Data Visualization Methods for Improving the Identification and Examination of Fraudulent Transactions in Online Payment Platforms

A Thesis

Submitted in partial fulfillment of the requirements

for the award of the Degree of

BACHELOR OF SCIENCE

IN

MATHEMATICS AND COMPUTING

BY

AYUSH KUMAR

IMH/10017/22

BIRLA INSTITUTE OF TECHNOLOGY

MESRA-835215, RANCHI

APPROVAL OF THE GUIDE

Recommended that the thesis entitled **"Data Visualization Methods for Improving the Identification and Examination of Fraudulent Transactions in Online Payment Platforms"**, submitted by **Ayush Kumar** under my supervision and guidance, be accepted as fulfilling the requirements for the award of the degree of **Integrated Master of Science**.

To the best of my knowledge, this work is original, has not been submitted for the award of any other academic degree, and contains no plagiarized content.

Date: ______________

Guide name: Dr. Payel Das
Department: Mathematics
Birla Institute of Technology, Mesra, Ranchi

DECLARATION CERTIFICATE

I hereby declare that:

1. The work presented in this thesis is my original research carried out under the guidance of my supervisor.
2. This thesis has not been submitted, either in part or in full, for the award of any other degree.
3. All the guidelines prescribed by the institute, including those related to ethics, have been strictly followed during the course of this work.
4. Proper acknowledgment has been made for all sources of information and assistance used in the preparation of this thesis.

Ayush Kumar
IMH/10017/22

CERTIFICATE OF APPROVAL

This is to certify that the work embodied in this thesis entitled **“Data Visualization Methods for Improving the Identification and Examination of Fraudulent Transactions in Online Payment Platforms”** is carried out by **Ayush Kumar (IMH/10017/22)** and sanctioned for the degree of **Integrated Master Of Science** at Birla Institute of Technology, Mesra, Ranchi.

Date: Place:

Internal Examiner External Examiner
(Chairman)

Head of Department

ABSTRACT

The growing number of online payment transactions has resulted in a parallel increase in fraudulent activities, and thus there is a critical need for efficient detection mechanisms. This project explores the use of a combination of data visualization methods and machine learning (ML) to improve the detection and analysis of fraud in online payment systems. The central idea is that the visualization of complex transaction data sets can uncover patterns and anomalies that may be hidden in conventional data formats, thus enhancing the capacity to detect fraudulent activity.

This work utilizes machine learning models to create models that are able to properly classify transactions as either genuine or phony. By examining transactions' attributes like transaction type, amount, time step, and origin/destination account balances, the ML models will try to learn the faint fraud indicators.

An integral aspect of this project involves incorporating data visualization throughout the process of detecting fraud. Graphical representations are used to examine the properties of individual variables, relationships between two variables, and sophisticated interactions among multiple variables. The visualizations are used to facilitate a better intuitive grasp of the data in order to assist in the identification of outliers, trends, and correlations that can be useful for fraud detection. In addition, seeing the result of the machine learning models can make them more interpretable and enable verification of their predictions. Finally, this project aims to illustrate the collaborative advantage of using data visualization with machine learning for detecting fraud. The aim is to create a stronger, more informative, and more actionable method of protecting online payment systems from fraud.

ACKNOWLEDGEMENT

I would like to take this moment to convey my warmest and sincerest gratitude to my respected project supervisor, Dr. Payel Das, for her constant encouragement, constant guidance, support, and constructive feedback she has offered me throughout the whole duration of this thesis process. Working under her expert guidance has been an extremely rewarding and life-altering experience that has greatly enhanced my knowledge of the subject matter and motivated me to explore it deeper and learn more about it.

I would like to convey my sincere appreciation to Mr. Abhinav Tandon, Department Head, Department of Mathematics and Computing, prestigious Birla Institute of Technology, Mesra, Ranchi. His untiring support and the provision of this research work through the academic environment and the infrastructure needed for this research work were the turning points in the successful completion of this research work.

I would like to express my heartfelt and genuine appreciation to all the committed members of the faculty of the Department of Mathematics for their ongoing support and encouragement. Their relentless efforts to provide useful knowledge and instill essential skills have been a part of my academic life and have contributed significantly to my success in a very real sense.

Lastly, I would like to thank my parents most sincerely for the encouragement, great love, and rock-solid support that they have given to me throughout the years.

Contents

List of Figures

1 Introduction

1.1 Background

The widespread adoption of online payment systems has brought about notable challenges in safeguarding transaction integrity. As the volume of digital financial exchanges continues to grow, fraudulent schemes have also become more intricate and prevalent. Traditional fraud detection techniques, which often rely on static rule-based frameworks, are increasingly insufficient against such dynamic threats. As a result, the integration of machine learning methods with data visualization has gained prominence as a more adaptive and insightful approach for detecting fraudulent transactions.

Visual analytics plays a pivotal role in interpreting large-scale datasets, especially in the financial sector. By transforming raw numerical data into graphical representations, it becomes significantly easier to spot trends, irregularities, and hidden structures that may not be immediately evident. This research, titled *"Data Visualization Methods for Improving the Identification and Examination of Fraudulent Transactions in Online Payment Platforms"*, leverages various visualization strategies to explore transactional data and uncover potential indicators of fraud.

1.2 Objectives

The primary aim of this work is to conduct a detailed analysis of financial transaction records to uncover signs of fraudulent behavior through data-centric techniques. The dataset under investigation includes attributes such as the nature of the transaction, the amount involved, time of occurrence, account balances before and after the transaction, and a binary flag indicating whether the transaction is fraudulent.[1]

To fulfill this main objective, several sub-goals are outlined:

- Investigate the distribution and frequency patterns of different transaction types.
- Detect anomalies and behavioral trends associated with fraudulent activities.
- Address challenges related to data imbalance and determine the relevance of individual features.
- Provide visual interpretations that aid in building and refining predictive machine learning models.

Visualization serves as a central element in each phase of the analysis. It facilitates understanding of the dataset's structure, guides the preprocessing of features, and supports the identification of meaningful patterns. For example, graphical tools can help determine if fraudulent activities are concentrated within specific time frames or associated with certain transaction categories.

1.3 Thesis Organization

This thesis is structured to reflect a logical progression of data analysis tasks, divided into four main sections:

1. **Univariate Analysis:** This section examines individual features independently to assess their statistical properties, distribution, and potential outliers. Such analysis helps in understanding each variable's role in the dataset.
2. **Bivariate Analysis:** This phase explores the interaction between pairs of variables, highlighting how they may influence one another. Techniques such as scatter plots and comparative charts are utilized to identify relationships, especially those that differentiate fraudulent from legitimate transactions.

3. **Multivariate Analysis:** This part involves the concurrent examination of three or more features to uncover complex patterns and correlations. Multivariate methods can reveal relationships that might remain hidden in simpler analyses.

4. **Correlation Analysis:** This component quantifies the strength and direction of associations between numerical variables using statistical measures like Pearson correlation. Visual representations such as heatmaps are employed to illustrate these connections clearly.

Collectively, these analytical techniques provide a well-rounded understanding of the dataset, assist in pinpointing features most relevant to fraud detection, and prepare the data for advanced predictive modeling. The emphasis on visualization throughout the process underscores its value in deciphering high-volume financial data and building effective fraud detection systems.

2 Univariate Analysis

Univariate analysis involves the study of each variable on its own to gain insights into its distribution, frequency, and inherent traits. In the context of online payment fraud detection, univariate visualizations such as bar graphs and histograms help reveal patterns in transaction types, fraud labels, amounts, and time steps. For example, a bar plot of the ‘type’ variable shows that ‘CASHOUT’ and ‘PAYMENT’ are the most common transaction types. Similarly, a histogram of the ‘amount’ variable reveals a heavily right-skewed distribution, indicating that while most transactions are of low value, there are some high-value transactions that might be suspicious. The class distribution of the ‘isFraud’ variable shows a significant imbalance, with fraudulent transactions forming a very small portion of the dataset. This highlights the need for careful handling of class imbalance during model training. The ‘step’ variable, when plotted, displays multiple peaks, suggesting certain hours are busier than others. Univariate analysis is crucial for initial data understanding, detecting outliers, and informing preprocessing steps. It also helps to highlight skewness and frequency-based anomalies which might contribute to fraudulent behavior. Overall, these insights are foundational before applying any machine learning models.

2.1 Distribution of Transaction Types

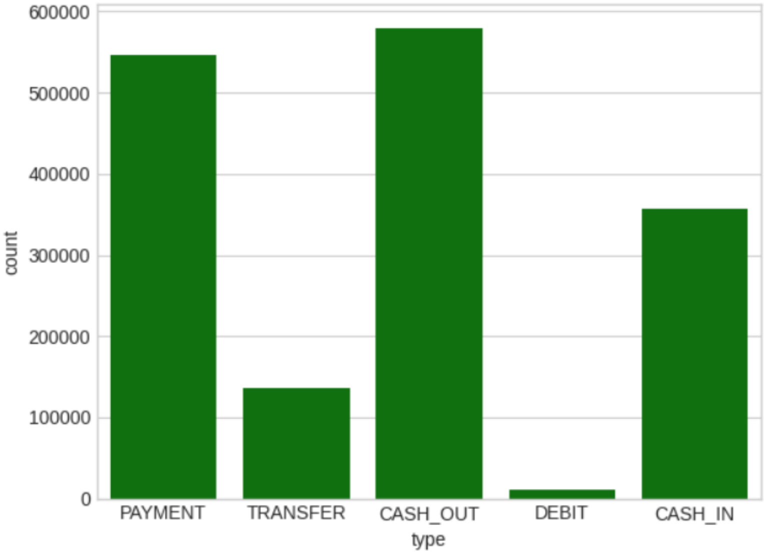


Figure 2.1: Distribution of Transaction Types

Understanding the distribution of transaction types in a financial dataset is critical for interpreting user behavior and identifying potential patterns linked to fraudulent activity. The count plot of transaction types presents the frequency of each type, such as `CASHOUT`, `PAYMENT`, `TRANSFER`, `CASHIN`, and others, recorded in the dataset [1]. This type of visualization is particularly valuable in exploratory data analysis as it provides an overview of how users interact with the financial system and highlights which transaction categories are most commonly used.

Analyzing the distribution of transaction types in a financial dataset offers valuable insights into user behavior and helps uncover potential indicators of fraudulent activity. A count plot depicting transaction types such as `CASHOUT`, `PAYMENT`, `TRANSFER`, and `CASHIN` reveals how frequently each type occurs, which assists in identifying commonly used financial operations [1]. This form of exploratory data visualization plays a key role in assessing system usage and in detecting anomalies that may signal fraud.

Mathematically, let P_r be the frequency of transaction type i, and M denote the total number of transactions. The relative frequency, expressed as $F_r = \frac{P_r}{M}$, provides a probability-based measure of observing a particular transaction type [2]. This allows for normalized comparisons across datasets, especially when dealing with varying data volumes.

The output of the plot shows that CASHOUT and PAYMENT transactions dominate the dataset [1]. These types are commonly associated with routine user activity but are also frequently exploited in fraudulent schemes, especially when combined with TRANSFER operations. Identifying this distribution helps prioritize which transaction types should undergo more rigorous fraud detection techniques. Furthermore, understanding these patterns supports model development and aids in feature selection for supervised learning algorithms aimed at fraud classification.

2.2 Distribution of Fraudulent Transactions

Figure 2.2 illustrates how fraudulent and non-fraudulent transactions are distributed.

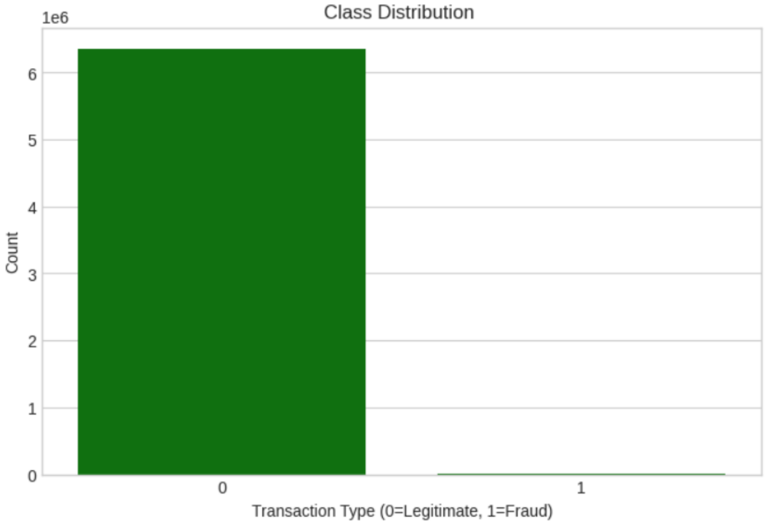


Figure 2.2: Distribution of Fraudulent Transactions

The count plot graphically demonstrates a severely imbalanced dataset, where the fraud transactions form a negligible proportion against the genuine ones. Imbalance is prevalent in actual fraud detection cases and poses a difficulty to machine learning algorithms, which may be skewed towards the majority class. The percentage of fraudulent transactions within the dataset can be calculated using this formula:

$$Fraud\% = (fraudulent/notfraudulent) * 100$$

2.3 Percentage of Fraudulent Transactions

Figure 2.3 Illustrates the percentage of fraudulent transactions.

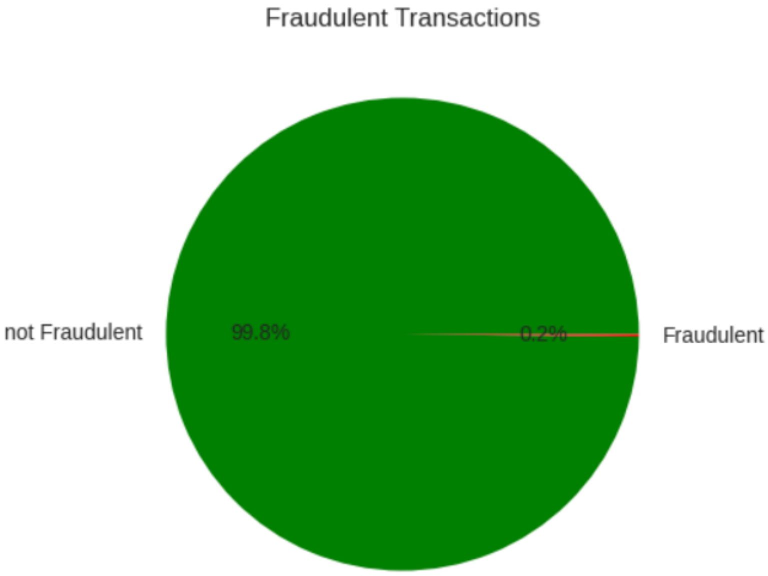


Figure 2.3: Percentage of Fraudulent Transactions

The pie chart shows the relative fraction of fraudulent and non-fraudulent transactions in the data set. It displays a sharp class imbalance in which fraudulent transactions are only a tiny fraction of the total. This is typical in real-world financial data sets, where valid transactions overwhelm fraudulent ones by orders of magnitude. This visualization highlights the need to address class imbalance in fraud detection tasks.

$$\text{Percentage} = \frac{\text{Count of Category}}{\text{Total Count}} \times 100$$

[3]

2.4 Distribution of Transaction Amounts

Figure 2.4 shows the distribution of transaction amounts.

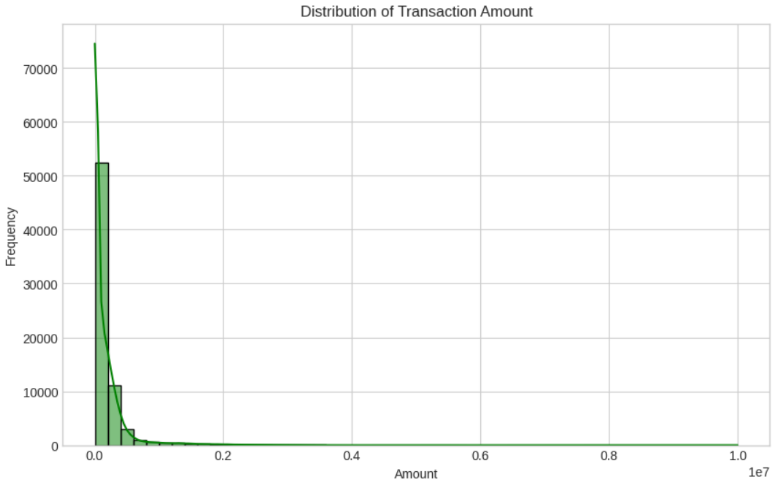


Figure 2.4: Distribution of Transaction Amounts

This histogram shows the distribution of different transaction values throughout the dataset. Most transactions are concentrated in the lower value range, with high-value transactions relatively less common, resulting in a right-skewed distribution. This form of skewness is common in financial datasets where typical payments are the norm and large transactions relatively less common. Visualizing the distribution makes it easier to spot patterns and anomalies in spending behaviors, and can also pick up outliers. Such outliers could be of special interest in fraud detection settings because unusually high or low transaction values can indicate fraud. Overall, this plot is critical to capturing the monetary dynamics of the dataset and provides helpful insights into downstream analytical and machine learning tasks on financial behavior and anomaly detection. If the amounts are $a_1, a_2, ..., a_n$, the mean amount $\bar{a}$ is:

$$\bar{a} = \frac{1}{n} \sum_{i=1}^{n} a_i$$

2.5 Distribution of Time Steps

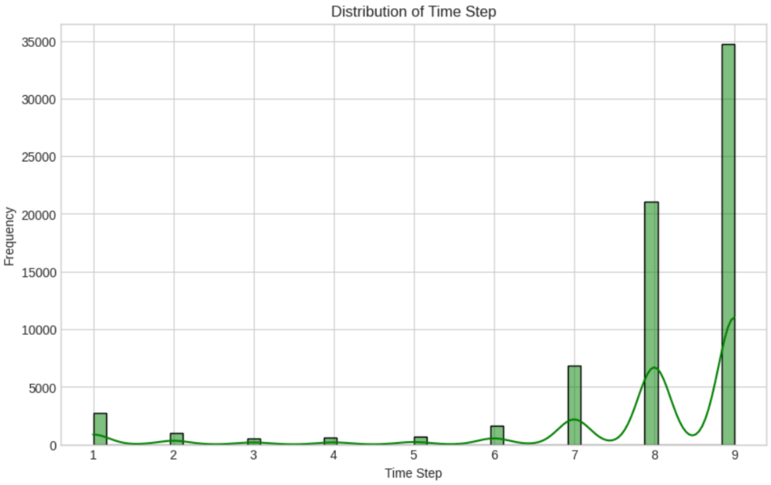


Figure 2.5: Distribution of Time Steps

The histogram gives a graphical view of the distribution of transaction values in the data. It is very evident that most transactions are of relatively small amounts, and transactions of high amounts are much more infrequent. The right-skewed nature of this distribution is common in financial transaction data, where typical consumer activity tends to have lower money values. The plot is helpful in the identification of outliers—very large or very small transactions—which can be indicative of fraud or data abnormalities. Knowing information about the general spread and central tendency of transaction amounts helps in designing proper preprocessing methods such as normalization or transformation for machine learning algorithms. Furthermore, this distribution is helpful in risk-based decision-making by identifying transactions which are very abnormal from typical behavior. With fraud detection, knowing information about where most of the transaction values are can help in the setting of monitoring thresholds as well as influence feature engineering processes.

Analysis: If the time steps are $t_1, t_2, \ldots, t_n$, the standard deviation σ_t is calculated as:

$$\sigma_t = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(t_i - \bar{t})^2}$$

where $\bar{t}$ is the mean of the time steps.[4]

2.6 Hourly Fraud Percentage

Figure 2.6 shows the percentage of fraudulent transactions for each hour of the day.

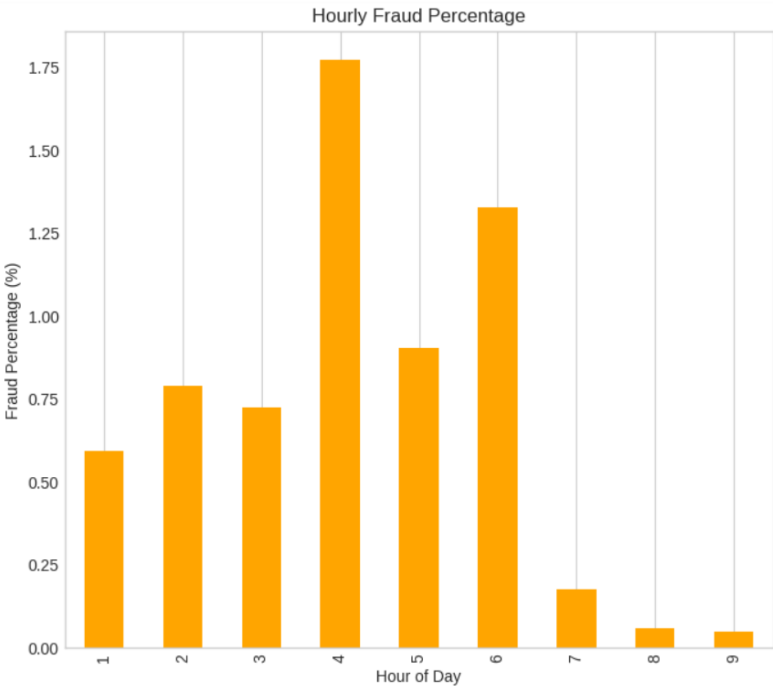


Figure 2.6: Hourly Fraud Percentage

Figure 2.6 is a bar chart that illustrates the percentage of fraudulent transactions that occur in every hour of the day. The x-axis is the hours (0 to 23), and the y-axis is the percentage of transactions that are labeled as fraudulent. This figure is generated by extracting the hour from the transaction time and calculating the fraud rate for every hour as a percentage of the total transactions in the specified time period.

The primary aim of this study is to assess if fraud tends to concentrate during certain periods. By determining high fraud periods, monitoring and control efforts can be directed towards such specific time frames. Patterns in the graphical plot can help identify the offenders' behavioral tendencies, such as exploiting off-peak times when systems' alertness or human monitoring could be diminished.

The hourly fraud percentage is mathematically calculated by dividing the number of fraudulent transactions that occur within an hour by the number of transactions within the same hour, and then multiplying by 100. This gives an even picture of fraud incidence in the course of the day.

The hourly fraud percentage is calculated as:

$$\text{Hourly Fraud Percentage}(h) = \frac{\text{Number of fraudulent transactions in hour } h}{\text{Total number of transactions in hour } h} \times 100\%$$

This approach, based on analyzing temporal fraud patterns, is inspired by methods discussed in prior research [5]. The analysis of fraud by time of day helps identify high-risk hours, allowing for more focused monitoring and mitigation efforts.

3 Bivariate Analysis

“Bivariate analysis is useful for revealing suspicious patterns between two variables, aiding in the detection of online payment fraud.” In the realm of detecting online payment fraud, this type of analysis is vital for comprehending how various elements interact and may lead to fraudulent behavior. This chapter delves into several important bivariate relationships within the dataset to shed light on these interactions.

3.1 Transaction Type vs. Fraud Status

Figure 3.1 shows the relationship between transaction type and fraud amount.

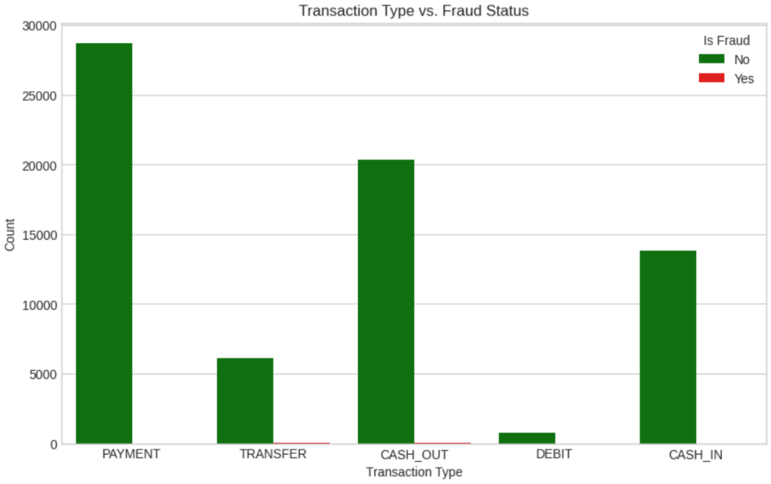


Figure 3.1: Count Plot of Transaction Type vs. Fraud amount

The graph illustrates the relationship between fraud status and transaction type. This bar plot provides information regarding the relationship between various types of transactions and fraudulent transactions. Each bar represents a different type of transaction, and the variation in color (hue) is whether the transaction was fraudulent or not. The height of the bars is the number of fraudulent transactions by type.

The primary goal of the visualization here is to figure out which type of transactions is most vulnerable to fraud. What one can observe from the plot here is what types have higher fraud cases and what types have higher financial risk. This will allow stakeholders to see where more attention or preventive measures can be taken.

The outcome of this analysis is the rate of fraud by transaction type. Higher fraud rates in specific types of transactions can be symptomatic of patterns or vulnerabilities that criminals exploit more often. Awareness of these trends is necessary to enable companies to distribute resources effectively and minimize the monetary risk of fraudulent transactions. This graph is an effective strategic decision-making tool for fraud prevention and detection.

Analysis: Let T represent a specific transaction type and F represent the fraud status (1 for fraudulent, 0 for legitimate). The total fraud amount for a specific transaction type T is:

$$\text{Total Fraud Amount for Type } T = \sum_{i=1}^{n} \text{amount}_i \quad \text{where} \quad \text{isFraud}_i = 1 \text{ and type}_i = T$$

3.2 Transaction Amount vs. Fraud Status

Figure 3.2 shows the relationship between transaction amount and fraud status using a scatter plot.

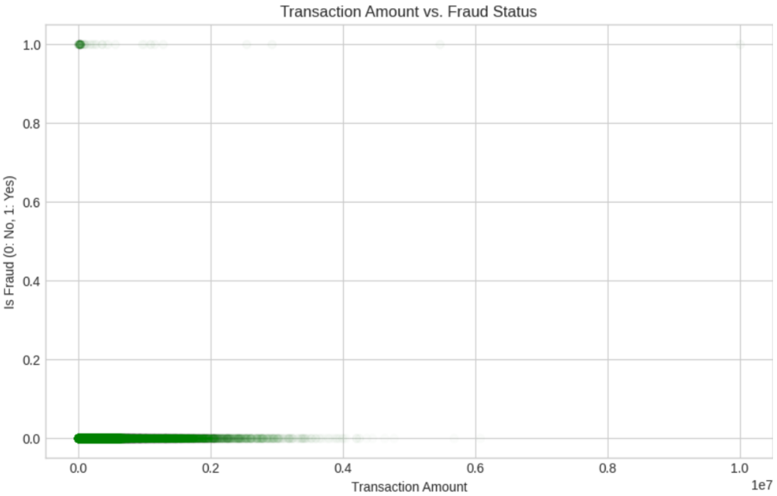


Figure 3.2: Scatter Plot of Transaction Amount vs. Fraud Status

This is a **scatter plot** visualizing individual transactions with their amounts and fraud status. The transparency is adjusted using the `alpha` parameter to make overlapping points more visible. The x-axis represents the transaction amount, while the y-axis indicates whether the transaction was fraudulent (1) or not (0).

The purpose of this plot is to examine whether there is a correlation between the transaction amount and the likelihood of fraud. Although the scatter plot does not directly compute a correlation, it allows for a visual assessment of any patterns that may suggest such a relationship.

Analysis: For a more quantitative analysis, one can calculate the Pearson correlation coefficient (r), which measures the linear relationship between the transaction amount (x) and fraud status (y).

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Where: - x is the transaction amount, - y is the fraud status (0 or 1), - $\bar{x}$ and $\bar{y}$ are the means of x and y respectively [6]. The output of this analysis may reveal if fraudulent transactions tend to cluster at certain transaction amounts, indicating areas of higher financial risk.

3.3 Time Step vs. Fraud Status

Figure 3.3 shows the relationship between the time step of a transaction and its fraud status using a scatter plot.

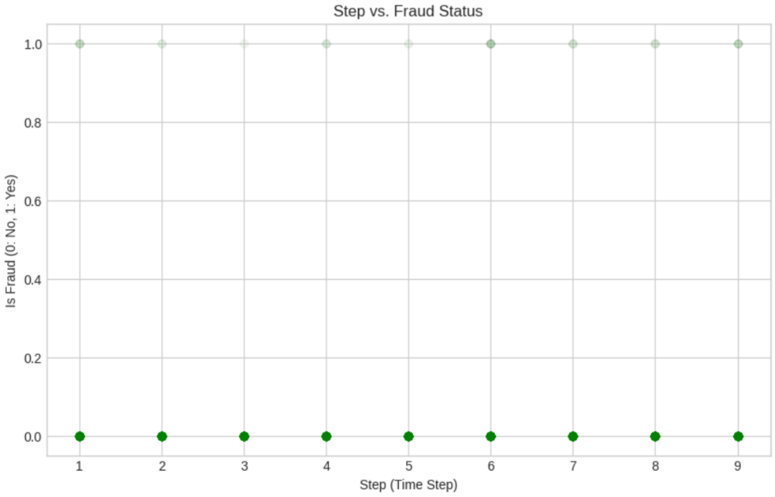


Figure 3.3: Scatter Plot of Time Step vs. Fraud Status

The figure displays the relationship between the transaction time step and fraud status using a scatter plot. The x-axis represents the time step, which indicates the sequence of transactions over time, while the y-axis shows the fraud status (0 for non-fraudulent, 1 for fraudulent). The purpose of this plot is to investigate whether fraudulent transactions are more prevalent at specific time steps, allowing us to identify potential patterns related to time. By visualizing the data in this manner, one can easily observe if certain time periods have a higher concentration of fraud incidents.

To gain a more detailed understanding, the Pearson correlation coefficient can be calculated to assess the linear relationship between the time step and fraud status. The metric thus can help quantify any potential correlation between the time of the transaction and the likelihood of fraud. The results of this analysis may reveal whether fraud tends to cluster at certain time intervals or if it is more evenly distributed, providing valuable insights into temporal patterns of fraudulent activity.

$$r = \frac{\sum_{i=1}^{n}(t_i - \bar{t})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(t_i - \bar{t})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Where t represents the timestep.[7]

3.4 Amount vs. Transaction Type

Figure 3.4 shows the average transaction amount for each transaction type using a bar plot.

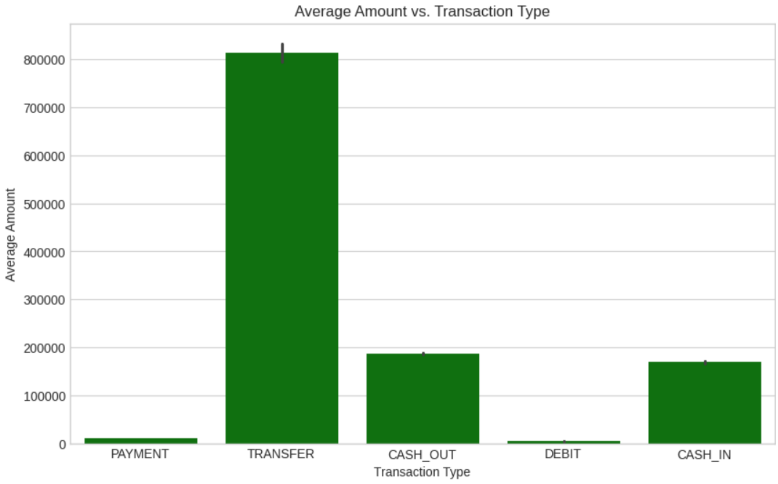


Figure 3.4: Bar Plot of Amount vs. Transaction Type

Figure 3.4 shows the average transaction amount for each transaction type using a bar plot. This plot visualizes how transaction amounts vary across different transaction types, providing insights into the typical value associated with each category. Each bar represents a transaction type, and its height reflects the average amount for that type.

The purpose of this bar plot is to compare the average transaction values for different transaction types. It helps identify if certain transaction types involve larger or smaller amounts, highlighting spending patterns or trends within specific categories. The average transaction amount for each type is computed as the ratio of the sum of transaction amounts of that type to the number of transactions of that type. This is mathematically represented as:

$$\text{Average Amount for Type } T = \frac{\sum_{i=1}^{n} (\text{Amount}_i \times I(\text{Type}_i = T))}{\sum_{i=1}^{n} I(\text{Type}_i = T)}$$

This approach is inspired by methods discussed in prior research [8]. The bar plot provides a clear overview of how different transaction types contribute to overall spending, helping stakeholders focus on areas with higher financial activity for further analysis or decision-making.

3.5 Cumulative Transaction Amounts

Figure 3.5 illustrates the cumulative amounts of all transactions as well as those identified as fraudulent over a period of time.

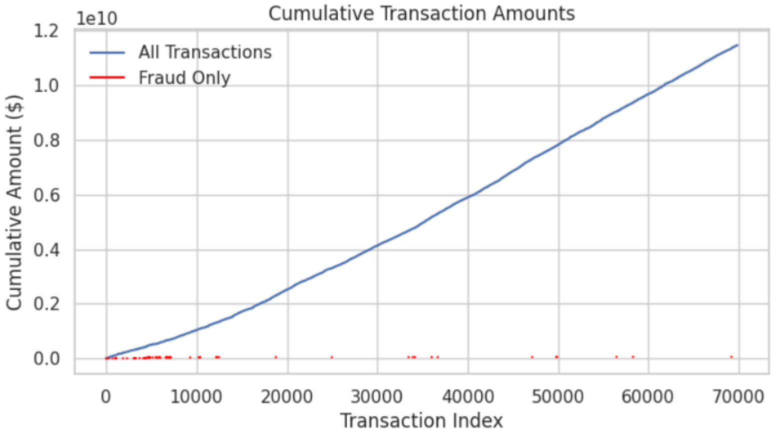


Figure 3.5: Cumulative Transaction Amounts

Figure 3.5 is a line chart charting the cumulative total of all transaction values against the cumulative total of fraud transaction values with respect to the transaction index. The x-axis illustrates the chronological order of transactions, while the y-axis indicates the cumulative monetary value accumulated over time. The graph contains two distinct lines: one for all transactions and one for fraud transactions, thus allowing for a comparison of their respective growth trend over time.

This visualization seeks to explore the buildup of the total value of fraudulent transactions in relation to the overall count of transactions. This study is necessary to identify the financial impact of fraud and identify the time periods when the impact of fraud on the total transaction value is largely highlighted. A rapidly rising trend in the fraud line compared to the line of overall transactions, for example, may indicate a rise in high-value fraudulent transactions.

Cumulative sums are computed by applying the cumulative summation function to the ‘amount‘ column for all transactions, and conditionally for fraudulent ones. This provides a stable and legible view of the dynamics of transactions.

Lastly, this graph points to the scale and speed at which fraudulent activities impact financial systems over time, providing useful evidence for the formulation of strategies for early detection and fraud mitigation.

Analysis: The cumulative amount for all transactions at any point i is calculated as:

$$\text{Cumulative Amount (All)}_i = \sum_{j=1}^{i} \text{amount}_j$$

The cumulative amount for fraudulent transactions at any point i is calculated as:

$$\text{Cumulative Amount (Fraud)}_i = \sum_{j=1}^{i} (\text{amount}_j \times I(\text{isFraud}_j = 1))$$

Where $I(\text{isFraud}_j = 1)$ is an indicator function that equals 1 if the j-th transaction is fraudulent and 0 otherwise. [9]

4 Correlation Analysis

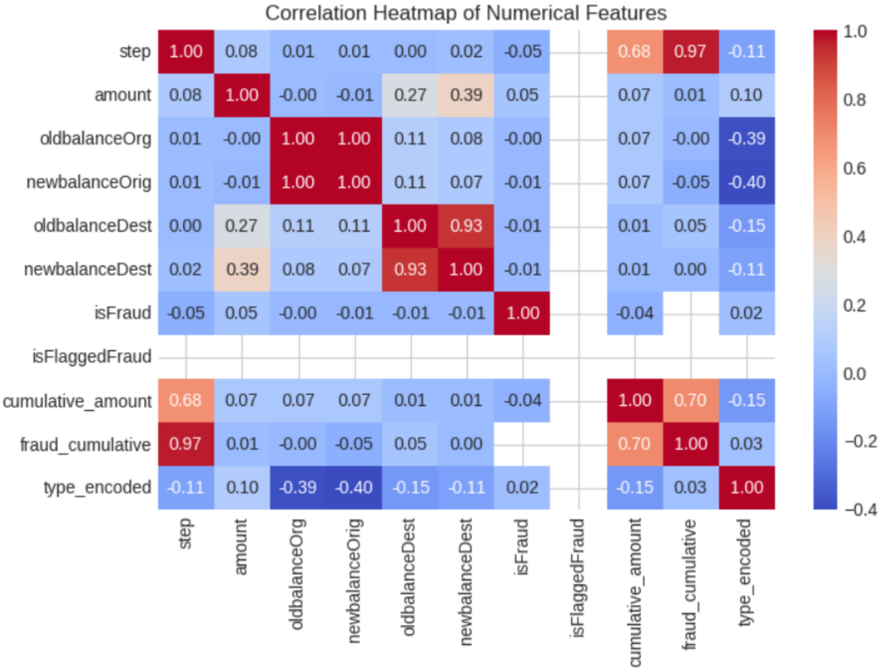
“In statistical research, correlation analysis not only quantifies the degree of linear association between two variables but also provides insight into potential predictive relationships.” The approach is important in pattern and interdependence discovery in data sets, particularly in fraud detection, where understanding the interrelationship between various features and fraud is of crucial importance. Through the analysis of correlations, one can uncover underlying relationships between features like transaction values, time factors, and fraud status, which can be beneficial towards the construction of predictive models.

In the fraud detection domain, correlation analysis plays a crucial role in determining the correlation between certain characteristics and fraud. For example, transaction value, time of day, or transaction type might be observed to have a high correlation with fraud status. Identification of such relationships can assist in designing special fraud detection systems. If a certain characteristic, in general, shows a correlation with fraud, it might be a significant indicator for the identification of future fraudulent transactions.

“A heatmap is a visual tool frequently used to illustrate correlations among multiple dataset variables.” This provides a quick and easy understanding of the relationships between features. In the case of this analysis, the heatmap is a graphical representation of the correlation coefficients between feature pairs, with darker shades indicating more substantial correlations. This technique not only indicates the possibility of relationships but also identifies redundant or highly correlated features, which may be eliminated during feature selection to make the model more efficient.

4.1 Correlation Analysis of Numerical Features

Figure 4.1 presents a heatmap illustrating the correlations among the dataset's numerical features.



	step	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud	cumulative_amount	fraud_cumulative	type_encoded
step	1.00	0.08	0.01	0.01	0.00	0.02	-0.05		0.68	0.97	-0.11
amount	0.08	1.00	-0.00	-0.01	0.27	0.39	0.05		0.07	0.01	0.10
oldbalanceOrg	0.01	-0.00	1.00	1.00	0.11	0.08	-0.00		0.07	-0.00	-0.39
newbalanceOrig	0.01	-0.01	1.00	1.00	0.11	0.07	-0.01		0.07	-0.05	-0.40
oldbalanceDest	0.00	0.27	0.11	0.11	1.00	0.93	-0.01		0.01	0.05	-0.15
newbalanceDest	0.02	0.39	0.08	0.07	0.93	1.00	-0.01		0.01	0.00	-0.11
isFraud	-0.05	0.05	-0.00	-0.01	-0.01	-0.01	1.00		-0.04		0.02
isFlaggedFraud											
cumulative_amount	0.68	0.07	0.07	0.07	0.01	0.01	-0.04		1.00	0.70	-0.15
fraud_cumulative	0.97	0.01	-0.00	-0.05	0.05	0.00			0.70	1.00	0.03
type_encoded	-0.11	0.10	-0.39	-0.40	-0.15	-0.11	0.02		-0.15	0.03	1.00

Figure 4.1: Correlation Heatmap of Numerical Features

This is a **Heatmap** that visualizes the correlation coefficients between numerical variables. The correlation coefficient r_{XY} measures the strength and direction of the linear relationship between two variables X and Y.

Purpose: To identify which numerical features are strongly correlated with each other and with the target variable . This can help in feature selection and understanding the relationships between variables.

Analysis: The Pearson correlation coefficient r is calculated as:

$$r_{XY} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$$

Where:

- x_i and y_i are the individual data points,
- $\bar{x}$ and $\bar{y}$ are the sample means of X and Y,

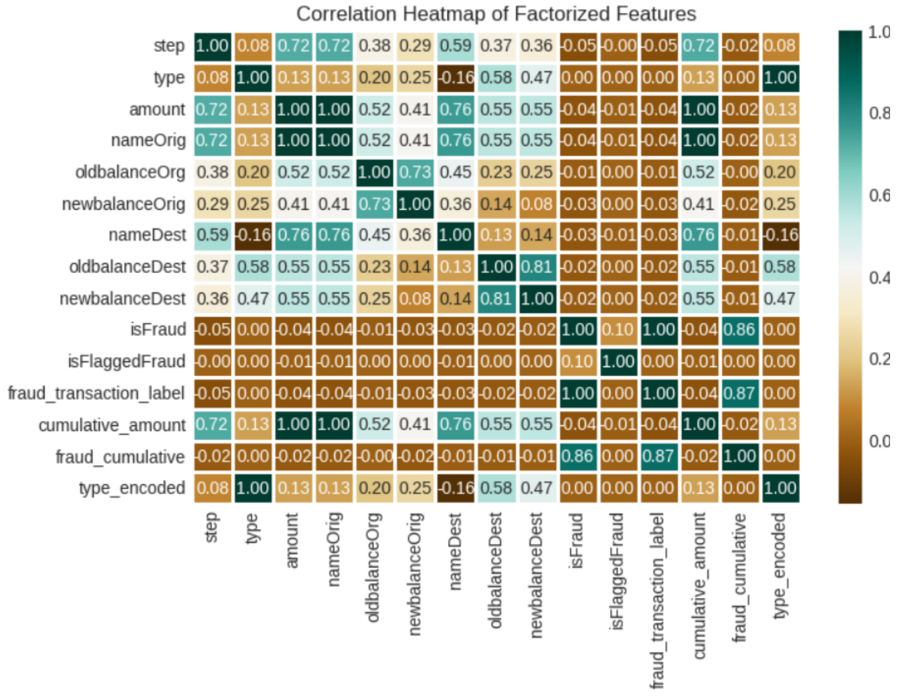
- N is the number of data points.

This formula measures the linear relationship between two variables, ranging from -1 (perfect negative correlation) to $+1$ (perfect positive correlation) [10].

Output: The heatmap shows the correlation coefficients, typically ranging from -1 to 1. A value of 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no linear correlation. [11]

4.2 Correlation Analysis of Factorized Features

Figure 4.2 shows the correlation heatmap for all features (including categorical), where categorical features have been factorized.


Correlation Heatmap of Factorized Features

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud	fraud_transaction_label	cumulative_amount	fraud_cumulative	type_encoded
step	1.00	0.08	0.72	0.72	0.38	0.29	0.59	0.37	0.36	-0.05	-0.00	-0.05	0.72	-0.02	0.08
type	0.08	1.00	0.13	0.13	0.20	0.25	-0.16	0.58	0.47	0.00	0.00	0.00	0.13	0.00	1.00
amount	0.72	0.13	1.00	1.00	0.52	0.41	0.76	0.55	0.55	-0.04	-0.01	-0.04	1.00	-0.02	0.13
nameOrig	0.72	0.13	1.00	1.00	0.52	0.41	0.76	0.55	0.55	-0.04	-0.01	-0.04	1.00	-0.02	0.13
oldbalanceOrg	0.38	0.20	0.52	0.52	1.00	0.73	0.45	0.23	0.25	-0.01	0.00	-0.01	0.52	-0.00	0.20
newbalanceOrig	0.29	0.25	0.41	0.41	0.73	1.00	0.36	0.14	0.08	-0.03	0.00	-0.03	0.41	-0.02	0.25
nameDest	0.59	-0.16	0.76	0.76	0.45	0.36	1.00	0.13	0.14	-0.03	-0.01	-0.03	0.76	-0.01	-0.16
oldbalanceDest	0.37	0.58	0.55	0.55	0.23	0.14	0.13	1.00	0.81	-0.02	0.00	-0.02	0.55	-0.01	0.58
newbalanceDest	0.36	0.47	0.55	0.55	0.25	0.08	0.14	0.81	1.00	-0.02	0.00	-0.02	0.55	-0.01	0.47
isFraud	-0.05	0.00	-0.04	-0.04	-0.01	-0.03	-0.03	-0.02	-0.02	1.00	0.10	1.00	-0.04	0.86	0.00
isFlaggedFraud	-0.00	0.00	-0.01	-0.01	0.00	0.00	-0.01	0.00	0.00	0.10	1.00	0.00	-0.01	0.00	0.00
fraud_transaction_label	-0.05	0.00	-0.04	-0.04	-0.01	-0.03	-0.03	-0.02	-0.02	1.00	0.00	1.00	-0.04	0.87	0.00
cumulative_amount	0.72	0.13	1.00	1.00	0.52	0.41	0.76	0.55	0.55	-0.04	-0.01	-0.04	1.00	-0.02	0.13
fraud_cumulative	-0.02	0.00	-0.02	-0.02	-0.00	-0.02	-0.01	-0.01	-0.01	0.86	0.00	0.87	-0.02	1.00	0.00
type_encoded	0.08	1.00	0.13	0.13	0.20	0.25	-0.16	0.58	0.47	0.00	0.00	0.00	0.13	0.00	1.00

Figure 4.2: Correlation Heatmap of Factorized Features

This is another **Heatmap** that visualizes the correlation between all features in the dataset after converting categorical features into numerical representations using factorization.

Purpose: To get a broader view of feature relationships, including how categorical variables might correlate with numerical ones. Factorization helps in including categorical variables in the correlation analysis.

Analysis: Factorization assigns a numerical code to each distinct category within a categorical variable. Subsequently, the Pearson correlation coefficient, as outlined in the previous section, is used to compute the correlation based on these numerical codes.

Output: The heatmap shows the correlation coefficients between all pairs of variables, providing insights into both numerical-numerical and numerical-categorical relationships.

5 Multivariate Analysis

“Multivariate analysis examines several variables at once to detect intricate relationships and dependencies within a dataset.” In the realm of fraud detection, this technique is particularly valuable, as fraudulent behavior often arises from intricate interactions among several features rather than from a single variable in isolation. Unlike univariate or bivariate approaches, which may overlook such interactions, multivariate analysis captures the combined influence of multiple factors on the likelihood of fraud.

This analytical method enables a broader and more nuanced understanding of how different features work together to signal potentially fraudulent activity. For example, a high transaction amount may not independently suggest fraud, but when paired with specific transaction types, unusual time steps, or suspicious account balance behaviors, it may indicate a greater risk. Identifying such compound relationships enhances the ability to detect fraud that is otherwise difficult to isolate.

In this chapter, multivariate analysis is applied to the transaction dataset to explore how combinations of features—such as transaction type, amount, origin and destination balances, and temporal attributes—correlate with fraudulent behavior. The objective is to identify recurring attribute patterns associated with fraud, thereby informing the development of more advanced fraud detection systems that are capable of identifying subtle and sophisticated fraudulent transactions.

5.1 Origin vs. Destination Balance (Fraud Cases)

Figure 5.1 illustrates the correlation between the balance of the origin account prior to the transaction (oldbalanceOrg) and the balance of the destination account before the transaction (oldbalanceDest) in cases of fraudulent transactions.

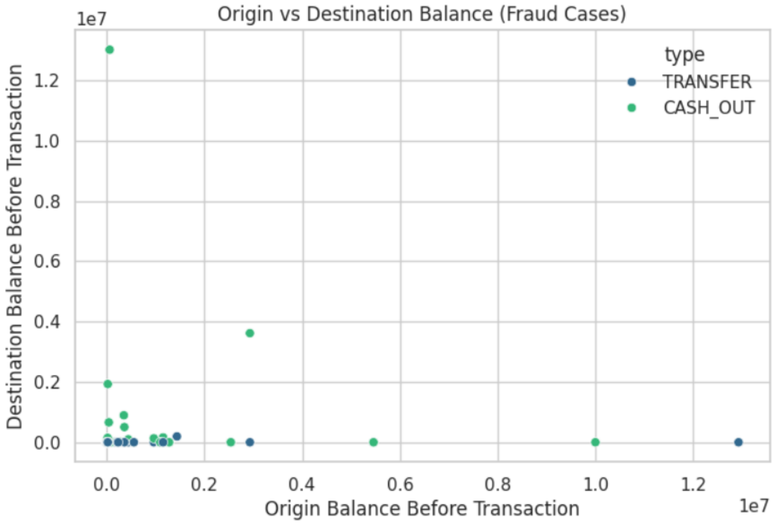


Figure 5.1: Origin vs. Destination Balance (Fraud Cases)

The graph is a scatter plot portraying the correlation between the origin account pre-transaction balance and the destination account pre-transaction balance for particular cases that are tagged as fraudulent. Each point that is plotted is an individual fraudulent transaction, with the x-coordinate being the pre-transaction origin account balance and the y-coordinate being the corresponding pre-transaction destination account balance. The color of each point differentiates the type of transaction, thereby enabling one to distinguish transaction behaviors for different types.

The main purpose of this graphical illustration is to investigate the relationship between the initial financial status of both the source and target accounts and their relationship with fraudulent transactions. This investigation allows for the identification of patterns or groupings that can signal common fraudulent behaviors in fraudulent transfers, such as the possible targeting or involvement of high-balance accounts.

By the selection of a smaller subset of suspect transactions, the graphical representation eliminates extraneous detail while still providing meaningful insights.

Observations of linear distributions, highly populated clusters, or prominent gaps might reveal valuable characteristics. For example, a data point cluster with high balances at the origin and destination might suggest a tactic used to hide large unauthorized transfers.

This analysis is very useful for fraud detection policies because it can identify balance behavior that is linked with fraud and then be applied to establish thresholds or guidelines for flagging suspicious behavior. [12]

6 Conclusion and Future Scope of Work

Conclusion

This project performed a comprehensive and intensive examination of a set of financial transactions with particular emphasis on identification and detection of fraudulent transactions using exploratory data analysis methods along with machine learning methods. A collection of visualizations was utilized in a strategic manner to gain a better understanding of the distribution of different types of transactions, the sizes of these transactions, and the patterns that emerge over time intervals. Some key findings that were derived from this analysis were that some types of transaction types like `CASHOUT` and `TRANSFER` exhibited a higher probability of fraudulent activity, frequent sudden and unexplained shifts in account balances that trigger warnings

The data set was thoroughly analyzed using different statistical metrics, and these revealed that there were very strong and significant relationships between some numerical features. This thorough analysis was very crucial during the feature selection process. There was a label encoding as well as factorization method used in order to convert categorical data into numerical data in order to ease visualization as well as model training processes. Towards this, different visualization methods were used, including histograms, pie charts, scatter plots, and heatmaps. These visualization methods were very critical in bringing out the normal transactional patterns as well as the suspicious ones in an easy way. Through this, it was very easy to achieve a clearer as well as a better understanding of fraud patterns over time.

Future Areas for Further Efforts

While this particular study has indeed brought forth a treasure of informative data through the effective application of data visualization techniques, It should be acknowledged that there is significant potential to elevate the work already done to even higher levels. Future enhancements to this study can definitely include the implementation of more sophisticated machine learning and deep learning techniques, which can include sophisticated methods such as Random Forests, XGBoost, or other types of Neural Networks, all of which can contribute towards the achievement of improved classification of fraudulent transactions. In addition, time-series models can also be thoroughly studied to identify anomalies that may occur in real-time streams of transactions better, thereby enhancing the effectiveness of the analysis as a whole.

In addition, the inclusion of other contextual information—some of which might be user demographics or location—can potentially greatly improve the accuracy of the predictions. One example of a real-world application of this work might be the development of a very advanced automated fraud detection system. Such a system would be programmed to trigger alerts when particular transactional activity is detected, which would then help financial institutions minimize their losses and improve their overall security protocols.

References

[1] Jainilcoder. Online payment fraud detection dataset. https://www.kaggle.com/datasets/jainilcoder/online-payment-fraud-detection, 2021. Accessed: 2025-05-04.

[2] David Freedman, Robert Pisani, and Roger Purves. *Statistics.* W. W. Norton & Company, 4th edition, 2007.

[3] Sara Makki, Zainab Assaghir, Yehia Taher, Rafiqul Haque, Mohand-Said Hacid, and Hassan Zeineddine. An experimental study with imbalanced classification approaches for credit card fraud detection. *Ieee Access*, 7:93010–93022, 2019.

[4] J.F. Kenney and E.S. Keeping. *Mathematics of Statistics, Part 1.* Van Nostrand, Princeton, NJ, 3rd edition, 1962.

[5] Barbara Carminati, Elisa Caron, Elena Ferrari, and Andrea Perego. Fraudbuster: a fraud detection system for e-commerce transactions. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 366–370. ACM, 2018.

[6] David S. Moore, George P. McCabe, and Bruce A. Craig. *Introduction to the Practice of Statistics.* W. H. Freeman and Company, 7th edition, 2012.

[7] Timothy R Derrick, Barry T Bates, and Janet S Dufek. Evaluation of time-series data sets using the pearson product-moment correlation coefficient. *Medicine and Science in Sports and Exercise*, 26(7):919–928, 1994.

[8] Checkout.com. What is average transaction value (atv) and why is it important?, 2023. Accessed: 2025-05-04.

[9] Andrey Pepelyshev and Aleksey S Polunchenko. Real-time financial surveillance via quickest change-point detection methods. *arXiv preprint arXiv:1509.01570*, 2015.

[10] Darren George and Paul Mallery. *IBM SPSS statistics 29 step by step: A simple guide and reference*. Routledge, 2024.

[11] Israel Cohen, Yiteng Huang, Jingdong Chen, Jacob Benesty, Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson correlation coefficient. *Noise reduction in speech processing*, pages 1–4, 2009.

[12] Roger S Debreceny and Glen L Gray. Data mining journal entries for fraud detection: An exploratory study. *International Journal of Accounting Information Systems*, 11(3):157–181, 2010.

A Appendix : Code Samples

A.1 [Python code for transaction analysis]

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder
df = pd.read_csv('/content/new_data.csv')
df

print(plt.style.available)
plt.style.use('seaborn-v0_8-whitegrid')
sns.set_palette(['green', 'red'])

#[caption="Code for Transaction Type Countplot"]
plt.figure(figsize=(8, 5))
sns.countplot(x='type', data=df)
plt.title("Countplot of Transaction Type")
plt.xlabel("Transaction Type")
plt.ylabel("Count")
plt.show()

#[caption="Code for Fraud Countplot"]
sns.countplot(x='isFraud', data=df)
plt.title("Class Distribution (isFraud)")
plt.xlabel("Transaction Type (0=Legitimate, 1=Fraud)")
plt.ylabel("Count")
plt.show()

```

```
#[caption="Code for Fraud Percentage Pie Chart"]
def Fraud(x):
    if x == 1:
        return "Fraudulent"
    else:
        return "not Fraudulent"

df["fraud_transaction_label"] = df["isFraud"].apply(Fraud)

plt.figure(figsize=(10, 5))
plt.title("Fraudulent Transactions")
fraud_counts = df.fraud_transaction_label.value_counts()
plt.pie(fraud_counts, labels=fraud_counts.index, autopct='%1.1f%%')
plt.show()

#[caption="Code for Transaction Amount Histogram"]
sns.histplot(df['amount'], bins=50, kde=True)
plt.title('Distribution of Transaction Amount')
plt.xlabel('Amount')
plt.ylabel('Frequency')
plt.show()

#[caption="Code for Time Step Histogram"]
sns.histplot(df['step'], bins=50, kde=True)
plt.title('Distribution of Time Step')
plt.xlabel('Time Step')
plt.ylabel('Frequency')
plt.show()

#[caption="Python Code for Bar Plot of Hourly Fraud Percentage"]
df['hour'] = df['step'] % 24
hourly_fraud = df.groupby('hour')['isFraud'].mean() * 100
plt.figure(figsize=(10, 6))
hourly_fraud.plot(kind='bar', color='orange')
plt.title("Hourly Fraud Percentage")
plt.ylabel("Fraud Percentage (%)")
plt.xlabel("Hour of Day")
plt.grid(axis='y')
plt.show()

#[caption="Python Code for Count Plot of Transaction Type vs. Fraud Status"]
plt.figure(figsize=(10, 6))
sns.countplot(x=’type’, hue=’isFraud’, data=df)
plt.title(’Transaction Type vs. Fraud Status’)
plt.xlabel(’Transaction Type’)
plt.ylabel(’Count’)
plt.legend(title=’Is Fraud’, labels=[’No’, ’Yes’])
plt.show()

#[caption="Python Code for Scatter Plot of Transaction Amount vs. Fraud Status"]
plt.figure(figsize=(10, 6))
plt.scatter(df[’amount’], df[’isFraud’], alpha=0.01) # Using alpha for transparency
plt.title(’Transaction Amount vs. Fraud Status’)
plt.xlabel(’Transaction Amount’)
plt.ylabel(’Is Fraud (0: No, 1: Yes)’)
plt.yscale(’linear’)
plt.show()

#[caption="Python Code for Scatter Plot of Time Step vs. Fraud Status"]
plt.scatter(df[’step’], df[’isFraud’], alpha=0.01)
plt.title(’Step vs. Fraud Status’)
plt.xlabel(’Step (Time Step)’)
plt.ylabel(’Is Fraud (0: No, 1: Yes)’)
plt.show()

#[caption="Python Code for Bar Plot of Amount vs. Transaction Type"]
sns.barplot(x=’type’, y=’amount’, data=df)
plt.title(’Average Amount vs. Transaction Type’)
plt.xlabel(’Transaction Type’)
plt.ylabel(’Average Amount’)
plt.show()

```

```
#[caption="Python Code for Line Plot of Cumulative Transaction Amounts"]
df[’cumulative_amount’] = df[’amount’].cumsum()
df[’fraud_cumulative’] = df[’amount’].where(df[’isFraud’] == 1).cumsum()
plt.figure(figsize=(12, 6))
plt.plot(df[’cumulative_amount’], label=’All Transactions’)
plt.plot(df[’fraud_cumulative’], label=’Fraud Only’, color=’red’)
plt.title("Cumulative Transaction Amounts")
plt.ylabel("Cumulative Amount ($)")
plt.xlabel("Transaction Index")
plt.legend()
plt.show()

#[caption="Python Code for Heatmap of Numerical Features"]
from sklearn.preprocessing import LabelEncoder
label_encoder = LabelEncoder()
df[’type_encoded’] = label_encoder.fit_transform(df[’type’])
corel = df.select_dtypes(include=np.number).corr()
# Heatmap of Correlation (Numerical Features)
plt.figure(figsize=(10, 8))
sns.heatmap(corel, annot=True, cmap=’coolwarm’, fmt=".2f")
plt.title("Correlation Heatmap of Numerical Features")
plt.show()

#[caption="Python Code for Heatmap of Factorized Features"]
label_encoder = LabelEncoder()
df[’type_encoded’] = label_encoder.fit_transform(df[’type’])
# Heatmap of Correlation (Factorized Features)
plt.figure(figsize=(12, 10))
sns.heatmap(df.apply(lambda x: pd.factorize(x)[0]).corr(),
cmap=’BrBG’,fmt=’.2f’,linewidths=2,annot=True)
plt.title("Correlation Heatmap of Factorized Features")
plt.show()

#[caption="Python Code for Scatterplot of Origin vs. Destination Balance"]
sns.set(style="whitegrid")
```

```
plt.figure(figsize=(10, 6))
fraud_cases = df[df[’isFraud’] == 1]
sample_size = 1000 if len(fraud_cases) > 1000 else len(
    fraud_cases)
sample_df = fraud_cases.sample(n=sample_size, random_state=42)
sns.scatterplot(
    x=’oldbalanceOrg’,
    y=’oldbalanceDest’,
    data=sample_df,
    hue=’type’,
    palette=’viridis’
)
plt.title("Origin vs Destination Balance (Fraud Cases)")
plt.xlabel("Origin Balance Before Transaction")
plt.ylabel("Destination Balance Before Transaction")
plt.show()
```