BANK TRANSACTIONS ANALYSIS

TABLE OF CONTENTS

1) Introduction
2) Descriptive Statistics
3) Logistic Regression
4) K-means Clustering
5) Conclusion

INTRODUCTION

This report presents an analysis of a bank transaction dataset obtained from Kaggle. The dataset, which can be accessed via the following link,

https://www.kaggle.com/datasets/valakhorasani/bank-transaction-dataset-for-fraud-detection/data

This has been used for applying and evaluating different descriptive statistics and machine learning methods, specifically Logistic Regression and K-means Clustering. The report aims to provide a comprehensive understanding of the dataset through detailed visualizations and explanations. Each section will walk you through the application of these methods, offering insights and interpretations for a clearer understanding of the underlying patterns and trends within the data.

DESCRIPTIVE STATISTICS

1.1 Introduction

Descriptive statistics provide an overview of the dataset by summarizing its main features through measures of central tendency, dispersion, and distribution.

1.2 Descriptive Statistics of the Dataset

In this section, we present the key descriptive statistics for the bank transaction dataset, focusing on the variables: Transaction Amount, Customer Age, Transaction Duration, Login Attempts, and Account Balance. These statistics offer valuable insights into the data distribution and help identify any patterns or anomalies that may influence further analysis.

	TransactionAmount	CustomerAge	TransactionDuration	LoginAttempts	AccountBalance
count	2512.000000	2512.000000	2512.000000	2512.000000	2512.000000
mean	297.593778	44.673965	119.643312	1.124602	5114.302966
std	291.946243	17.792198	69.963757	0.602662	3900.942499
min	0.260000	18.000000	10.000000	1.000000	101.250000
25%	81.885000	27.000000	63.000000	1.000000	1504.370000
50%	211.140000	45.000000	112.500000	1.000000	4735.510000
75%	414.527500	59.000000	161.000000	1.000000	7678.820000
max	1919.110000	80.000000	300.000000	5.000000	14977.990000

The statistics displayed above are explained in detail below, providing a deeper understanding of the dataset's key characteristics:

- **Transaction Amount:** The average transaction amount is 297.59, with a wide range from 0.26 to 1919.11, indicating significant variability. Most transactions are below 414.53.
- **Customer Age:** The average age is 44.67, with a range from 18 to 80, covering a broad age spectrum, with most customers being middle-aged.
- **Transaction Duration:** The average transaction duration is 119.64 seconds, with a range of 10 to 300 seconds, reflecting varying transaction complexities.
- **Login Attempts:** The average number of login attempts is 1.12, with a low standard deviation, suggesting most customers make few attempts, with some possibly retrying after failures.

- **Account Balance:** The average account balance is 5114.30, with a significant range from 101.25 to 14,977.99, indicating considerable variation, with most balances below 7,678.82.

The descriptive statistics provide key insights into the bank transaction dataset, highlighting the wide range of transaction amounts, the distribution of customer ages, and the variability in account balances. Most transactions are of moderate size, and the customer base is primarily middle-aged, with diverse transaction durations. These insights form a crucial foundation for further analysis, such as fraud detection and customer segmentation, by helping to uncover underlying patterns in the data.

LOGISTIC REGRESSION

2.1 Introduction

Logistic Regression is a method used to understand the relationship between a dependent variable (the outcome we want to predict) and one or more independent variables (the factors that might influence that outcome). It's especially useful when the outcome is binary meaning there are only two possible outcomes, like "yes" or "no," "success" or "failure." In the case of a bank transaction dataset, this could be something like "fraud" or "non-fraud."

When applied to a bank's data, Logistic Regression can predict whether a transaction is likely to be fraudulent or legitimate. It takes into account factors like the transaction amount, customer age, and account balance to make its prediction. The model then gives a probability between 0 and 1, closer to 1 means a higher chance the transaction is fraud, and closer to 0 means it's more likely to be legitimate.

So, using Logistic Regression in this context helps the bank make smarter decisions about flagging transactions, potentially stopping fraud before it happens.

2.2 Determining the Accuracy

From the dataset, 80% of the data was used for training the model, while the remaining 20% was allocated for testing. The following outputs were obtained from the analysis:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	484
1	0.95	1.00	0.97	19
accuracy			1.00	503
macro avg	0.97	1.00	0.99	503
weighted avg	1.00	1.00	1.00	503

The following metrics provide a comprehensive evaluation of the model's performance in predicting fraudulent and non-fraudulent transactions based on the dataset:

1. **Precision, Recall, F1-Score, and Support:**

- Precision (0): 1.00 - For the class labeled as 0 (likely non-fraudulent transactions), this means that when the model predicted a transaction as class 0, it was correct 100% of the time.
- Recall (0): 1.00 – This indicates that the model identified all the actual class 0 transactions correctly.

- F1-Score (0): 1.00 - The F1-score is the harmonic mean of precision and recall. A perfect score of 1.00 means that the model's performance for class 0 is optimal.
- Precision (1): 0.95 - For class 1 (likely fraudulent transactions), the precision is 0.95, meaning 95% of the time when the model predicted a fraud, it was correct.
- Recall (1): 1.00 - The recall for class 1 is 1.00, indicating that the model correctly identified all fraudulent transactions.
- F1-Score (1): 0.97 - The F1-score for class 1 is 0.97, which is also very good, balancing the precision and recall for the minority class (fraudulent transactions).

2. Accuracy:

The overall accuracy of the model is 1.00, meaning that it correctly predicted 100% of the instances across both classes. However, this can be misleading if the dataset is imbalanced (which it seems to be, given the low number of fraudulent transactions).

3. Macro Average:

Macro Average gives an average of precision, recall, and F1-score across all classes, treating each class equally regardless of how many instances it has. Here, the macro averages show:

1) Precision: 0.97
2) Recall: 1.00
3) F1-Score: 0.99

4. Weighted Average:

Weighted Average takes into account the class distribution (i.e., how many instances there are in each class). Since there are more non-fraudulent transactions (0), the weighted averages are very high:

1) Precision: 1.00
2) Recall: 1.00
3) F1-Score: 1.00

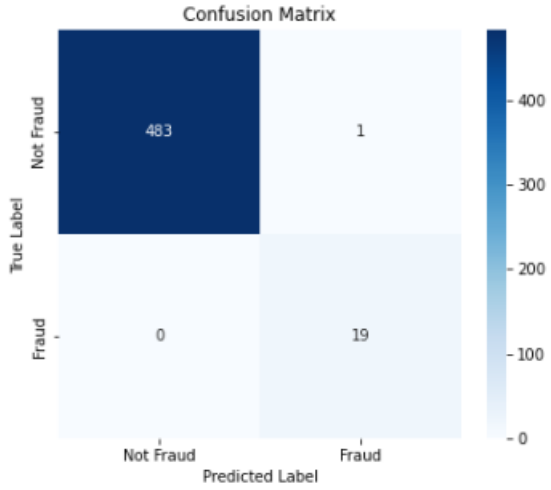
5. ROC AUC (Area Under the Curve):

The ROC AUC score is 0.99897, which is very close to 1, indicating that the model has excellent discriminatory ability and can distinguish between fraudulent and non-fraudulent transactions very well.

These results highlight the model's strong performance, particularly in identifying fraudulent transactions (Class 1) while maintaining perfect precision, recall, and F1-scores for non-fraudulent transactions (Class 0). The high ROC AUC score further confirms the model's ability to effectively discriminate between the two classes, making it a reliable tool for fraud detection.

2.3 Analysis using the Confusion Matrix

The analysis is based on the confusion matrix, which is a table used to evaluate the performance of a classification model by comparing actual and predicted labels, and is explained in detail below:



- **High Accuracy:**

The model performs exceptionally well, with only 1 false positive and no false negatives. It correctly identifies almost all "Not Fraud" and "Fraud" transactions.

- **Recall for Fraud:**

Since there are no false negatives, the recall (sensitivity) for detecting fraud is 100%.

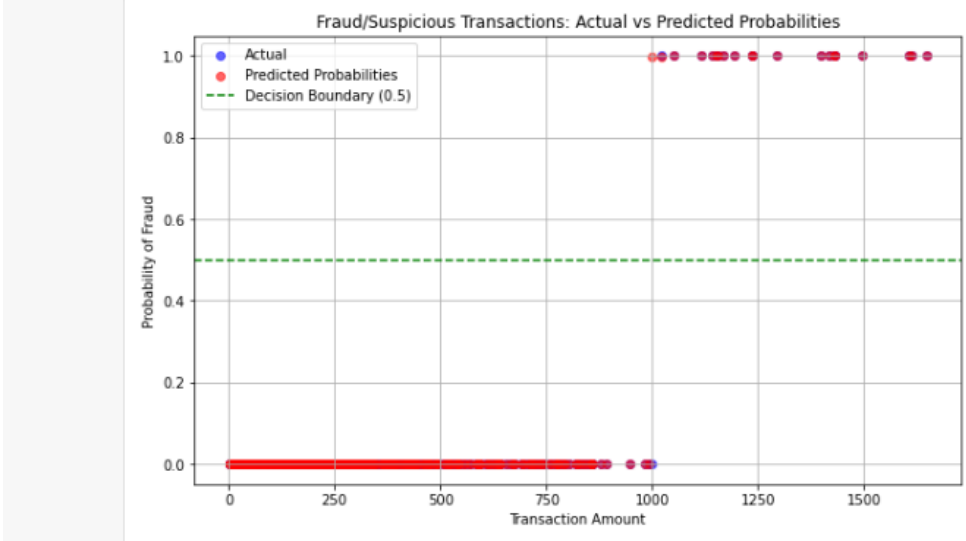
- **Precision for Fraud:**

With 1 false positive, the precision for detecting fraud is slightly less than perfect but still very high.

In conclusion we can see that the confusion matrix confirms that your model is highly effective at classifying transactions as fraudulent or not. However, given the dataset's class imbalance (many more "Not Fraud" than "Fraud"), be cautious of overfitting. The model may be overly confident in predicting "Not Fraud" due to the larger class size.

2.4 Analysis using the Scatterplot

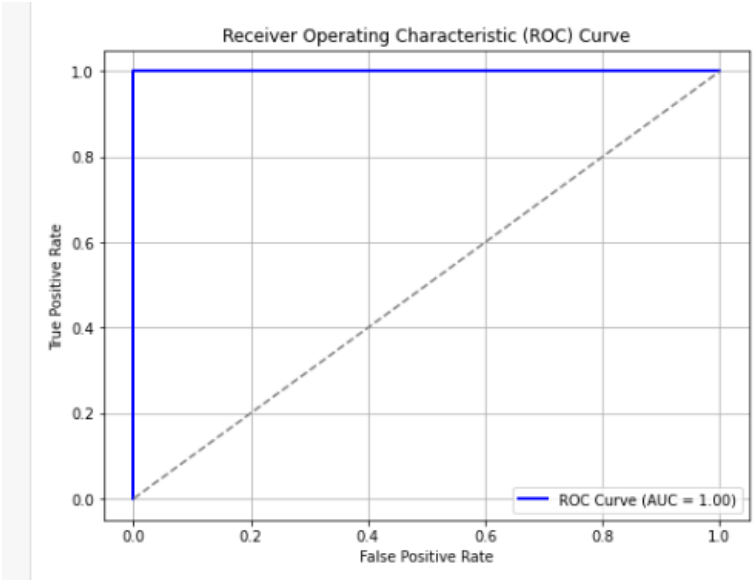
This section presents the scatterplot, illustrating its role in the analysis and demonstrating the accuracy of the model.



The scatter plot reveals that most data points are concentrated near the probability extremes (0 or 1), indicating the model's high confidence in its predictions. Notably, transactions with a 'TransactionAmount' exceeding the fraud threshold (1000 in this case) exhibit predicted probabilities close to 1, suggesting a strong correlation with potential fraudulent activity.

2.5 Analysis using the ROC Curve

This stage showcases the ROC Curve, emphasizing its significance in the analysis and reflecting the model's precision.



The ROC curve approaches the top-left corner, indicating that the model achieves a high True Positive Rate (TPR) while maintaining a low False Positive Rate (FPR). An AUC score of 1.00 signifies that the model perfectly distinguishes between fraudulent and non-fraudulent cases within this dataset.

K-means Clustering

3.1 Introduction

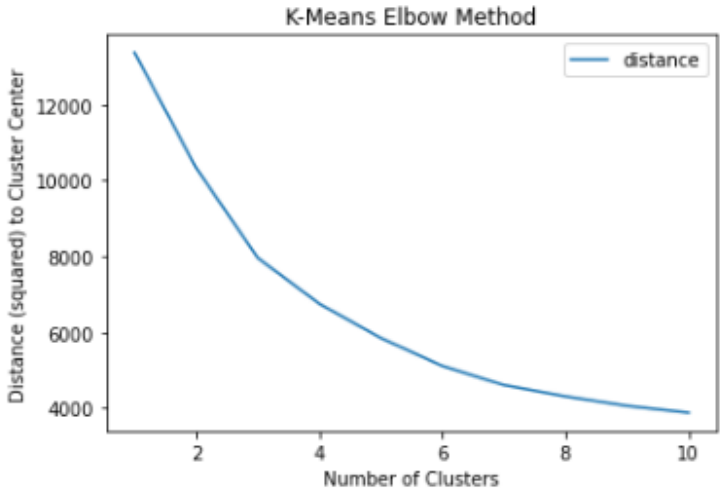
The banking dataset under analysis includes variables such as Transaction Amount, Customer Age, Transaction Duration, Login Attempts, and Account Balance. Applying K-Means clustering to this data can help identify groups of customers with similar behaviors and financial profiles. These insights can aid in addressing key objectives, such as:

- Customer Segmentation: Grouping customers into clusters for tailored marketing strategies.
- Risk Assessment: Identifying clusters that might represent high-risk customers.
- Fraud Detection: Detecting unusual transaction patterns that deviate from established clusters.
- Service Optimization: Offering personalized services based on cluster-specific behaviors.

By leveraging K-Means clustering, this analysis aims to provide a clearer understanding of customer behavior, optimize banking operations, and enhance decision-making processes. The results will support the development of strategies that align with customer needs and business goals.

3.2 Analysis using the Elbow Curve

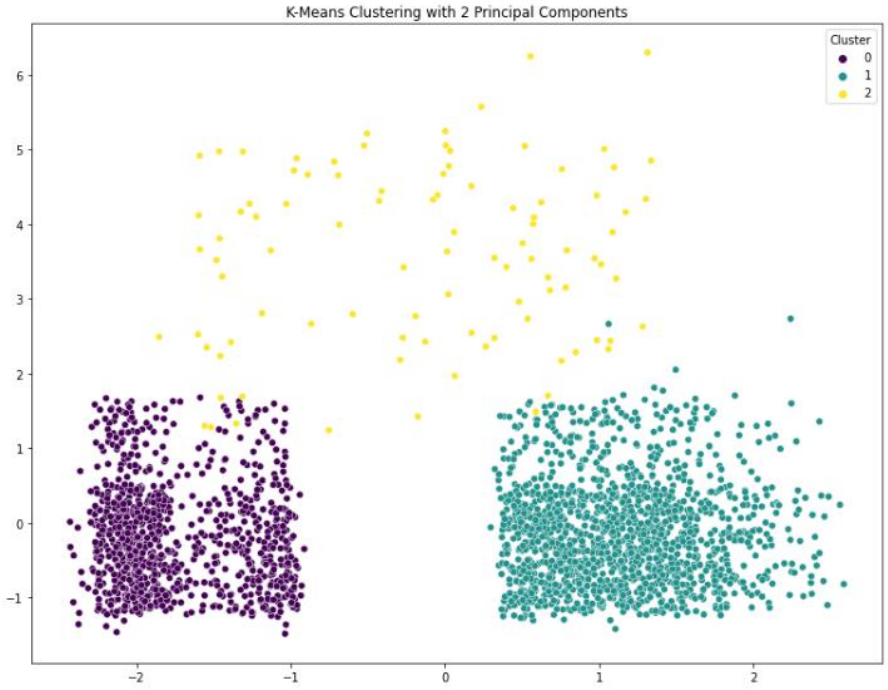
The Elbow Curve is used in this analysis to determine the optimal number of clusters by plotting the variance explained against the number of clusters, helping to identify the point where adding more clusters provides diminishing returns.



The analysis using the Elbow Curve reveals a steep decline in the Within Cluster Sum of Squares (WCSS) as the number of clusters increases from 1 to around 3 or 4, indicating that adding clusters reduces the variance within the clusters. The "elbow" point, where the rate of decrease in WCSS slows down, occurs around 3 or 4 clusters, suggesting that increasing the number of clusters beyond this point does not significantly enhance clustering quality. Thus, the optimal number of clusters is typically chosen as 3 or 4, as it strikes a balance between simplicity and performance. This indicates that the dataset likely contains 3 or 4 meaningful clusters, and further increasing the number of clusters may lead to overfitting without substantial improvement.

3.3 Analysis using the Scatterplot

The analysis using the scatterplot visualizes the results of the K-Means clustering algorithm, displaying how the data points are grouped into distinct clusters. Each cluster is represented by a different color, allowing for an intuitive understanding of how the algorithm has segmented the data based on the chosen number of clusters. This visualization helps assess the effectiveness of the clustering process and provides insights into the distribution and separation of the data.



Cluster 0 represents a group of customers with similar behaviors, characterized by moderate transaction amounts, younger customer ages, mid-range account balances, and shorter transaction durations. Cluster 1, although stacked, could represent customers with slightly different patterns, such as slightly older individuals with larger transaction amounts, longer

durations, and higher account balances. Cluster 2, being well-separated, may represent a distinct group with larger transaction amounts, more varied customer ages, higher account balances, and different transaction durations, possibly including outliers such as premium customers or business accounts.

CONCLUSION

This analysis of the bank transaction dataset provided valuable insights into customer behavior and fraud detection. Descriptive statistics revealed key trends in transaction amounts, customer age, and account balances, setting the stage for further analysis. Logistic Regression demonstrated excellent performance in predicting fraudulent transactions with high accuracy, precision, and recall, particularly for the minority class (fraudulent transactions). K-Means clustering identified distinct customer segments, with the optimal number of clusters being 3 or 4, aiding in risk assessment and targeted marketing. Overall, the combination of Logistic Regression and K-Means clustering proved effective for fraud detection, customer segmentation, and improving banking services.