

Steps of the credit card fault detection case study

1. Imported necessary packages: Pandas, NumPy, Seaborn, Plotly Express, and Matplotlib.
2. Loaded the dataset named "Fraud.csv" into a DataFrame and displayed its shape and info.
3. Checked for null values in the dataset.
4. Analyzed the distribution of the 'type' column using value counts and a countplot.
5. Analyzed the 'isFlaggedFraud' column, which had only 0 values except for a single instance of 1.
6. Analyzed the 'isFraud' column, which represents whether a transaction is fraudulent (1) or not (0).
7. Dropped the 'isFlaggedFraud' column as it had very little variation and wouldn't contribute significantly to the analysis.
8. Created a new DataFrame 'fraud_df' containing only the transactions marked as fraudulent.
9. Visualized the distribution of fraudulent transactions based on the 'type' column using a histogram and a countplot.
10. Replaced all transaction types other than "TRANSFER" and "CASH_OUT" with "OTHER" in the 'type' column.
11. Created two new columns 'NaOr' and 'NaDe' by extracting the first character from 'nameOrig' and 'nameDest' columns, respectively.
12. Explored the types of transactions happening between customers and merchants in the 'NaDe' column.
13. Dropped irrelevant columns ('NaOr', 'nameOrig', 'nameDest', 'step', 'oldbalanceOrg', 'newbalanceDest') from the DataFrame.
14. Encoded categorical variables using one-hot encoding for the 'type' column.
15. Split the data into training and testing sets and performed SMOTE oversampling on the training data to handle class imbalance.
16. Preprocessed the data using StandardScaler for scaling numerical features.
17. Built and evaluated four different classifiers: Logistic Regression, Decision Tree, Random Forest, and K-NN.