# CREDIT CARD FRAUD DETECTION ANALYSIS REPORT

**By:**
**Name – Saanya Jain**
**Date – 05/13/2024**

# Table of contents


| Sections | Page no. |
|---|---|
| Executive summary | 3 |
| Data description | 4 |
| Data cleaning | 8 |
| Variable creation | 11 |
| Feature selection | 19 |
| Model exploration | 21 |
| Final model performance | 26 |
| Financial curves | 29 |
| Summary | 30 |
| Appendix | 32 |

## Executive summary

In this project, a predictive model was developed, focusing on the identification of fraudulent transactions within the company's operations. A rigorous false discovery rate (FDR) of 3% was applied to the out-of-time (oot) sample, ensuring a high level of reliability in detecting fraud. This approach is projected to result in annual savings of approximately $48 million. The significant financial impact stems from enhanced fraud detection capabilities, which serve to minimize losses and optimize resource allocation.

# Data Description

Data overview:
The dataset contains transaction records from card payments, capturing a wide array of attributes including transaction amounts, merchant details, and fraud indicators. The data comes from real-world financial transactions over 1 year and includes both numerical and categorical fields. It contains 10 fields and 97,852 records and is designed for analytical exploration and fraud detection model development.

Statistics tables:

1. Numerical Fields Table

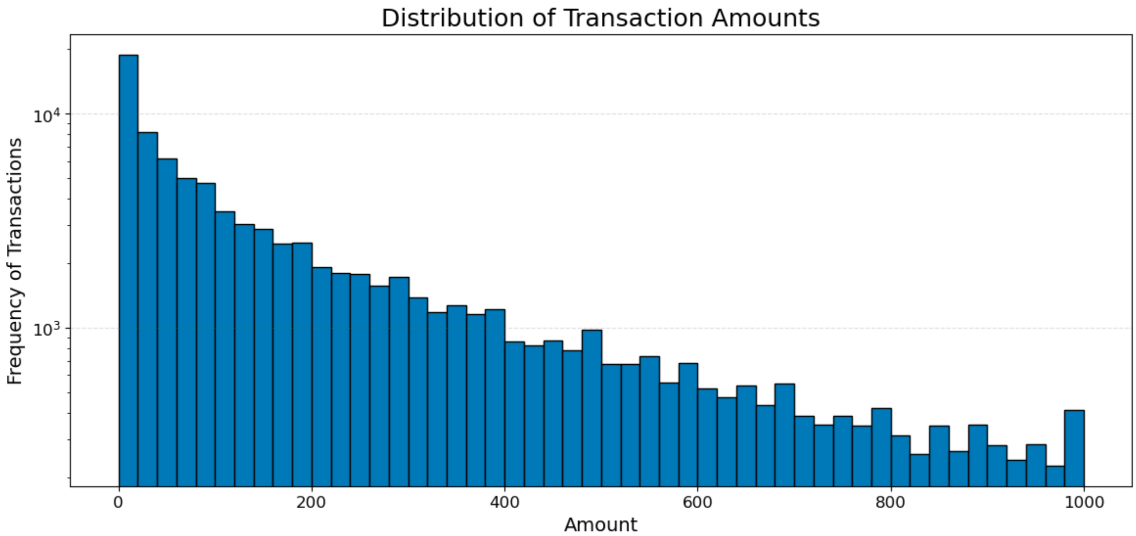
| | Field Name | Field Type | # Records Have Values | % Populated | # Zeros | Min | Max | Mean | Standard Deviation | Most Common |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Amount | numeric | 97852 | 100.0% | 0 | 0.01 | 3102045.53 | 425.466438 | 9949.8 | 3.62 |

2. Categorical Fields Table

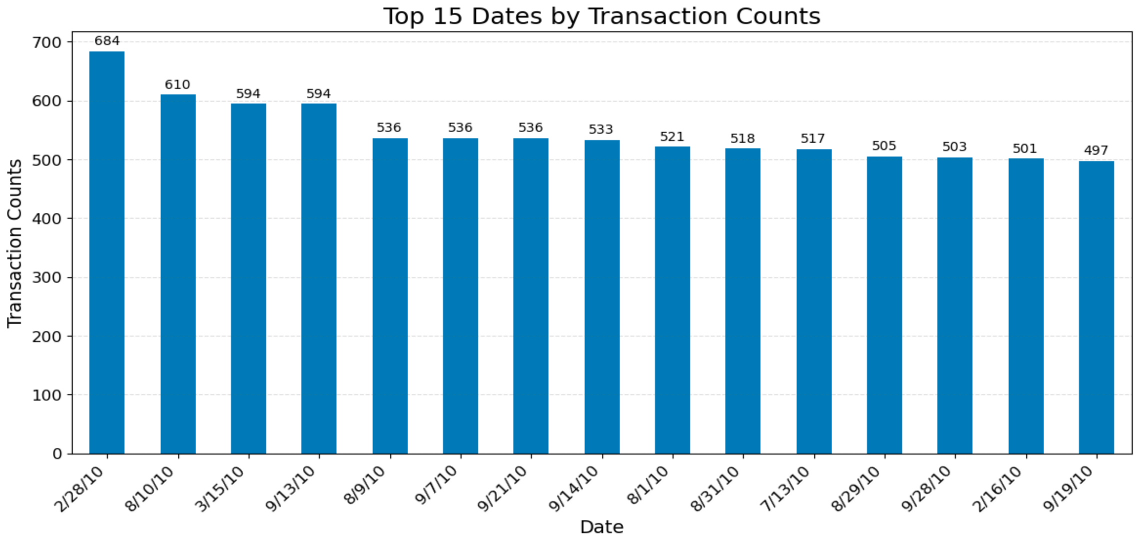
| | Field Name | Field Type | # Records Have Values | % Populated | # Zeros | # Unique Values | Most Common |
|---|---|---|---|---|---|---|---|
| 0 | Date | categorical | 97852 | 100.0% | 0 | 365 | 2/28/10 |
| 1 | Merchnum | categorical | 94455 | 96.5% | 0 | 13091 | 930090121224 |
| 2 | Merch description | categorical | 97852 | 100.0% | 0 | 13126 | GSA-FSS-ADV |
| 3 | Merch state | categorical | 96649 | 98.8% | 0 | 227 | TN |
| 4 | Transtype | categorical | 97852 | 100.0% | 0 | 4 | P |
| 5 | Recnum | categorical | 97852 | 100.0% | 0 | 97852 | 1 |
| 6 | Fraud | categorical | 97852 | 100.0% | 95805 | 2 | 0 |
| 7 | Cardnum | categorical | 97852 | 100.0% | 0 | 1645 | 5142148452 |
| 8 | Merch zip | categorical | 93149 | 95.2% | 0 | 4567 | 38118 |

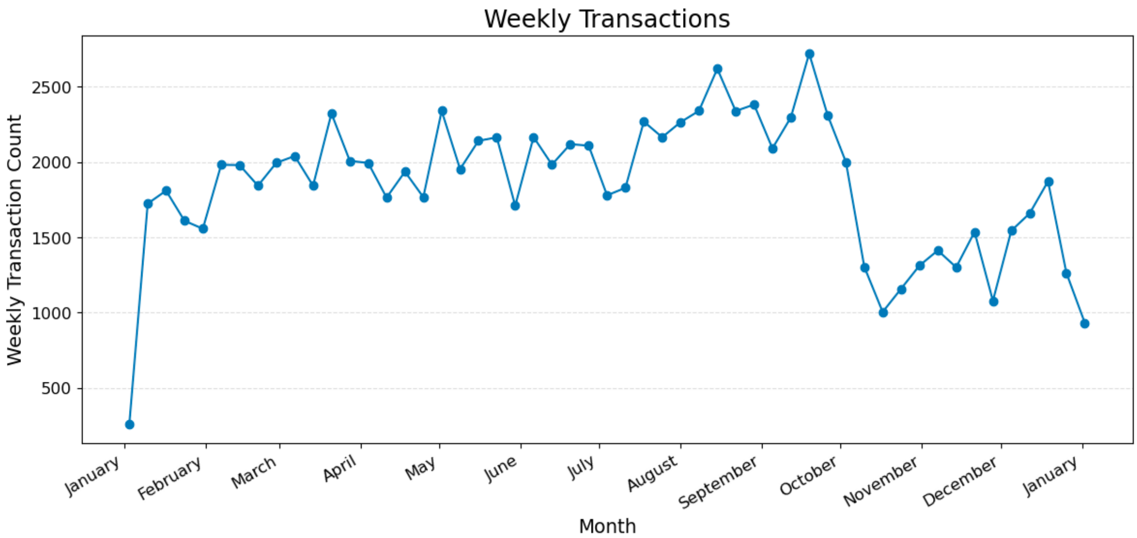
Field distributions:

1. Field Name: Amount
   Description: The dollar amount of the transaction varies widely from as little as $0.01 to over $3 million, showcasing a vast range of transaction values. The first graph shows the distribution of transaction amounts ranging from $0 to $1000 since the maximum number of transactions occurred in this range. The second graph shows a box plot with several high-value outliers.

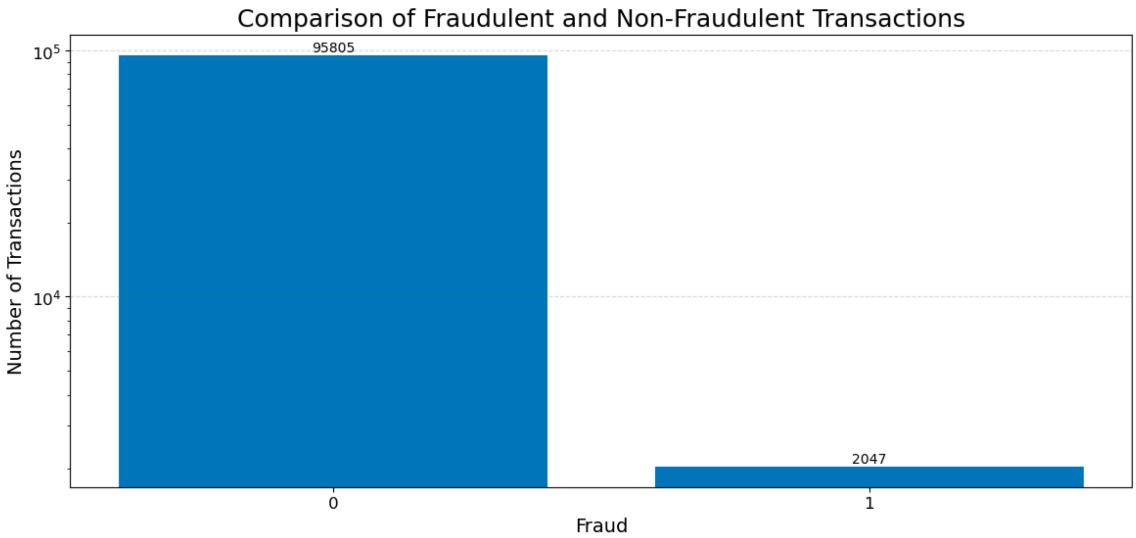


2. Field Name: Date
   Description: Date of the transaction, spanning a period from 1st January 2010 to 31st December 2010, with the most transactions recorded on 28th February.
   The first graph shows the top 15 dates when transactions occurred. The second graph visualizes the number of weekly transactions over time.



3. Field Name: Fraud
   Description: Fraud identification label. Fraud = 0 (Not fraudulent), Fraud = 1 (Fraudulent). The total count of fraud = 0 is 95,805. The total count of fraud = 1 is 2,047.



## Data Cleaning

1. Exclusions: The dataset reveals four distinct types of transactions: P (97,497), A (181), D (173), and Y (1). The majority are type P, which likely indicates a purchase, while the other types could signify authorizations or declined transactions. For clarity and focus in the analysis, only transactions categorized as type P will be considered, and all other types will be excluded.

2. Outlier Treatment: There's one record in the data with a transaction amount exceeding $3,000,000, which significantly surpasses the next-highest transaction of $47,900. This outlier stems from a transaction with a Mexican retailer and is not flagged as fraudulent. After careful consideration, it has been determined that this particular record will be removed from the analysis to avoid skewing the results.

3. Imputation process for the required fields:
    1) Merchnum: The dataset contains 3,279 records where the Merchnum field was absent, necessitating the estimation of reasonable values to fill these gaps. The Merch description field was used to assign the most fitting Merchnum to each corresponding description. For entries labeled as "RETAIL CREDIT ADJUSTMENT" and "RETAIL DEBIT ADJUSTMENT," the Merchnum was set as "unknown." To address the remaining records without a Merchnum, unique new Merchnum values were assigned based on the 515 unique

Merch descriptions, ensuring complete data integrity in the Merchnum field.

2) Merch state: To address gaps in the Merch state field where 1,028 records were missing, a methodical imputation process was implemented using relationships between Merch zip codes and Merch state to establish a zip_state mapping. This initial step allowed for the imputation of several states, but subsequent mappings using Merchnum and Merch description yielded limited success. Entries categorized as “RETAIL CREDIT ADJUSTMENT” and “RETAIL DEBIT ADJUSTMENT” were explicitly labeled as “unknown” to handle non-standard transactions. Furthermore, non-U.S. locations were tagged as ‘foreign’ based on a list of U.S. states and territories, with remaining gaps ultimately labeled as “unknown” to ensure complete data coverage in the Merch state field.

3) Merch zip: To address the initial 4,347 missing Merch zip values in the dataset, a systematic imputation approach was employed, utilizing internal and external data sources for comprehensive coverage. Foundational mappings were created by linking Merchnum and Merch description to existing Merch zip records, which significantly reduced the number of missing values. Additional steps included designating "unknown" for specific entries such as "RETAIL CREDIT ADJUSTMENT" or "RETAIL DEBIT ADJUSTMENT,"

and using the most populous zip codes from known Merch states for further imputation. The process concluded with the remaining gaps being labeled "unknown," ensuring that no records in the dataset lacked a Merch zip field entry.

## Variable creation

In the development of the predictive model for identifying fraudulent transactions, the creation of new variables was crucial to enrich the dataset and enhance the model's ability to discern patterns indicative of fraud. The variables were meticulously designed to capture various aspects of transaction behavior and entity profiles, reflecting both historical data trends and predictive indicators of fraud. This approach aimed to improve model accuracy by integrating a broader context and deeper insights into each transaction to identify fraudulent activities.

High-level description of reasoning:

1. Temporal Variables: These include day of the week and time of day, which were introduced based on the hypothesis that fraudulent activities could follow specific temporal patterns.

2. Risk Scoring Variables: Variables such as 'Risk for Day of Week' were generated to quantify the risk associated with transactions on particular days, derived from historical fraud incidence rates.

3. Transaction Frequency and Amount Variables: Multiple variables were crafted to monitor the frequency and amounts of transactions over different time windows (e.g., last 1, 3, 7, 14, 30, 60 days). These variables help in understanding the short-term and long-term spending behaviors of entities and detecting anomalies.

4. Categorical Encoding: Techniques like target encoding were applied to categorical fields such as merchant categories, turning potentially informative but unwieldy categorical data into a format suitable for modeling.

5. Specialized Financial Indicators: Variables like 'Transaction Count Ratios' and 'Transaction Amount Ratios' compare recent activity to historical patterns, highlighting unusual deviations.

| **Description** | **# Variables_Created** |
|---|---|
| **Day of week:**<br>The name of the weekday extracted from the Date column, indicating the specific day on which a transaction occurred | 1 |
| **Risk for Day of week:**<br>Risk score associated with each day of the week | 1 |
| **Target Encoded:**<br>Numeric representations of categorical features based on the mean target (fraud) value per category | 3 |
| **Day Since:**<br>Tracks days since the last transaction per entity | 23 |
| | |

| | |
|---|---|
| **Transaction Count:** Counts transactions per entity over the last {0, 1, 3, 7, 14, 30, 60} days | 161 |
| **Average Transaction Amount:** Computes average spending per entity over the last {0, 1, 3, 7, 14, 30, 60} days | 161 |
| **Maximum Transaction Amount:** Identifies the highest spending per entity in the past {0, 1, 3, 7, 14, 30, 60} days | 161 |
| **Median Transaction Amount:** Determines the median spending per entity in the last {0, 1, 3, 7, 14, 30, 60} days | 161 |
| **Total Transaction Amount:** Sums transaction amounts per entity over the last {0, 1, 3, 7, 14, 30, 60} days | 161 |
| | |

| | |
|---|---|
| **Transaction Amount Ratios:** Compares individual transactions in the last {0, 1, 3, 7, 14, 30, 60} days to the average, max, median and total transactions in the last {0, 1, 3, 7, 14, 30, 60} days | 644 |
| **Transaction Count Ratios:** Number of transactions for all entities in the last {0, 1} days divided by the number of transactions for the entities in the last {7, 14, 30, 60} days, normalized by the last {7, 14, 30, 60} days | 184 |
| **Total Transaction Amount Ratios:** Total transaction amount for all entities in in the last {0, 1} days divided by the total transaction amount for the entities in the last {7, 14, 30, 60} days, normalized by the last {7, 14, 30, 60} days | 184 |
| | |

| | |
|---|---|
| **Transaction Velocity Ratios:** Ratios of transaction frequency over the last {0, 1} days compared to the recency of transactions normalized over the last {7, 14, 30, 60} days for all entities | 184 |
| **Average Transaction Variability:** Measures the average difference in transaction amounts for each entity over the last {0, 1, 3, 7, 14, 30 days} | 138 |
| **Maximum Transaction Variability:** Captures the largest single change in transaction amounts for each entity in the last {0, 1, 3, 7, 14, 30 days} | 138 |
| **Median Transaction Variability:** Calculates the median difference in transaction amounts for each entity in the last {0, 1, 3, 7, 14, 30 days} | 138 |
| | |

| | |
|---|---|
| **Unique Interaction Counts:** Variables measure the unique interactions between pairs of entities in the last {1, 3, 7, 14, 30, 60} days | 696 |
| **Squared Transaction Count Ratios:** Number of transactions in the last {0, 1} days divided by the total number of transactions in the last {7, 14, 30, 60} days, divided by the square of {7, 14, 30, 60} for each entity | 184 |
| **Amount Categories:** Segments transaction amounts into five evenly populated quantiles, labeled from 1 to 5 | 1 |
| **Foreign Zip Codes:** Introduces a binary indicator to flag merchant zip codes not found in the US zip code database, distinguishing between domestic (0) and international (1) transactions | 1 |
| | |

| New variable categories | |
|---|---|
| **Time Weighted Transaction Frequency:** Calculates the frequency of transactions for each entity, adjusted by a decay factor that weights more recent transactions higher | 23 |
| **Weekday vs. Weekend Spending Ratio:** Computes the ratio of total spending on weekends to weekdays for each entity | 23 |
| **Change in Spending Behavior Over Time:** Measures the percentage change in transaction amounts over time for each entity | 23 |
| **High-Value Transaction Rate:** Identifies the proportion of transactions that are in the top 90$^{th}$ percentile of amounts for each entity | 23 |
| **Average Transaction Value Classification:** Segments entities into categories based on their average transaction amount (low, medium, high) | 23 |
| | |

| | |
|---|---|
| **Loyalty Score:**<br>Assesses the activity frequency relative to the lifespan of each entity's transactions | 23 |
| **Time of Day Analysis:**<br>Categorizes transactions into time bands (morning, afternoon, evening, night) | 1 |
| **Days to Nearest Special Date:**<br>Calculates the number of days until the next significant date (e.g., Christmas) | 1 |
| **Rolling Variability of Transaction Amounts:**<br>Analyzes the 30-day rolling standard deviation of transaction amounts for each entity | 23 |
| **Most Common Transaction Hour:**<br>Determines the hour of the day when most transactions occur for each entity | 23 |
| **New vs. Returning Customer Analysis:**<br>Flags transactions as either from new or returning customers | 46 |

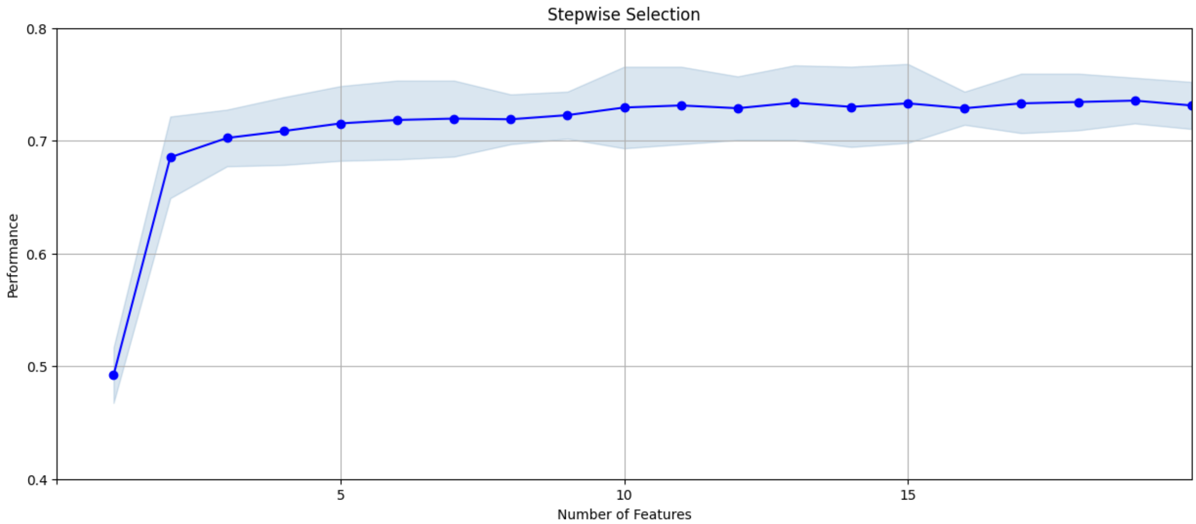
## Feature Selection

Description:
The feature selection process played a crucial role in enhancing the predictive accuracy and efficiency of the fraud detection model. This process utilized a combination of filtering and multiple wrapper methods, including Random Forest, LightGBM, and Catboost, to refine the selection of the most effective predictors of fraudulent transactions. Initially, a large set of candidate variables was subjected to a filtering method based on their univariate scores, assessing each variable's individual predictive power. After filtering, various wrapper models were applied to evaluate the collective performance of the variables. Ultimately, the Catboost model, with num_filter=200 and num_wrapper=20, was selected for its superior ability to refine the selection and optimize the model's performance.

## List of final variables with the univariate filter score:

| wrapper order | variable | filter score |
|---|---|---|
| 1 | Cardnum_unique_count_for_card_state_1 | 0.47606661 |
| 2 | Card_Merchdesc_total_3 | 0.31967518 |
| 3 | card_state_max_3 | 0.34132338 |
| 4 | card_state_max_1 | 0.33479740 |
| 5 | Cardnum_vdratio_0by14 | 0.37903676 |
| 6 | Card_dow_actual/max_7 | 0.34899988 |
| 7 | Cardnum_count_14 | 0.44544343 |
| 8 | card_state_max_14 | 0.30594589 |
| 9 | Card_dow_unique_count_for_merch_state_60 | 0.32057982 |
| 10 | card_merch_total_14 | 0.32902312 |
| 11 | Cardnum_count_0_by_60_sq | 0.31787144 |
| 12 | Card_dow_unique_count_for_state_des_14 | 0.37433364 |
| 13 | Cardnum_avg_0 | 0.36315040 |
| 14 | Card_dow_avg_7 | 0.32609082 |
| 15 | card_zip_total_7 | 0.32580738 |
| 16 | merch_state_total_1 | 0.30489321 |
| 17 | Merchnum_total_1 | 0.30486816 |
| 18 | Card_dow_actual/toal_7 | 0.38928809 |
| 19 | Cardnum_avg_1 | 0.35252902 |
| 20 | Card_dow_count_30 | 0.39045359 |

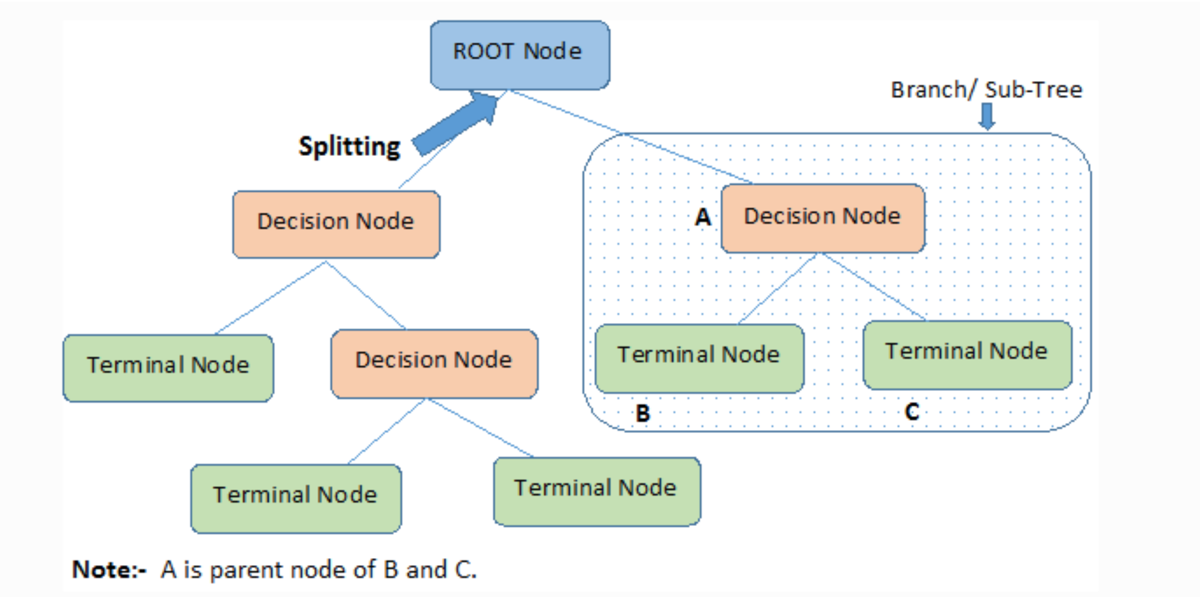
## Plot of performance vs number of variables:



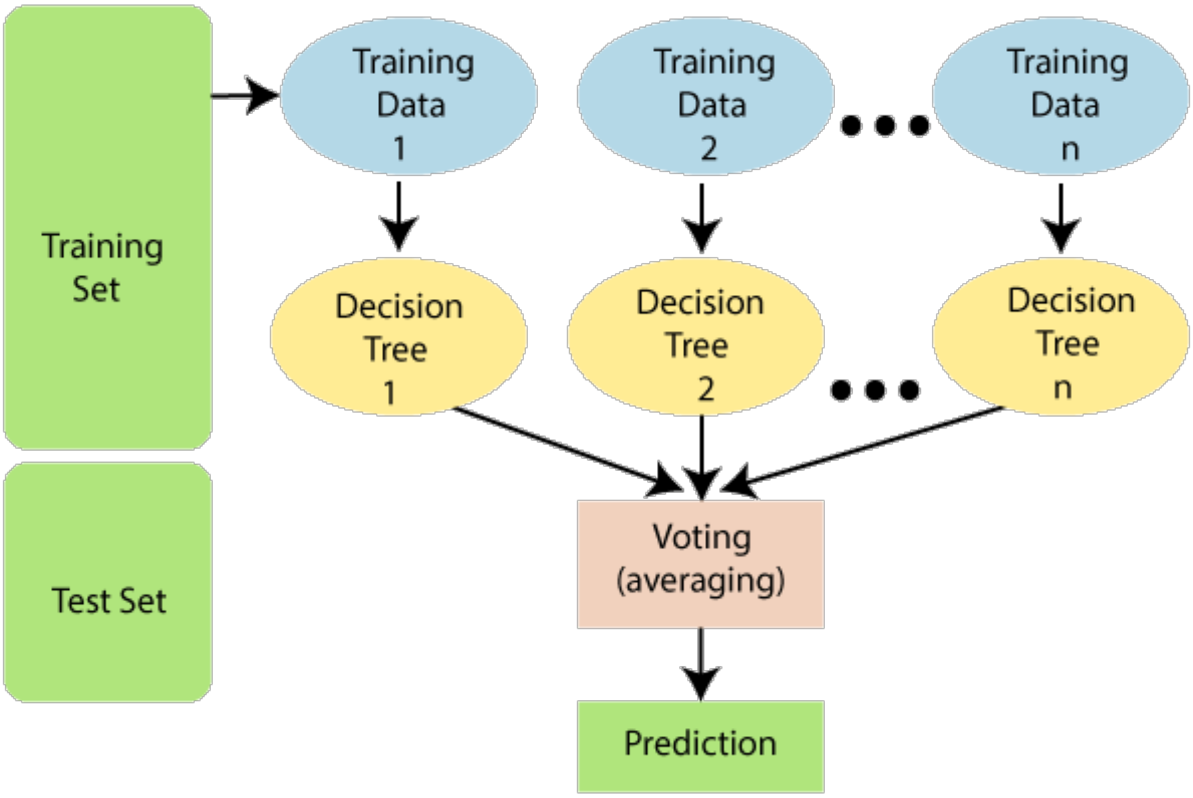
# Model exploration

High-level description:

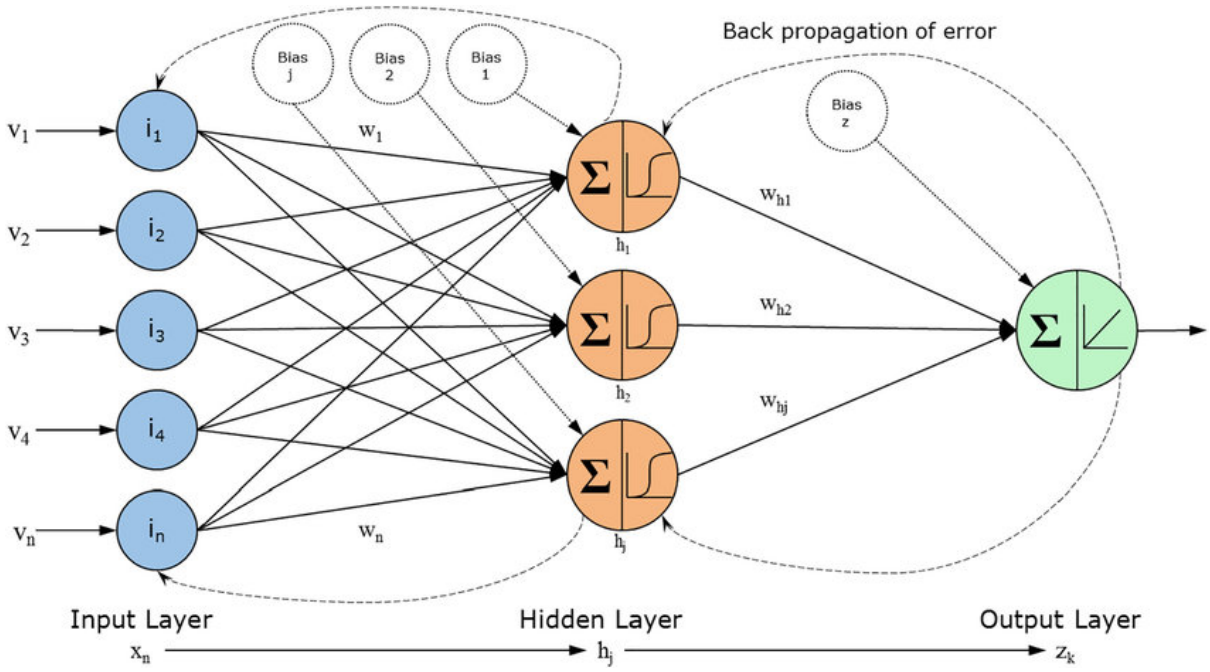
1. Decision Tree: Decision trees are a type of supervised learning algorithm predominantly used for classification and regression tasks. They work by splitting the data into branches at decision nodes, which are based on feature values. Each decision node in the tree represents a test on a specific attribute, and each branch represents an outcome of that test. This process results in a tree-like structure of decisions, where each leaf node represents a class label or a continuous outcome. Decision trees are easy to interpret and can handle both numerical and categorical data, but they are prone to overfitting, especially with complex datasets.



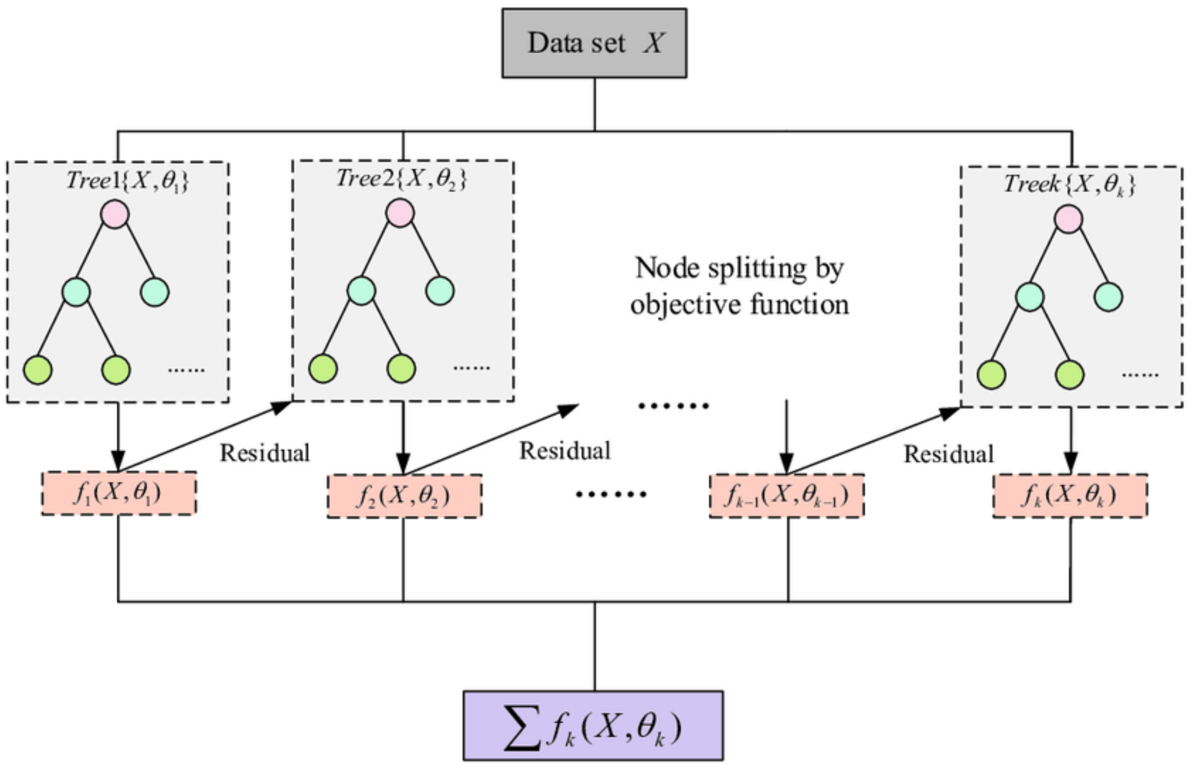
2. Random Forest: An ensemble method that builds on the simplicity of decision trees, random forests improve model accuracy and robustness by creating a ‘forest’ of decision trees and merging their outputs. Each tree in a random forest is built from a random subset of data points and features, leading to high variance but low bias. The final prediction is typically made by averaging the predictions (for regression) or using a majority vote (for classification) from all trees. This technique is effective in reducing overfitting and is highly versatile for various types of data.



3. Neural Network: Neural networks are a set of algorithms modeled loosely after the human brain, designed to recognize patterns. They interpret sensory data through a kind of machine perception, labeling, or clustering raw input. The networks use layers of nodes, or neurons, each of which is a mathematical operation. Data passes through interconnected layers where the outputs of one layer become inputs for the next, thus 'learning' from data features. Neural networks are particularly powerful for complex problems like image recognition, natural language processing, and time series prediction, but require substantial data and computational power.



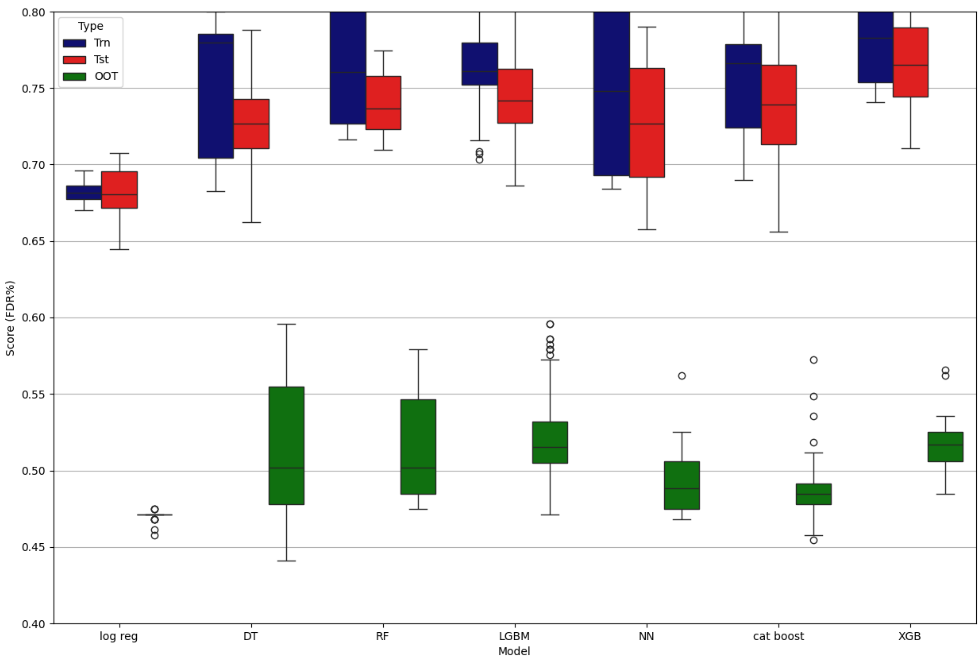
4. XGBoost: XGBoost (Extreme Gradient Boosting) is an advanced implementation of gradient boosting that is both efficient and effective in predictive accuracy. It uses a gradient boosting framework, constructing new models that predict the residuals or errors of prior models and then combining them into a final ensemble model. XGBoost is well-regarded for its performance and speed in training, capabilities of handling various types of predictive modeling problems, and its scalability across multiple scenarios. It has been successfully used in numerous machine learning competitions due to its ability to handle sparse data and its flexibility in tuning model parameters.



# Table of tests:

| Model | | Parameters | | | | | Average FDR at 3% | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Logistic Regression | Iteration | penalty | C | solver | | max_iter | Trn | Tst | OOT | |
| | 1 (default) | l2 | 1 | lbfgs | | 100 | 0.683787 | 0.675251 | 0.464983 | |
| | 2 | l2 | 0.01 | liblinear | | 100 | 0.684388 | 0.681365 | 0.469697 | |
| | 3 | l2 | 0.01 | liblinear | | 50 | 0.685658 | 0.678556 | 0.470707 | |
| | 4 | l1 | 0.01 | liblinear | | 50 | 0.68221 | 0.679498 | 0.472054 | *Best set of hyperparameters |
| | 5 | None | 0.01 | lbfgs | | 100 | 0.681466 | 0.679827 | 0.465657 | |
| Decision Tree | Iteration | max_depth | min_samples_split | min_samples_leaf | criterion | | Trn | Tst | OOT | |
| | 1 (default) | None | 2 | 1 | gini | | 1 | 0.65498 | 0.383838 | |
| | 2 | 5 | 40 | 20 | gini | | 0.709536 | 0.693863 | 0.487879 | |
| | 3 | 5 | 40 | 20 | entropy | | 0.723638 | 0.707039 | 0.475758 | |
| | 4 | 5 | 50 | 25 | entropy | | 0.724742 | 0.712436 | 0.482155 | |
| | 5 | 10 | 50 | 25 | gini | | 0.784493 | 0.744269 | 0.535354 | |
| Random Forest | Iteration | n_estimators | max_depth | min_samples_split | min_samples_leaf | max_features | Trn | Tst | OOT | |
| | 1 (default) | 100 | None | 2 | 1 | sqrt | 1 | 0.815368 | 0.545118 | |
| | 2 | 100 | 10 | 40 | 20 | sqrt | 0.800958 | 0.752882 | 0.529293 | |
| | 3 | 100 | 5 | 40 | 20 | log2 | 0.728275 | 0.721602 | 0.487542 | *Best set of hyperparameters |
| | 4 | 300 | 10 | 40 | 20 | sqrt | 0.793343 | 0.760005 | 0.522896 | |
| | 5 | 200 | 10 | 50 | 25 | sqrt | 0.791589 | 0.745382 | 0.512795 | |
| LightGBM | Iteration | num_leaves | max_depth | learning_rate | n_estimators | | Trn | Tst | OOT | |
| | 1 (default) | 31 | -1 | 0.1 | 100 | | 0.984888 | 0.808584 | 0.510774 | |
| | 2 | 31 | 10 | 0.01 | 100 | | 0.839603 | 0.776373 | 0.536027 | |
| | 3 | 50 | 10 | 0.001 | 100 | | 0.808371 | 0.760915 | 0.512121 | |
| | 4 | 30 | 5 | 0.01 | 100 | | 0.78263 | 0.743032 | 0.547811 | |
| | 5 | 50 | 5 | 0.001 | 300 | | 0.758806 | 0.741943 | 0.52862 | *Best set of hyperparameters |
| Neural Network | Iteration | hidden_layer_sizes | | activation | solver | learning_rate | Trn | Tst | OOT | |
| | 1 (default) | (100,) | | relu | adam | constant | 0.799217 | 0.763179 | 0.523906 | |
| | 2 | (1,) | | relu | adam | constant | 0.684141 | 0.684818 | 0.47037 | |
| | 3 | (100, ) | | relu | sgd | adaptive | 0.694615 | 0.683059 | 0.474074 | *Best set of hyperparameters |
| | 4 | (100,) | | relu | lbfgs | adaptive | 0.808769 | 0.764673 | 0.510438 | |
| | 5 | (100,) | | tanh | adam | constant | 0.810706 | 0.758073 | 0.511111 | |
| Catboost | Iteration | iterations | learning_rate | depth | bootstrap_type | | Trn | Tst | OOT | |
| | 1 (default) | 1000 | None | 6 | Bayesian | | 0.928662 | 0.813864 | 0.543771 | |
| | 2 | 1000 | 0.01 | 6 | Bayesian | | 0.810715 | 0.778627 | 0.513805 | |
| | 3 | 1000 | 0.01 | 6 | Bernoulli | | 0.812551 | 0.783077 | 0.514478 | |
| | 4 | 500 | 0.01 | 3 | Bayesian | | 0.725214 | 0.718544 | 0.482828 | *Best set of hyperparameters |
| | 5 | 1000 | 0.1 | 3 | Bayesian | | 0.837019 | 0.797159 | 0.525926 | |
| XGB | Iteration | booster | n_estimators | max_depth | learning_rate | | Trn | Tst | OOT | |
| | 1 (default) | gbtree | 100 | 6 | 0.3 | | 0.980182 | 0.819913 | 0.503704 | |
| | 2 | gblinear | 100 | 6 | 0.3 | | 0.681479 | 0.681252 | 0.465657 | |
| | 3 | gbtree | 100 | 5 | 0.01 | | 0.754919 | 0.730523 | 0.518855 | *Best set of hyperparameters |
| | 4 | gbtree | 200 | 3 | 0.1 | | 0.816518 | 0.787114 | 0.50404 | |
| | 5 | gbtree | 200 | 3 | 0.01 | | 0.722687 | 0.702626 | 0.486532 | |

# Box plot:



# Final model performance

For the final model in the project, an XGBoost classifier was employed, which is well-suited for handling large datasets and providing robust predictive power. Here's a detailed description of the model configuration and the non-default hyperparameters used:

Final Model: XGBoost Classifier

Hyperparameters:

1. booster: 'gbtree'
   Description: Specifies the type of model to run at each iteration. 'gbtree' uses tree-based models as base learners. This is the default setting for XGBoost but is explicitly stated here to clarify the model choice.

2. n_estimators: 70
   Description: The number of boosting rounds or trees to build. Though the default is typically set around 100, it was adjusted to 70 in this model to balance overfitting and underfitting, optimizing the model's complexity and computational efficiency.

3. max_depth: 3
   Description: The maximum depth of a tree. Limiting the depth to 3 helps prevent the model from becoming overly complex and overfitting to the training data. This is shallower than the default depth to ensure the model generalizes well over unseen data.

4. learning_rate: 0.1
   Description: Also known as the "eta" parameter, the learning rate shrinks the feature weights to make the boosting process more conservative. A rate of 0.1 reduces the risk of overfitting and improves the final model's robustness. The default is typically set at 0.3, so this represents a more conservative approach to updating weights.

Summary tables:

1. Training

| Training | # Records | | # Goods | | # Bads | | Fraud Rate | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 59684 | | 58467 | | 1217 | | 0.020390724 | | | | | |
| | Bin Statistics | | | | | Cumulative Statistics | | | | | | |
| Population Bin % | # Records | # Goods | # Bads | % Goods | % Bads | Total # Records | Cumulative Goods | Cumulative Bads | % Goods | % Bads (FDR) | KS | FPR |
| 1 | 597 | 43 | 554 | 7.20% | 92.80% | 597 | 43 | 554 | 0.07% | 45.52% | 45.45 | 0.08 |
| 2 | 597 | 291 | 306 | 48.74% | 51.26% | 1194 | 334 | 860 | 0.57% | 70.67% | 70.09 | 0.39 |
| 3 | 597 | 496 | 101 | 83.08% | 16.92% | 1791 | 830 | 961 | 1.42% | 78.96% | 77.55 | 0.86 |
| 4 | 596 | 557 | 39 | 93.46% | 6.54% | 2387 | 1387 | 1000 | 2.37% | 82.17% | 79.8 | 1.39 |
| 5 | 597 | 580 | 17 | 97.15% | 2.85% | 2984 | 1967 | 1017 | 3.36% | 83.57% | 80.2 | 1.93 |
| 6 | 597 | 581 | 16 | 97.32% | 2.68% | 3581 | 2548 | 1033 | 4.36% | 84.88% | 80.52 | 2.47 |
| 7 | 597 | 583 | 14 | 97.65% | 2.35% | 4178 | 3131 | 1047 | 5.36% | 86.03% | 80.68 | 2.99 |
| 8 | 597 | 582 | 15 | 97.49% | 2.51% | 4775 | 3713 | 1062 | 6.35% | 87.26% | 80.91 | 3.5 |
| 9 | 597 | 585 | 12 | 97.99% | 2.01% | 5372 | 4298 | 1074 | 7.35% | 88.25% | 80.9 | 4 |
| 10 | 596 | 582 | 14 | 97.65% | 2.35% | 5968 | 4880 | 1088 | 8.35% | 89.40% | 81.05 | 4.49 |
| 11 | 597 | 581 | 16 | 97.32% | 2.68% | 6565 | 5461 | 1104 | 9.34% | 90.71% | 81.37 | 4.95 |
| 12 | 597 | 589 | 8 | 98.66% | 1.34% | 7162 | 6050 | 1112 | 10.35% | 91.37% | 81.02 | 5.44 |
| 13 | 597 | 591 | 6 | 98.99% | 1.01% | 7759 | 6641 | 1118 | 11.36% | 91.87% | 80.51 | 5.94 |
| 14 | 597 | 590 | 7 | 98.83% | 1.17% | 8356 | 7231 | 1125 | 12.37% | 92.44% | 80.07 | 6.43 |
| 15 | 597 | 592 | 5 | 99.16% | 0.84% | 8953 | 7823 | 1130 | 13.38% | 92.85% | 79.47 | 6.92 |
| 16 | 596 | 595 | 1 | 99.83% | 0.17% | 9549 | 8418 | 1131 | 14.40% | 92.93% | 78.54 | 7.44 |
| 17 | 597 | 594 | 3 | 99.50% | 0.50% | 10146 | 9012 | 1134 | 15.41% | 93.18% | 77.77 | 7.95 |
| 18 | 597 | 595 | 2 | 99.66% | 0.34% | 10743 | 9607 | 1136 | 16.43% | 93.34% | 76.91 | 8.46 |
| 19 | 597 | 594 | 3 | 99.50% | 0.50% | 11340 | 10201 | 1139 | 17.45% | 93.59% | 76.14 | 8.96 |
| 20 | 597 | 589 | 8 | 98.66% | 1.34% | 11937 | 10790 | 1147 | 18.45% | 94.25% | 75.79 | 9.41 |

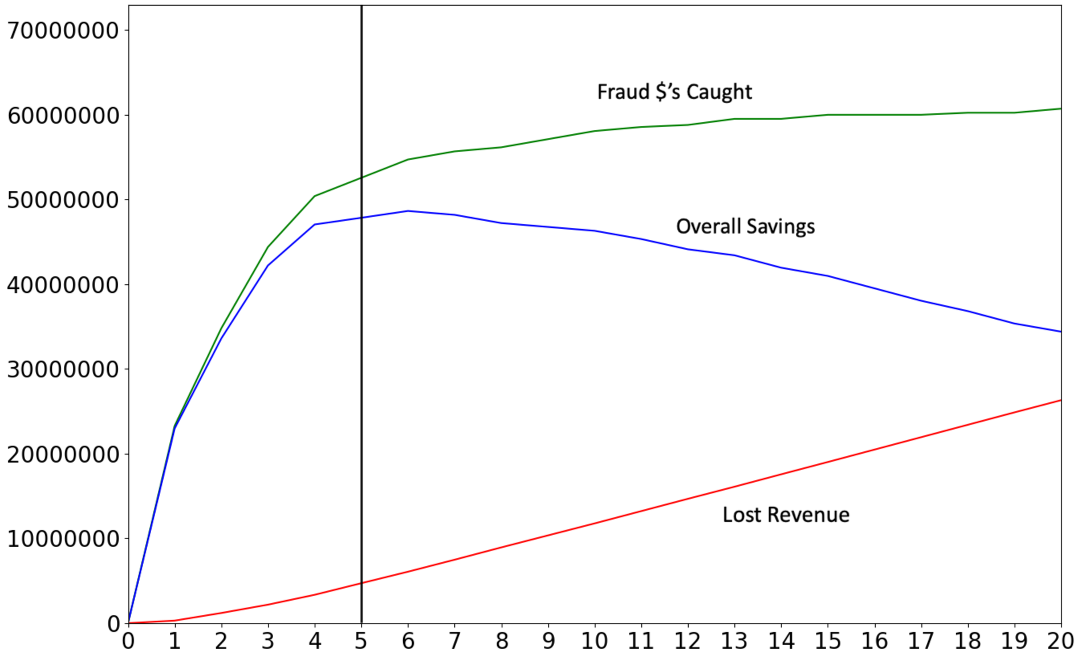
## 2. Testing

| Testing | # Records | | # Goods | | # Bads | | Fraud Rate | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 25580 | | 25047 | | 533 | | 0.020836591 | | | | |
| | Bin Statistics | | | | | Cumulative Statistics | | | | | |
| Population Bin % | # Records | # Goods | # Bads | % Goods | % Bads | Total # Records | Cumulative Goods | Cumulative Bads | % Goods | % Bads (FDR) | KS | FPR |
| 1 | 256 | 37 | 219 | 14.45% | 85.55% | 256 | 37 | 219 | 0.15% | 41.09% | 40.94 | 0.17 |
| 2 | 256 | 130 | 126 | 50.78% | 49.22% | 512 | 167 | 345 | 0.67% | 64.73% | 64.06 | 0.48 |
| 3 | 255 | 206 | 49 | 80.78% | 19.22% | 767 | 373 | 394 | 1.49% | 73.92% | 72.43 | 0.95 |
| 4 | 256 | 240 | 16 | 93.75% | 6.25% | 1023 | 613 | 410 | 2.45% | 76.92% | 74.48 | 1.5 |
| 5 | 256 | 241 | 15 | 94.14% | 5.86% | 1279 | 854 | 425 | 3.41% | 79.74% | 76.33 | 2.01 |
| 6 | 256 | 247 | 9 | 96.48% | 3.52% | 1535 | 1101 | 434 | 4.40% | 81.43% | 77.03 | 2.54 |
| 7 | 256 | 247 | 9 | 96.48% | 3.52% | 1791 | 1348 | 443 | 5.38% | 83.11% | 77.73 | 3.04 |
| 8 | 255 | 250 | 5 | 98.04% | 1.96% | 2046 | 1598 | 448 | 6.38% | 84.05% | 77.67 | 3.57 |
| 9 | 256 | 249 | 7 | 97.27% | 2.73% | 2302 | 1847 | 455 | 7.37% | 85.37% | 77.99 | 4.06 |
| 10 | 256 | 249 | 7 | 97.27% | 2.73% | 2558 | 2096 | 462 | 8.37% | 86.68% | 78.31 | 4.54 |
| 11 | 256 | 252 | 4 | 98.44% | 1.56% | 2814 | 2348 | 466 | 9.37% | 87.43% | 78.06 | 5.04 |
| 12 | 256 | 254 | 2 | 99.22% | 0.78% | 3070 | 2602 | 468 | 10.39% | 87.80% | 77.42 | 5.56 |
| 13 | 255 | 254 | 1 | 99.61% | 0.39% | 3325 | 2856 | 469 | 11.40% | 87.99% | 76.59 | 6.09 |
| 14 | 256 | 255 | 1 | 99.61% | 0.39% | 3581 | 3111 | 470 | 12.42% | 88.18% | 75.76 | 6.62 |
| 15 | 256 | 251 | 5 | 98.05% | 1.95% | 3837 | 3362 | 475 | 13.42% | 89.12% | 75.7 | 7.08 |
| 16 | 256 | 256 | 0 | 100.00% | 0.00% | 4093 | 3618 | 475 | 14.44% | 89.12% | 74.67 | 7.62 |
| 17 | 256 | 256 | 0 | 100.00% | 0.00% | 4349 | 3874 | 475 | 15.47% | 89.12% | 73.65 | 8.16 |
| 18 | 255 | 251 | 4 | 98.43% | 1.57% | 4604 | 4125 | 479 | 16.47% | 89.87% | 73.4 | 8.61 |
| 19 | 256 | 253 | 3 | 98.83% | 1.17% | 4860 | 4378 | 482 | 17.48% | 90.43% | 72.95 | 9.08 |
| 20 | 256 | 254 | 2 | 99.22% | 0.78% | 5116 | 4632 | 484 | 18.49% | 90.81% | 72.31 | 9.57 |

## 3. OOT

| OOT | # Records | | # Goods | | # Bads | | Fraud Rate | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 12232 | | 11935 | | 297 | | 0.024280576 | | | | | |
| | Bin Statistics | | | | | Cumulative Statistics | | | | | | |
| Population Bin % | # Records | # Goods | # Bads | % Goods | % Bads | Total # Records | Cumulative Goods | Cumulative Bads | % Goods | % Bads (FDR) | KS | FPR |
| 1 | 122 | 25 | 97 | 20.49% | 79.51% | 122 | 25 | 97 | 0.21% | 32.66% | 32.45 | 0.26 |
| 2 | 123 | 75 | 48 | 60.98% | 39.02% | 245 | 100 | 145 | 0.84% | 48.82% | 47.98 | 0.69 |
| 3 | 122 | 82 | 40 | 67.21% | 32.79% | 367 | 182 | 185 | 1.52% | 62.29% | 60.76 | 0.98 |
| 4 | 122 | 97 | 25 | 79.51% | 20.49% | 489 | 279 | 210 | 2.34% | 70.71% | 68.37 | 1.33 |
| 5 | 123 | 114 | 9 | 92.68% | 7.32% | 612 | 393 | 219 | 3.29% | 73.74% | 70.44 | 1.79 |
| 6 | 122 | 113 | 9 | 92.62% | 7.38% | 734 | 506 | 228 | 4.24% | 76.77% | 72.53 | 2.22 |
| 7 | 122 | 118 | 4 | 96.72% | 3.28% | 856 | 624 | 232 | 5.23% | 78.11% | 72.89 | 2.69 |
| 8 | 123 | 121 | 2 | 98.37% | 1.63% | 979 | 745 | 234 | 6.24% | 78.79% | 72.55 | 3.18 |
| 9 | 122 | 118 | 4 | 96.72% | 3.28% | 1101 | 863 | 238 | 7.23% | 80.13% | 72.90 | 3.63 |
| 10 | 122 | 118 | 4 | 96.72% | 3.28% | 1223 | 981 | 242 | 8.22% | 81.48% | 73.26 | 4.05 |
| 11 | 123 | 121 | 2 | 98.37% | 1.63% | 1346 | 1102 | 244 | 9.23% | 82.15% | 72.92 | 4.52 |
| 12 | 122 | 121 | 1 | 99.18% | 0.82% | 1468 | 1223 | 245 | 10.25% | 82.49% | 72.24 | 4.99 |
| 13 | 122 | 119 | 3 | 97.54% | 2.46% | 1590 | 1342 | 248 | 11.24% | 83.50% | 72.26 | 5.41 |
| 14 | 122 | 122 | 0 | 100.00% | 0.00% | 1712 | 1464 | 248 | 12.27% | 83.50% | 71.24 | 5.90 |
| 15 | 123 | 121 | 2 | 98.37% | 1.63% | 1835 | 1585 | 250 | 13.28% | 84.18% | 70.89 | 6.34 |
| 16 | 122 | 122 | 0 | 100.00% | 0.00% | 1957 | 1707 | 250 | 14.30% | 84.18% | 69.87 | 6.83 |
| 17 | 122 | 122 | 0 | 100.00% | 0.00% | 2079 | 1829 | 250 | 15.32% | 84.18% | 68.85 | 7.32 |
| 18 | 123 | 122 | 1 | 99.19% | 0.81% | 2202 | 1951 | 251 | 16.35% | 84.51% | 68.16 | 7.77 |
| 19 | 122 | 122 | 0 | 100.00% | 0.00% | 2324 | 2073 | 251 | 17.37% | 84.51% | 67.14 | 8.26 |
| 20 | 122 | 120 | 2 | 98.36% | 1.64% | 2446 | 2193 | 253 | 18.37% | 85.19% | 66.81 | 8.67 |

# Financial curves




---

Recommended cutoff:
Based on the plot, a recommended cutoff could be around 5%. This point maximizes savings while controlling costs, making it an optimal trade-off point.

Description of the logic:
The chosen cutoff is recommended due to its optimal balance between maximizing cost savings (as seen in the blue curve) and minimizing losses or costs (as seen in the red curve). The green curve supports this choice by showing that increases beyond this point do not yield significant additional benefits. This cutoff ensures that the model effectively identifies fraudulent transactions while maintaining operational efficiency and cost-effectiveness.

## Summary

The project entailed developing a predictive model specifically tailored to detect fraudulent transactions within credit card operations. The process began with a thorough data description phase where transaction data was meticulously analyzed, including both numerical and categorical attributes related to transaction amounts, merchant details, and fraud indicators. Following this, the data cleaning phase addressed issues such as outlier removal and imputation for missing values in fields like Merchnum, Merch state, and Merch zip, ensuring data integrity for modeling.

Variable creation was strategically undertaken to enrich the dataset, including the development of variables to capture temporal patterns, risk scores, and transaction behaviors, which are critical for identifying fraud. The feature selection was robust, utilizing filtering and multiple wrapper methods, including Random Forest, LightGBM, and Catboost, to refine the variable set to those most predictive of fraud.

Model exploration involved evaluating several machine learning models, ultimately selecting the XGBoost model due to its superior performance. The chosen model was fine-tuned with specific hyperparameters such as the number of estimators and maximum depth, ensuring optimal model complexity and performance.

Model performance:
The final model, an XGBoost classifier, was configured with carefully selected hyperparameters to balance the detection of fraudulent transactions against the risk of overfitting. The model demonstrated high predictive accuracy, with a rigorous application of a 3% false discovery rate in out-of-time validation samples, reflecting its robustness and reliability. The implementation of this model is projected to save approximately $48 million annually by enhancing fraud detection capabilities and optimizing resource allocation. The performance metrics from the training, testing, and out-of-time datasets underscored the model's effectiveness across various scenarios, confirming its practical utility in operational environments.

This comprehensive approach, from data preparation through to final model selection and validation, exemplifies a structured and data-driven methodology for tackling fraud detection in financial transactions.

# Appendix

Data Description: The dataset contains transaction records from card payments, capturing a wide array of attributes including transaction amounts, merchant details, and fraud indicators. The data comes from real-world financial transactions over 1 year and includes both numerical and categorical fields. It contains 10 fields and 97,852 records and is designed for analytical exploration and fraud detection model development.

Summary Tables:

1. Numeric Fields Table

| | Field Name | Field Type | # Records Have Values | % Populated | # Zeros | Min | Max | Mean | Standard Deviation | Most Common |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Amount | numeric | 97852 | 100.0% | 0 | 0.01 | 3102045.53 | 425.466438 | 9949.8 | 3.62 |

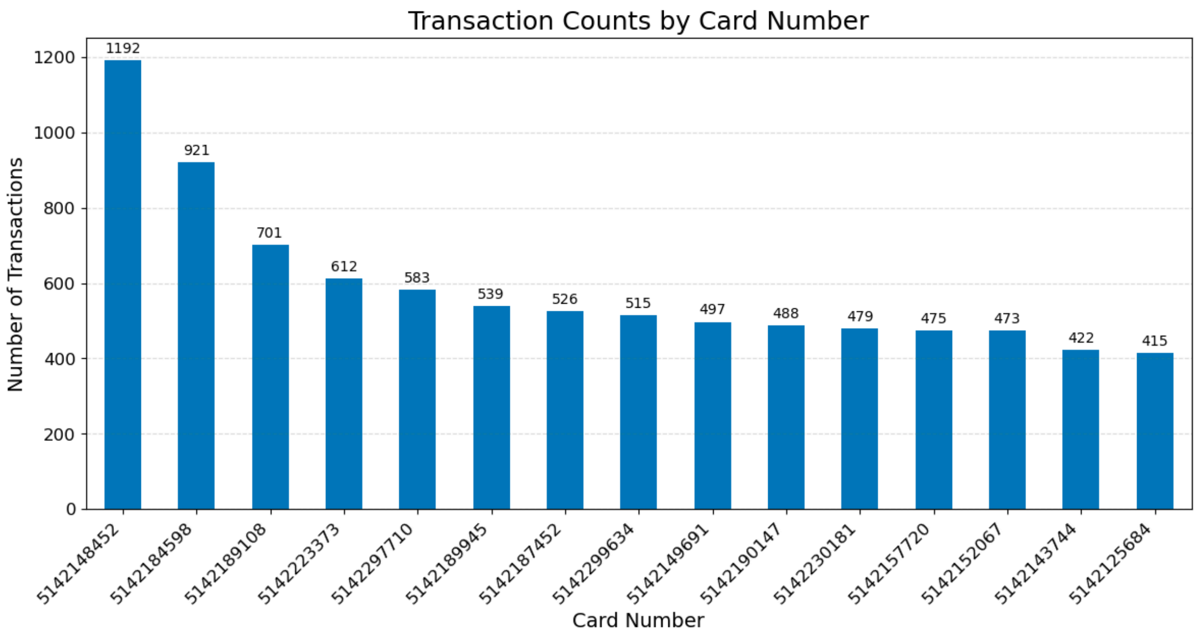
2. Categorical Fields Table

| | Field Name | Field Type | # Records Have Values | % Populated | # Zeros | # Unique Values | Most Common |
|---|---|---|---|---|---|---|---|
| 0 | Date | categorical | 97852 | 100.0% | 0 | 365 | 2/28/10 |
| 1 | Merchnum | categorical | 94455 | 96.5% | 0 | 13091 | 930090121224 |
| 2 | Merch description | categorical | 97852 | 100.0% | 0 | 13126 | GSA-FSS-ADV |
| 3 | Merch state | categorical | 96649 | 98.8% | 0 | 227 | TN |
| 4 | Transtype | categorical | 97852 | 100.0% | 0 | 4 | P |
| 5 | Recnum | categorical | 97852 | 100.0% | 0 | 97852 | 1 |
| 6 | Fraud | categorical | 97852 | 100.0% | 95805 | 2 | 0 |
| 7 | Cardnum | categorical | 97852 | 100.0% | 0 | 1645 | 5142148452 |
| 8 | Merch zip | categorical | 93149 | 95.2% | 0 | 4567 | 38118 |

Visualization of Each Field:

1) Field Name: Recnum
   Description: Ordinal unique positive integer for each transaction record, from 1 to 97,852.

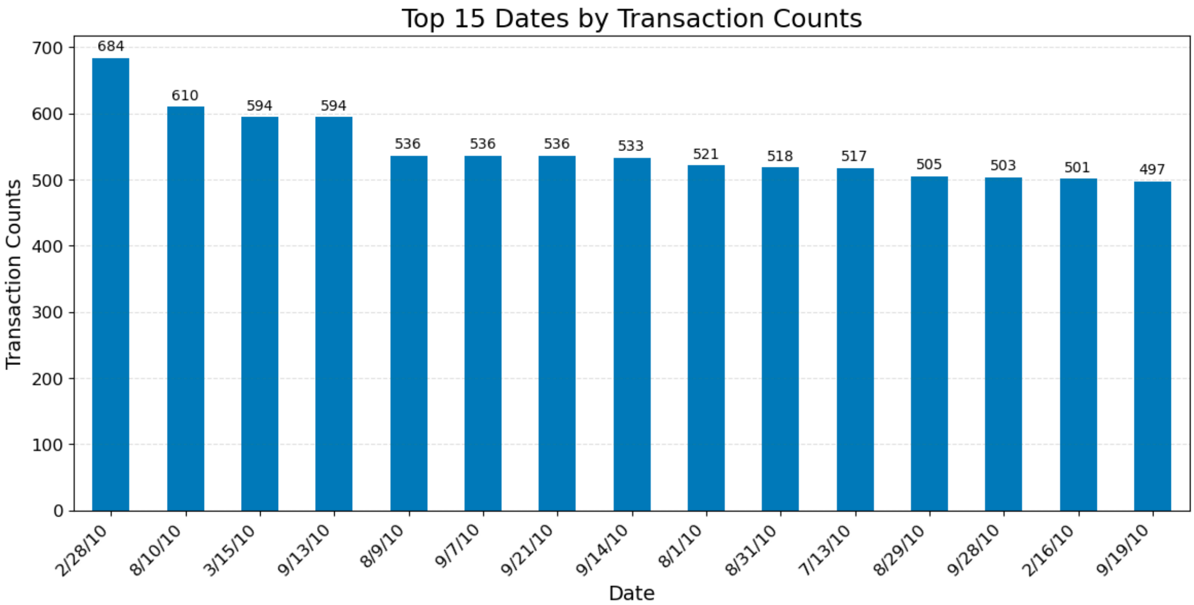
2) Field Name: Cardnum
   Description: Applicant's card number. The distribution shows the top 15 field values of card numbers. The most used card number is 5142148452, with a total count of 1,192.

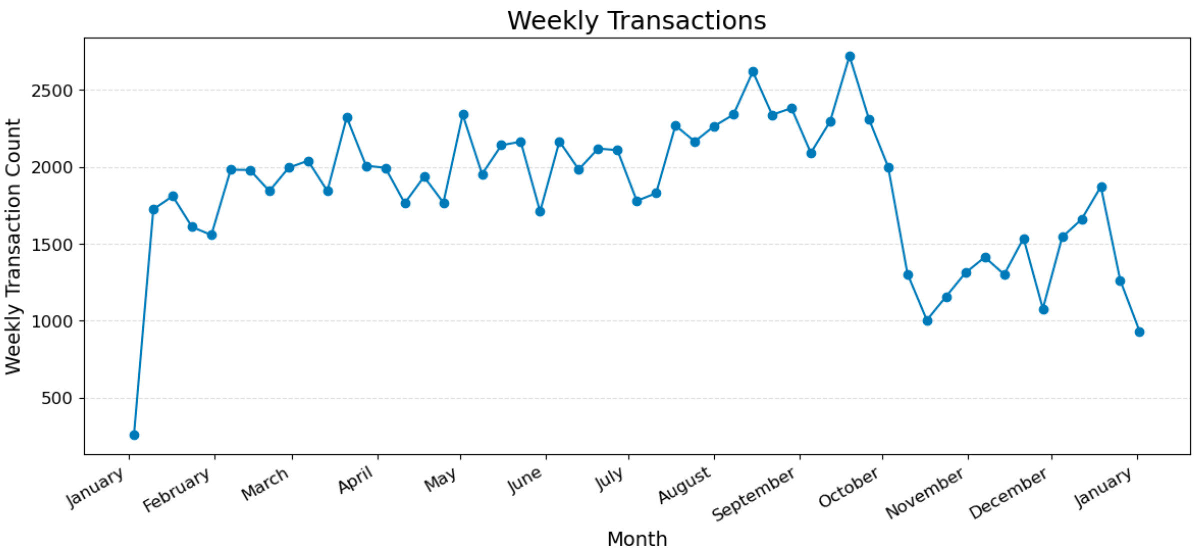


3) Field Name: Date

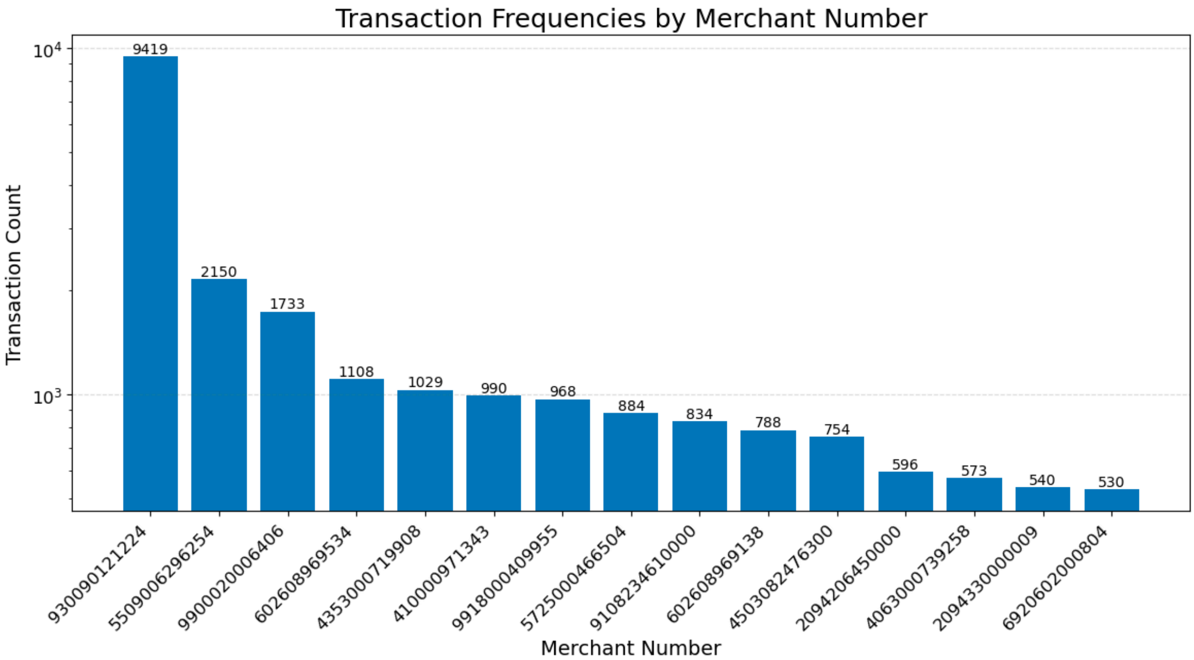
Description: Date of the transaction, spanning a period from 1st January 2010 to 31st December 2010, with the most transactions recorded on 28th February.

The first graph shows the top 15 dates when transactions occurred. The second graph visualizes the number of weekly transactions over time.

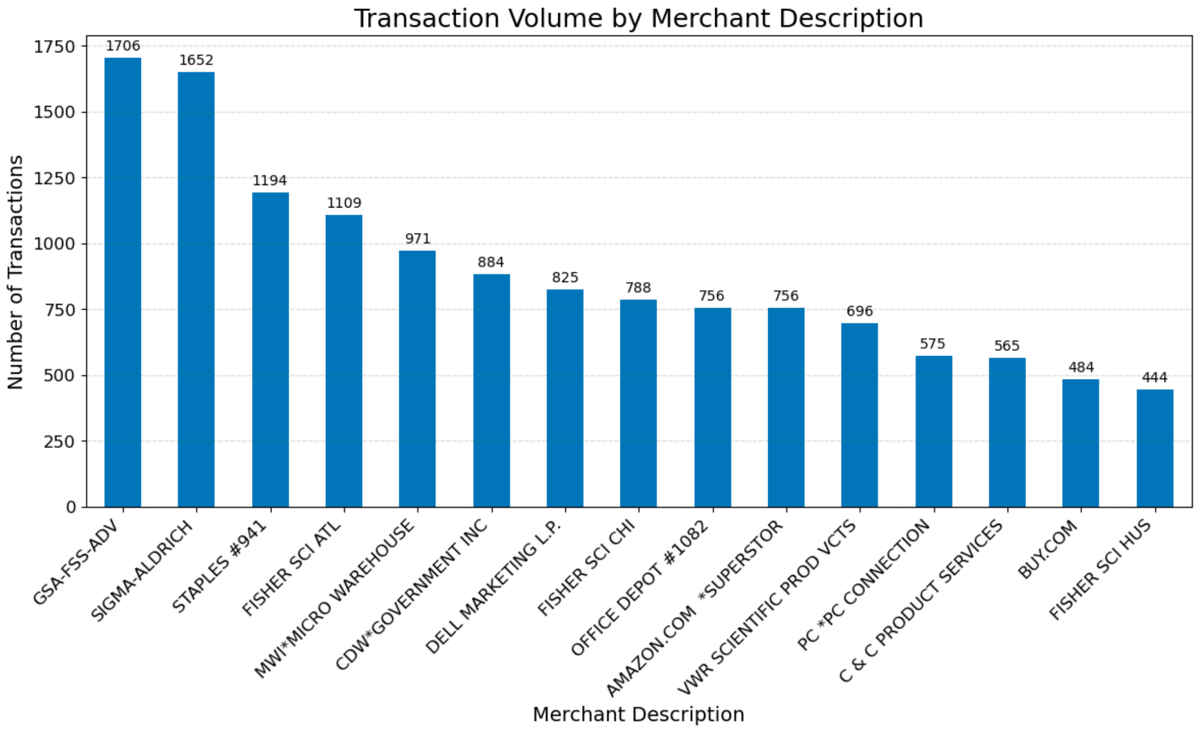


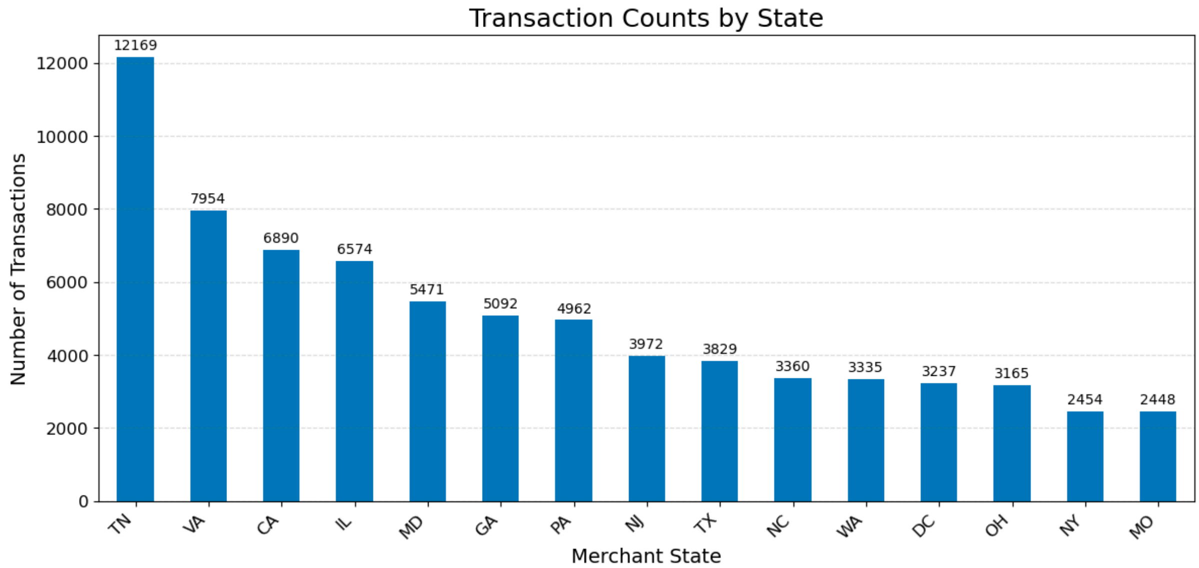
4) Field Name: Merchnum
Description: Merchant number. The distribution shows the top 15 merchants that received the most transactions by merchant number, and the most frequent merchant number having 9,419 transactions.



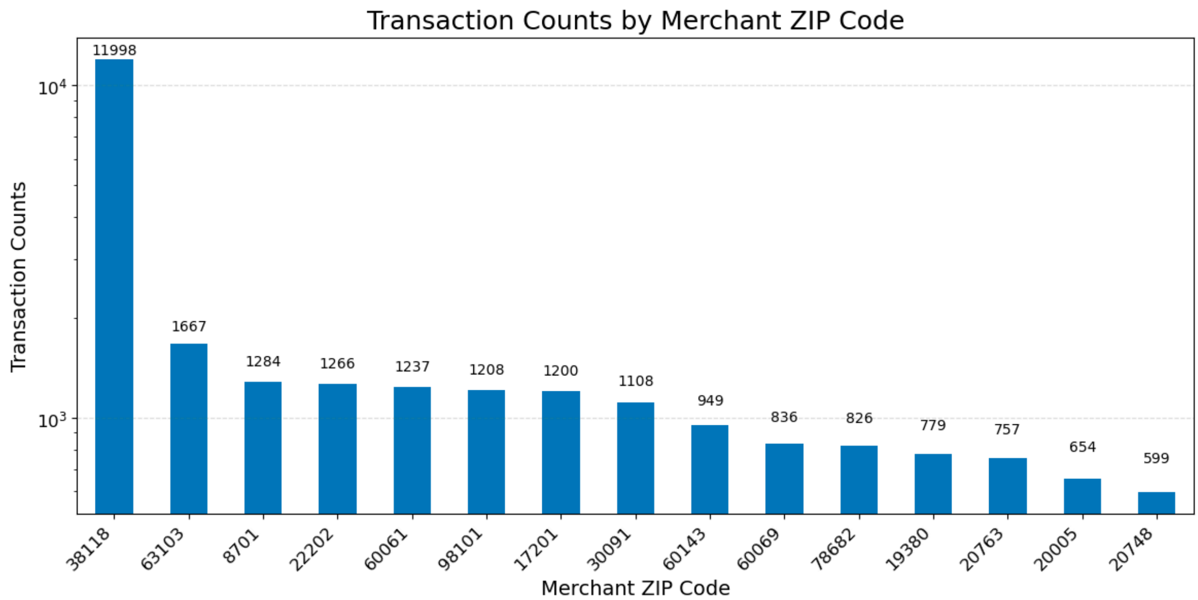
5) Field Name: Merch description
Description: Description of the merchant. The distribution shows the top 15 merchants that received the most transactions by merchant description, with 'GSA-FSS-ADV' appearing most frequently at 1,706 times.



6) Field Name: Merch state
Description: State where the merchant is located, with transactions occurring across 227 different states, most commonly in Tennessee.

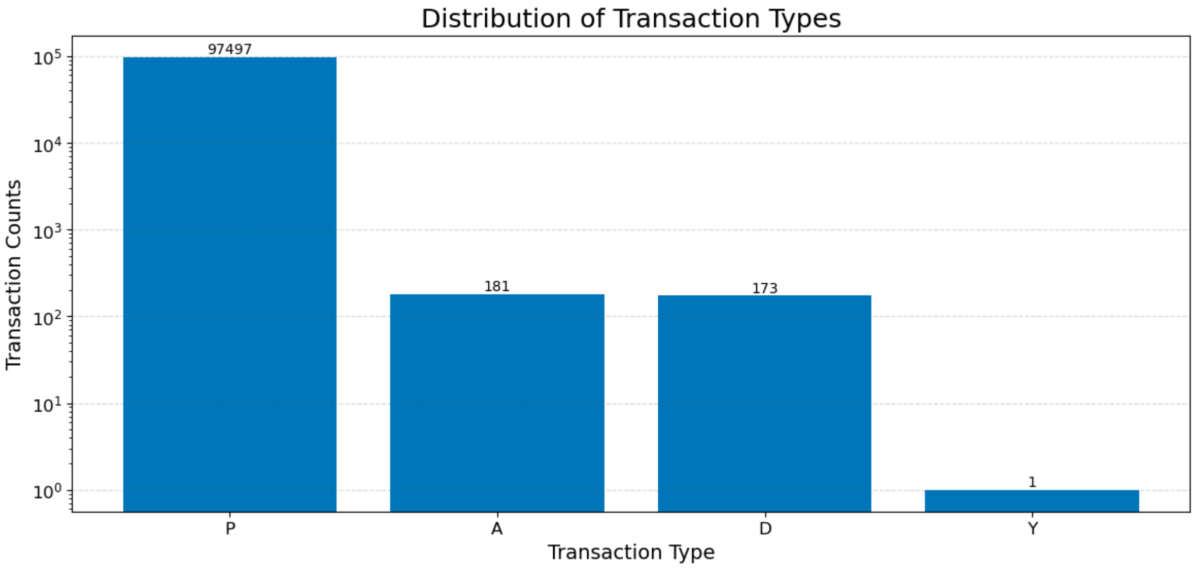



7) Field Name: Merch zip
Description: ZIP code of the merchant, with 4,567 unique ZIP codes in the dataset, indicating a wide geographical spread.



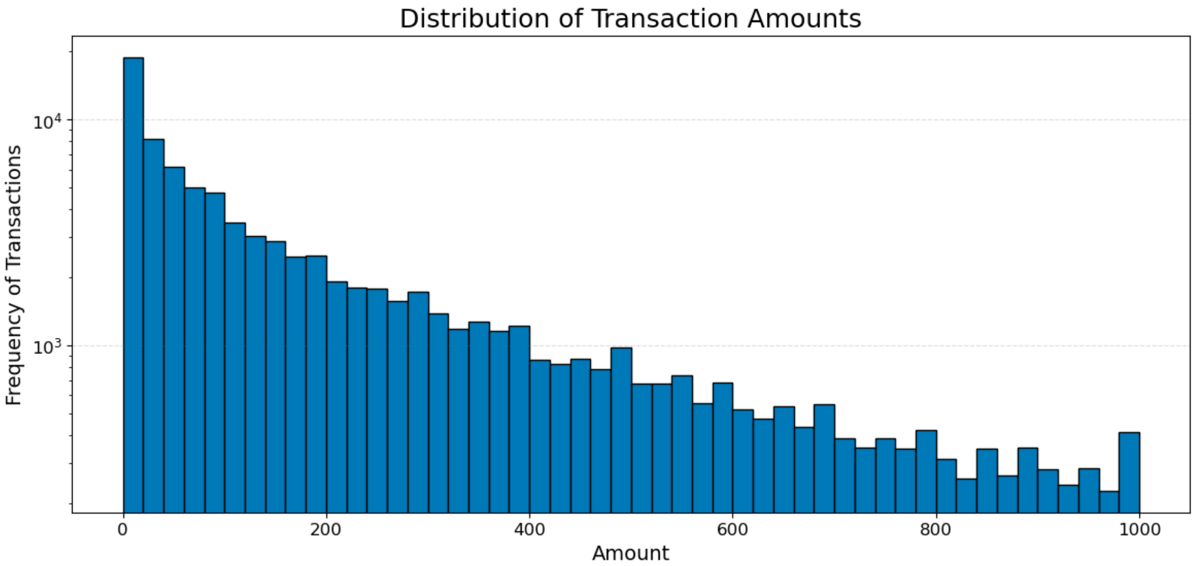
8) Field Name: Transtype

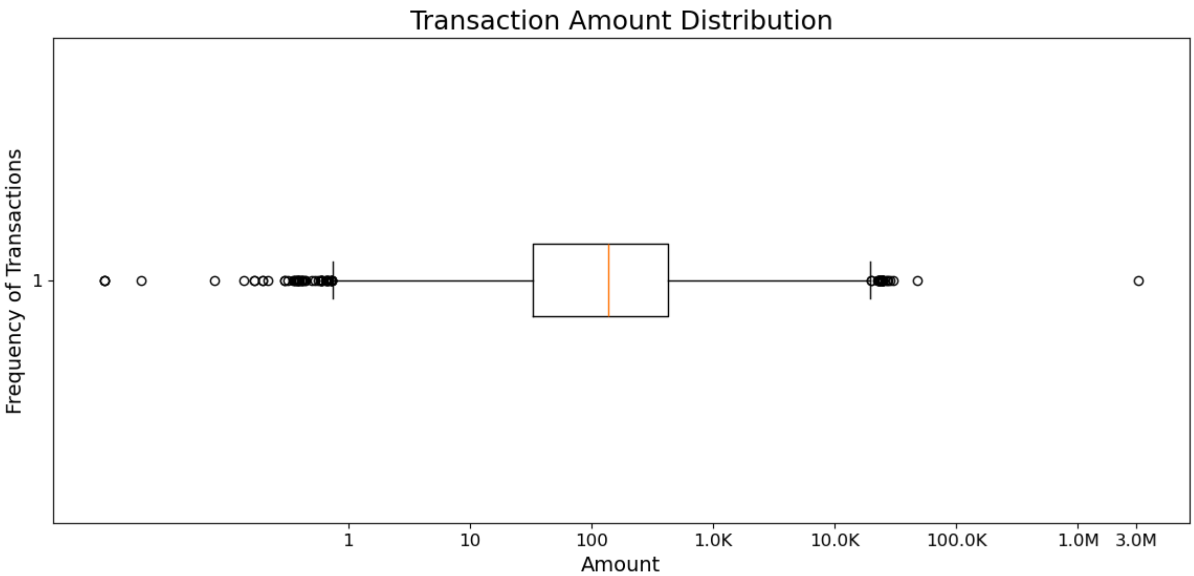
Description: Type of transactions with four unique types. The 'P' or purchase type is by far the most common, with over 97,000 transaction records.



9) Field Name: Amount

Description: The dollar amount of the transaction varies widely from as little as $0.01 to over $3 million, showcasing a vast range of transaction values. The first graph shows the distribution of transaction amounts ranging from $0 to $1000 since the maximum number of transactions occurred in this range. The second graph shows a box plot with several high-value outliers.



10) Field Name: Fraud

Description: Fraud identification label. Fraud = 0 (Not fraudulent), Fraud = 1 (Fraudulent). The total count of fraud = 0 is 95,805. The total count of fraud = 1 is 2,047.