**Anomaly Detection in Credit Card Transactions using Power BI**

#### 

**Objective:**

* Develop a Power BI dashboard to analyze credit card transactions, detect anomalies, and visualize transaction patterns.

#### **Data Preprocessing:**

**1. Data Cleaning:**

* Handling Missing Values: Utilize Power Query Editor to remove, replace, or impute missing values using functions like "Replace Values" or "Fill Down".
* Removing Duplicates: In Power Query Editor, use the "Remove Duplicates" function to eliminate duplicate rows based on selected columns.
* Ensuring Data Integrity: Apply data validation rules and constraints within Power Query Editor or Power BI Data Model to ensure data consistency and accuracy, such as defining data types, setting relationships, and creating calculated columns to validate data integrity.

**2. Data Modeling:**

* In this particular data model there exists only single table which consist of the following fields:  
  step,type,Amount,nameOrig,oldbalanceOrg ,newbalanceOrig ,nameDest   
  oldbalanceDest,newbalanceDest ,isFraud

#### **DAX Function: 1.Row Context and Filter Context** Row Context: Refers to the current row being evaluated in a calculation in DAX, processed row by row. Filter Context: Refers to the set of filters applied to the data affecting DAX expression calculations. DAX functions related to filtering: - FILTER - CALCULATE

**2.DAX Functions:**

* Calculated average transaction amount for normal and fraudulent transactions.
* Counted total credit card transactions and fraudulent transactions.
* Determined the highest fraud transaction amount.
* Compared maximum transaction amounts for normal and fraudulent transactions.
* Calculated the percentage of fraudulent transactions.
* Created a clustered column chart for the distribution of transaction amounts.

#### DAX Formulas Explanation:

#### DAX Function 1: Average Transaction Amount for Normal vs. Fraudulent Transactions

Average Amount Normal = CALCULATE(AVERAGE('YourTableName'[amount]), 'YourTableName'[isFraud] = 0)

Average Amount Fraudulent = CALCULATE(AVERAGE('YourTableName'[amount]), 'YourTableName'[isFraud] = 1)

#### DAX Function 2: Count of Total and Fraudulent Transactions

Total Transactions = COUNTROWS('YourTableName')

Fraudulent Transactions = CALCULATE(COUNTROWS('YourTableName'), 'YourTableName'[isFraud] = 1)

#### DAX Function 3: Highest Fraud Transaction Amount

Highest Fraud Amount = MAXX(FILTER('YourTableName', 'YourTableName'[isFraud] = 1), 'YourTableName'[amount])

#### DAX Function 4: Maximum Transaction Amounts for Normal vs. Fraudulent Transactions

Max Amount Normal = MAX('YourTableName'[amount])

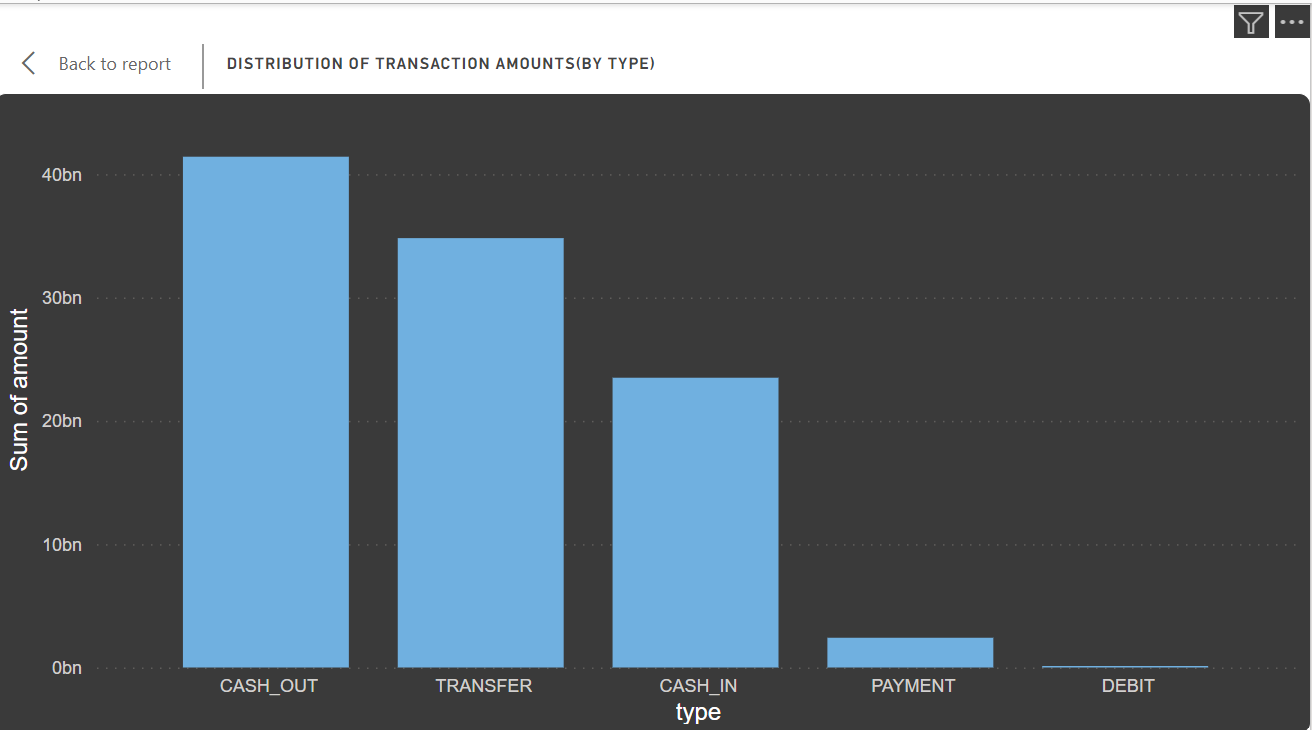
Max Amount Fraudulent = MAXX(FILTER('YourTableName', 'YourTableName'[isFraud] = 1), 'YourTableName'[amount])

#### DAX Function 5: Percentage of Fraudulent Transactions

Percentage Fraudulent Transactions = DIVIDE([Fraudulent Transactions], [Total Transactions])

#### DAX Function 6: Distribution of Transaction Amounts (Clustered Column Chart)

Transaction Amount Bins = ROUNDUP('YourTableName'[amount]/100, 0)\*100

Transaction Amount Distribution = COUNTROWS(FILTER(ALL('YourTableName'), 'YourTableName'[Transaction Amount Bins] = EARLIER('YourTableName'[Transaction Amount Bins])))  
  


These DAX formulas provide calculations for the specified questions in the Power BI dashboard, enabling the generation of insightful visualizations and metrics. Adjust the table and column names in the formulas based on your actual dataset structure.  
  
  
 **Anomaly Visualization:**  
 **1. Field Parameters and Slicers:**  
Field Parameters: Allow dynamic changing of analyzed fields in Power BI visuals or calculations.

Usage: Useful for comparing metrics, analyzing different dimensions, or customizing visualizations based on user preferences.

In the current project: Field parameters can be created with the usages of calculated measures like average of normal transaction amount , average of fraudulent transactions amount and count of transactions based on each type of transaction.

Slicer: Interactive filtering tool in Power BI for narrowing down data displayed in visuals.

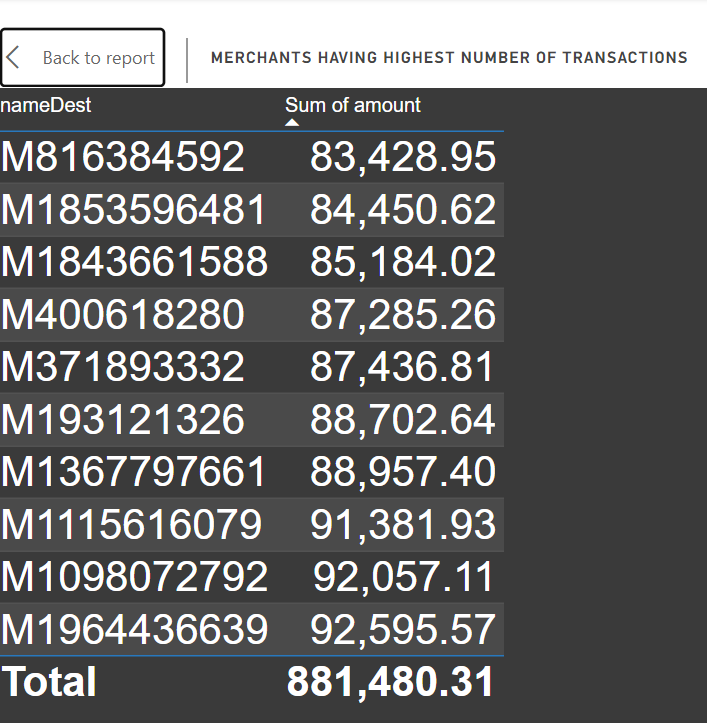
Usage: Provides users control over filtering data in reports, focusing on specific subsets dynamically.

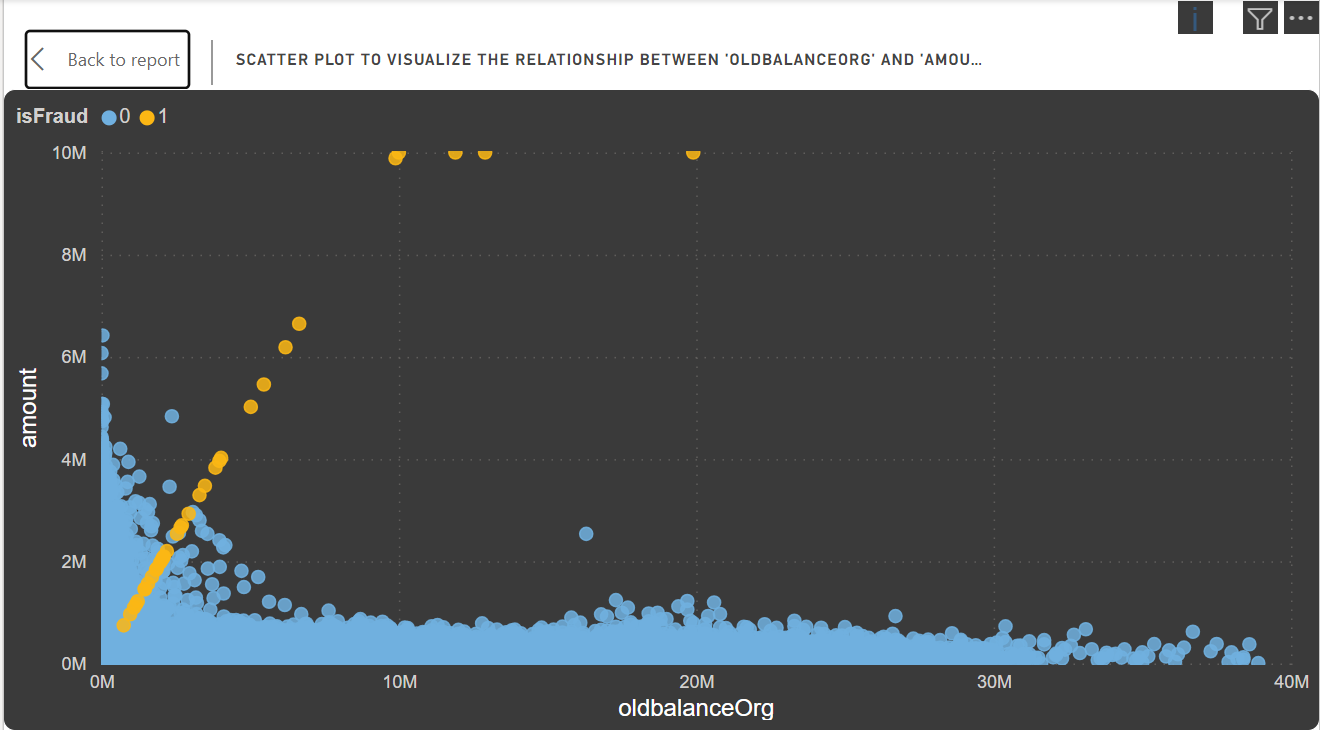
In the current project: Added slicer from Visualizations pane for transaction type and Isfraud fields ,customized appearance and behavior using Power BI formatting options.

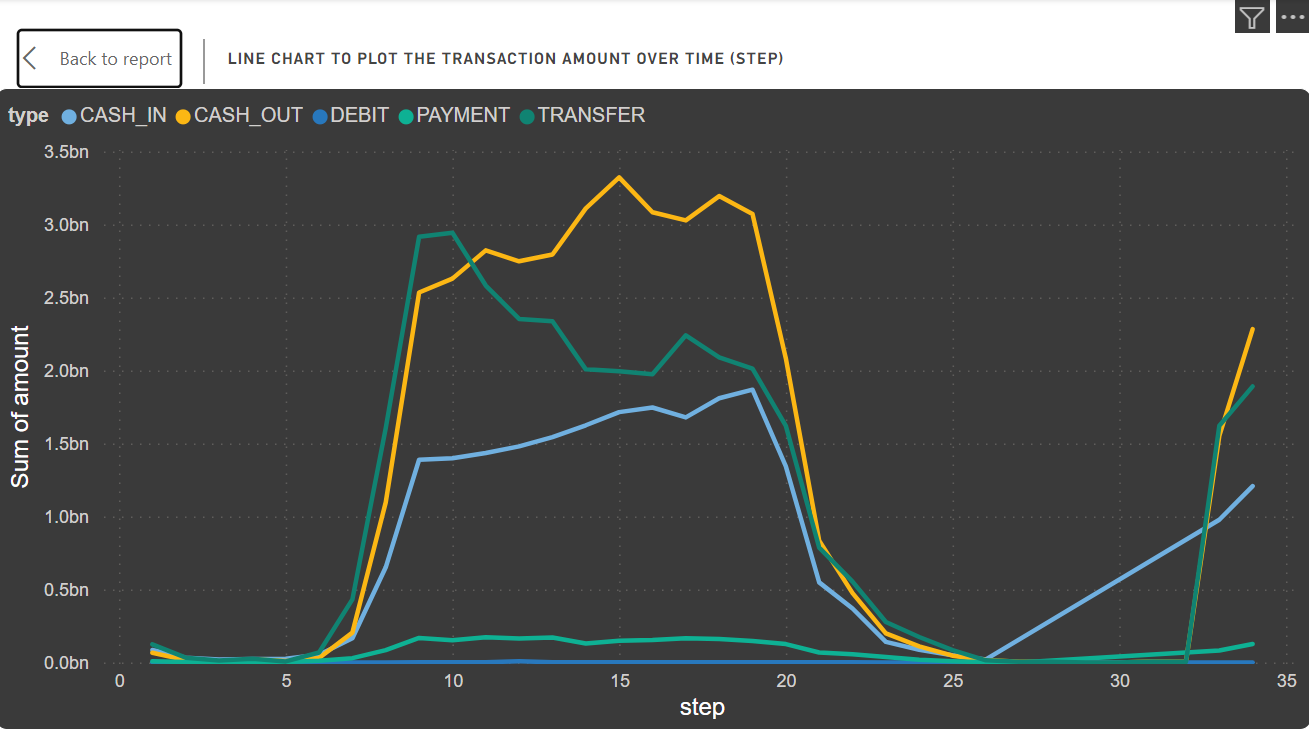
**2. Anomaly Visualization**

* Implemented visualizations (line charts, scatter plots, heat maps) to highlight potential anomalies and outliers in credit card transactions.

#### Key Insights:

1. Merchant Transactions:
   1. Setup:
      1. Fields: 'dest' (merchant names), Sum of amounts (total transaction amounts)
      2. Advanced Filtering: Name starts with 'm', Top 10
   2. Explanation:
      1. Field: 'dest' for merchant names, Sum of amounts for total transaction amounts
      2. Filter: Focus on merchants starting with 'm', Top 10 selection
   3. Interpretation:
      1. Concise view of top-contributing merchants
      2. Useful for quick comparison
      3. Enables exploring transaction patterns for specific merchants.
2. Scatter Plot (oldbalanceOrg vs. amount):
   1. Purpose:
      1. Explore relationship between 'oldbalanceOrg' and 'amount'
   2. Why:
      1. Visualize distribution and relationship between two continuous variables
      2. Identify patterns, clusters, and outliers
   3. Interpretation:
      1. Clustering of points indicates common transaction behaviors
      2. Outliers or patterns may suggest fraudulent activities



1. Line Chart (Transaction Amount Over Time):
   1. Purpose:
      1. Illustrate trends in transaction amounts over time
   2. Why:
      1. Suitable for visualizing trends over time
      2. Identify unusual spikes or drops in transaction amounts
   3. Interpretation:
      1. Sudden spikes or drops may represent abnormal transaction behavior
      2. Consistent trends indicate regular transaction patterns  
         
2. Table for Merchants with High Occurrence of Fraudulent Transactions:
   1. Setup:
      1. Fields: 'dest' (merchant names), Sum of amounts (total transaction amounts), 'isFraud' (fraud indicator)
      2. -Advanced Filtering: Name starts with 'm', 'isFraud' equals 1
   2. Explanation:
      1. Field: 'dest' for merchant names, Sum of amounts for total transaction amounts, 'isFraud' for indicating fraudulent transactions
      2. Filter: Conditions set for merchants starting with 'm' and 'isFraud' equal to 1
   3. Interpretation:
      1. No output suggests no 'm'-starting merchants have high fraudulent transaction occurrences
      2. Next Steps: Check data for 'm'-starting merchants with valid 'isFraud' values, adjust filtering if needed

### **Conclusion:**

* A total of 630,894 transactions were recorded, including 383 fraudulent transactions, which account for approximately 0.06% of the total transactions.

* Transactions are classified into five categories: cash in, cash out, debit, payment, and transfer. Notably, fraudulent activities predominantly occurred in 'transfer' and 'cash out' transactions, with around $169 million and $168 million, respectively, totaling $337.65 million in fraudulent amounts. This emphasizes the importance of enhancing security measures for these transaction types.
* 'Cash out' transactions were the most prevalent, totaling 224,013 transactions amounting to roughly $41 billion. Conversely, 'debit' transactions were the least common, comprising only 4,769 transactions with a total value of approximately $29 million.
* Analysis of transaction volumes reveals a concentration within the 10 to 20 step range, representing the highest total transaction value at approximately $7.3 billion. A significant portion of this amount, approximately $3.3 billion, falls under the 'cash out' category, indicating a substantial subset of high-value transactions.
* The top 10 merchants with the highest number of transactions include 'M1964436639', 'M1098072792', and 'M1115616079', with the highest transaction amounts of $93K, $92K, and $91K, respectively.