WEEK-6 FINAL REPORT

# Analysis on Bati Bank

JUNE 9, 2024

BATI-BANK
Yodahe Teshome (https://github.com/jodahe1/Bati-Bank.git)

# Bati Bank

## The Business Objective

The business objective of the project is to create a Credit Scoring Model for Bati Bank's partnership with an eCommerce company. The objective is to enable a buy-now-pay-later service for customers, providing them with the ability to buy products on credit if they qualify for the service. The specific goals of the project are as follows:

Define a Proxy Variable: Create a proxy variable that can categorize users as high risk (bad) or low risk (good) based on their creditworthiness. This variable will serve as a measure of the likelihood of default.

Select Predictive Features: Identify observable features that demonstrate a high correlation with the proxy variable defined in the previous step. These features will be used as inputs to the credit scoring model.

Develop Risk Probability Model: Build a model that assigns a risk probability for a new customer based on the selected features. This model will estimate the likelihood of default for each customer, helping the bank evaluate their creditworthiness.

Develop Credit Scoring Model: Create a model that assigns a credit score to each customer based on the risk probability estimates. The credit score will provide a standardized measure of the customer's creditworthiness and help in the decision-making process for loan approvals.

Optimal Loan Amount and Duration Prediction: Develop a model that predicts the optimal amount and duration of the loan for each customer. This model will consider the risk probability, credit score, and other relevant factors to determine the suitable loan terms for the customer.

So Overall aim is to leverage data analytics and statistical techniques to assess the creditworthiness of potential borrowers accurately. By implementing the Credit Scoring Model, Bati Bank aims to make informed decisions regarding loan approvals, minimize the risk of defaults, and provide a reliable buy-now-pay-later service to customers in collaboration with the eCommerce platform.

## 1. Introduction to Credit Risk

Credit risk refers to the potential that a borrower or counterparty will fail to meet its obligations in accordance with agreed terms. This risk is inherent in any financial institution's lending and investment activities and is managed to ensure the institution's stability and profitability.

## 2. Credit Risk Management Framework

According to the Basel III framework, the objective of credit risk management is to maximize a bank's risk-adjusted rate of return by maintaining credit risk exposure within acceptable parameters. This involves managing the credit risk in individual credits or transactions, as well as the entire portfolio.

Key elements of a robust credit risk management framework include:

- **Credit Risk Assessment:** Evaluating the creditworthiness of counterparties.
- **Credit Risk Monitoring:** Ongoing tracking of credit exposures.
- **Credit Risk Mitigation:** Utilizing collateral, guarantees, and other measures.
- **Stress Testing:** Assessing the impact of adverse scenarios on credit risk exposure.

## 3. Credit Scoring and Modeling Approaches

Credit scoring models are essential tools for assessing the credit risk of individual borrowers. These models use statistical techniques to predict the probability of default (PD) based on historical data and borrower characteristics.

### Traditional Credit Scoring Models

Traditional models, such as logistic regression, use factors like credit history, income, and employment status to predict default risk. These models are widely used due to their simplicity and interpretability.

### Alternative Credit Scoring Models

Emerging approaches leverage alternative data sources, such as social media activity, utility payments, and mobile phone usage. These models employ machine learning techniques to enhance prediction accuracy, especially for individuals with limited credit histories.

## 4. Developing a Credit Risk Model and Scorecard (Towards Data Science)

A credit risk model and scorecard can be developed using the following steps:

- **Data Collection:** Gather relevant data on borrowers, including traditional financial metrics and alternative data.
- **Data Preprocessing:** Clean and preprocess the data to handle missing values and outliers.
- **Feature Selection:** Identify the most relevant features for predicting default.
- **Model Building:** Use statistical or machine learning models to develop the credit risk model.
- **Model Validation:** Validate the model using out-of-sample testing to ensure its accuracy and robustness.
- **Scorecard Development:** Convert the model output into a scorecard format that can be used for decision-making.

## 5. Case Studies and Guidelines

### World Bank Guidelines

The World Bank provides comprehensive guidelines on credit scoring approaches, emphasizing the importance of using both traditional and alternative data sources. These guidelines highlight best practices in model development, validation, and implementation to ensure the models are fair, accurate, and robust.

### Hong Kong Monetary Authority (HKMA)

The HKMA discusses the potential of alternative credit scoring models in financial inclusion. These models can help extend credit to underserved populations by utilizing non-traditional data sources. The HKMA emphasizes the need for regulatory oversight to ensure these models are used responsibly.

### Statistica Sinica Study

A study published in Statistica Sinica explores advanced statistical methods for credit risk modeling. The paper highlights the use of complex algorithms and machine learning techniques to improve prediction accuracy and model performance.

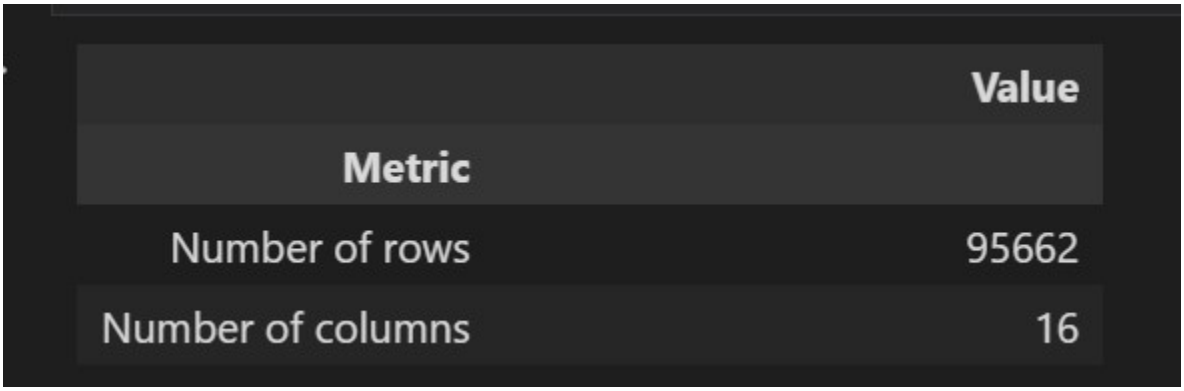
## 6. Conclusion

Effective credit risk management and credit scoring are critical for the stability and profitability of financial institutions. By leveraging traditional methods and embracing alternative data and advanced modeling techniques, institutions can enhance their credit risk assessment capabilities, improve decision-making, and promote financial inclusion.

**Bati Bank**

# EDA

First let's see the rows and columns we got to understand the data better. As we see on the image, we have 95662 rows and 16 columns.



| Metric | Value |
|---|---|
| Number of rows | 95662 |
| Number of columns | 16 |

Then I try to remove non numeric values from the columns mentioned on the image.

Extract Only Numeric Parts From Columns

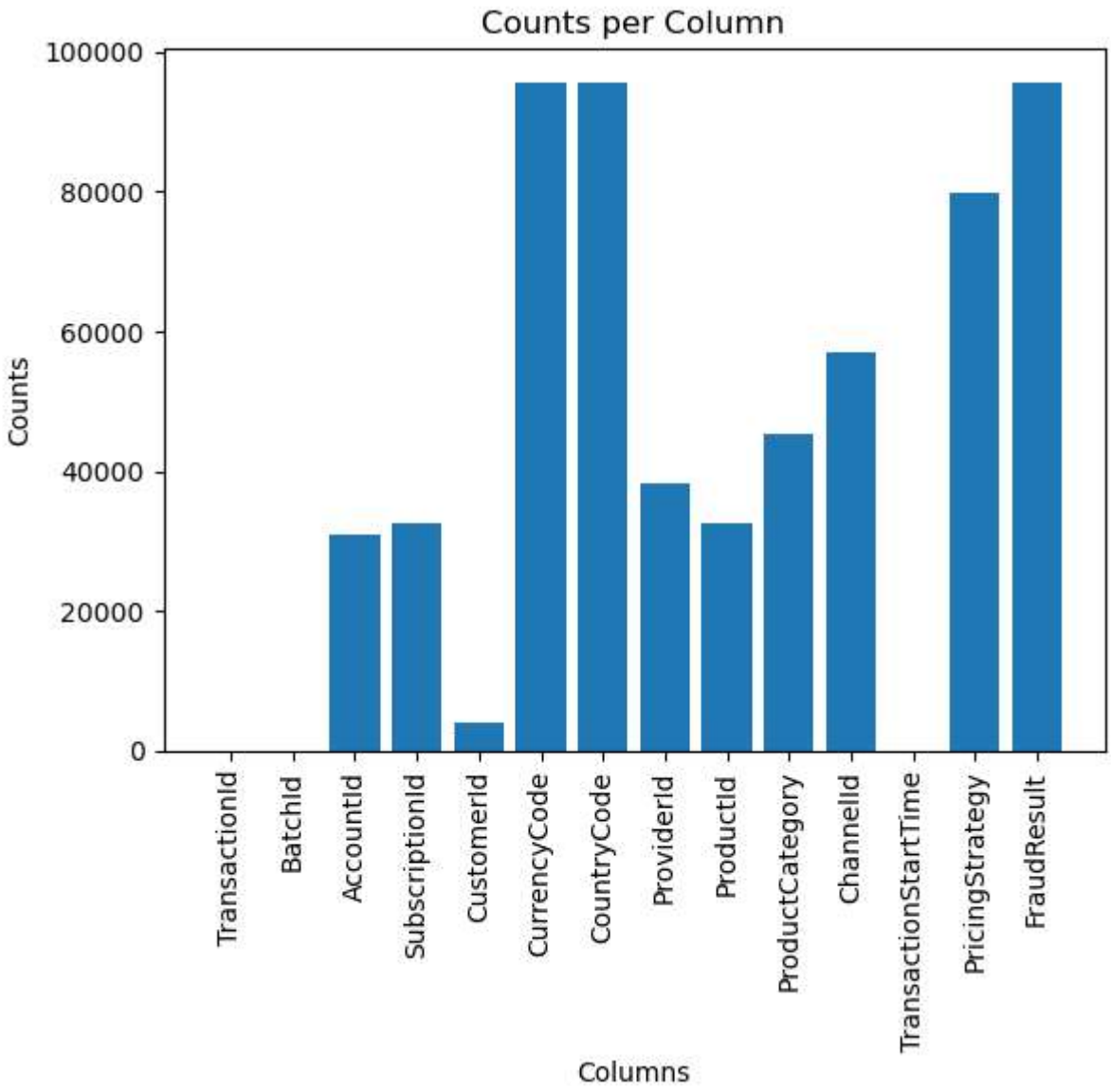
+ Code + Markdown

```
df['TransactionId'] = df['TransactionId'].str.extract('(\d+)') # Extract only numeric part
df['BatchId'] = df['BatchId'].str.extract('(\d+)') # Extract only numeric part
df['AccountId'] = df['AccountId'].str.extract('(\d+)') # Extract only numeric part
df['SubscriptionId'] = df['SubscriptionId'].str.extract('(\d+)') # Extract only numeric part
df['CustomerId'] = df['CustomerId'].str.extract('(\d+)') # Extract only numeric part
df['ProviderId'] = df['ProviderId'].str.extract('(\d+)') # Extract only numeric part
df['ProductId'] = df['ProductId'].str.extract('(\d+)') # Extract only numeric part
df['ChannelId'] = df['ChannelId'].str.extract('(\d+)') # Extract only numeric part
```

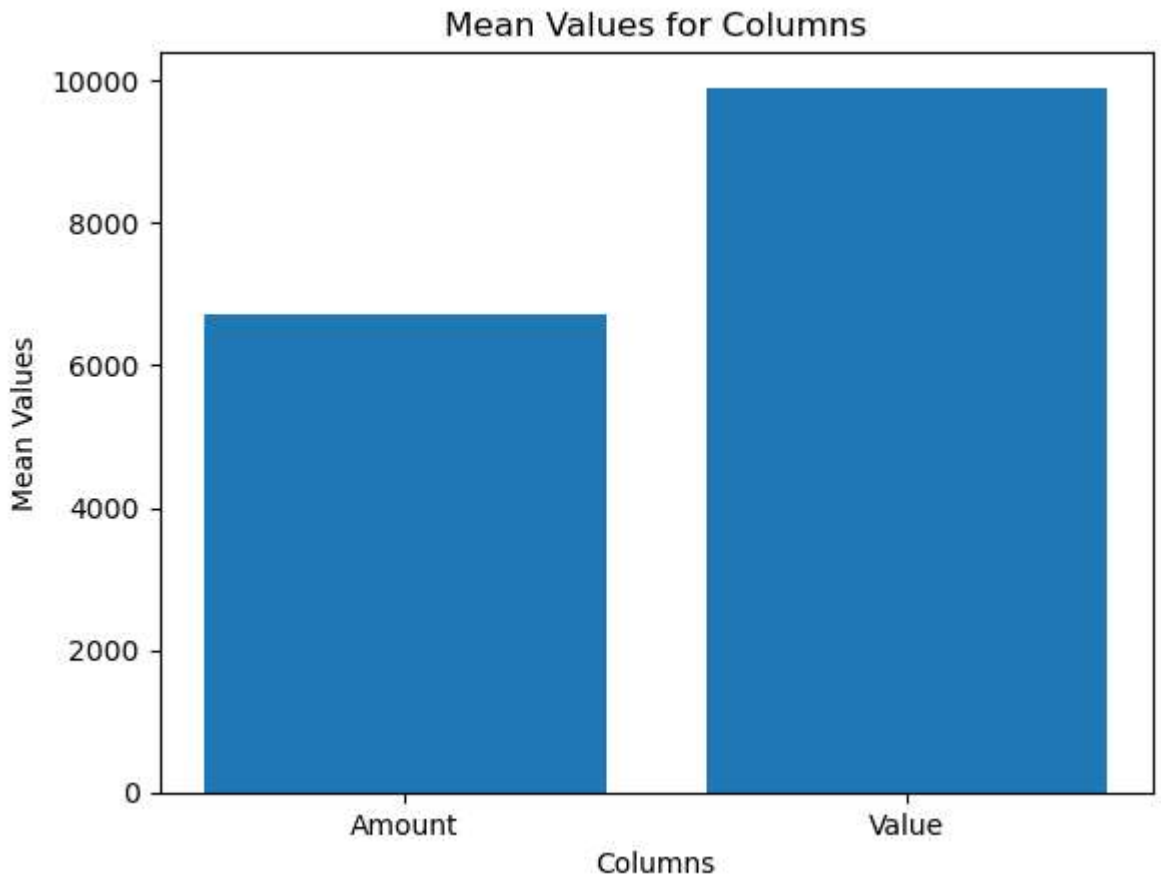
5]

Python

I use bar chart to show most frequent items as we saw in the chart transaction id mode (frequent is null) which tell us it's the unique or primary key for the data set.




The above chart was for non-numeric columns or categorical data to analyze numeric data's I calculate mean.




So, from the above two chart's we can learn: -

**TransactionId**: There is only one transaction with a TransactionId of 1. This indicates that the TransactionId might be a unique identifier for each transaction, ensuring data integrity and avoiding duplicate entries.

**BatchId**: There are 28 transactions with a BatchId of 67019. This suggests that these transactions might be grouped together for some specific processing or reporting purposes.

**AccountId**: There are 30,893 transactions associated with the AccountId 4841. This indicates that AccountId 4841 is quite active and has a significant number of transactions, possibly representing a regular customer or an important account.

**SubscriptionId**: There are 32,630 transactions with the SubscriptionId 3829. This suggests that SubscriptionId 3829 might correspond to a popular or widely used subscription service.

**CustomerId**: There are 4,091 transactions associated with the CustomerId 7343. This indicates that CustomerId 7343 might be a relatively active customer, engaging in multiple transactions.

**CurrencyCode** and **CountryCode**: The currency code UGX and country code 256.0 appear in 95,662 transactions, indicating that these transactions are likely from Uganda, as the currency code UGX represents the Ugandan shilling so we can say most people participate from UGANDA.

**ProviderId**: There are 38,189 transactions associated with the ProviderId 4. This suggests that ProviderId 4 is a significant provider and is involved in a large number of transactions.

**ProductId and ProductCategory**: There are 32,635 transactions associated with the ProductId 6 and the product category "financial_services" appears in 45,405 transactions. This indicates that financial services (represented by the product category) and ProductId 6 are popular among customers, potentially indicating a high demand for financial products.

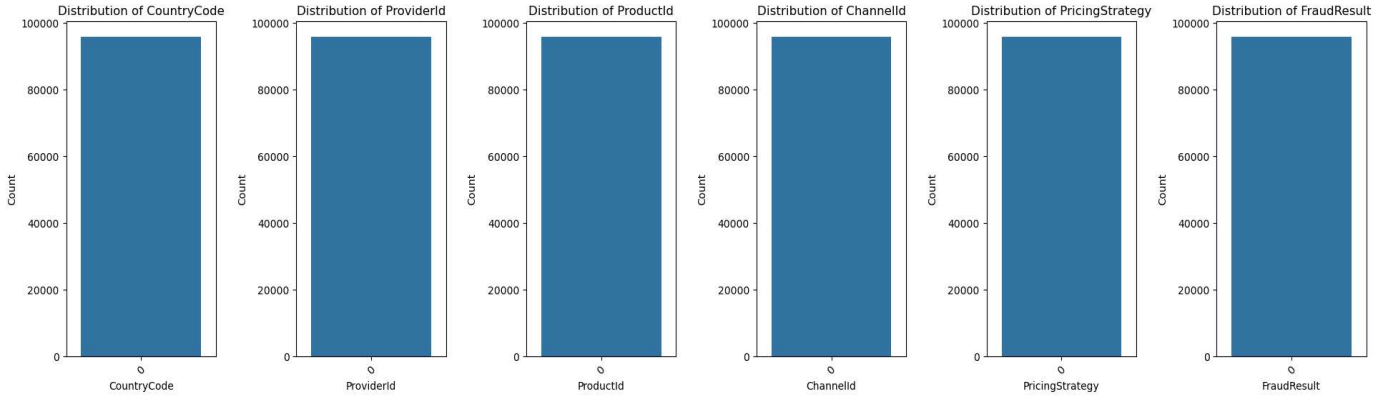
**ChannelId**: There are 56,935 transactions associated with the ChannelId 3. This suggests that ChannelId 3 is a predominant channel through which transactions are conducted, possibly representing a specific distribution channel or platform.

**TransactionStartTime**: There are 17 transactions with a TransactionStartTime of 2018-12-24T16:30:13Z. This specific timestamp may indicate a particular event, such as a system update or a batch processing time.

**PricingStrategy**: There are 79,848 transactions with a PricingStrategy of 2. This implies that PricingStrategy 2 is commonly used, potentially indicating a specific pricing model or strategy.

**FraudResult**: There are 95,469 transactions classified as non-fraudulent (FraudResult = 0). This suggests that the majority of the transactions in the dataset are classified as non-fraudulent, highlighting the importance of fraud detection and prevention measures.

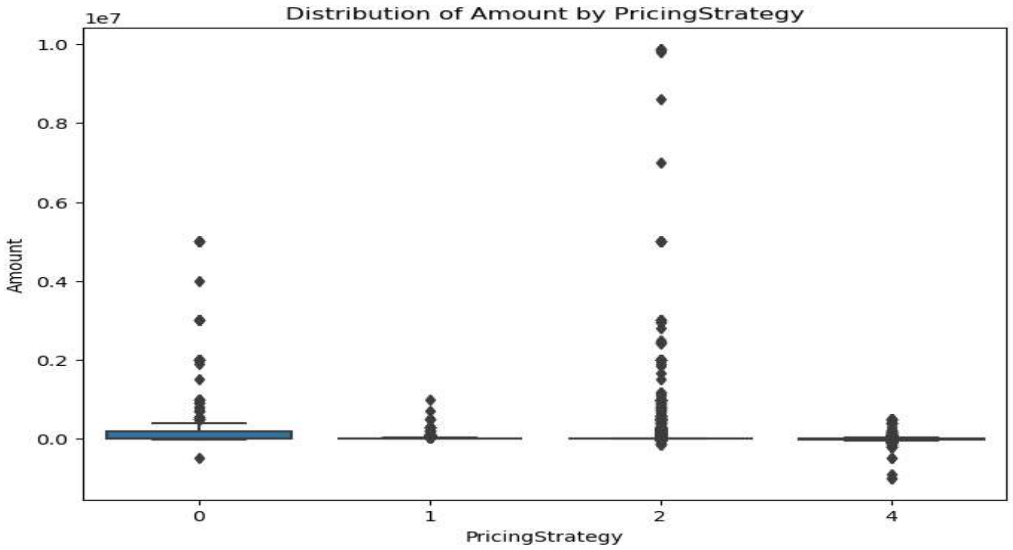
Here another chart to show distribution of categorical data

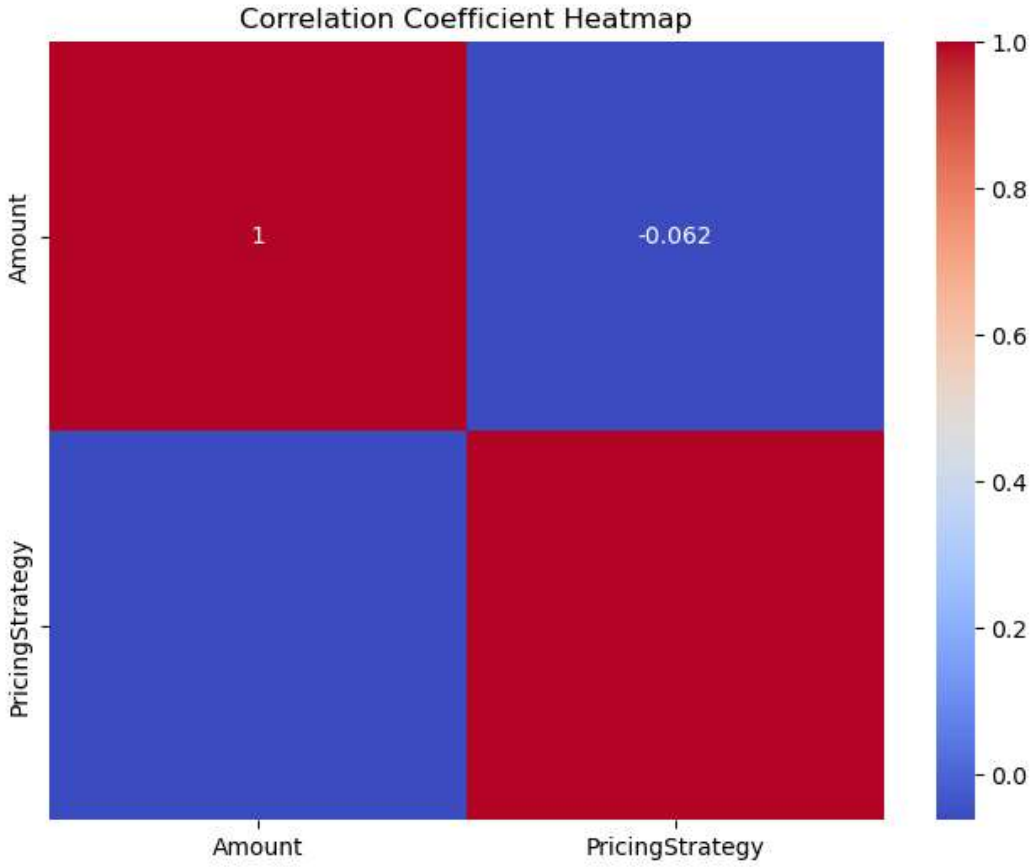


## Correlation Analysis

I tried to show correlation among Amount vs Price strategy And Amount vs Fraud Result.

Amount vs Price strategy



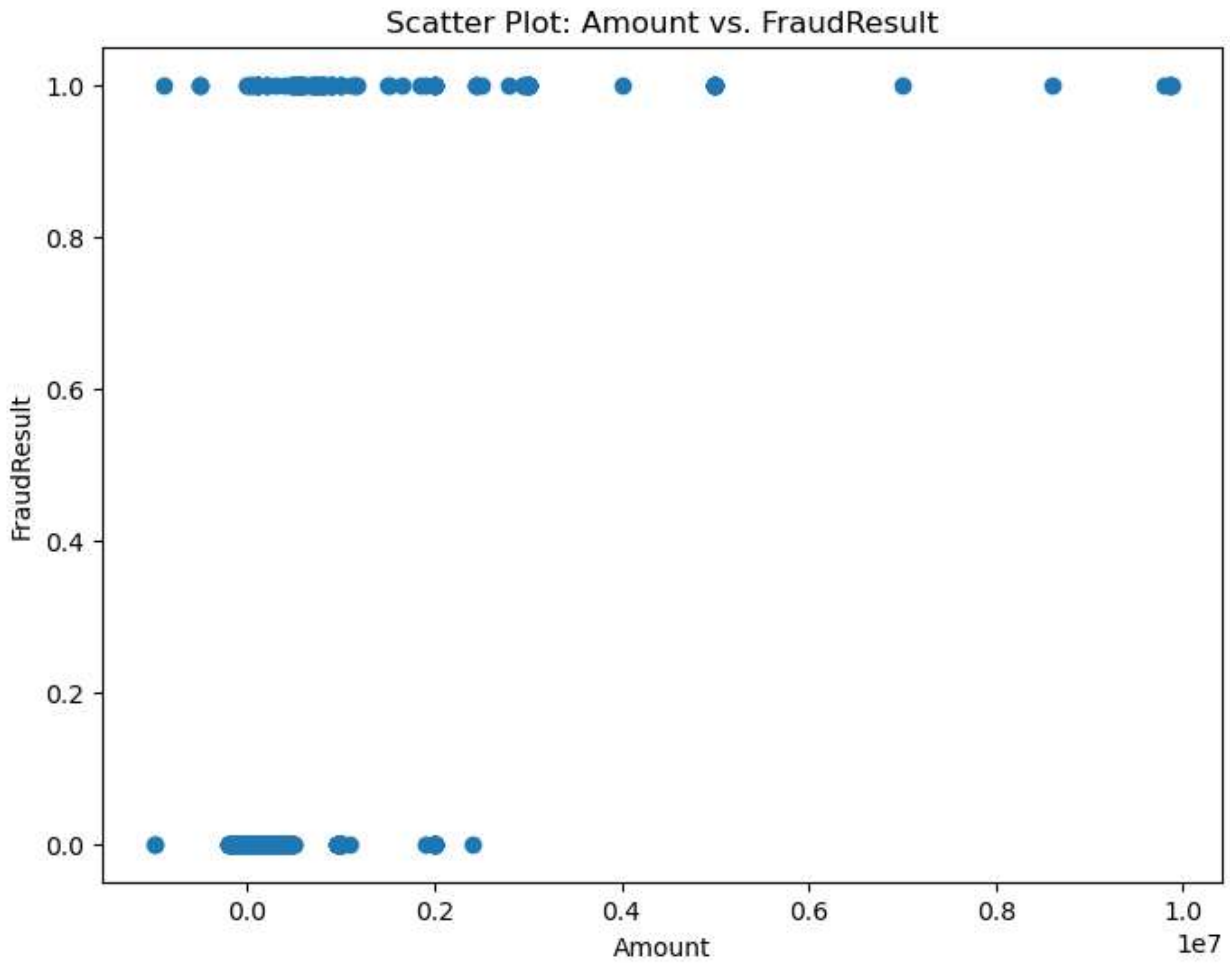
The correlation coefficient of -0.061930792420875715 between the 'Amount' and 'PricingStrategy' from a statistical perspective.

The correlation coefficient measures the strength and direction of the linear relationship between two variables. In this case, the correlation coefficient of -0.0619 suggests a weak negative correlation between 'Amount' and 'PricingStrategy'.

A negative correlation means that as the 'PricingStrategy' increases, the 'Amount' tends to decrease slightly, although the relationship is weak. However, it's essential to note that the correlation coefficient is close to zero, indicating a very weak relationship.

In practical terms, this means that there might be a subtle tendency for lower pricing strategies to have slightly higher transaction amounts.

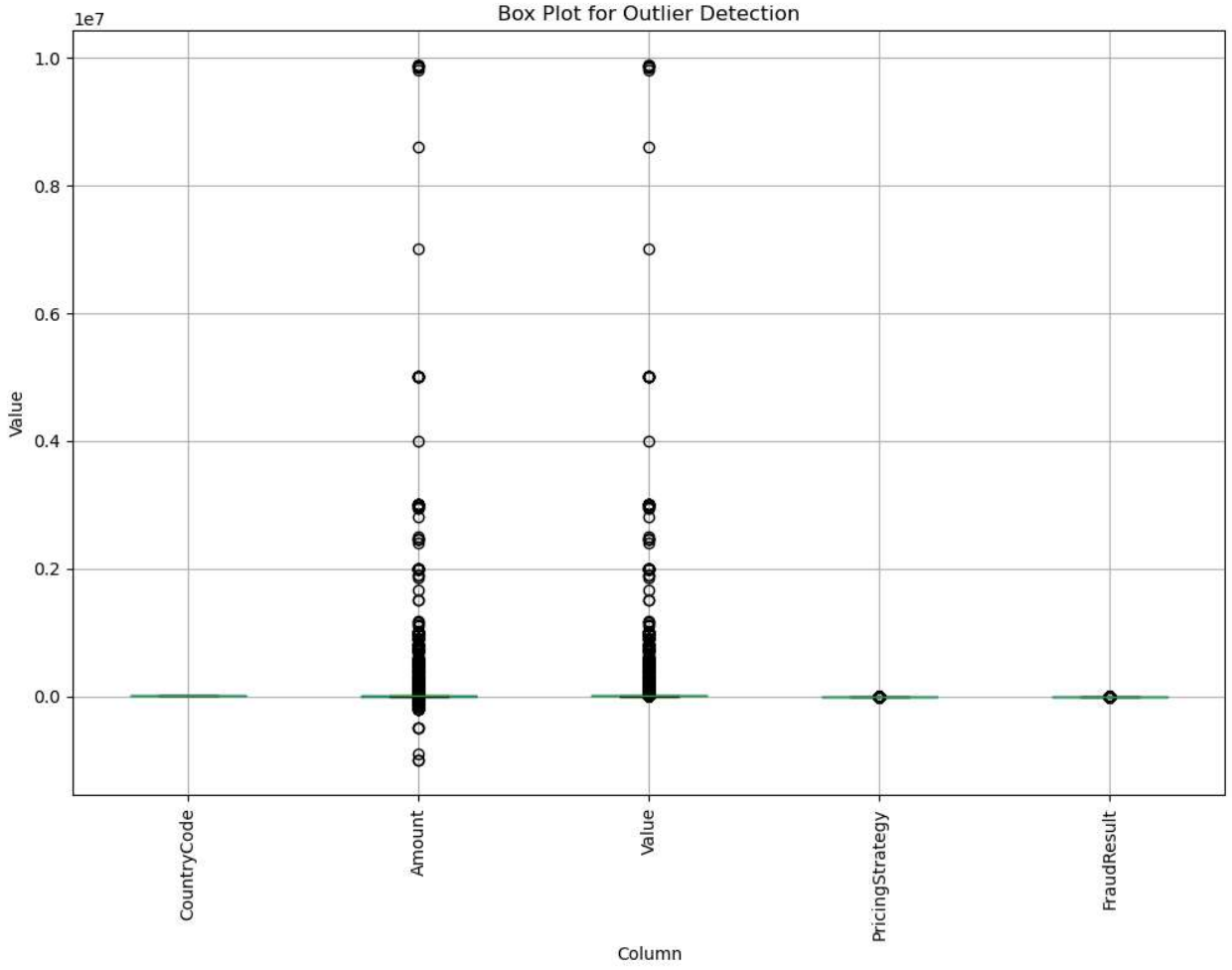
**Amount vs Fraud Result**



The correlation coefficient of 0.5573700909352298 between the 'Amount' and 'FraudResult' features, there is a moderate positive relationship between these two variables. This indicates that higher transaction amounts are more likely to be associated with fraudulent transactions

## Outlier Detection

I tried to use Box Plot to show outliers



# Feature Engineering

Feature engineering and data preprocessing are crucial steps in the data analysis pipeline as they help transform raw data into a format suitable for model training and analysis. These steps ensure that the data is in a structured and informative state, facilitating accurate predictions and insights.

### Create Aggregate Features

Aggregate features summarize individual customer transaction data to provide a holistic view of their financial behavior. The following examples illustrate some of the aggregate features that can be created:

Total Transaction Amount: The sum of all transaction amounts for each customer, providing an indication of their overall spending habits.

Average Transaction Amount: The average transaction amount per customer, representing their typical spending patterns.

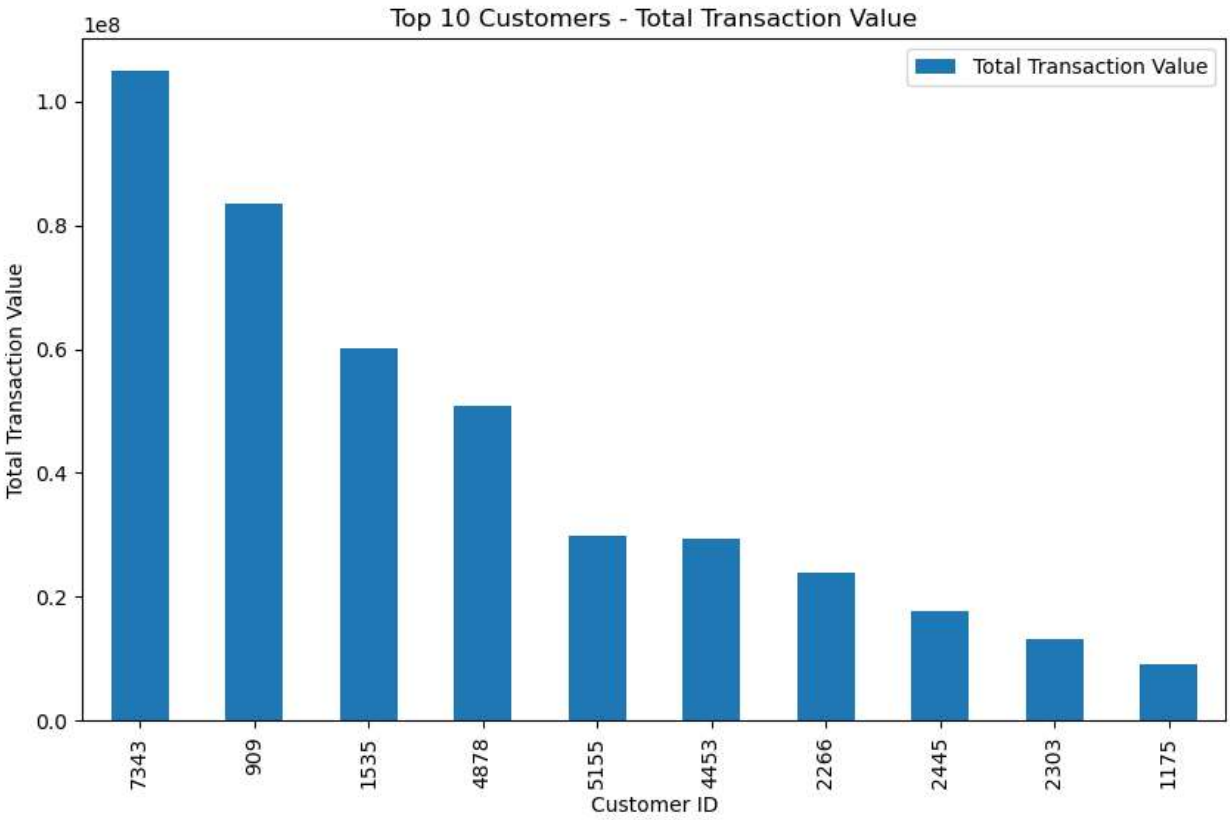
Transaction Count: The number of transactions made by each customer, reflecting their engagement with the eCommerce platform.

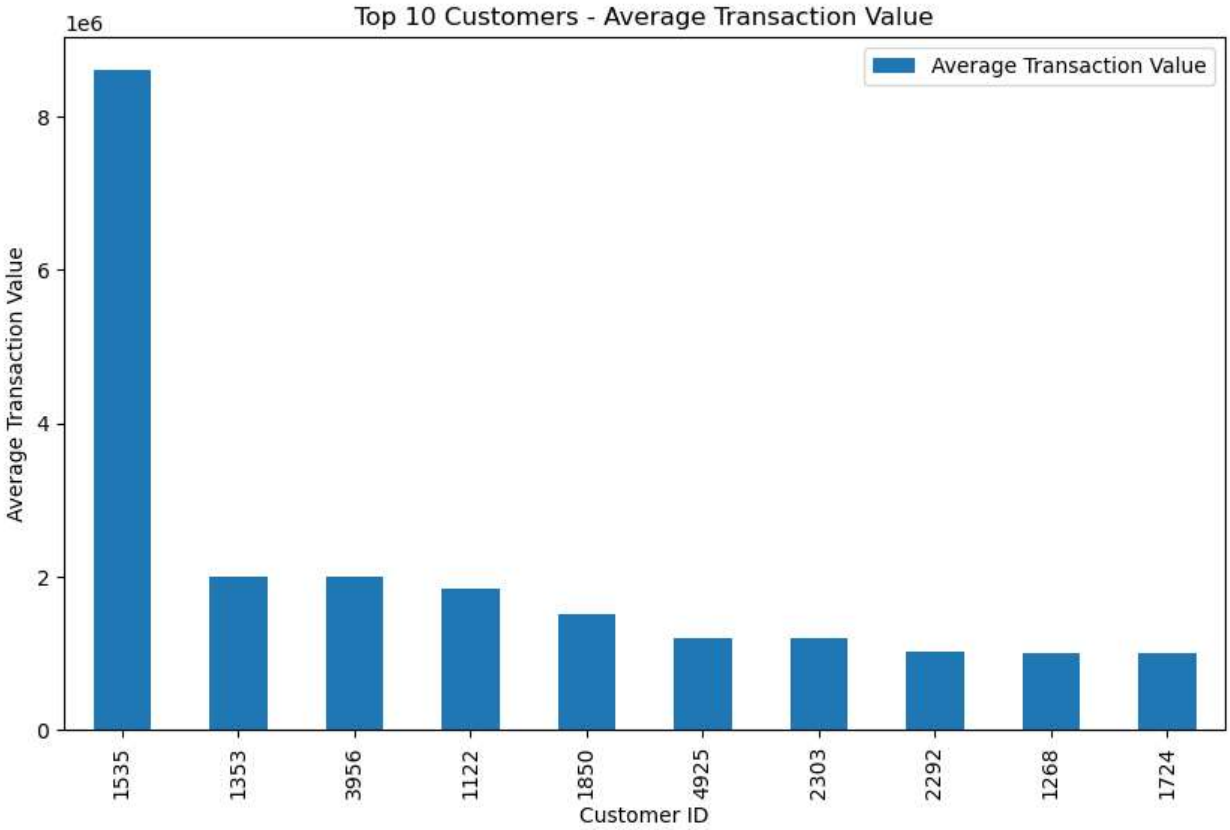
Standard Deviation of Transaction Amounts: The variability in transaction amounts per customer, capturing the diversity of their purchases.

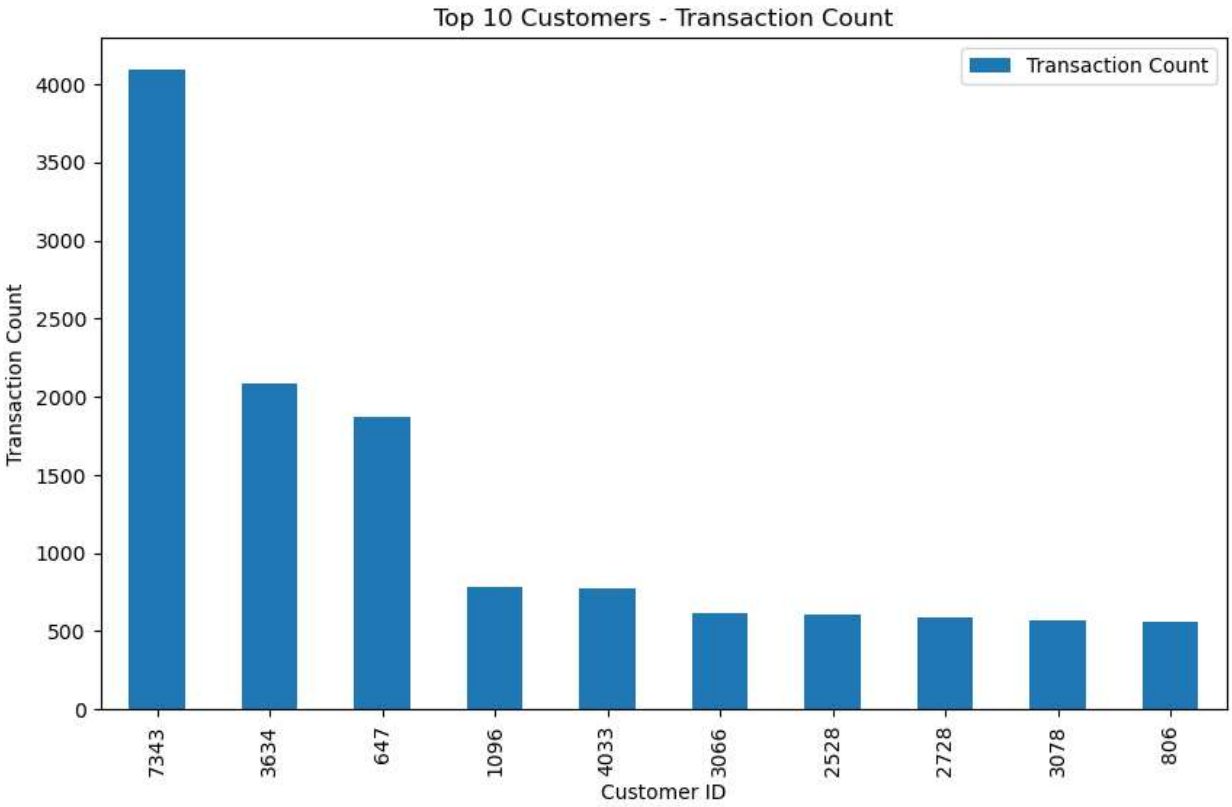
These aggregate features provide a consolidated representation of customer behavior, enabling the Credit Scoring Model to capture essential patterns and trends.

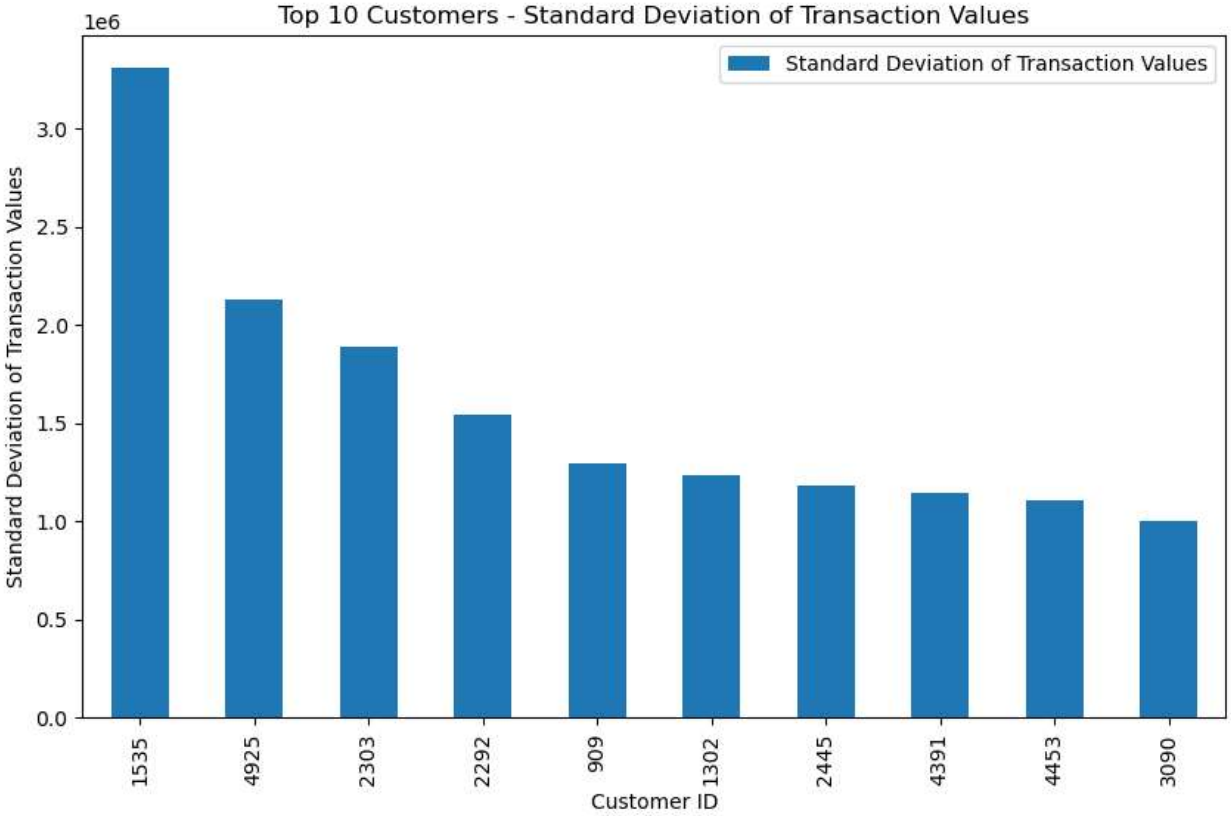
| CustomerId | Total Transaction Value | Average Transaction Value | Transaction Count | Standard Deviation of Transaction Values |
|---|---|---|---|---|
| 1 | 10000 | 10000.000000 | 1 | 0.000000 |
| 10 | 10000 | 10000.000000 | 1 | 0.000000 |
| 1001 | 30400 | 6080.000000 | 5 | 4100.243895 |
| 1002 | 4775 | 434.090909 | 11 | 518.805446 |
| 1003 | 32000 | 5333.333333 | 6 | 3945.461528 |
| ... | ... | ... | ... | ... |
| 992 | 32000 | 5333.333333 | 6 | 4033.195590 |
| 993 | 32000 | 6400.000000 | 5 | 3781.534080 |
| 994 | 614077 | 6079.970297 | 101 | 14537.733039 |
| 996 | 151000 | 8882.352941 | 17 | 2619.216317 |
| 998 | 163000 | 7409.090909 | 22 | 3168.431953 |

3742 rows × 4 columns


Top 10 Customers - Total Transaction Value
1e8
Total Transaction Value
1.0
0.8
0.6
0.4
0.2
0.0
Total Transaction Value
7343
909
1535
4878
5155
4453
2266
2445
2303
1175
Customer ID


Top 10 Customers - Average Transaction Value
1e6
Average Transaction Value
8
6
4
2
0
Average Transaction Value
1535
1353
3956
1122
1850
4925
2303
2292
1268
1724
Customer ID




the plots are self-descriptive we use them to know about our customers.

## Extract Features

Extracting relevant features from the available data can enhance the predictive power of the Credit Scoring Model. Extracted features can include temporal information that captures the context of transactions. Examples of extracted features are:

Transaction Hour: The hour of the day when the transaction occurred, allowing for analysis of time-specific spending patterns.

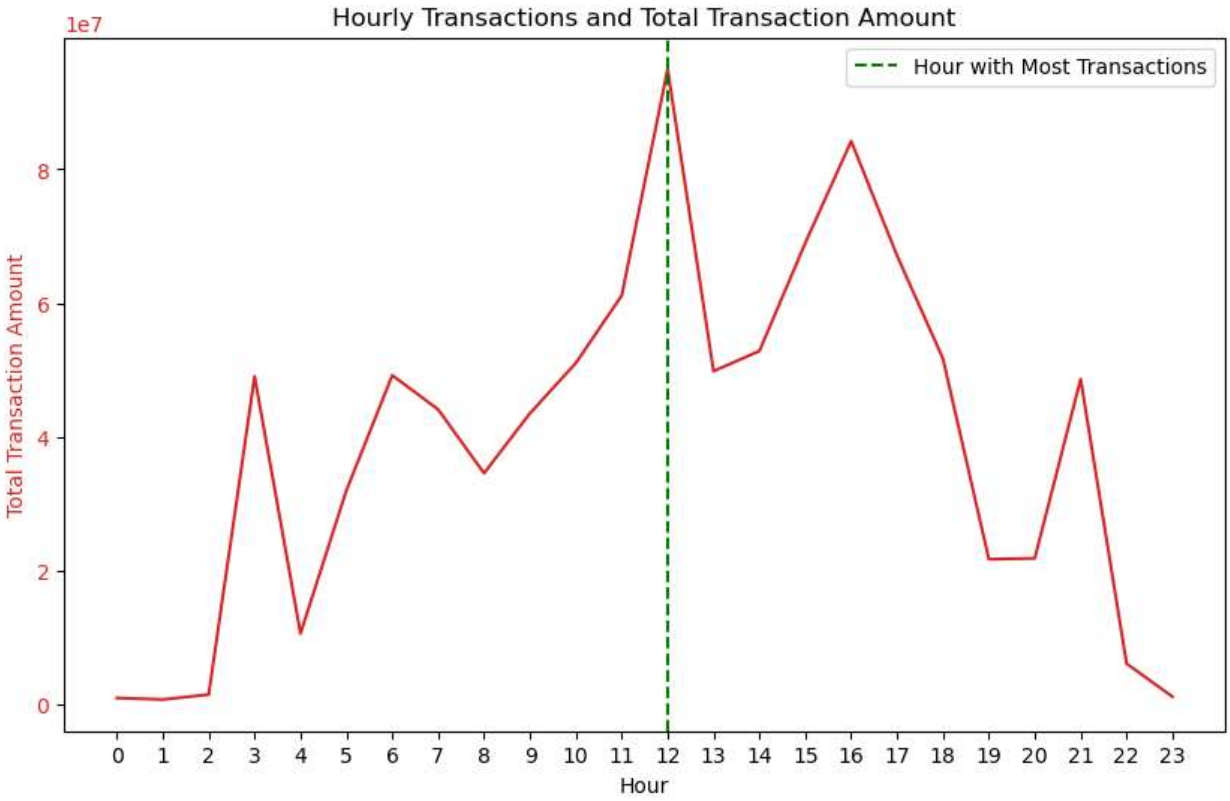
Transaction Day: The day of the month when the transaction occurred, providing insights into spending behavior at different times of the month.

Transaction Month: The month when the transaction occurred, helping identify seasonal trends in customer activity.

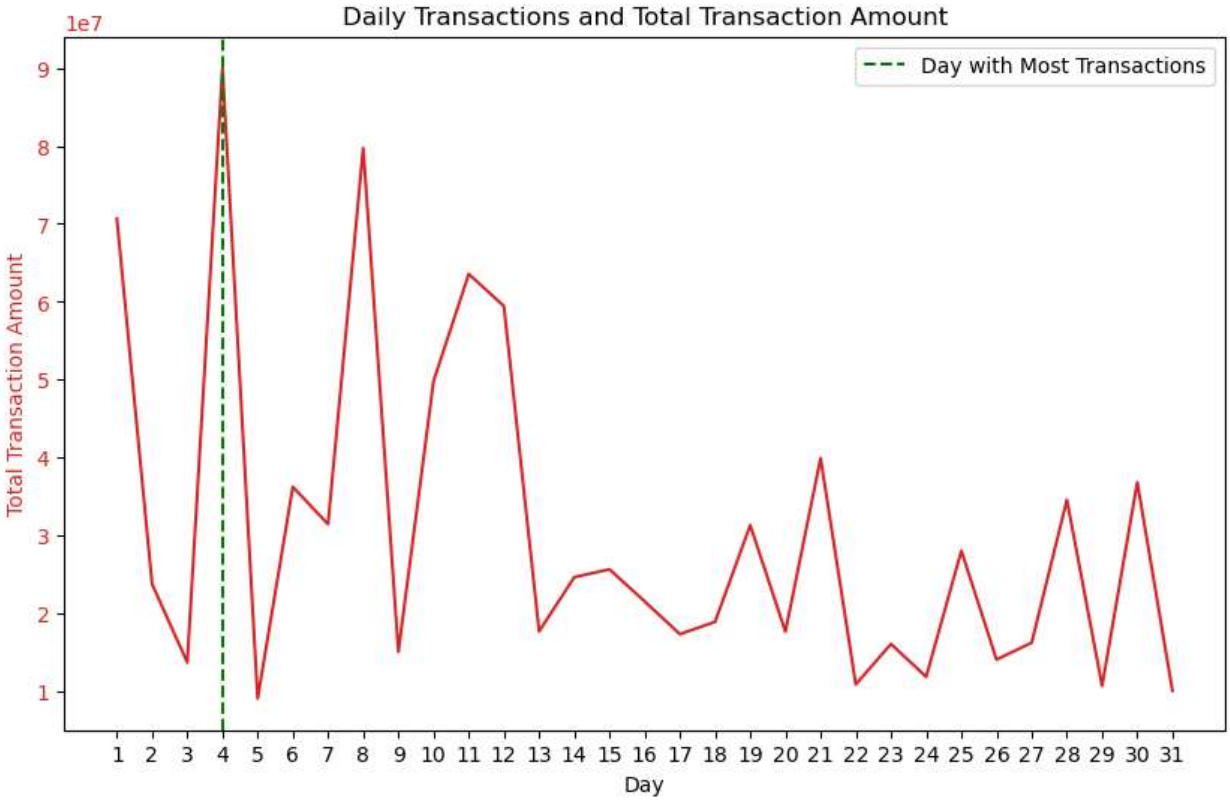
Transaction Year: The year when the transaction occurred, enabling analysis of long-term trends and changes in customer behavior.

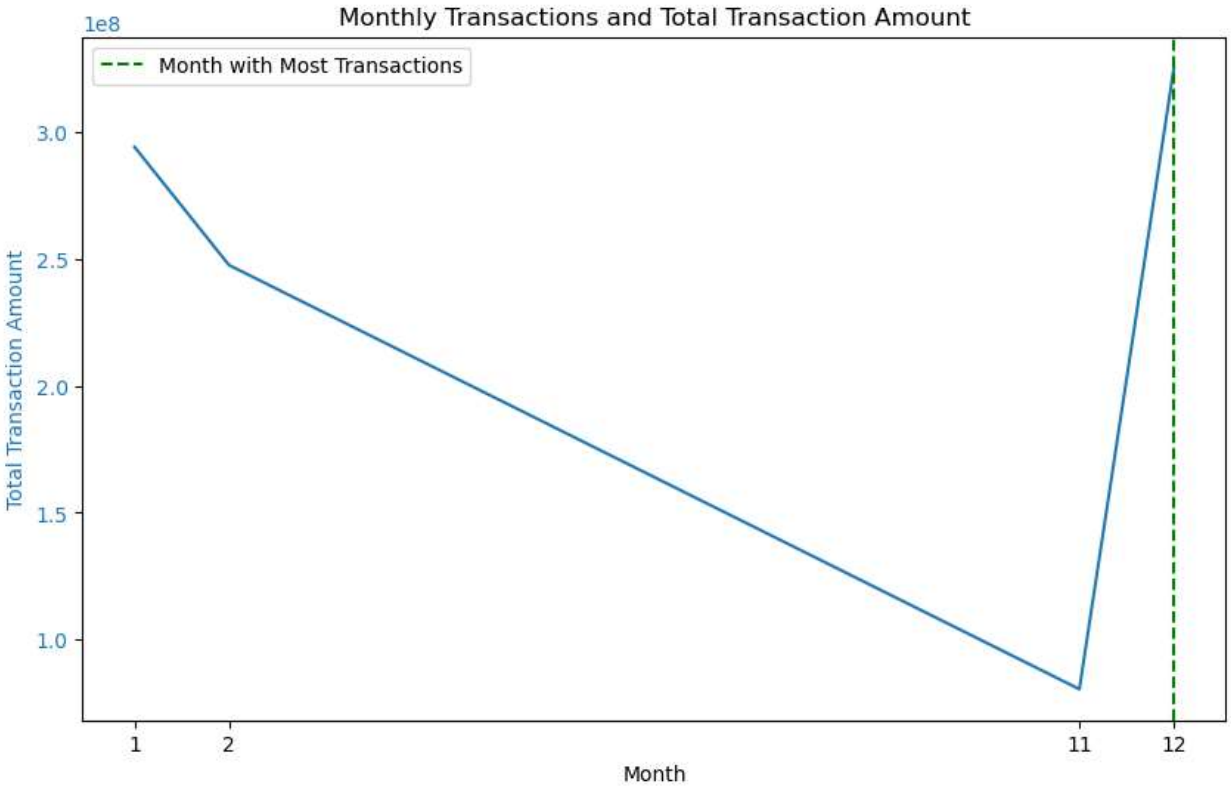
By incorporating these extracted features, the Credit Scoring Model can capture temporal dynamics and improve its predictive accuracy.

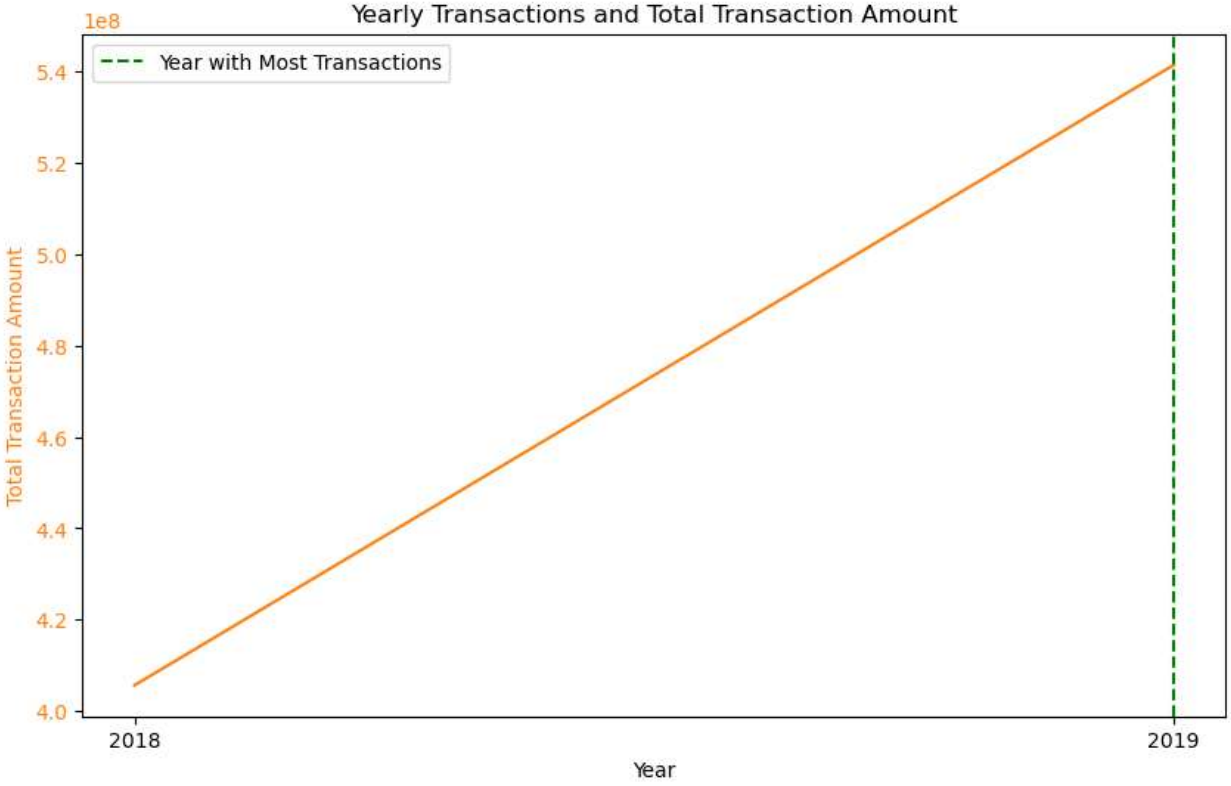
| ›ductCategory | ChannelId | Amount | Value | TransactionStartTime | PricingStrategy | FraudResult | TransactionHour | TransactionDay | TransactionMonth | TransactionYear |
|---|---|---|---|---|---|---|---|---|---|---|
| airtime | 3 | 1000.0 | 1000 | 2018-11-15 02:18:49+00:00 | 2 | 0 | 2 | 15 | 11 | 2018 |
| ıancial_services | 2 | -20.0 | 20 | 2018-11-15 02:19:08+00:00 | 2 | 0 | 2 | 15 | 11 | 2018 |
| airtime | 3 | 500.0 | 500 | 2018-11-15 02:44:21+00:00 | 2 | 0 | 2 | 15 | 11 | 2018 |
| utility_bill | 3 | 20000.0 | 21800 | 2018-11-15 03:32:55+00:00 | 2 | 0 | 3 | 15 | 11 | 2018 |
| ıancial_services | 2 | -644.0 | 644 | 2018-11-15 03:34:21+00:00 | 2 | 0 | 3 | 15 | 11 | 2018 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| ıancial_services | 2 | -1000.0 | 1000 | 2019-02-13 09:54:09+00:00 | 2 | 0 | 9 | 13 | 2 | 2019 |
| airtime | 3 | 1000.0 | 1000 | 2019-02-13 09:54:25+00:00 | 2 | 0 | 9 | 13 | 2 | 2019 |
| ıancial_services | 2 | -20.0 | 20 | 2019-02-13 09:54:35+00:00 | 2 | 0 | 9 | 13 | 2 | 2019 |
| tv | 3 | 3000.0 | 3000 | 2019-02-13 10:01:10+00:00 | 2 | 0 | 10 | 13 | 2 | 2019 |
| ıancial_services | 2 | -60.0 | 60 | 2019-02-13 10:01:28+00:00 | 2 | 0 | 10 | 13 | 2 | 2019 |




The above plot can show us in which hour most transaction is computed which is 12.






We can see transactions are increasing each year so fraud might increase same way

### Encode Categorical Variables

Categorical variables, need to be converted into numerical format for model compatibility. Two common techniques for encoding categorical variables are:

One-Hot Encoding: This method converts categorical values into binary vectors, where each category is represented by a binary column. This technique allows the model to understand and utilize categorical information effectively.

Label Encoding: Label encoding assigns a unique integer value to each category, transforming categorical variables into ordinal numerical representations. However, caution must be exercised to prevent the model from assigning arbitrary ordinal relationships to the categories.

By encoding categorical variables, the Credit Scoring Model can effectively utilize the information contained within these variables.

So, I encode the data and used it in different parts of the cod and saved it as separate csv file.

### Handle Missing Values

Missing values are a common occurrence in real-world datasets and need to be addressed to ensure accurate model training. Two approaches to handle missing values are:

Imputation: Imputation involves filling missing values with estimated values based on other observed data. Common imputation methods include using mean, median, mode, or more sophisticated techniques like KNN imputation. Care should be taken to select the appropriate imputation method based on the nature of the missing data and the specific context of the problem.

Removal: In some cases, if the number of missing values is small relative to the dataset size, removing rows or columns with missing values can be a viable option. However, this approach should be carefully considered, as it may result in data loss and potential information gaps.

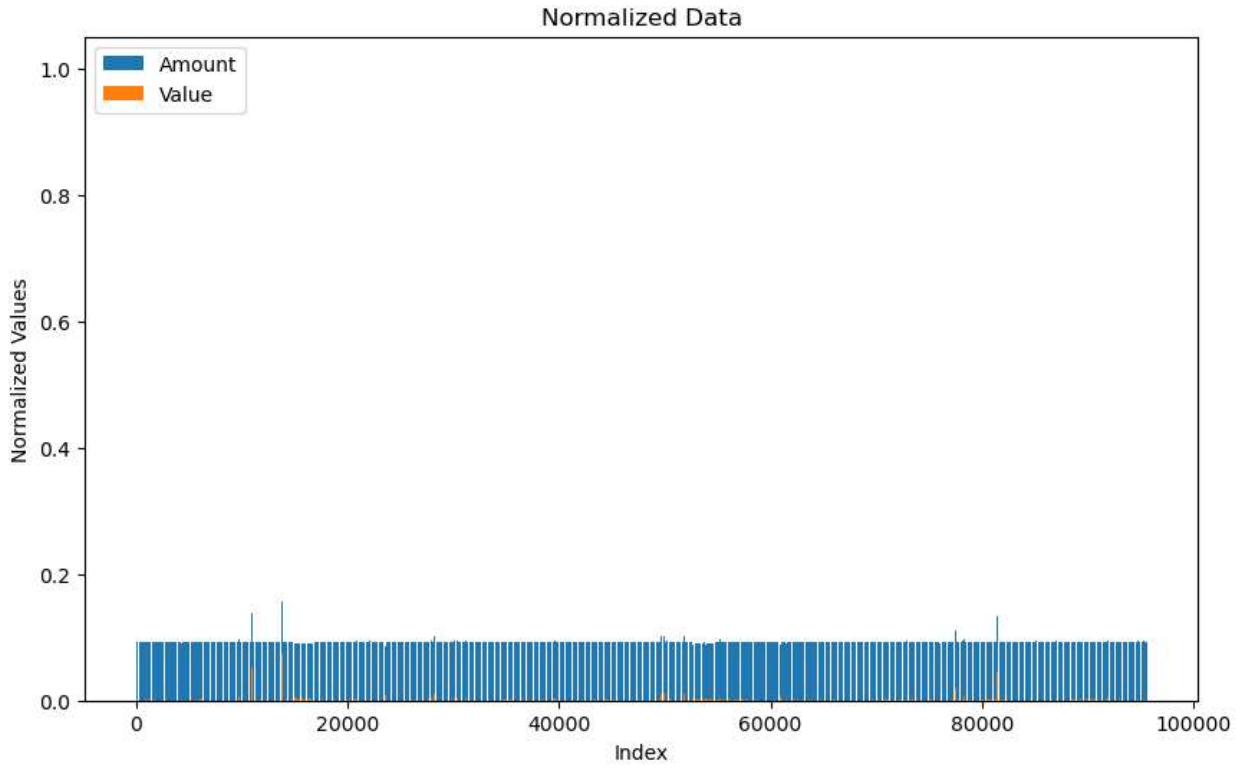
### Normalize/Standardize Numerical Features

Normalization and standardization are scaling techniques used to bring numerical features onto a similar scale, eliminating biases introduced by differing units or magnitude. Two common techniques are:

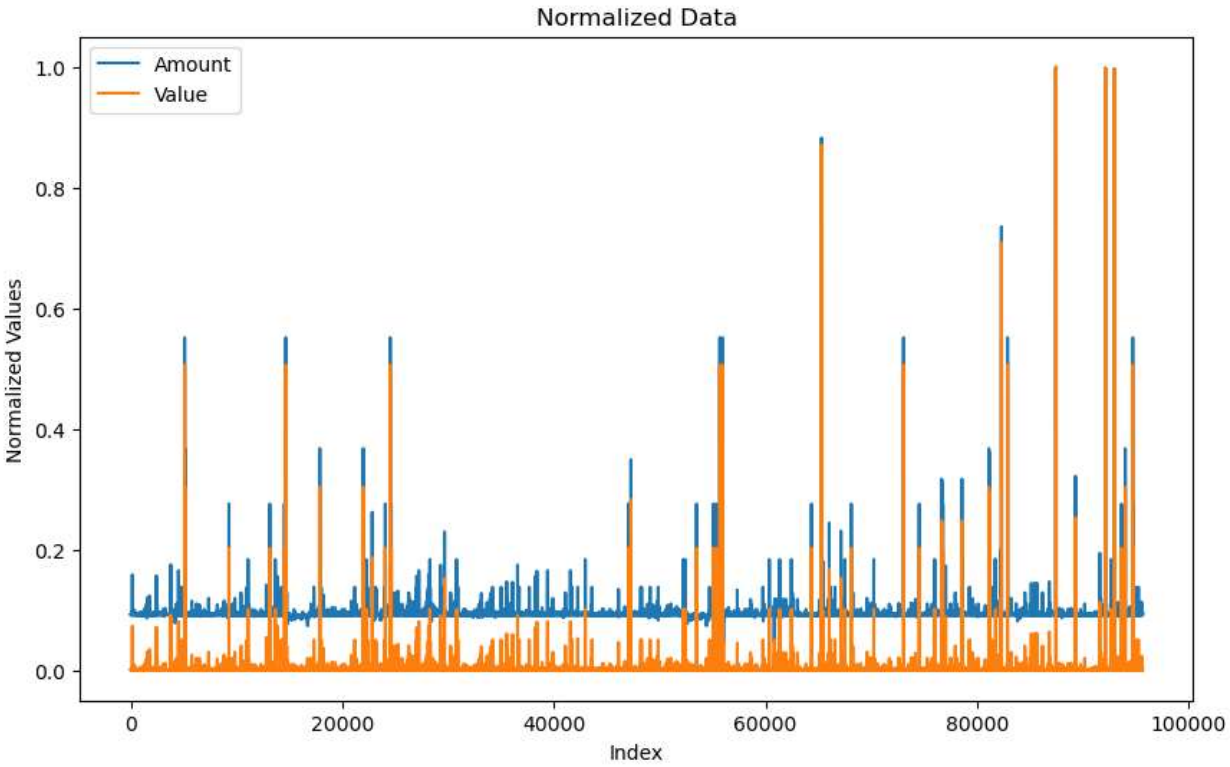
Normalization: This technique scales the data to a range of [0, 1]. It is particularly useful when the absolute values of the features are not as important as their relative proportions.

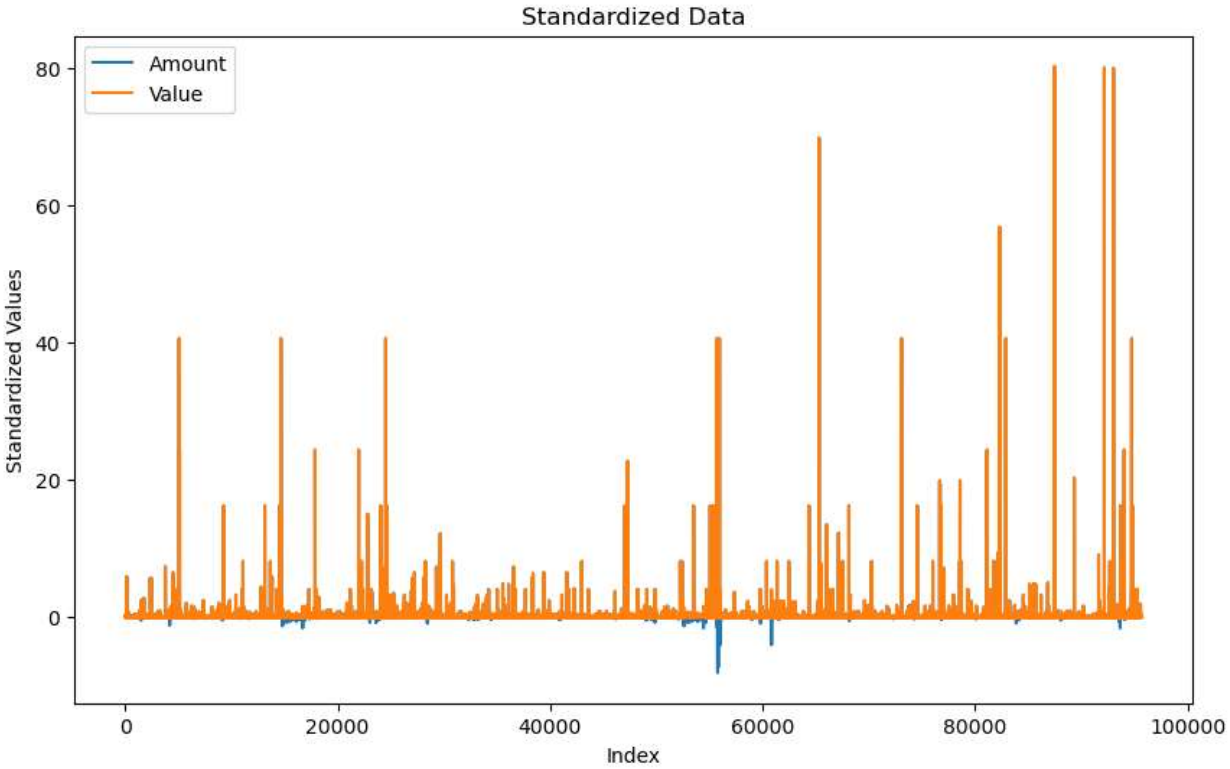
Standardization: Standardization scales the data to have a mean of 0 and a standard deviation of 1. It is beneficial when the absolute values and distributions of the features are crucial for accurate modeling.

By normalizing or standardizing numerical features, the Credit Scoring Model can effectively compare and analyze different features without being influenced by their original scales.



# Bati Bank


Normalized Data
Amount
Value
Normalized Values
Index
0.0
0.2
0.4
0.6
0.8
1.0
0
20000
40000
60000
80000
100000


Standardized Data
Amount
Value
Standardized Values
Index
0
20
40
60
80
0
20000
40000
60000
80000
100000

In conclusion, feature engineering and data preprocessing play a crucial role in building a robust Credit Scoring Model for Bati Bank's buy-now-pay-later service. These techniques enable the model to capture relevant patterns, ensure compatibility with machine learning algorithms, handle missing values appropriately, and eliminate biases caused by feature scales. By implementing these steps effectively, Bati Bank can enhance the accuracy and reliability of its credit scoring system, facilitating informed decision-making and risk management.

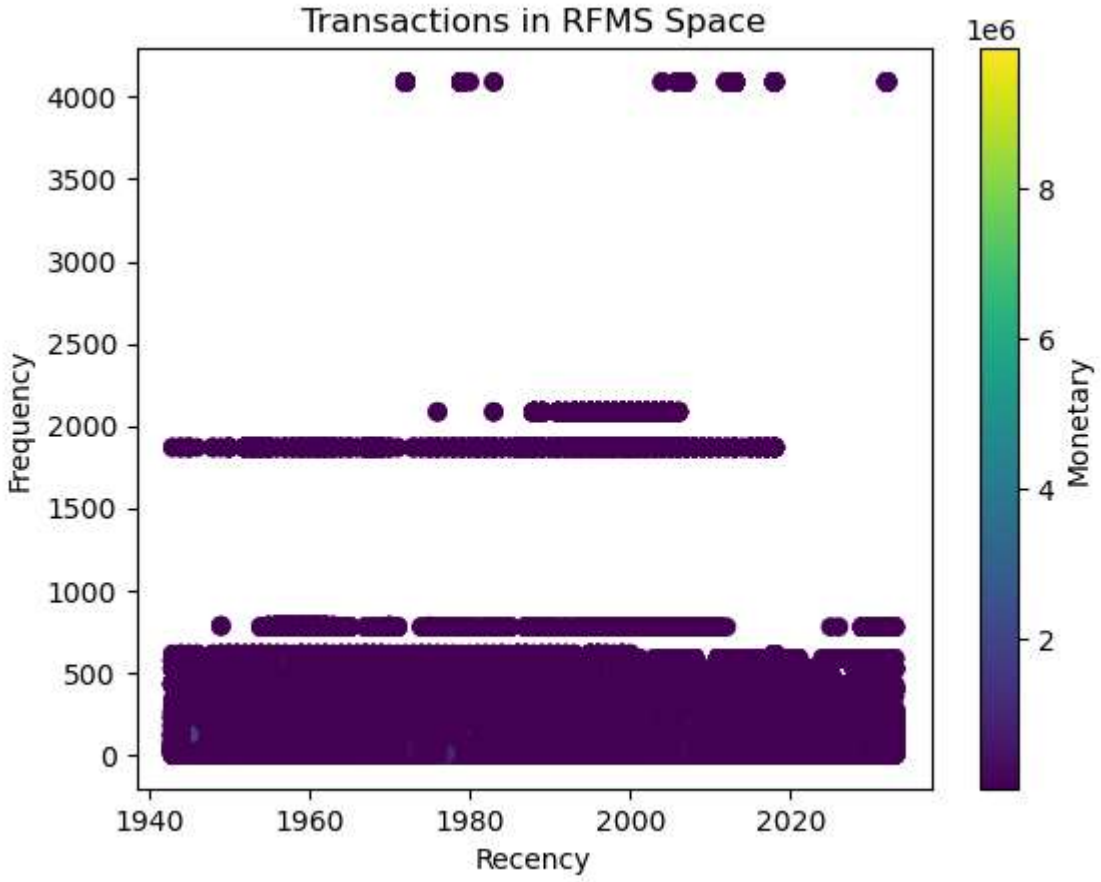
# Default estimator and WoE binning

This report aims to outline the process and objectives of constructing a default estimator (proxy) using the RFMS (Recency, Frequency, Monetary, and Score) formalism and performing Weight of Evidence (WoE) binning for credit scoring purposes.

### Constructing a Default Estimator (Proxy)

The primary objective of a credit scoring system is to classify users into high-risk or low-risk groups based on their likelihood of defaulting on loan payments. The RFMS formalism provides a framework for analyzing user behavior and establishing a boundary that separates high RFMS score (good) from low RFMS score (bad) users.

To construct a default estimator (proxy), we visualize all transactions in the RFMS space. By analyzing the distribution of RFMS scores, we can identify a threshold or boundary that effectively separates high RFMS score users (indicating a lower risk of default) from low RFMS score users (indicating a higher risk of default). This boundary serves as the basis for classifying users into good and bad categories.



### The RFMS formalism considers the following dimensions

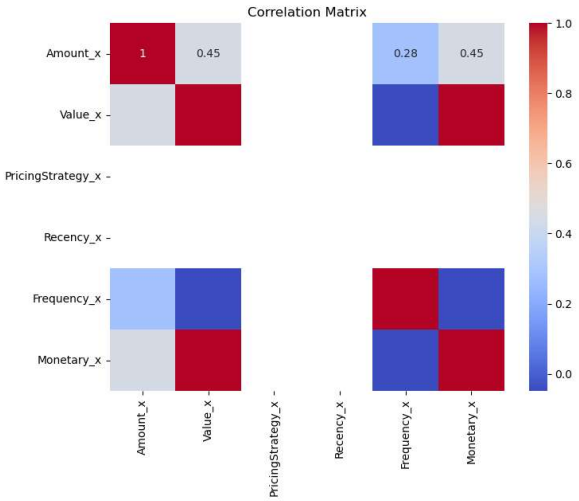
Recency: How recently a user has made transactions.

Frequency: How frequently a user engages in transactions.

Monetary: The monetary value of the user's transactions.

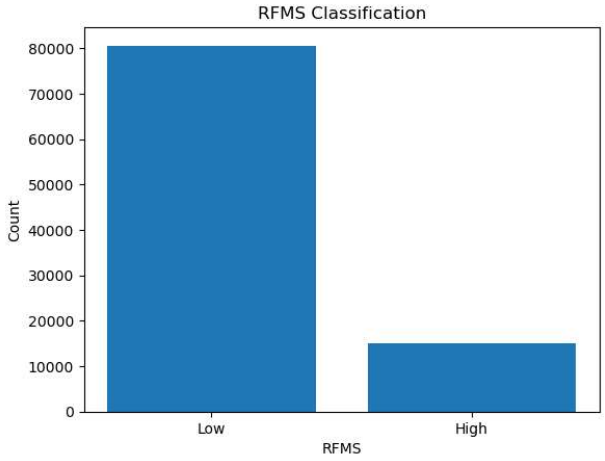
Score: An aggregate score that represents the overall creditworthiness of the user.

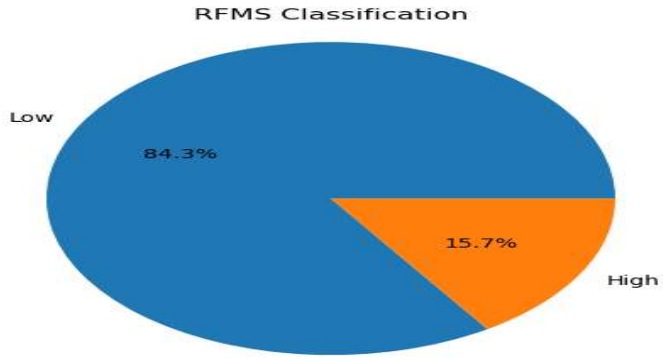
By assessing user behavior across these dimensions, we can create a default estimator that effectively categorizes users into high and low RFMS score groups.

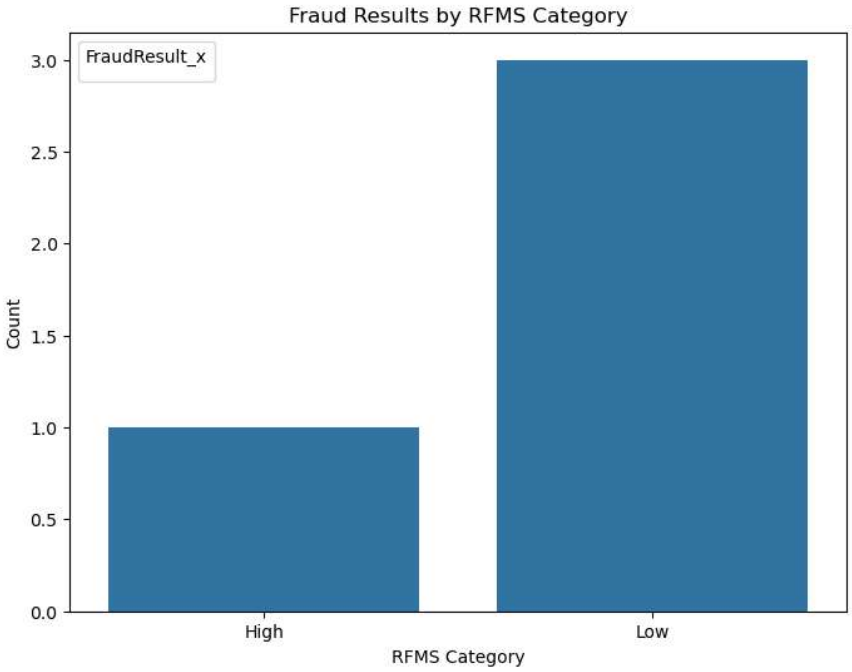



### Assigning Good and Bad Labels

Once the boundary is established, all users can be assigned a label of either good or bad based on their RFMS scores. Users falling above the boundary are considered good, indicating a lower risk of default, while users below the boundary are classified as bad, indicating a higher risk of default.






Assigning these labels allows for the segmentation of users into distinct risk categories, enabling the credit scoring model to make informed decisions about loan approvals and risk management.

### Performing Weight of Evidence (WoE) Binning

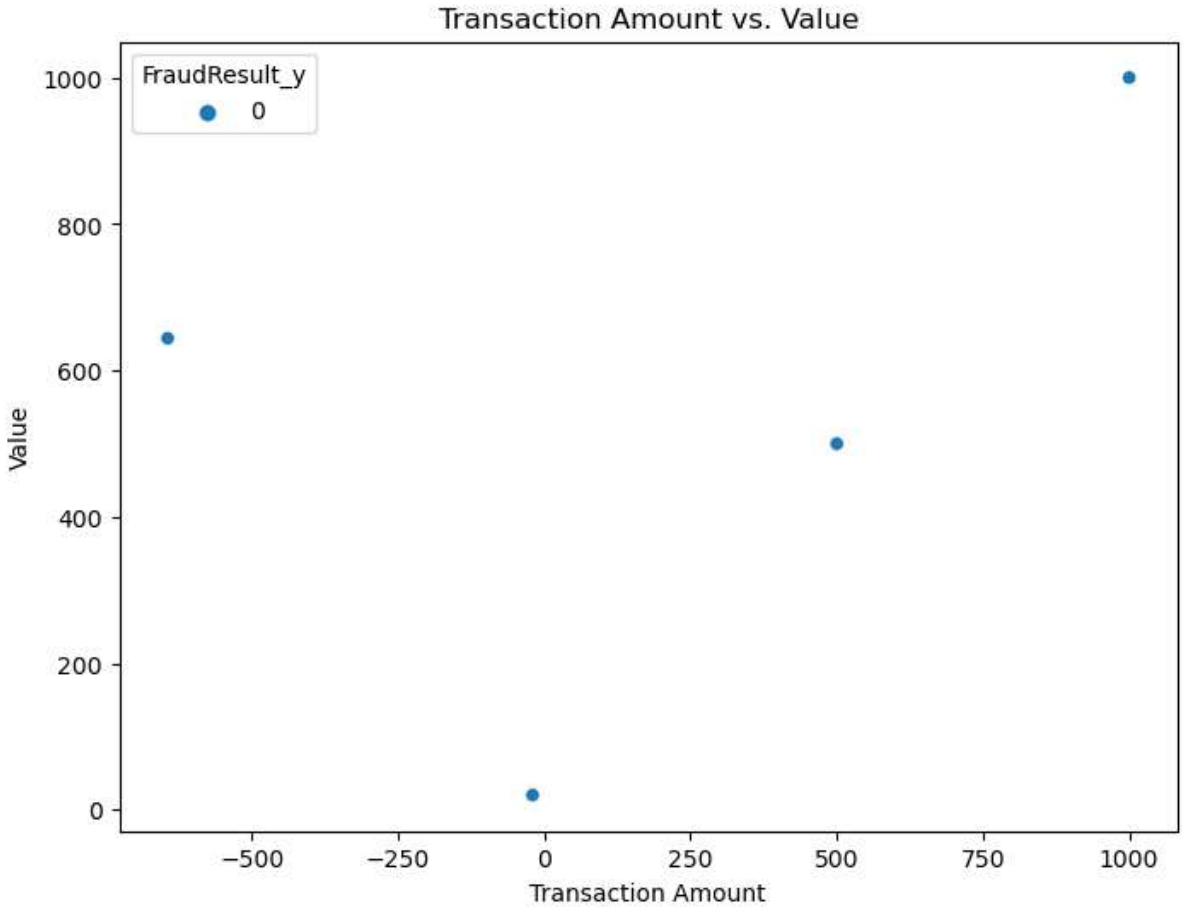
Weight of Evidence (WoE) binning is a technique used to transform continuous variables into discrete bins, based on the WoE values calculated for each bin. The WoE values provide insights into the predictive power of each bin in relation to the default variable.

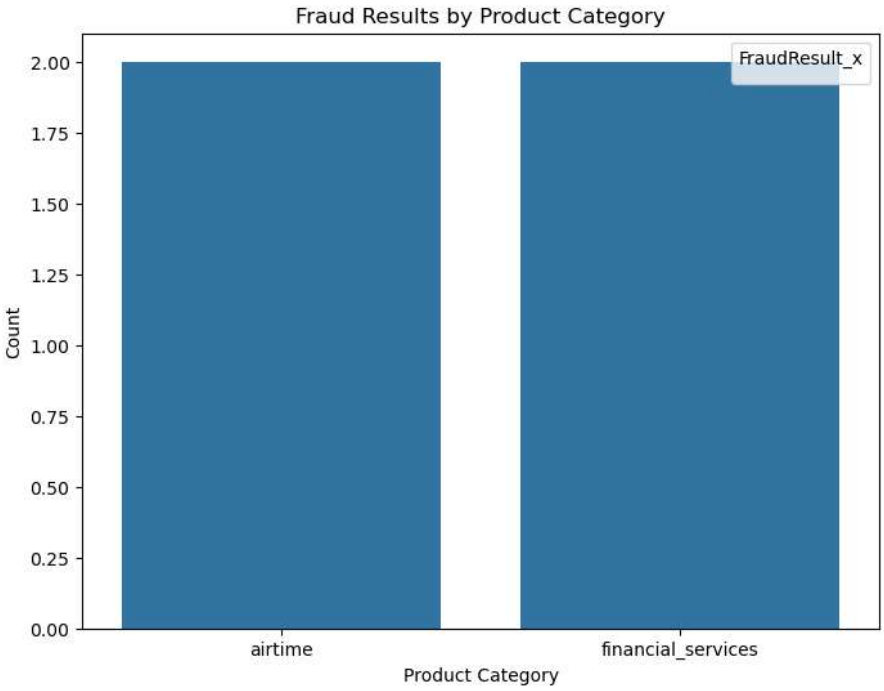
By applying WoE binning to the RFMS variables, we can further enhance the credit scoring model's ability to differentiate between high-risk and low-risk users. The WoE values for each bin indicate the relative predictive strength of that bin in

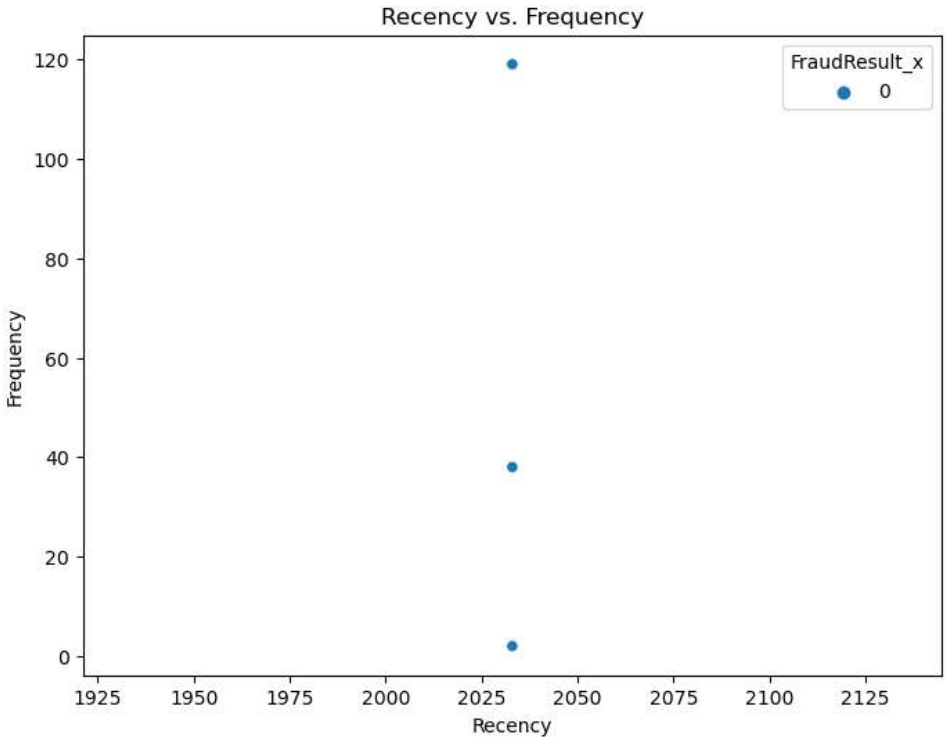
determining the likelihood of default. This information assists in assigning appropriate weights to each bin during the credit scoring process.

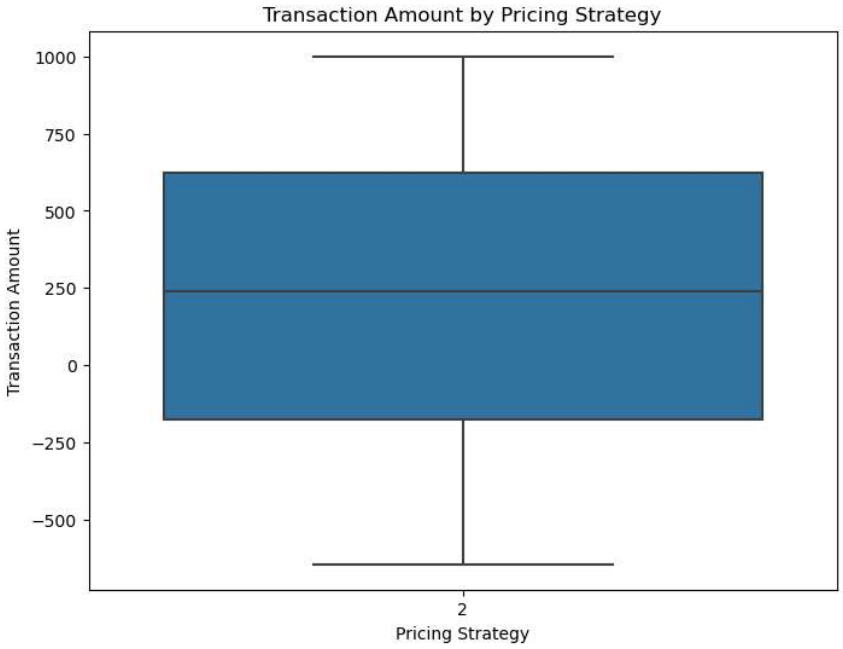
By utilizing WoE binning, we can effectively capture the information contained in the RFMS variables and translate it into a format that is easily interpretable by the credit scoring model.

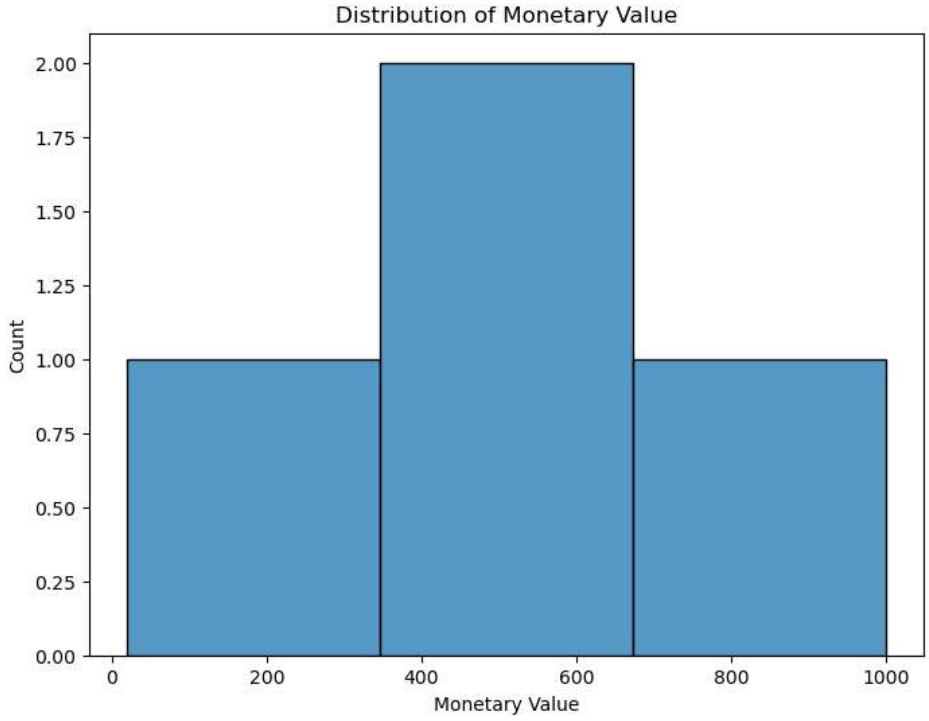
Different Plots are here to support my ideas.

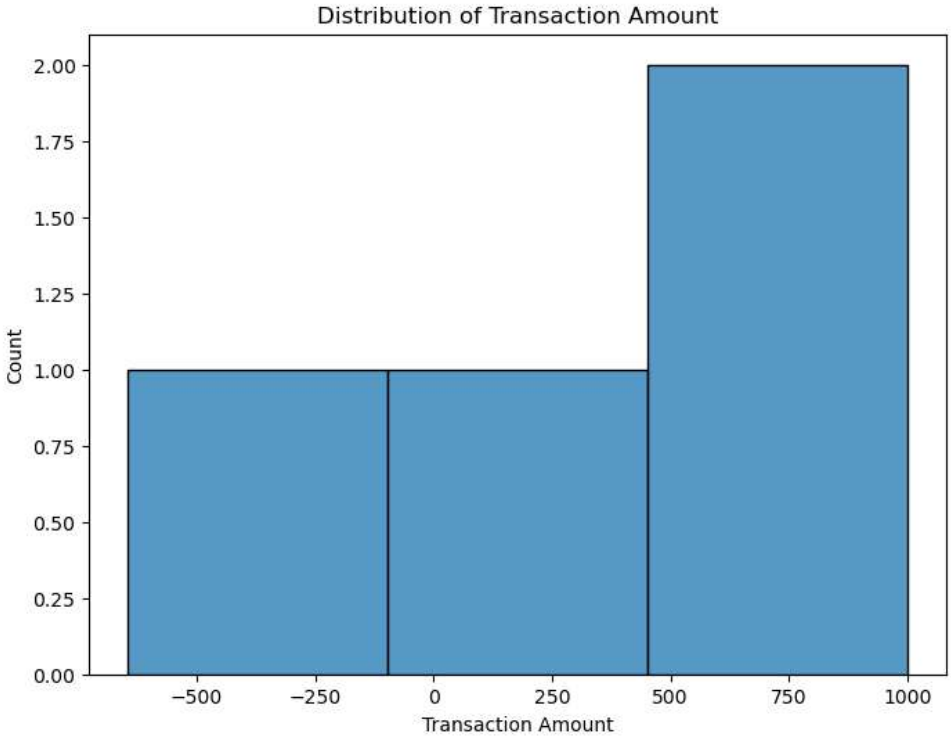


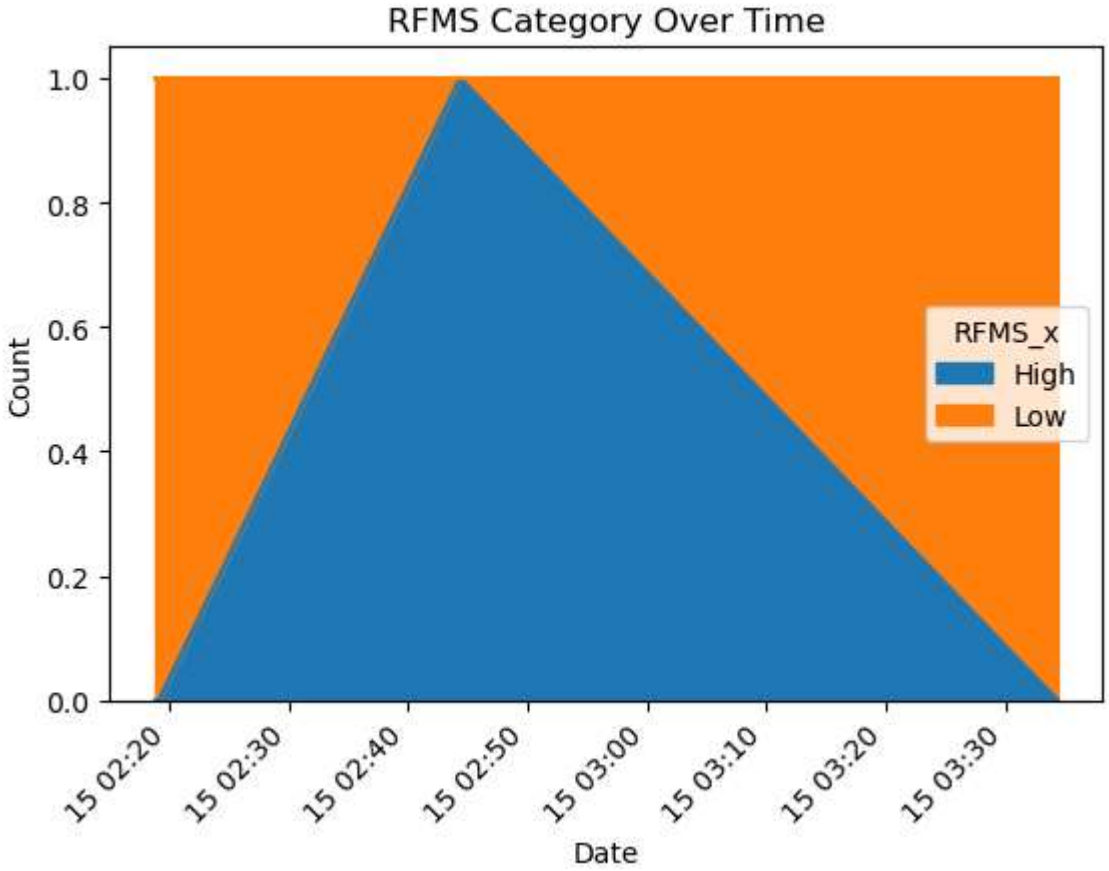

Fraud Results by Product Category
FraudResult_x
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Count
airtime
financial_services
Product Category


Recency vs. Frequency
FraudResult_x
0
120
100
80
60
40
20
0
Frequency
1925
1950
1975
2000
2025
2050
2075
2100
2125
Recency


Transaction Amount by Pricing Strategy
Transaction Amount
1000
750
500
250
0
−250
−500
2
Pricing Strategy


Distribution of Monetary Value
Count
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0
200
400
600
800
1000
Monetary Value


Distribution of Transaction Amount
Count
Transaction Amount
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
−500
−250
0
250
500
750
1000


RFMS Category Over Time
Count
Date
RFMS_x
High
Low
1.0
0.8
0.6
0.4
0.2
0.0
15 02:20
15 02:30
15 02:40
15 02:50
15 03:00
15 03:10
15 03:20
15 03:30

# Modelling

This report aims to outline the process and objectives of model selection, training, and evaluation for credit scoring using various machine learning algorithms. The following steps are involved:

### Splitting the Data

To assess the performance of the models on unseen data, it is essential to split the available data into training and testing sets. The training set will be used for model training, while the testing set will serve as a benchmark to evaluate the models' performance. This separation helps in identifying potential issues of overfitting and ensures the generalizability of the models.

### Choosing Models

For credit scoring, it is recommended to select at least two models from the following options:

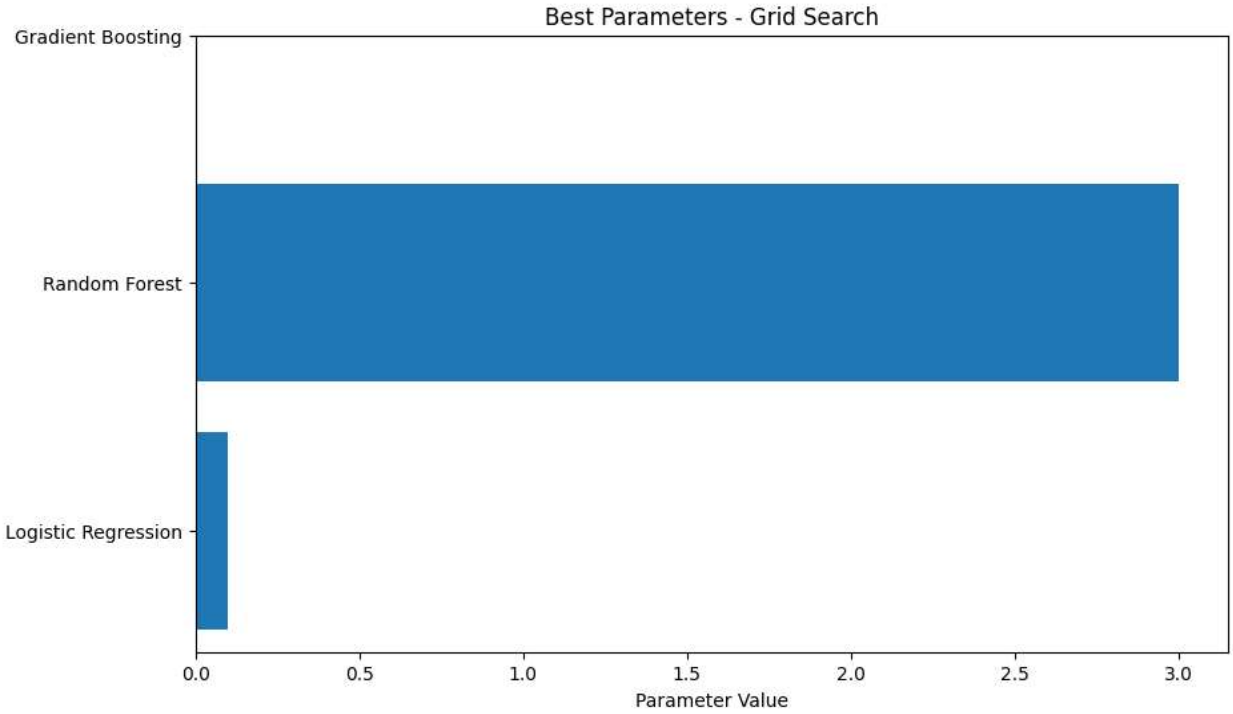
Logistic Regression: A linear model that predicts the probability of an event occurring based on the input variables. It is commonly used for binary classification problems.

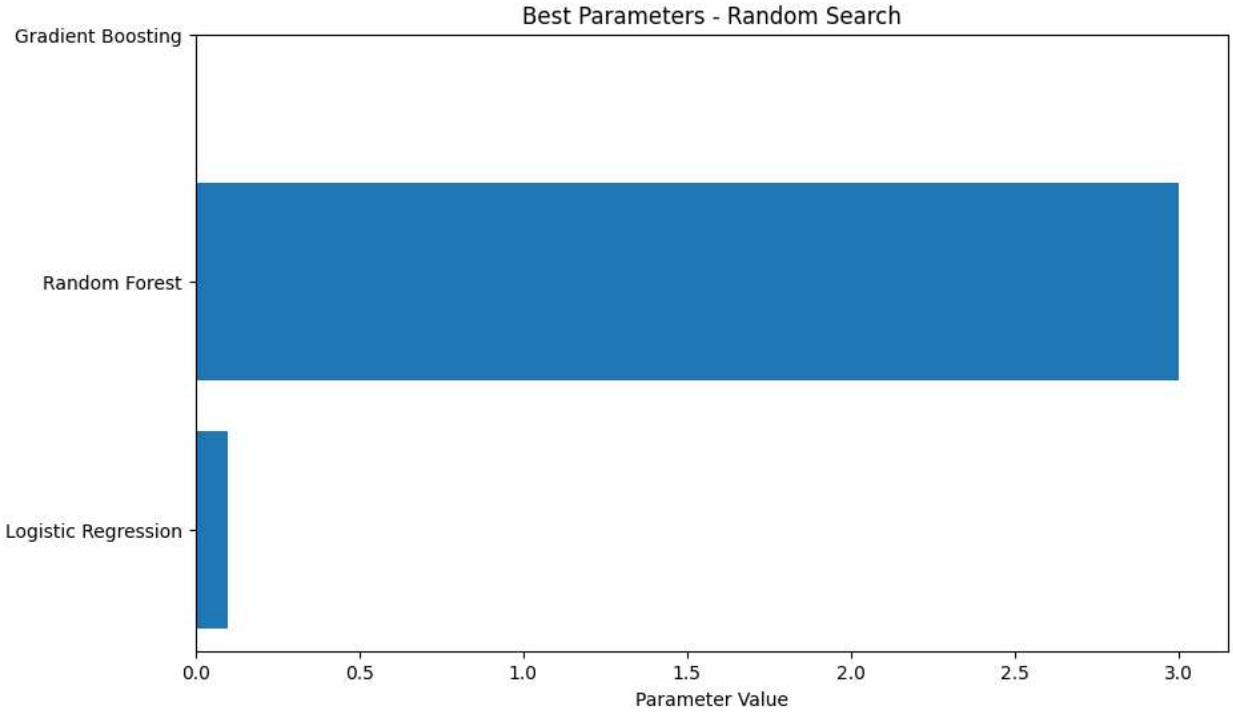
Decision Trees: A tree-like structure where internal nodes represent tests on features, and leaf nodes represent class labels. Decision trees are easy to interpret and can handle both categorical and numerical data.

Random Forest: An ensemble of decision trees that combines their predictions to improve accuracy and reduce overfitting.

Gradient Boosting Machines (GBM): A boosting algorithm that combines multiple weak learners, such as decision trees, to create a strong predictive model.

The selection of models should be based on their suitability for the credit scoring task, interpretability, computational efficiency, and previous research or industry best practices.




Training the Models:

Once the models are chosen, they need to be trained on the training data. During the training process, the models learn patterns and relationships between the input

features and the target variable, which, in this case, is the credit default status. The training process involves adjusting model parameters to minimize the prediction error.

Hyperparameter Tuning:

To improve model performance, hyperparameter tuning is performed. Hyperparameters are settings or configurations that are not learned during the training process but affect the model's performance. Techniques like Grid Search and Random Search can be utilized to explore different combinations of hyperparameters and identify the optimal settings that maximize the model's performance.

Hyperparameter tuning helps in fine-tuning the models, finding the right balance between underfitting and overfitting, and optimizing the models for credit scoring.

**Model Evaluation**

After training and tuning the models, their performance needs to be evaluated using appropriate metrics. The following metrics are commonly used for model evaluation in credit scoring:

Accuracy: The ratio of correctly predicted observations to the total observations, providing an overall measure of the model's correctness.

Precision: The ratio of correctly predicted positive observations to the total predicted positives, indicating the model's ability to accurately identify positive cases.

Recall (Sensitivity): The ratio of correctly predicted positive observations to all observations in the actual positive class, measuring the model's ability to identify all positive cases.

F1 Score: The weighted average of Precision and Recall, providing a balanced measure of the model's accuracy and completeness.

ROC-AUC: Area Under the Receiver Operating Characteristic Curve, which quantifies the model's ability to distinguish between the positive and negative classes. A higher ROC-AUC indicates better model performance.

By assessing these metrics, the performance and effectiveness of the models can be evaluated, enabling informed decisions regarding the selection of the final model for credit scoring.

# Bati Bank

## Summary

The project's business objective is to create a Credit Scoring Model for Bati Bank's partnership with an eCommerce company. The goal is to offer a buy-now-pay-later service to customers, contingent upon their creditworthiness. To achieve this objective, the project has defined specific goals:

1. Defining a Proxy Variable: Creating a proxy variable that categorizes users as high risk or low risk based on their creditworthiness, serving as an indicator of default likelihood.

2. Selecting Predictive Features: Identifying observable features that exhibit a strong correlation with the proxy variable defined in the previous step. These features will be utilized as inputs for the credit scoring model.

3. Developing a Risk Probability Model: Building a model that assigns a risk probability to new customers based on the selected features. This model will estimate the likelihood of default, aiding the bank in assessing the creditworthiness of each customer.

4. Developing a Credit Scoring Model: Creating a model that assigns a credit score to each customer using the risk probability estimates. The credit score will provide a standardized measure of the customer's creditworthiness, facilitating loan approval decisions.

5. Predicting Optimal Loan Amount and Duration: Developing a model that predicts the ideal loan amount and duration for each customer. This model will consider factors such as risk probability, credit score, and other pertinent variables to determine suitable loan terms.

The project aims to leverage data analytics and statistical techniques to accurately evaluate the creditworthiness of potential borrowers. By implementing the Credit Scoring Model, Bati Bank seeks to make informed decisions regarding loan approvals, mitigate the risk of defaults, and offer a reliable buy-now-pay-later service in collaboration with the eCommerce platform.

This report highlights the importance and objectives of feature engineering and data preprocessing in the context of developing a Credit Scoring Model for Bati Bank's buy-now-pay-later service. The key points covered in the report include:

Creating aggregate features that summarize individual customer transaction data, providing a holistic view of their financial behavior.

Extracting relevant features, such as temporal information, to capture time-specific patterns and trends in customer activity.

Encoding categorical variables using techniques like one-hot encoding or label encoding to convert them into numerical format for model compatibility.

Handling missing values through imputation or removal to ensure accurate model training without disregarding valuable data.

Normalizing or standardizing numerical features to bring them onto a similar scale, eliminating biases caused by differing units or magnitude.

Also, this report discusses the process of model selection, training, and evaluation for credit scoring. The key steps involved include splitting the data into training and testing sets, choosing suitable models such as logistic regression, decision trees, random forest, or gradient boosting machines, training the models on the training data, tuning hyperparameters to optimize performance, and evaluating the model's using metrics like accuracy, precision, recall, F1 score, and ROC-AUC.

The report emphasizes the importance of selecting models based on their suitability, interpretability, and computational efficiency. It also highlights the significance of evaluating model performance to identify the most accurate and reliable model for credit scoring. By following these steps, Bati Bank can make informed decisions regarding loan approvals, risk management, and enhance the efficiency of their credit scoring system.