Toronto Metropolitan University

Master of Science in Data Science and Analytics
MRP - Literature Review and Exploratory Data Analysis

Optimizing Supervised Machine Learning for Enhanced Transaction Fraud Detection Accuracy

Submitted to:

MRP Supervisor – Dr. Shengkun Xie
MRP Second Reader – Dr. Ceni Babaoglu

Submitted by:

Nguyen Duy Anh Luong (Student ID – 500968520)

June 16, 2025

Abstract

Transaction fraud remains a serious concern for banks, businesses, and consumers, especially as online transactions continue to rise. In this study, several papers (listed in the references) were first reviewed to explore supervised machine learning techniques for detecting fraud. The review focuses on practical strategies for model selection, addressing data imbalance, and improving predictive performance. It also discusses some common challenges such as the lack of fraud cases, anonymous features, and the ongoing trade-off between limiting false positives and increasing fraud detection in real-time systems. Many of the reviewed studies highlight the effectiveness of ensemble models, neural networks, and data sampling techniques in improving overall detection accuracy.

In the second part, an exploratory data analysis (EDA) was conducted on the IEEE-CIS Fraud Detection dataset. Using visual and statistical tools, the EDA revealed patterns in missing data, feature relationships, and the timing of fraudulent transactions. Additional findings include the distribution of transaction amounts by fraud class, the fraud rate across different product categories, and how certain high-value transactions are more closely linked to fraud. One major finding was the significant class imbalance, as fraudulent cases make up only 3.5% of the data, highlighting the importance of using appropriate techniques in future modeling stages.

Literature Review

This literature review focuses on supervised machine learning approaches for transaction fraud detection, especially in real-world scenarios with imbalanced and anonymized data. As online transactions become part of daily life, whether for shopping, paying bills, or transferring money, fraud has become a growing concern for businesses and consumers. Traditional rule-based systems are no longer effective against evolving fraud tactics, leading to increased interest in machine learning methods that learn from historical data.

In this Major Research Project (MRP), the IEEE-CIS Fraud Detection dataset is used, which was released by Vesta through a Kaggle competition. It includes masked transaction and identity features, adding complexity to preprocessing and modeling. Insights are drawn from fifteen peer-reviewed journal and conference papers, examining techniques for handling class imbalance, feature engineering, model selection, and the use of identity-based variables to improve fraud detection performance.

1. Supervised Learning Models in Fraud Detection

The exploration began with how supervised learning models are applied in fraud detection, as they form the foundation of most modern systems. These models are trained on historical data, where each transaction is labeled as either fraudulent or legitimate. By learning from these patterns, predictions can be made on new, unseen transactions.

One of the most notable approaches reviewed is the Distributed Deep Neural Network (DDNN) model proposed by Lei et al. (2023), which focuses on both accuracy and user privacy [1]. Instead of centralizing all transaction data, this method allows financial institutions to train local models on companies' own data while only sharing model parameters with a central server. This approach not only protects user privacy but also

reduces data handling costs and improves training efficiency through parallel computing. Experimental results showed that the DDNN model outperformed centralized models in terms of accuracy, precision, recall, and F1-score.

In addition, more interpretable models such as decision trees and random forests were explored. The 2023 study Credit Card Fraud Detection using Decision Tree and Random Forest by Shah and Sharma examined these two algorithms using a simulated credit card transaction dataset [2]. The results showed that while decision trees are easy to understand, they tend to overfit the training data. Random forests, which combine multiple decision trees, performed better overall, particularly after hyperparameter tuning. However, both models continued to struggle with class imbalance, which limited their effectiveness in accurately detecting fraudulent transactions.

To explore this challenge further, a large-scale study by Alfaiz and Fati (2022) was examined, in which 66 combinations of nine machine learning algorithms and nineteen resampling techniques were evaluated [3]. This approach was particularly valuable, as a real-world dataset was used and each combination was systematically compared. The best results were achieved by combining CatBoost, an advanced gradient boosting model, with the AllKNN undersampling method. This combination produced an F1-score of 87.40%, a recall of 95.91%, and an AUC of 97.94%, outperforming many traditional methods. The findings emphasized the importance of selecting both an effective model and an appropriate data balancing strategy when working with imbalanced fraud datasets.

Overall, these studies provided a clearer understanding of the strengths and limitations of different supervised learning models. Deep learning approaches like DDNN have demonstrated strong performance, particularly in scenarios where privacy and scalability are critical. In contrast, interpretable models such as random forests and CatBoost have also proven to be highly effective when supported by thoughtful preprocessing and well-selected data balancing techniques.

2. Imbalanced Data: Core Challenge and Solutions

As supervised learning models were explored more deeply, class imbalance quickly stood out as one of the biggest challenges in fraud detection. In most real-world datasets, fraudulent transactions make up only a small fraction of the total, which often causes models to favor legitimate transactions and miss fraud cases. To build a system that performs well, this imbalance must be addressed so that rare fraud cases can be detected without generating too many false positives.

In the paper titled Hybrid Undersampling and Oversampling for Handling Imbalanced Credit Card Data, Alamri and Ykhlef (2022) introduced a hybrid resampling technique called BCB-SMOTE, which combines Tomek links for noise reduction, BIRCH clustering, and Borderline SMOTE to generate synthetic samples near the decision boundary [4]. When tested with a random forest classifier, the proposed method achieved an F1-score of 85.2% and successfully addressed common oversampling issues such as data overlap and overfitting. Similarly, in Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost, Ileberi et al. (2021) demonstrated that combining SMOTE with AdaBoost has a positive impact on the performance across several machine learning models [5].

Building on these ideas, the study Credit Card Fraud Detection in Imbalanced Datasets: A Comparative Analysis of Machine Learning Techniques by Lahbiss and Chtouki (2024) evaluated traditional, ensemble, and deep learning models using SMOTE and SMOTE-ENN [6]. It was found that Random Forest combined with SMOTE-ENN achieved an AUC-ROC of 0.85, while LSTM with SMOTE-ENN reached an even higher AUC-ROC of 0.90. These results indicate that the combination of resampling strategies with strong classifiers can significantly improve fraud detection performance.

Together, the reviewed studies demonstrate that no one-size-fits-all solution exists. Instead, effective fraud detection in imbalanced datasets is often achieved by combining resampling techniques with models such as Random Forest or LSTM to enhance predictive accuracy.

3. Feature Engineering and Identity-Based Enhancements

After the impact of class imbalance was explored, attention was turned to feature engineering. Even the most advanced algorithms cannot perform well without meaningful input features. In fraud detection, basic variables such as transaction amount or card type provide limited predictive value on their own. Significant improvements are typically achieved by transforming features and incorporating identity-related data that more accurately reflects user behavior.

One study that demonstrated the value of feature engineering was conducted by Lei et al. (2020), where XGBoost was applied to the IEEE-CIS dataset [7]. Emphasis was placed on careful data cleaning, handling of missing values, consistent label encoding, and feature elimination. By combining transaction data with identity fields such as device type and email domain, model performance was significantly improved, achieving a ROC-AUC of 0.942 and an accuracy of 97.6%. Another paper by Lucas et al. (2019) introduced a feature engineering method based on Hidden Markov Models (HMMs) [8]. Instead of using standard transaction aggregates, sequences were modeled using eight combinations derived from three binary perspectives: whether the sequence was linked to a cardholder or a terminal, whether it was based on transaction amounts or time elapsed, and whether the history was genuine or fraudulent. When the multi-perspective HMM-based features were incorporated into a Random Forest classifier, a 15.1% boost in Precision-Recall AUC was observed.

In addition, Bahnsen et al. (2016) proposed a strategy combining transaction aggregation with periodic behavior modeling [9]. Features were created based on customer spending patterns, and the von Mises distribution was used to model typical transaction times. These features led to a 13% increase in savings, reflecting improved prevention or recovery of losses due to fraud.

Overall, these studies show that advanced feature engineering, such as the use of identity-based data and behavioral patterns, plays a crucial role in improving model accuracy, adaptability, and robustness in real-world fraud detection.

4. Ensemble Learning and Hybrid Models

After the performance of individual models and the impact of feature engineering were examined, attention was directed toward ensemble learning to evaluate whether combining models could lead to improved results. Ensemble methods are designed to combine multiple classifiers to enhance accuracy, reduce bias, and better capture the complexity of fraud patterns, which often vary across users and evolve over time.

A study conducted by Carcillo et al. (2021) introduced a hybrid approach that integrated unsupervised and supervised learning techniques [10]. In this approach, outlier scores were first generated using unsupervised models and then passed into a supervised classifier to make the final decision. Fraud was evaluated at multiple levels, such as globally across the dataset, locally for individual cardholders, and within clusters of similar customers, enabling the model to detect a broader range of fraud behaviors more effectively. In a related study, a dynamic ensemble selection method was proposed by Achakzai and Peng (2023), where the most effective classifiers were selected for each transaction based on local competence [11]. This dynamic approach consistently outperformed static ensemble classifiers in both accuracy and overall performance.

Further supporting the value of ensemble approaches, Jahnavi et al. (2024) proposed a hybrid model that combines decision trees with logistic regression for fraud detection [12]. An accuracy of 98.1% was achieved, with strong performance observed in both sensitivity and flexibility. By combining logistic regression with the decision-making structure of decision trees, a model was developed that can adapt to evolving fraud tactics. In a related study, several ensemble ML models, including Decision Tree, Random Forest, XGBoost, CatBoost, and Gradient Boosting, were compared by Chaurasia et al. (2024) [13]. Both balanced and imbalanced versions of the European cardholder dataset were used, and XGBoost was found to deliver the best performance in terms of F1-score and recall, making it particularly effective at identifying rare fraud cases.

Together, these studies highlight that selecting an appropriate ensemble approach and combining it with effective data balancing strategies can significantly improve the accuracy and reliability of fraud detection systems.

5. Comparative Reviews

In addition to individual models and techniques, broader research comparing multiple machine learning approaches for fraud detection has also been examined. These comparative reviews provide a clearer understanding of which methods tend to perform well across different scenarios and offer practical insights into model selection.

A comparative review conducted by Patel et al. (2024) evaluated several supervised machine learning models for credit card fraud detection, including artificial neural networks (ANN), logistic regression (LR), Naive Bayes (NB), and K-nearest neighbors (KNN) [14]. The results showed that ANN achieved the highest accuracy at 98.9%, followed by logistic regression at 98.6%, Naive Bayes at 98.4%, and KNN at 96.6%. Although ANN delivered the top performance, the study also emphasized that models such as logistic regression and Naive Bayes demonstrated strong performance across multiple evaluation metrics.

In a related study, deep learning methods were compared with traditional machine learning techniques, such as support vector machines, random forests, and logistic regression, by Jyoti et al. (2024), using three datasets [15]. A deep neural network (DNN), trained with the Adam optimizer, achieved an accuracy of 99.4% on the European credit card dataset. The advantages of using Adam were emphasized, including computational efficiency, low memory usage, and suitability for large datasets. Fraudulent transactions were accurately classified, and error rates were kept low.

Together, these studies indicate that although deep learning models often achieve the highest accuracy, traditional models such as logistic regression continue to offer strong and dependable performance. This highlights the importance of choosing models based on both effectiveness and practicality in real-world fraud detection.

6. Conclusion: Gaps and Contributions of This Project

In summary, this project builds on existing research in transaction fraud detection by developing an optimized supervised machine learning algorithm designed to perform effectively on a large, complex, and imbalanced real-world dataset. Although many previous studies have explored different models and techniques, few have addressed all the key challenges in combination, such as class imbalance, meaningful feature selection, and the use of identity-related data like device type and email domain. For this Major Research Project, several machine learning models will be compared, including logistic regression, random forest, artificial neural networks, deep neural networks, and XGBoost, to determine the most effective approach. Through the application of feature engineering, resampling techniques, and ensemble strategies, the system is intended to improve fraud detection accuracy while minimizing false positives. The final objective is to develop a reliable, scalable, and interpretable model that enhances fraud prevention in real-world financial applications.

Data Description - Exploratory Data Analysis (EDA)

In this section, patterns in fraudulent behavior were explored using a combination of visualizations and statistical analysis. The goal was to uncover meaningful insights that could inform model development and improve fraud detection accuracy. Each analysis was aligned with the previously defined research questions, with a focus on identifying trends, anomalies, or potential feature transformations that could enhance the model's ability to detect fraud effectively.

Data Source

For this project, the IEEE-CIS Fraud Detection dataset was used, which was made available through a Kaggle competition hosted in collaboration with Vesta, a global fraud prevention company. The dataset represents real-world e-commerce transaction environments and is intended to support the development of machine learning models for fraud detection.
Source: `https://www.kaggle.com/competitions/ieee-fraud-detection/data`

Data Acquisition

The dataset was downloaded directly from Kaggle. It includes separate files for transaction and identity information, which are joined using the `TransactionID` key. The training set contains labeled examples (with the binary `isFraud` target), while the test set includes similar features but does not contain fraud labels.

Data Files

- `train_transaction.csv` and `train_identity.csv`: Training data with features and the isFraud label (1 = fraud, 0 = non-fraud)
- `test_transaction.csv` and `test_identity.csv`: Test data without labels

Transaction Features:

- `TransactionDT`: Relative timestamp
- `TransactionAmt`: Transaction amount in USD
- `ProductCD`: Product category code
- `card1`–`card6`: Card-related attributes (e.g., type, issuer)
- `addr1`, `addr2`: Geographic location codes
- `C1`–`C14`, `D1`–`D15`, `V1`–`V339`: Engineered and anonymized features
- `P_emaildomain`, `R_emaildomain`: Purchaser and recipient email domains
- `M1`–`M9`: Binary match indicators

Identity Features:

- `DeviceType`, `DeviceInfo`: Device metadata
- `id_12`–`id_38`: Browser, OS, network, and identity-related signals

Data Constraints and Preprocessing

Before the analysis was conducted, the raw transaction and identity datasets were cleaned to make them easier to work with. Column names containing hyphens and extra spaces were modified by replacing hyphens with underscores and removing any leading or trailing whitespace to avoid issues in Python. Missing values also required significant attention during preprocessing. For columns containing text or categorical data, missing values were replaced with the word "missing" to clearly indicate that information was originally not provided. For numerical columns, missing values were retained as None, allowing them to be handled more carefully during the analysis or modeling phase. This preprocessing step was performed using a simple Python script that cleaned both `train_transaction.csv` and `train_identity.csv`. The cleaned versions, `cleaned_train_transaction.csv` and `cleaned_train_identity.csv`, were then saved and used for all subsequent steps in the exploratory data analysis (EDA).

Descriptive Statistics

Before visual exploration was performed, basic statistics were reviewed to better understand the structure of the dataset. The transaction data contains over 590,000 records, with only 3.5% labeled as fraudulent. The variable `TransactionAmt` ranges from a few cents to several thousands of dollars, with a median of around $68. Some features contain a high amount of missing values, particularly within the identity data. Categorical variables such as `ProductCD` and `card4` have a small number of unique values, while anonymized numerical features (`V1`–`V339`) have wide distributions. Overall, these initial findings helped guide the direction of the subsequent exploratory data analysis (EDA).

1. Class Imbalance and Baseline Performance

When examining the dataset for the first time, one of the most noticeable patterns is the large imbalance between legitimate and fraudulent transactions. Out of a total of 590,540 records, only 20,663 transactions (3.50%) are labeled as fraudulent, while the remaining 96.5% are legitimate. This shows that fraud is not only rare but also potentially very costly when it occurs.

A simple bar chart (Figure 1) was created to show the distribution of the two classes. As expected, legitimate transactions dominate the chart. Percentages were added above each bar to highlight the imbalance: 96.5% legitimate vs. 3.5% fraudulent.

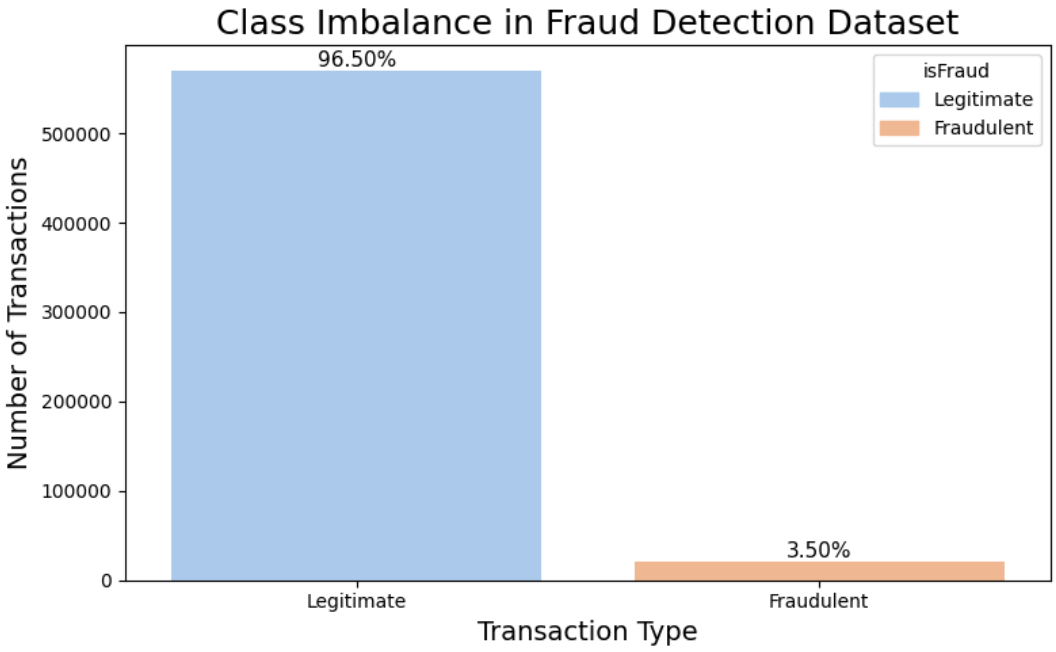

Figure 1: Fraud Distribution by Transaction Type

To demonstrate why this imbalance is a serious issue for machine learning, a basic test was done using a dummy classifier that always predicts the majority class (not fraud). The model achieved 96.5% accuracy, which might seem impressive at first, but it failed to catch any fraud cases. Precision, recall, and F1-score were all 0.000, making the model completely ineffective for detecting fraud.

To put this into a real-world perspective, consider an estimate where each missed fraud costs about $500. Missing 20,663 fraud cases would translate to over $10 million in potential losses. This highlights why accuracy alone is not a reliable performance measure for fraud detection.

Overall, this analysis confirms that fraud detection cannot be treated as a standard classification problem. Because of how rare and high-impact fraud cases are, special techniques such as resampling (e.g., SMOTE), class weighting, or ensemble models like XGBoost are needed to address the imbalance and improve detection performance.

2. Transaction Amount and Product Category Patterns

The distribution of TransactionAmt was first visualized for both fraudulent and legitimate transactions. While the majority of transactions in both classes occurred at lower amounts, fraudulent transactions were more widely distributed, especially in the $100 to $500 range. This pattern was clearly visible in the KDE plot (Figure 2) below.

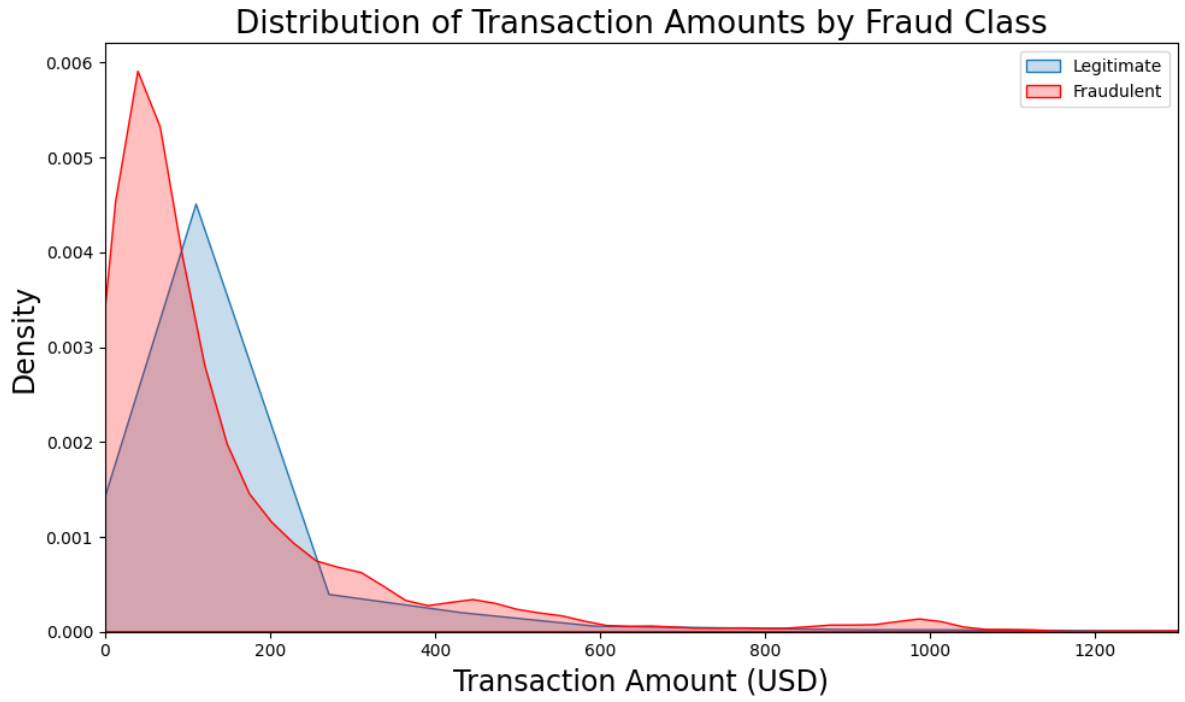

Figure 2: Distribution of Transaction Amounts by Fraud Class

The boxplot (Figure 3) also provided additional insight. Fraudulent transactions showed a greater number of high-value outliers compared to legitimate ones. This supports the idea that certain fraud attempts involve larger purchases, possibly with the intent of maximizing value before detection.

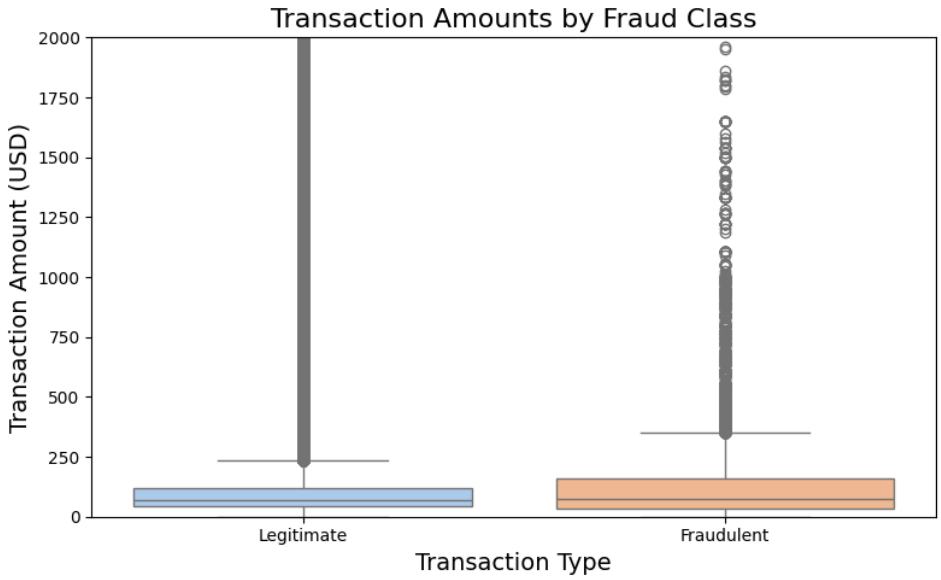

Figure 3: Transaction Amounts by Fraud Class (Boxplot)

Additionally, the variable ProductCD, which represents product or service types, was also examined. Fraud rates varied noticeably across categories:

- Product C had the highest fraud rate at approximately 11.7%.
- Product W had the lowest, around 2.1%.

This suggests that some product categories are more frequently targeted in fraud attempts, offering valuable information for feature selection.

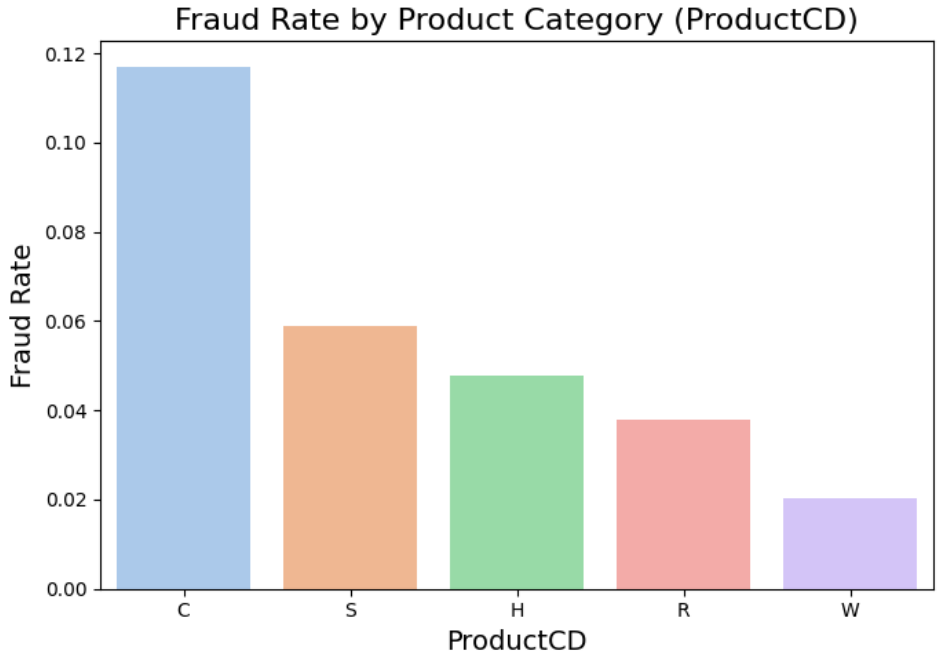

Figure 4: Fraud Rate by Product Category (ProductCD)

To further explore the relationship between transaction amount and product type, a new feature called amt_over_500_risky_product was created, which flags transactions over $500 in the three product categories with the highest fraud rates (C, S, and H). This group showed a slightly higher fraud rate (3.64%) than the rest of the data (3.50%), which suggests the combination does capture a modestly higher-risk segment.

Two key insights emerge from this analysis. First, both transaction amount and product type provide useful fraud signals, particularly when considered in combination. Second, feature engineering can uncover important patterns, but assumptions should always be validated with data, as not every engineered feature yields significant improvement.

Overall, these findings will help inform the selection and refinement of input features for machine learning models. Models such as XGBoost and Deep Neural Networks are especially likely to benefit from well-designed features that expose meaningful fraud-related patterns.

3. Temporal Patterns in Fraudulent Activity

This part of the analysis focused on uncovering time-based patterns in fraudulent transactions using the TransactionDT field. Since this field represents a time delta in seconds, it was transformed into more interpretable units: transaction day, hour of day, and a weekend indicator. These transformations supported the investigation of whether fraud is more likely to occur at specific times, aligning with RQ5: How does feature engineering affect fraud detection?

The fraud rate by hour of day revealed a clear and meaningful pattern. Fraudulent activity was most frequent in the early morning hours, peaking sharply around 7 AM with a fraud rate exceeding 10%, and gradually declining throughout the rest of the day. This suggests that early hours may be targeted by fraudsters, possibly due to lower detection activity during off-peak periods.

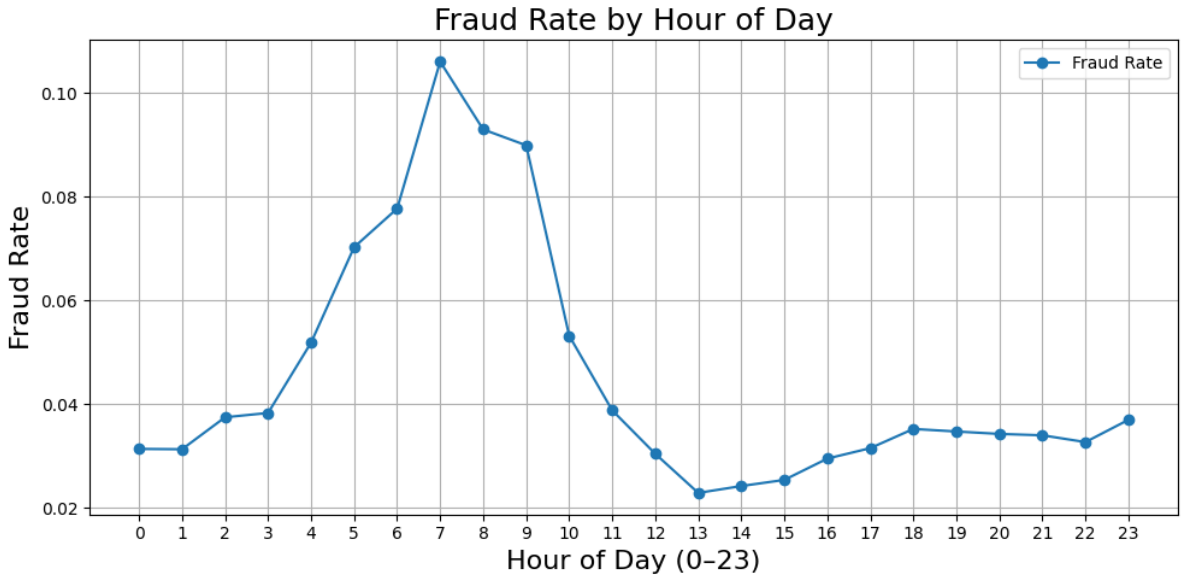

Figure 5: Fraud Rate by Hour of Day

Fraud trends by transaction day were also examined. While the pattern was less consistent, several noticeable spikes were observed in the fraud rate over time. These spikes may indicate periodic fraud campaigns or temporary system vulnerabilities that were exploited.

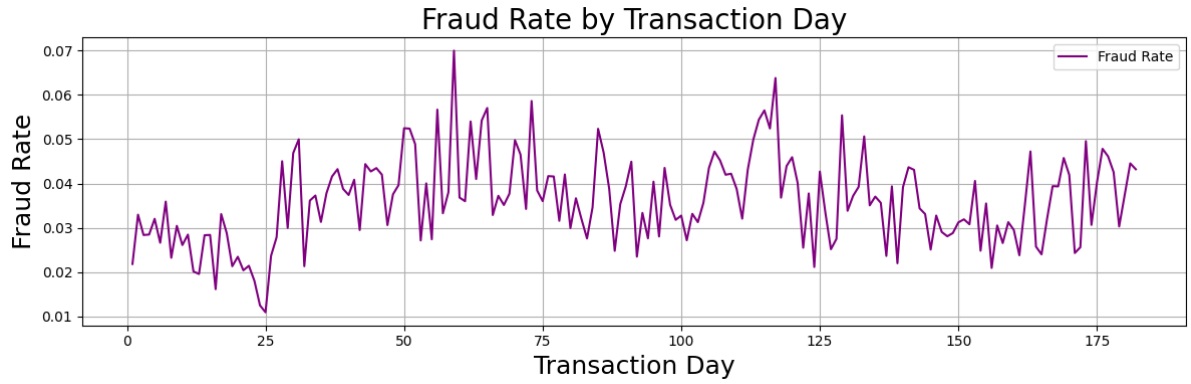

Figure 6: Fraud Rate by Transaction Day

Additionally, fraud rates were compared between weekdays and weekends. A slightly higher fraud rate was observed on weekdays (3.55%) compared to weekends (3.38%), though the difference was modest. This may reflect higher transaction volumes and more varied behavior during business days.

Overall, the results suggest that time-based features contain valuable behavioral signals for fraud detection. Patterns observed in this analysis can guide future feature engineering, and variables such as transaction_hour or is_weekend may contribute meaningfully to the performance of supervised models like XGBoost or deep neural networks.

4. Regional and Geolocation Signals

This analysis examined whether the location-based features addr1 (region) and addr2 (country) show patterns related to fraud. This supports RQ5, which focuses on identifying features, particularly engineered ones, that enhance model performance.

Fraud rates were found to vary significantly across different addr1 regions. For example, in Figure 7, billing region 465 showed fraud rates above 8%, while many other regions remained considerably lower. This suggests that regional fraud risk may be influenced by geographic, demographic, or institutional factors.

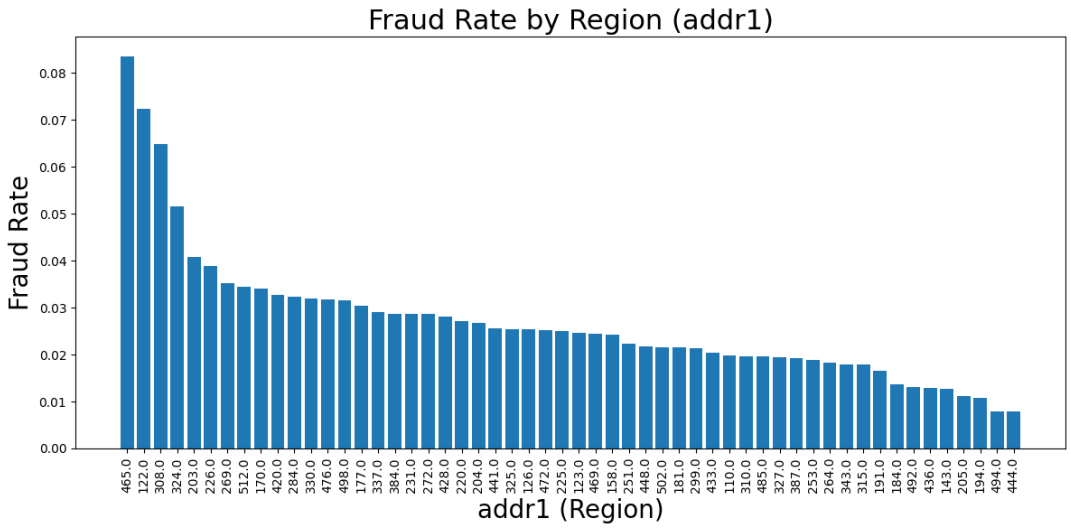

Figure 7: Fraud Rate by Region (addr1)

In the case of addr2, a sharp contrast in fraud rates across country codes was observed. Transactions from country code 87 (likely representing domestic U.S. cards) had a relatively low fraud rate of 2.4%, while transactions associated with other codes, such as 60 and 96, showed significantly higher rates ranging from 8.9% to 13.9%.

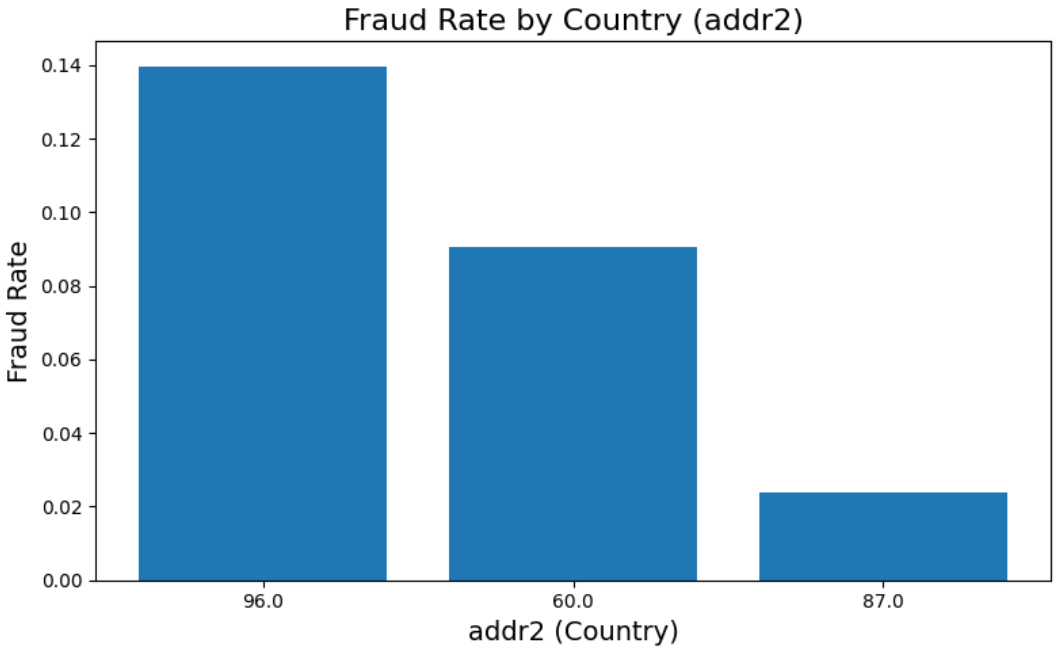

Figure 8: Fraud Rate by Country (addr2)

To further investigate this pattern, a new binary feature named foreign_transaction_flag was created, labeling transactions as foreign if addr2 is not equal to 87. This feature revealed a strong signal: foreign transactions had a fraud rate of 11.7%, nearly five times higher than that of domestic transactions (2.40%).

These results indicate that location-based fields such as addr1 and addr2 carry meaningful fraud-related information. Insights from this analysis can guide the development of new features, such as region_risk_flags or foreign_transaction_indicators, that can improve model performance in later stages, especially for models like XGBoost and DNNs that benefit from meaningful categorical or binary inputs.

5. Identity-Based Features: Email Domain and Device Type

This analysis focused on identity-related features such as email domain and device type to assess whether they contribute to fraud detection. The results directly support RQ2, which investigates whether merging identity data with transaction data can improve model performance.

Purchaser email domains displayed notable differences in fraud risk. For example, transactions associated with outlook.com had a fraud rate of approximately 9.5%, while hotmail.com and gmail.com followed with rates of 5.3% and 4.4%, respectively. In contrast, domains such as yahoo.com and aol.com showed significantly lower fraud rates, typically around 2.1%. Even domains labeled as anonymous or missing carried a fraud risk, with rates ranging from 2.5% to 2.9%, indicating that they may still provide useful signals.

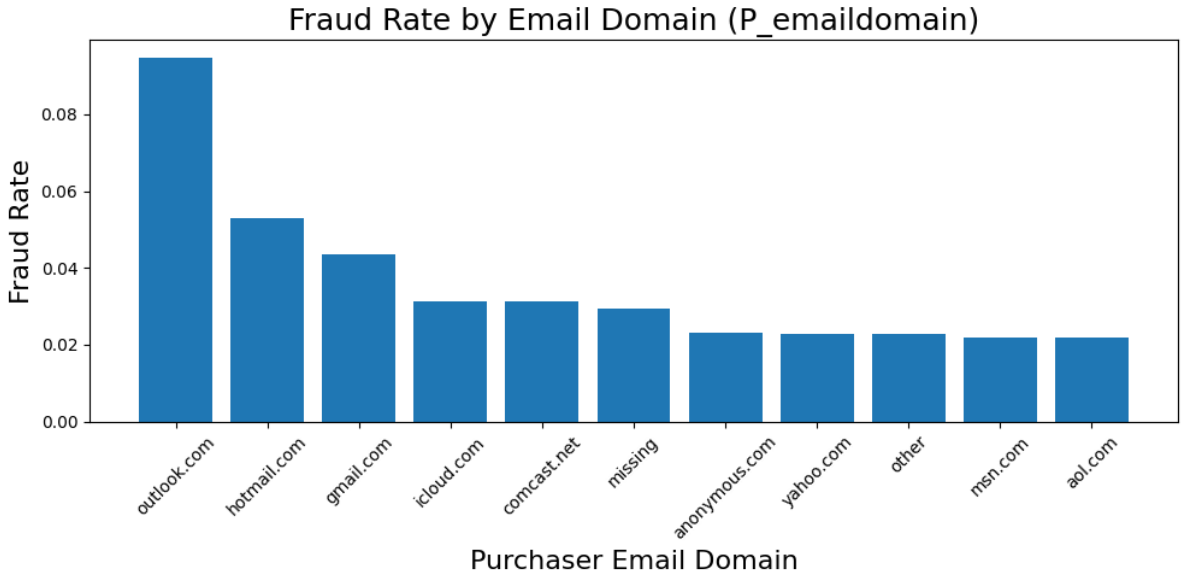

Figure 9: Fraud Rate by Email Domain (P_emaildomain)

Additionally, device type also revealed strong patterns related to fraud. Transactions conducted on mobile devices had the highest fraud rate at 10.1%, followed by desktop transactions at 6.5%. Records with missing device type information showed the lowest fraud rate at 3.1%.

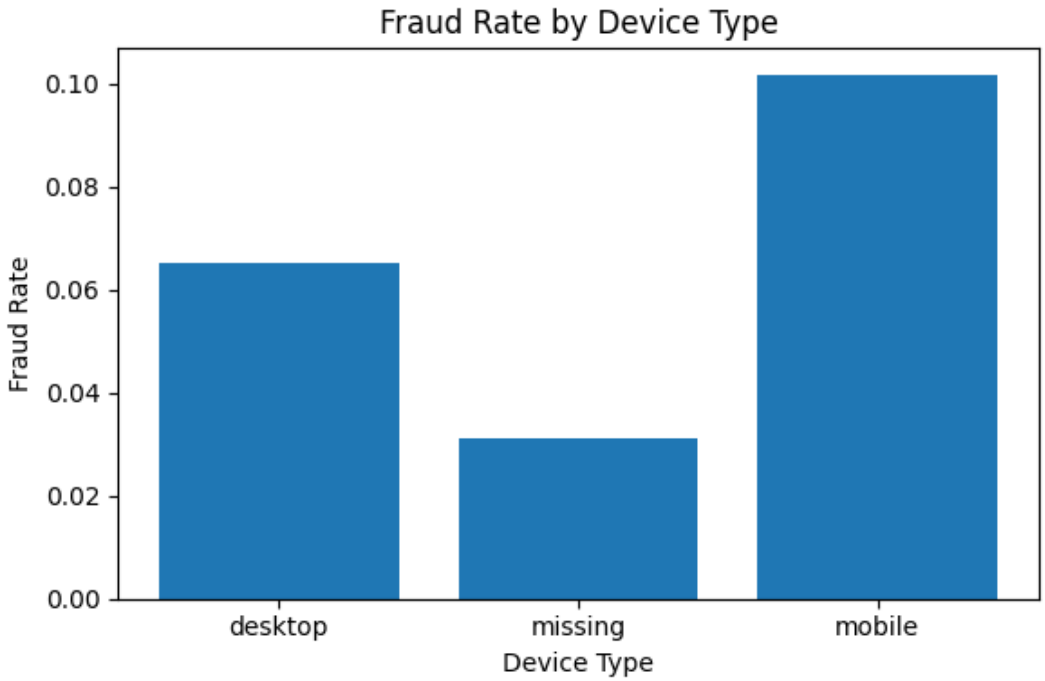

Figure 10: Fraud Rate by Device Type

Further insights were found in the DeviceInfo field. Certain Android models, such as "SM-J700M Build/MMB29K", showed fraud rates above 11%. In contrast, devices running macOS or browsers like Trident/7.0 showed much lower fraud rates, generally under 3%. Windows and iOS devices fell in between, with rates ranging from 6% to 8%.

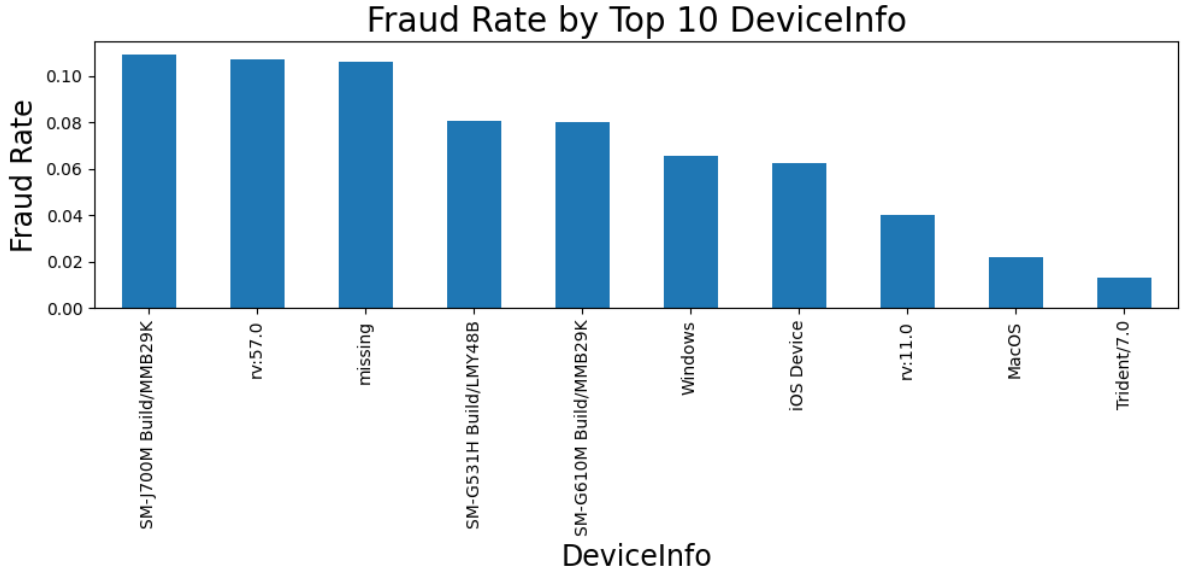

Figure 11: Fraud Rate by Top 10 DeviceInfo Values

These insights support several feature engineering opportunities that can directly applied into model development. For example, a binary variable such as is_disposable_email could be used to flag risky or missing email domains, while is_mobile_device could capture elevated risk in mobile-based transactions. Grouping DeviceInfo into a device_risk_level feature or simplifying email domains using a variable like email_domain_grouped could also help reduce cardinality and noise, making it easier for machine learning models to learn from these inputs. Overall, the analysis demonstrates that identity-related fields contain meaningful behavioral patterns and contribute valuable information for fraud detection.

6. High-Correlation Numeric Features

This part of the analysis focused on identifying which numerical features in the dataset are most strongly associated with fraud. Because the dataset contains many anonymized variables, including the V-series, Pearson correlation was used to measure the relationship between each numeric feature and the target variable, isFraud. This approach supports both RQ1 (which features best predict fraud) and RQ5 (how feature selection and engineering affect performance), and helps to identify features that may be valuable in early model development, particularly for models that rely heavily on numerical inputs, such as logistic regression.

Among the numeric features, V45 showed the strongest correlation with fraud, with a value of approximately 0.2818. This is considered relatively high in fraud detection tasks, especially given the significant class imbalance. The next highest correlation was observed for V44 (0.2604), followed closely by V86 and V87 (both just above 0.251). Other notable features included V52 (0.2395) and V51 (0.2232), with V40, V39, V38, and V43 all showing correlations around or slightly above 0.20. While these values are not extremely high, they are still meaningful and should be considered during feature selection.

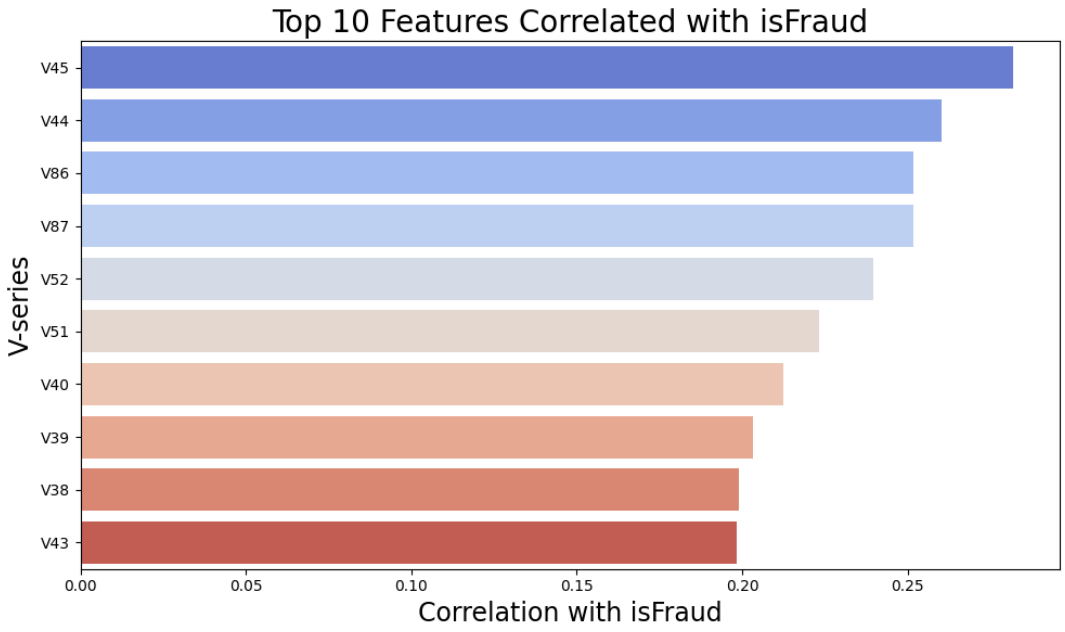

Figure 12: Top 10 Features Correlated with isFraud

To better understand the behavior of these features, the distributions of V44 and V45 were visualized by fraud class.

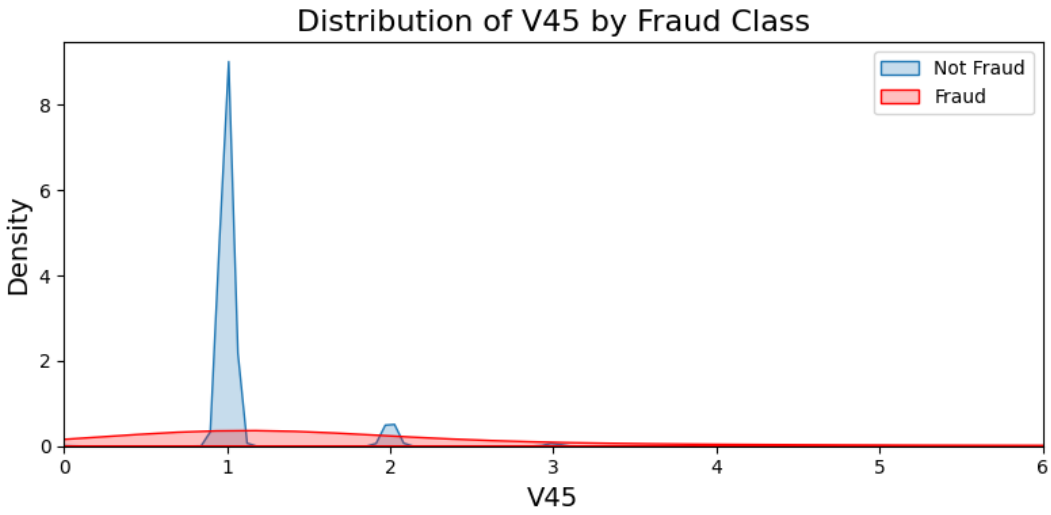

Figure 13: Distribution of V45 by Fraud Class

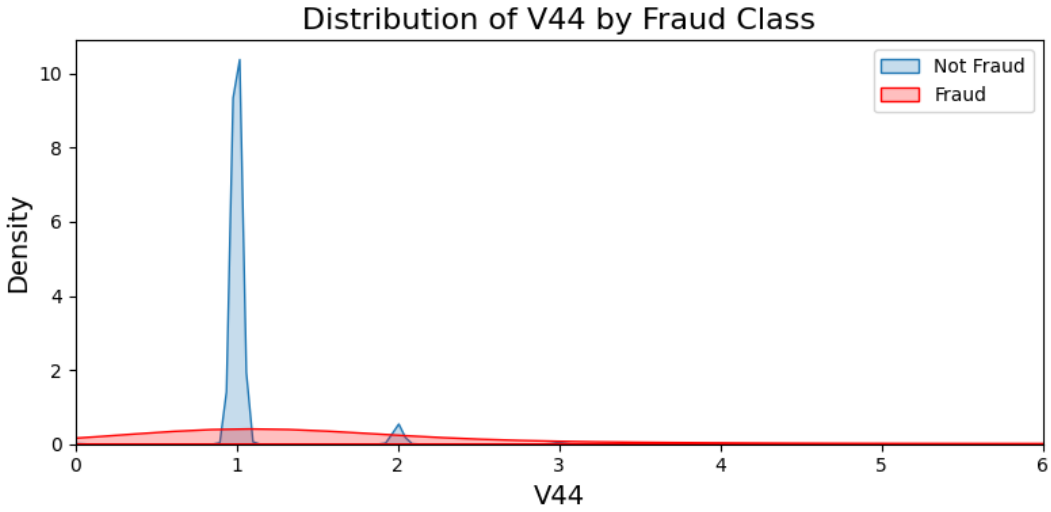

Figure 14: Distribution of V44 by Fraud Class

The plots showed clear separation between fraudulent and non-fraudulent transactions. In both cases, non-fraudulent transactions were tightly clustered near one, while fraudulent ones were more broadly distributed. These distributional differences reinforce the correlation findings and indicate that the V-series features contain valuable fraud-related signals despite being anonymized.

Insights from this analysis are especially useful at this stage, as early-stage models such as logistic regression benefit significantly from high-quality numeric inputs. Based on the findings, potential feature engineering steps include creating interaction terms (e.g., V45 $\times$ V44) and binning variables like V45 into quantile-based risk tiers to reduce skew and improve interpretability.

Overall, this analysis produced a focused list of informative numeric features. These features can help simplify the modeling process and guide the selection of inputs that are most likely to contribute to effective fraud detection.

7. Categorical Feature Patterns

This part of the analysis focused on three features that may provide important fraud-related signals: card4, card5, and M6. The fields card1 to card6 contain various payment card details, including card type, card category, issuing bank, and possibly the cardholder's country. Among these, card4 and card5 were selected for deeper analysis, along with M6, which indicates whether the billing address matches the cardholder's information.

Starting with card4, which represents the card network (such as Visa, Mastercard, or Discover), fraud rates were found to vary across different networks. Discover had the highest fraud rate at approximately 7.8%, while Visa, Mastercard, and American Express ranged between 3% and 3.5%. This suggests that certain card networks may be more frequently targeted in fraudulent transactions, potentially due to differences in verification protocols or fraud detection systems.

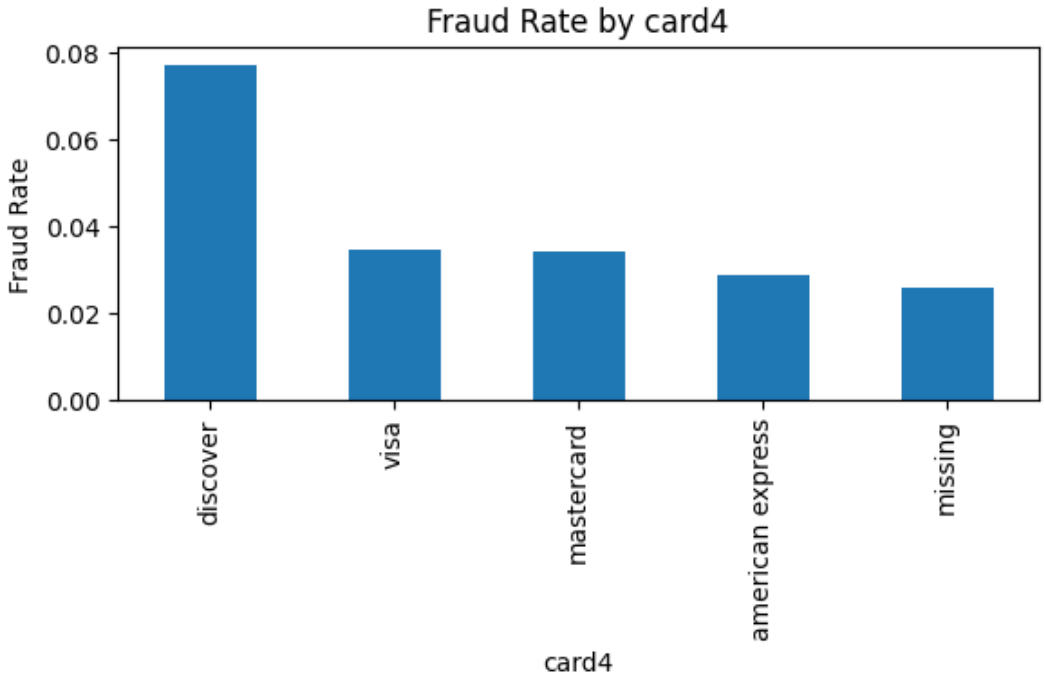

Figure 15: Fraud Rate by card4

The card5 feature, while technically numeric, behaves like a categorical variable. The analysis was narrowed to the ten most frequent values. One particular value, 137, exhibited a fraud rate exceeding 14%, while others ranged from 2% to 9%. Although the specific meanings of these values are anonymized, they likely correspond to different issuing banks or card tiers. This feature may be a strong candidate for binary transformation, such as flagging high-risk card5 values or grouping them based on observed fraud rates.

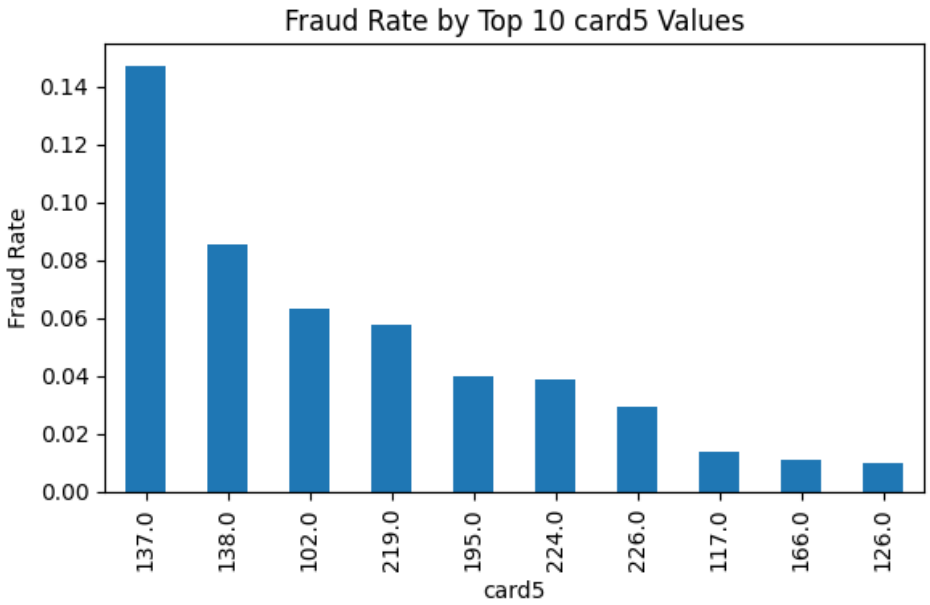

Figure 16: Fraud Rate by Top 10 card5 Values

The M6 field, which indicates whether the billing address matches the one on file, provided clearer insights. When M6 = T (true match), the fraud rate was 1.7%. It increased to 2.4% for F (false match), and spiked to 7.1% when the value was missing. This suggests that missing or mismatched billing addresses are associated with elevated fraud risk. A binary feature such as M6_is_missing could help capture this signal more effectively.

Figure 17: Fraud Rate by M6

Overall, these three features are interpretable and practical for modeling. They offer clear opportunities for binary flags, groupings, or one-hot encodings, all of which can help supervised models like XGBoost and deep neural networks detect subtle patterns in fraudulent behavior.

8. Digital Fingerprinting from Identity Data

This part of the analysis focused on selected features from the identity dataset to assess whether digital fingerprinting data provides predictive value for fraud detection. Specifically, id_02 (a numerical identity score), id_30 (operating system), and id_31 (browser) were examined. These fields represent device- or session-level metadata collected by Vesta and its partners and are likely used to flag suspicious behavior.

The distribution of id_02 by fraud class revealed subtle but useful separation. Most legitimate transactions were concentrated around lower values, while fraudulent transactions showed a broader distribution, particularly in higher ranges. Although the distinction was not sharp, the density shift suggests that id_02 may reflect behavioral traits or identity scores associated with increased fraud risk.

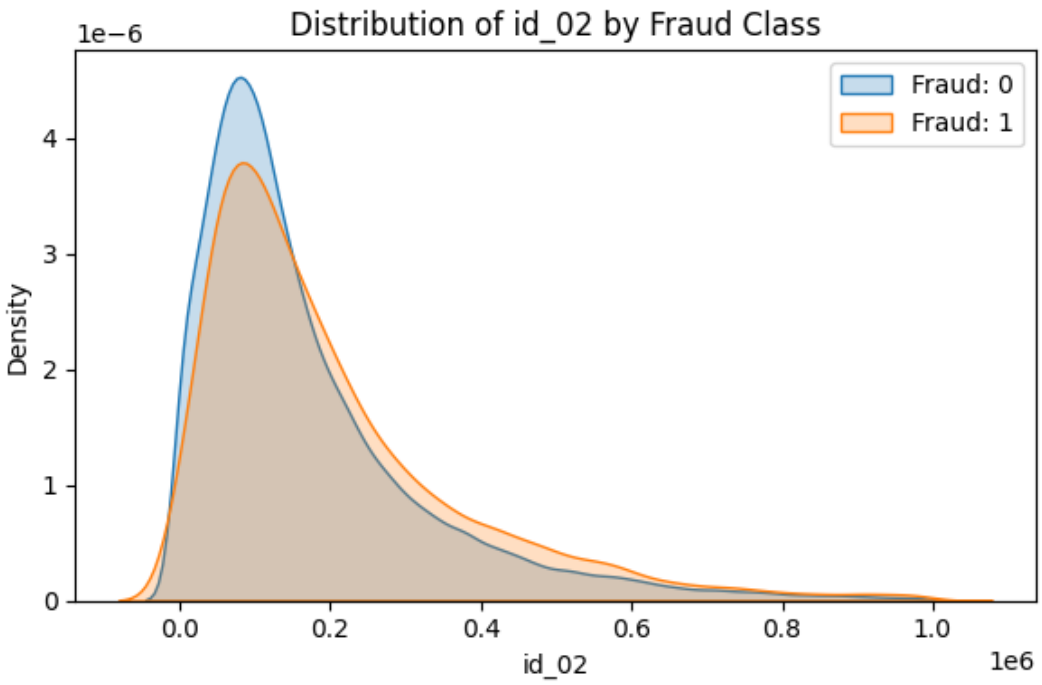

Figure 18: Distribution of id_02 by Fraud Class

In the case of id_30, which records the operating system, the highest fraud rate was observed for missing values, peaking at approximately 11.5%. This indicates that transactions where the OS could not be identified may be riskier. Among the detected operating systems, moderately elevated fraud rates were observed for older or mobile platforms such as Windows 8.1, iOS 11.3.0, and Android 7.0. These systems may represent less secure or more vulnerable environments.

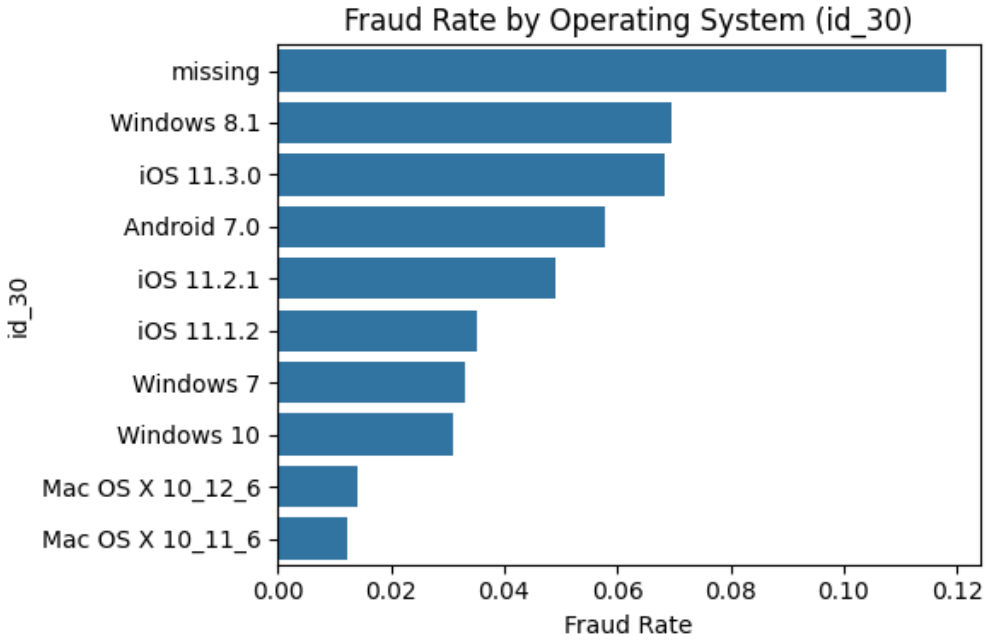

Figure 19: Fraud Rate by Operating System (id_30)

For id_31, which captures browser information, the observed pattern was even more noticeable. Generic browser labels such as "chrome generic" and "mobile safari generic" had the highest fraud rates, with "chrome generic" exceeding 17%. This suggests that fraudsters may be using tools or spoofing techniques that mask their actual browser identity. In contrast, more specific labels like "ie 11.0 for desktop" had much lower fraud rates, typically under 3%.

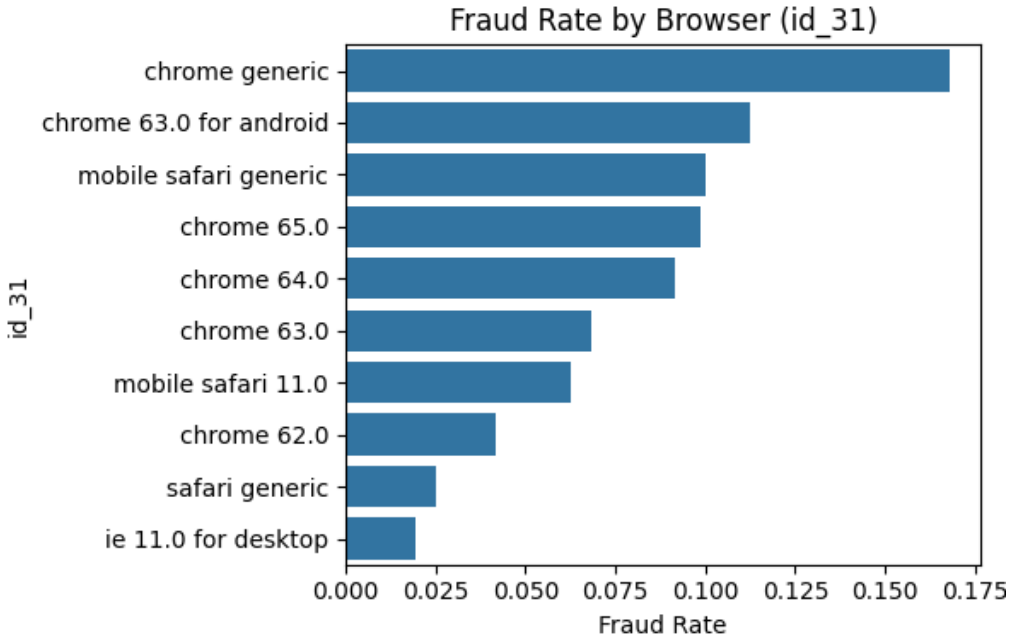

Figure 20: Fraud Rate by Browser (id_31)

Overall, these findings provide strong support for merging the identity dataset with the transaction data. They also highlight the potential of engineered features such as id_30_missing_flag or categorizing id_31 values as "generic" versus "specific." Digital fingerprinting signals found in identity data can contribute significantly to model performance, particularly in advanced algorithms such as XGBoost and deep neural networks.

9. Spatial and Behavioral Risk Indicators

This part of the analysis examined two distance-based features, dist1 and dist2, along with one count-based feature, C13, to identify potential fraud-related patterns. Although the exact meaning of C13 is anonymized, these features may capture location inconsistencies or behavioral anomalies.

For dist1, fraud rates increased steadily with distance and peaked in the 200 to 500 mile range. This suggests that mid-range geographic discrepancies may be more indicative of fraud than either very short or very long distances. Beyond 500 miles, the fraud rate declined slightly.

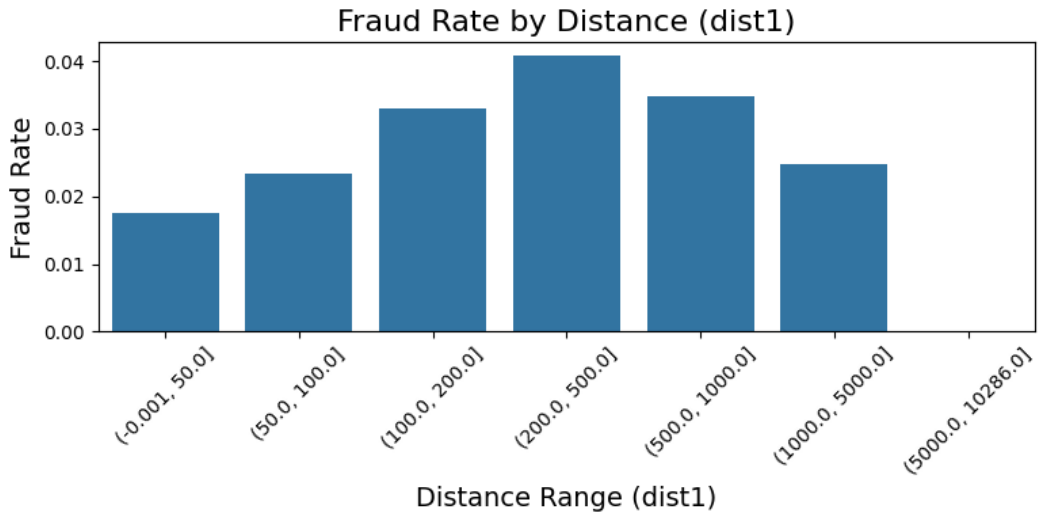

Figure 21: Fraud Rate by Distance Range (dist1)

dist2 showed an even stronger relationship with fraud. Higher fraud rates were observed in the 100 to 1000 mile range, with a sharp increase for distances exceeding 5000 miles, where the fraud rate exceeded 13%. This pattern may reflect the use of spoofed IP addresses, proxies, or VPNs, which can introduce large mismatches between the billing and access locations.

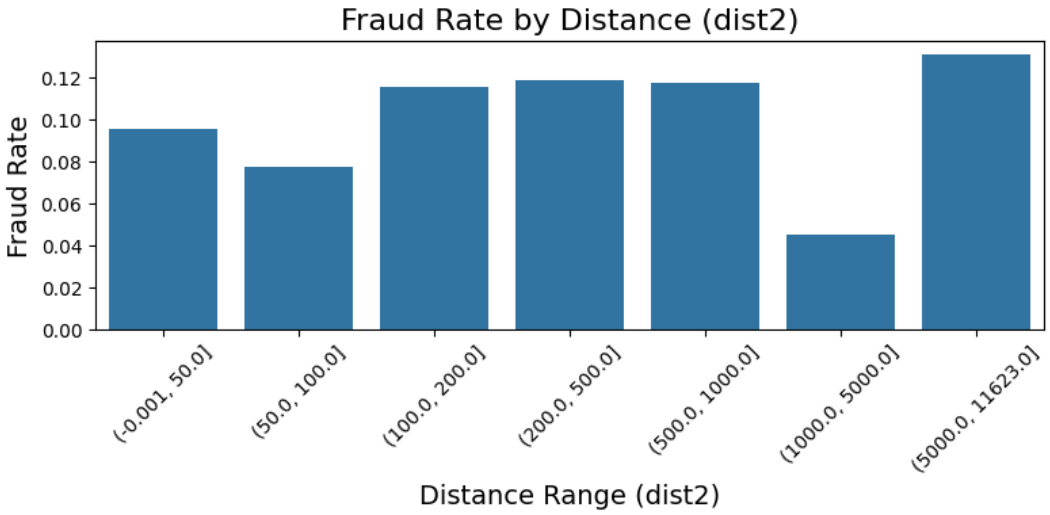

Figure 22: Fraud Rate by Distance Range (dist2)

The feature C13 showed an inverse trend compared to typical count-based variables. Fraud was most prevalent when C13 had low values, and the rate decreased as the count increased. While the specific nature of C13 is unknown, this behavior suggests that transactions with limited historical usage or rare patterns may carry a higher fraud risk compared to more frequent or established ones.

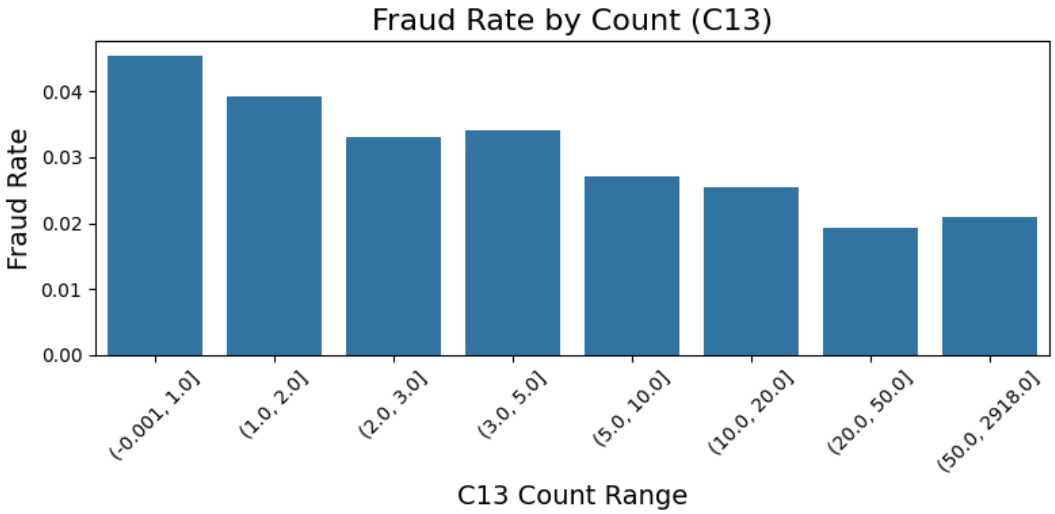

Figure 23: Fraud Rate by Count Range (C13)

Overall, creating simple flags like dist2_over_5000 or C13_low could help models capture these non-obvious patterns more directly. In addition, binning or transforming these features can also reduce noise and improve the model's ability to learn useful fraud signals.

Project Approach

This project began with a review of fifteen peer-reviewed journal and conference papers on fraud detection to gain an overview of how supervised machine learning is applied and to understand the common challenges encountered by researchers. These challenges include extreme class imbalance, anonymized features, and the trade-off between identifying fraudulent transactions and minimizing false positives. The literature review helped define the direction of the project by highlighting effective strategies such as ensemble learning, identity feature engineering, and data balancing methods.

Following the background research, an exploratory data analysis (EDA) was conducted on the IEEE-CIS Fraud Detection dataset. This step provided a deeper understanding of the dataset's structure, quality, and key features, with particular attention given to missing values and the distribution of fraud cases. Initial preprocessing involved cleaning the dataset by handling missing data and standardizing column names to ensure consistency and usability in Python-based workflows.

Together, the literature review and EDA provide a solid foundation for the development and evaluation of machine learning models in the next phase of the project. The diagram below outlines the overall approach from research to modeling.

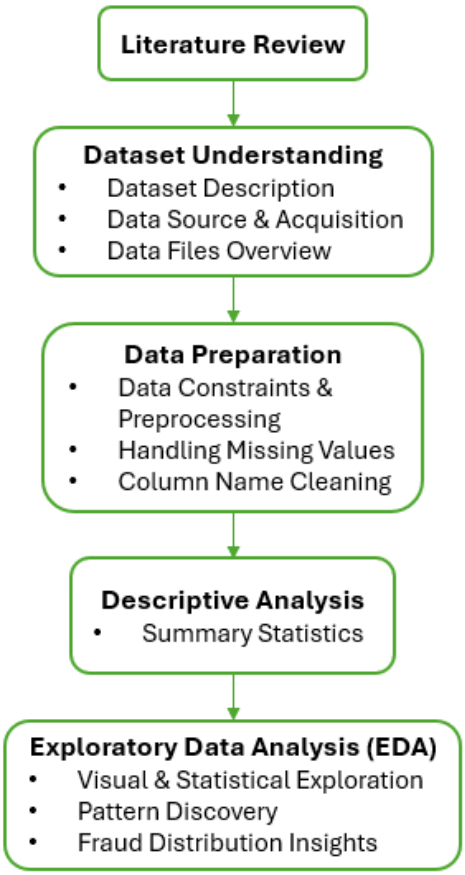

Figure 24: Overall Project Methodology

GitHub Repository

All codes, visualizations, and results for this Literature Review and EDA stage are available at the following GitHub repository:
Link: https://github.com/nguyenduyanhluong/TMU-MRP-2025/tree/main/literature_review_and_eda

References

[1] Y.-T. Lei, C.-Q. Ma, Y.-S. Ren, X.-Q. Chen, S. Narayan, and A. N. Q. Huynh, "A distributed deep neural network model for credit card fraud detection," *Finance Research Letters*, vol. 58, p. 104547, 2023. https://doi.org/10.1016/j.frl.2023.104547

[2] D. Shah and L. K. Sharma, "Credit Card Fraud Detection using Decision Tree and Random Forest," *ITM Web of Conferences*, vol. 53, p. 2012, 2023. https://doi.org/10.1051/itmconf/20235302012

[3] N. S. Alfaiz and S. M. Fati, "Enhanced Credit Card Fraud Detection Model Using Machine Learning," *Electronics*, vol. 11, no. 4, p. 662, 2022. https://doi.org/10.3390/electronics11040662

[4] M. Alamri and M. Ykhlef, "Hybrid Undersampling and Oversampling for Handling Imbalanced Credit Card Data," *IEEE Access*, vol. 12, pp. 14050–14060, 2024. https://doi.org/10.1109/ACCESS.2024.3357091

[5] E. Ileberi, Y. Sun, and Z. Wang, "Performance Evaluation of Machine Learning Methods for Credit Card Fraud Detection Using SMOTE and AdaBoost," *IEEE Access*, vol. 9, pp. 165286–165294, 2021. https://doi.org/10.1109/ACCESS.2021.3134330

[6] M. M. Lahbiss and Y. Chtouki, "Credit Card Fraud Detection in Imbalanced Datasets: A Comparative Analysis of Machine Learning Techniques," *2024 International Conference on Computer and Applications (ICCA)*, pp. 1–6, 2024. https://doi.org/10.1109/ICCA62237.2024.10927865

[7] S. Lei, K. Xu, Y. Huang, and X. Sha, "An Xgboost based system for financial fraud detection," *E3S Web of Conferences*, vol. 214, p. 2042, 2020. https://doi.org/10.1051/e3sconf/202021402042

[8] Y. Lucas et al., "Multiple perspectives HMM-based feature engineering for credit card fraud detection," *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pp. 1359–1361, 2019. https://doi.org/10.1145/3297280.3297586

[9] A. Correa Bahnsen, D. Aouada, A. Stojanovic, and B. Ottersten, "Feature engineering strategies for credit card fraud detection," *Expert Systems with Applications*, vol. 51, pp. 134–142, 2016. https://doi.org/10.1016/j.eswa.2015.12.030

[10] F. Carcillo et al., "Combining unsupervised and supervised learning in credit card fraud detection," *Information Sciences*, vol. 557, pp. 317–331, 2021. https://doi.org/10.1016/j.ins.2019.05.042

[11] M. A. K. Achakzai and J. Peng, "Detecting financial statement fraud using dynamic ensemble machine learning," *International Review of Financial Analysis*, vol. 89, p. 102827, 2023. https://doi.org/10.1016/j.irfa.2023.102827

[12] D. Jahnavi et al., "Robust Hybrid Machine Learning Model for Financial Fraud Detection in Credit Card Transactions," *2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, pp. 680–686, 2024. https://doi.org/10.1109/IDCIoT59759.2024.10467340

[13] S. Chaurasia et al., "Analysis of Ensemble Machine Learning Models for Fraud Detection," *2024 International Conference on Intelligent Systems for Cybersecurity (ISCS)*, pp. 1–6, 2024. https://doi.org/10.1109/ISCS61804.2024.10581076

[14] A. Patel, M. Patel, and P. Patel, "Exploring Supervised Machine Learning Techniques for Detecting Credit Card Fraud: An Investigative Review," *ITM Web of Conferences*, vol. 65, p. 3006, 2024. https://doi.org/10.1051/itmconf/20246503006

[15] Jyoti, K. Bhardwaj, G. Garima, M. Kumar, R. Verma, and D. Kumar, "Machine Learning and Deep Learning for Credit Card Fraud Detection: A Comparative Analysis," *2024 International Conference on Artificial Intelligence and Emerging Technology (Global AI Summit)*, pp. 131–136, 2024. https://doi.org/10.1109/GlobalAISummit62156.2024.10947915