Project Assignment

Data Cleaning and Preprocessing

Scenario

You are working as a data analyst in a customer analytics team for an e-commerce company. The raw data collected from multiple sources has issues such as missing values, outliers, unscaled data, and lack of derived features. Your job is to clean, preprocess, and enrich the data for downstream analysis or machine learning pipelines.

You are given a file: `customer_insights_raw.csv` containing 50 rows of customer data.

Section 1: Missing Value Handling

Objective: Identify and handle missing values through appropriate techniques.

Tasks:

1. Load the dataset using pandas and generate a summary using `.info()` and `.isnull().sum()`.
2. Identify columns with missing values and classify them as:
   - Numerical
   - Categorical
3. Apply the following imputation strategies:
   - For numerical columns: Use median and mean imputation separately and compare.
   - For categorical columns: Use mode imputation.
4. Drop records (rows) where more than 2 values are missing.
5. Save the cleaned version as `customer_cleaned_imputed.csv`.

Bonus Task: Create a heatmap to visualize missing values before and after imputation.

Section 2: Outlier Detection and Handling

Objective: Detect and handle outliers using Z-Score, IQR, and Capping techniques.

Tasks:

1. Select the following numerical columns:
    - `AnnualIncome`
    - `AvgTransactionValue`
    - `LoginFrequency`
2. Apply **Z-score method** to detect outliers. Define threshold = 3.
3. Apply **IQR method** and print the lower and upper bounds for each column.
4. Use **capping and flooring** to treat extreme values based on calculated bounds.
5. Visualize each column's distribution **before and after outlier handling** using boxplots.
6. Save the transformed dataset as `customer_outliers_handled.csv`.

Bonus Task: Create a function `handle_outliers(df, column)` that accepts a column and applies IQR-based treatment.

Section 3: Data Normalization

Objective: Normalize the dataset to bring all numerical features to a comparable scale.

Tasks:

1. Select the following columns:
    - `AnnualIncome`
    - `AvgTransactionValue`
    - `LoginFrequency`
2. Apply **Min-Max scaling** on a copy of the dataset.

3. Apply **Z-score standardization** (StandardScaler) on another copy.
4. Compare the histograms of original vs scaled vs standardized values.
5. Save both versions as:
   - `customer_minmax_scaled.csv`
   - `customer_standard_scaled.csv`

Bonus Task: Write a function that returns a DataFrame after applying the specified scaling method: `"minmax"` or `"standard"`.

Section 4: Feature Engineering

Objective: Derive new features from existing data to improve insights or modeling.

Tasks:

1. Create a new feature `TotalSpend = TotalTransactions * AvgTransactionValue`.
2. Create age bands such as: `18-25`, `26-35`, `36-45`, `46+`.
3. Convert `IsChurned` column to binary: `1` for Yes, `0` for No.
4. Create a flag column `HighValueCustomer` if `TotalSpend` > 90th percentile.
5. Calculate average transaction frequency per month using:
   `MonthlyTransactionRate = TotalTransactions / (LoginFrequency / 4)`
6. Derive a new score:
   `CustomerScore = (TotalSpend * SatisfactionScore) / CartAbandonRate`

Bonus Task: Export a final version of enriched dataset as `customer_features_enriched.csv`.

Submission Expectations

- Submit all 4 CSV files mentioned in the tasks.
- Submit a Python script with:
  - Step-by-step implementation
  - Clear comments
  - Visualizations for each step

- Write a short document (or markdown) summarizing:
    - What preprocessing was done
    - Which columns were most impacted
    - Any interesting insight discovered during the process