Mini Project: Feature Engineering on Car Sales Dataset

Subject: Data Science
Name : Mohd Shahrukh
Roll No: 243301022
class : MCA 3rd semester

Introduction

Feature Engineering is a key process in data science that transforms raw data into meaningful features that can improve machine learning models.
In this mini-project, I selected a **Car Sales Dataset** containing columns like `Brand`, `Price`, `Body`, `Mileage`, `EngineV`, `Engine Type`, `Registration`, `Year`, and `Model`.
The focus of this project is on **data preprocessing and feature engineering** — not on model training or prediction accuracy.

Step 1: Data Cleaning

Definition:
Data cleaning is the process of identifying and correcting errors or inconsistencies in the dataset. It ensures the data is accurate, complete, and reliable for analysis.

Why it is needed:
Raw data often contains missing values, duplicates, outliers, or incorrect entries. Without cleaning, the analysis and feature engineering results could be misleading.

What I did:

- Checked for **missing values** and replaced missing `Price` values with the median and missing `Engine Type` with the most frequent type.
- Removed rows where the `Model` value was missing.
- Deleted **duplicate records** based on all columns.
- Filtered out **noisy data**, such as cars with unrealistic years or prices.

- Treated **outliers** in `Price` using the **IQR (Interquartile Range)** method to remove extreme values.

Example:
If a car's `Year` was 1975 or `Price` was extremely high compared to others, those records were removed to make the dataset more realistic.

Step 2: Data Integration

Definition:

Data integration combines multiple data sources into one consistent dataset by aligning columns, units, and entities.

Why it is needed:

If multiple datasets are used (e.g., sales from different regions), integration ensures consistent naming, units, and formats.

What I did:

Since I used a **single dataset**, this step was not required.

Step 3: Data Transformation

Definition:
Data transformation converts data into suitable formats or scales for analysis or model input. It includes encoding, scaling, and transforming features.

Why it is needed:
Most machine learning models require numeric and scaled input data. Transformation helps handle categorical values, normalize numerical ranges, and make distributions more uniform.

What I did:

- Applied **Label Encoding** on the `Model` column.
- Used **One-Hot Encoding** for categorical columns like `Brand`, `Body`, `Engine Type`, and `Registration`.
- Scaled numerical features (`Price`, `Mileage`, `EngineV`, `Year`) using **Min-Max Scaling**.
- Applied **Log Transformation** on `Price` and `Mileage` to reduce skewness.
- Created a new feature `Price_Level` by dividing prices into **Low**, **Medium**, and **High** categories (Discretization).

Example:
`Brand = Toyota` and `Body = SUV` were converted into numeric binary columns (0 or 1). `Price` values were scaled between 0 and 1 for uniform comparison.

Step 4: Data Reduction

Definition:
Data reduction simplifies the dataset by removing unnecessary features or compressing information while keeping important patterns.

Why it is needed:
Large datasets with many features can slow down computation and introduce redundancy. Reduction helps improve model efficiency and interpretability.

What I did:

- Removed less informative columns such as `Model`.
- Performed **correlation analysis** to identify and remove highly correlated variables.
- Removed columns with zero variance (same value for all rows).
- Applied **Principal Component Analysis (PCA)** to reduce dimensionality while retaining 95% of the data variance.

Example:
If `EngineV` and `Price` were highly correlated, one of them was dropped.
PCA converted all numeric features into a smaller set of new features that represent most of the dataset's information.