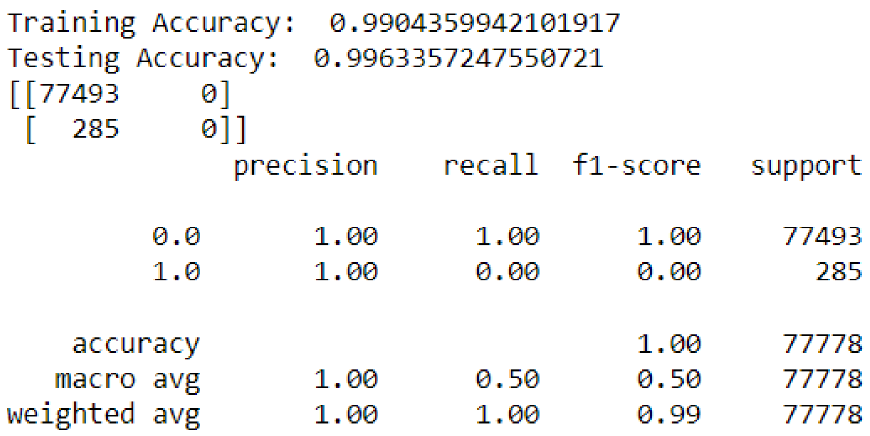
It can help improve run time and storage problems by reducing the number of training data samples when the training data set is huge.

Disadvantages

It can discard potentially useful information which could be important for building rule classifiers. The sample chosen by random under sampling may be a biased sample. And it will not be an accurate representative of the population. Thereby, resulting in inaccurate results with the actual test data set.

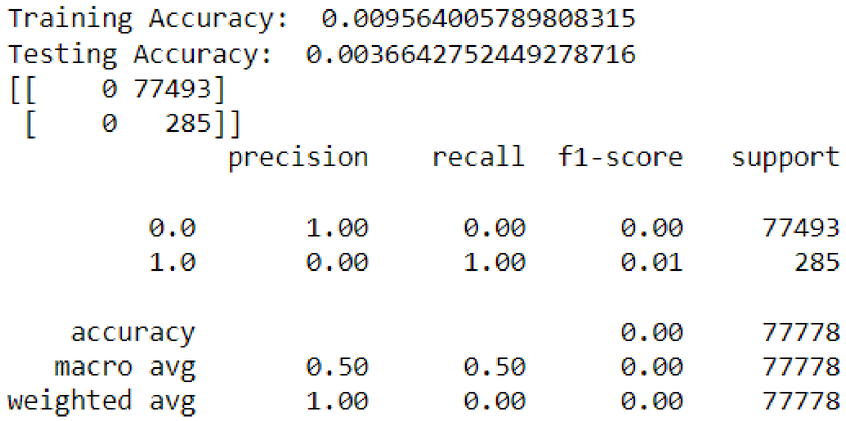
Data Mining Models/Methods

Logistic Regression Classification results without balancing class



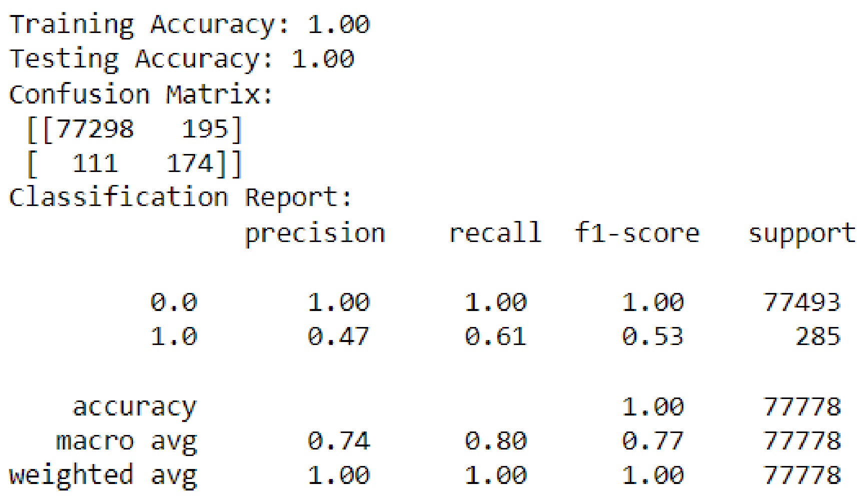
```
Training Accuracy:  0.9904359942101917
Testing Accuracy:  0.9963357247550721
[[77493     0]
 [  285     0]]
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00     77493
         1.0       1.00      0.00      0.00       285

    accuracy                           1.00     77778
   macro avg       1.00      0.50      0.50     77778
weighted avg       1.00      1.00      0.99     77778
```

Logistic Regression Classification Under Sampling



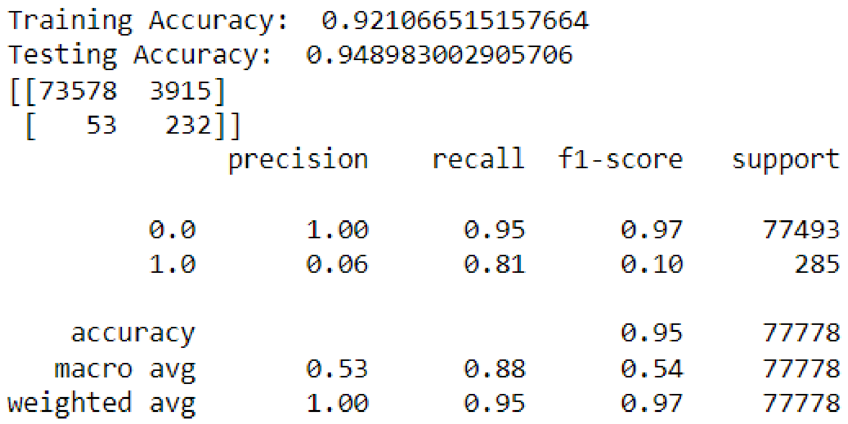
```
Training Accuracy:  0.009564005789808315
Testing Accuracy:  0.0036642752449278716
[[    0 77493]
 [    0   285]]
              precision    recall  f1-score   support

         0.0       1.00      0.00      0.00     77493
         1.0       0.00      1.00      0.01       285

    accuracy                           0.00     77778
   macro avg       0.50      0.50      0.00     77778
weighted avg       1.00      0.00      0.00     77778
```

Decision Tree Classification results without balancing class



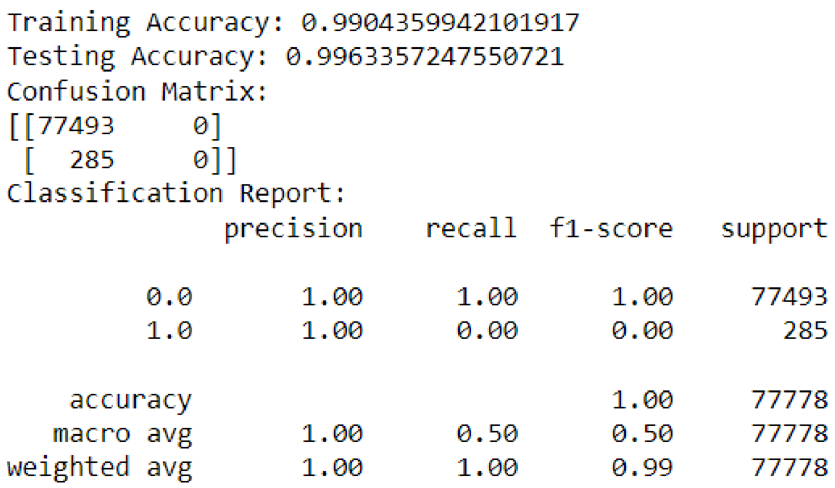
```
Training Accuracy: 1.00
Testing Accuracy: 1.00
Confusion Matrix:
 [[77298   195]
 [  111   174]]
Classification Report:
               precision    recall  f1-score   support

          0.0       1.00      1.00      1.00     77493
          1.0       0.47      0.61      0.53       285

     accuracy                           1.00     77778
    macro avg       0.74      0.80      0.77     77778
 weighted avg       1.00      1.00      1.00     77778
```

Decision tree Classification Under Sampling



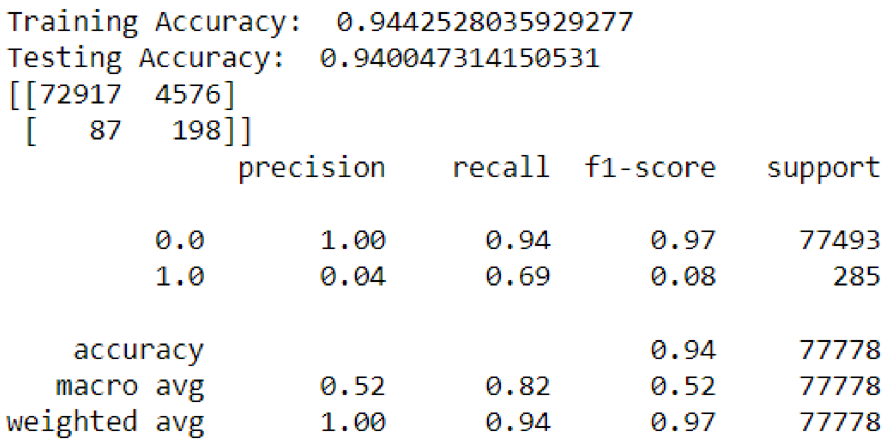
```
Training Accuracy:  0.921066515157664
Testing Accuracy:  0.948983002905706
[[73578  3915]
 [   53   232]]
               precision    recall  f1-score   support

          0.0       1.00      0.95      0.97     77493
          1.0       0.06      0.81      0.10       285

     accuracy                           0.95     77778
    macro avg       0.53      0.88      0.54     77778
 weighted avg       1.00      0.95      0.97     77778
```

Random Forest Classifier results without balancing class



```
Training Accuracy: 0.9904359942101917
Testing Accuracy: 0.9963357247550721
Confusion Matrix:
[[77493     0]
 [  285     0]]
Classification Report:
              precision    recall  f1-score   support

         0.0       1.00      1.00      1.00     77493
         1.0       1.00      0.00      0.00       285

    accuracy                           1.00     77778
   macro avg       1.00      0.50      0.50     77778
weighted avg       1.00      1.00      0.99     77778
```

Random Forest Classifier results Under Sampling

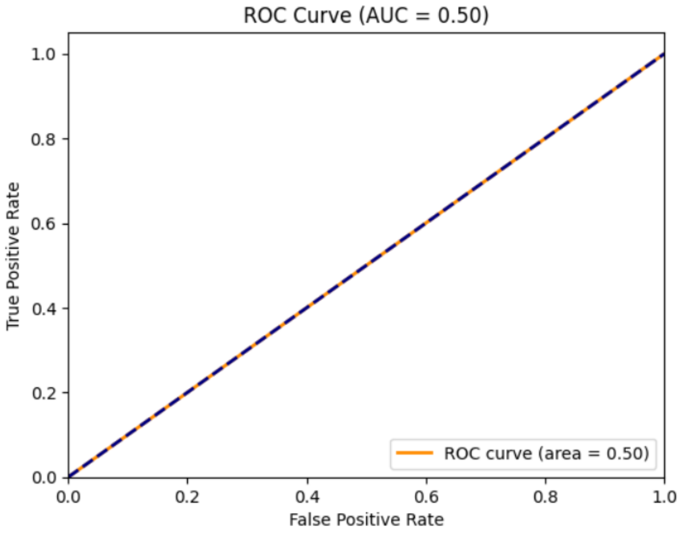


```
Training Accuracy:  0.9442528035929277
Testing Accuracy:  0.940047314150531
[[72917  4576]
 [   87   198]]
              precision    recall  f1-score   support

         0.0       1.00      0.94      0.97     77493
         1.0       0.04      0.69      0.08       285

    accuracy                           0.94     77778
   macro avg       0.52      0.82      0.52     77778
weighted avg       1.00      0.94      0.97     77778
```

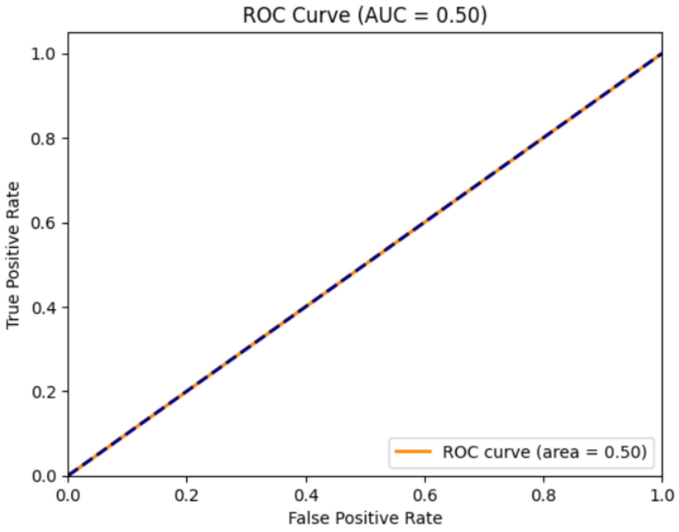
Performance Evaluation

Linear Regression

Before Under – Sampling

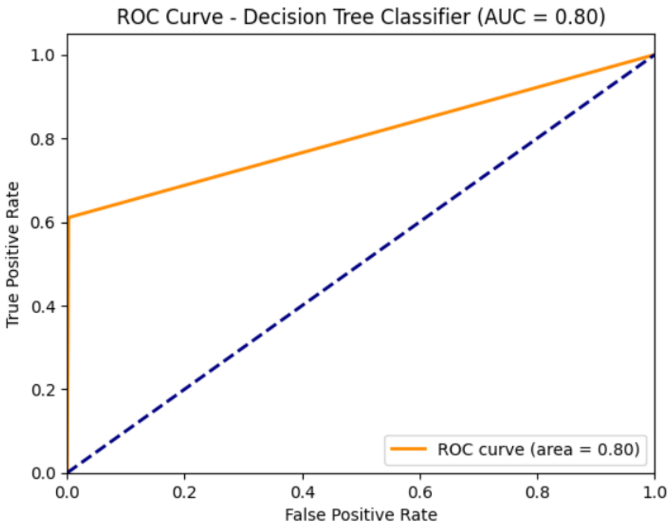


After Under – Sampling

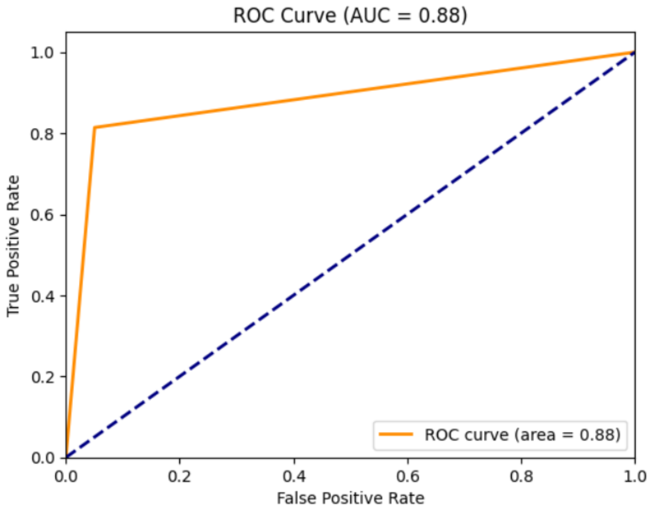


Decision Tree

Before Under sampling

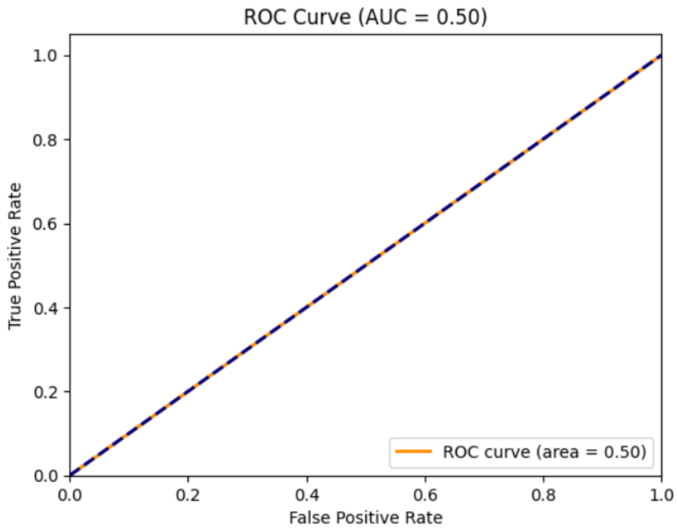


After Under- Sampling

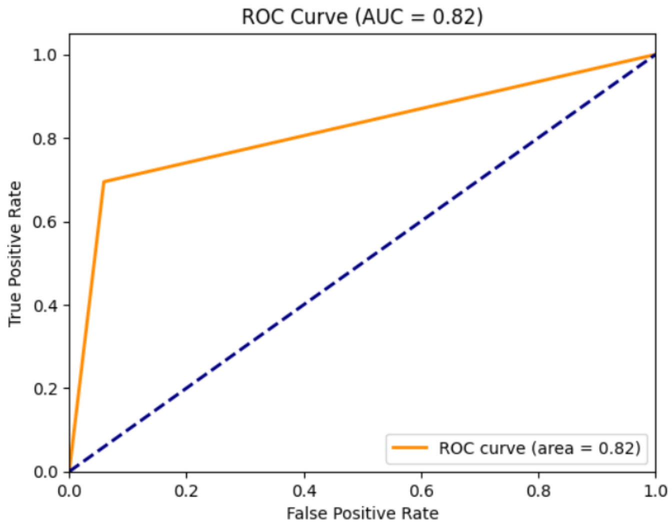


Random Forest Model

Before Under – Sampling



After Under – Sampling



Project Results

The logistic regression model demonstrates extremely low accuracy and F1-score, indicating poor performance in classifying fraudulent transactions.

In contrast, both decision tree and random forest models outperform logistic regression, achieving significantly higher accuracy, precision, and recall for detecting fraudulent transactions.

The decision tree model exhibits the highest AUC, indicating better overall performance in distinguishing between fraudulent and non-fraudulent transactions compared to the other models.

Based on these findings, the logistic regression model may not be suitable for fraud detection tasks due to its poor performance compared to decision tree and random forest models.

Given the superior performance of the decision tree and random forest models, it is recommended to deploy either of these models for fraud detection in production environments. Continuous monitoring and refinement of the chosen model are essential to adapt to evolving fraud patterns and maintain effectiveness over time.

Impact of the Project Outcomes

The outcomes of the project, particularly in fraud detection for credit card transactions, can have significant impacts on various stakeholders:

- **Financial Institutions:** Financial institutions, including banks and credit card companies, stand to benefit from more effective fraud detection systems. By implementing robust machine learning models, they can reduce financial losses associated with fraudulent transactions, improve customer trust, and enhance the overall security of their services.
- **Customers:** Customers will experience increased security and confidence in using credit cards for transactions. With improved fraud detection systems in place, they are less likely to fall victim to fraudulent activities, resulting in fewer disputes, inconvenience, and potential financial losses.
- **Regulatory Authorities:** Regulatory bodies overseeing the financial sector may benefit from reduced instances of fraud and financial crimes. Enhanced fraud detection capabilities can contribute to maintaining the integrity and stability of the financial system, potentially leading to better regulatory compliance and consumer protection.
- **Law Enforcement Agencies:** Law enforcement agencies tasked with investigating financial crimes can leverage the insights and data provided by fraud detection systems to identify patterns, trends, and perpetrators of fraudulent activities. This can lead to more

targeted and effective enforcement efforts, resulting in a safer and more secure financial environment.

- **Technology Providers:** Companies developing fraud detection solutions and machine learning algorithms stand to gain from the adoption of their technologies by financial institutions and other stakeholders. Continual improvements in machine learning techniques and data analytics can further drive innovation in fraud prevention and detection.

Overall, the successful implementation of advanced fraud detection systems can have far-reaching implications, ranging from financial security and consumer trust to regulatory compliance and law enforcement effectiveness. By leveraging data-driven approaches and cutting-edge technologies, stakeholders can collectively work towards minimizing the impact of fraudulent activities on the financial ecosystem.