MOVEINSYNC ASSIGNMENT

Problem Statement:

A company plans to expand into new markets with five existing products. Based on market research, the new market's behavior is expected to resemble their current market.
In their current market, customers are divided into four segments (A, B, C, D), and targeted outreach for each segment has proven effective. The company now wants to apply the same segmentation strategy to 2,627 potential customers in the new market.
Your task is to assist in predicting the appropriate segment for each new customer.
Output should be filled csv file(test2.csv).

Dataset:

The dataset consists of 8068 entries, with 11 features describing demographic, professional, and lifestyle attributes of individuals, along with a segmentation label. Below is a summary of the dataset's columns and characteristics:

Columns:

- ID: A unique identifier for each individual (integer, no missing values).
- Gender: The gender of the individual (categorical, no missing values).
- Ever_Married: Indicates if the individual has ever been married (categorical, 7928 non-null, 140 missing values).
- Age: The age of the individual (integer, no missing values).
- Graduated: Whether the individual has graduated (categorical, 7990 non-null, 78 missing values).
- Profession: The profession of the individual (categorical, 7944 non-null, 124 missing values).
- Work_Experience: Number of years of work experience (float, 7239 non-null, 829 missing values).
- Spending_Score: The spending behavior of the individual, categorized (categorical, no missing values).

- Family_Size: The size of the individual's family (float, 7733 non-null, 335 missing values).
- Var_1: An anonymized variable (categorical, 7992 non-null, 76 missing values).
- Segmentation: The target variable, representing customer segmentation (categorical, no missing values).

Preprocessing and EDA:

- The ID column just serves as a unique identifier for each row and does not carry any predictive info.
- Missing value in train2 dataset:

```
Gender              0
Ever_Married      140
Age                 0
Graduated          78
Profession        124
Work_Experience   829
Spending_Score      0
Family_Size       335
Var_1              76
Segmentation        0
dtype: int64
```

Number of missing values for each columns

Initially, missing values in the dataset were identified. The columns Work_Experience, Family_Size, and Var_1 contained missing numerical values, while columns like Ever_Married, Graduated, and Profession had missing categorical values.
For numerical columns, the missing values were imputed using the median of each respective column.
For categorical columns, the missing values were imputed using the most frequent value (mode) of the respective column.

- Data transformation:
  The Var_1 column, initially a string with numeric characters, was transformed by extracting the numerical part and converting it into a float type.

The categorical columns Gender, Ever_Married, and Graduated were encoded into numerical values. This was done using the pd.Categorical method, where categories were encoded as integers (Male as 0, Female as 1, etc.).

The Spending_Score column, which had categories like 'Low', 'Average', and 'High', was mapped to numerical values (1, 2, 3, respectively).

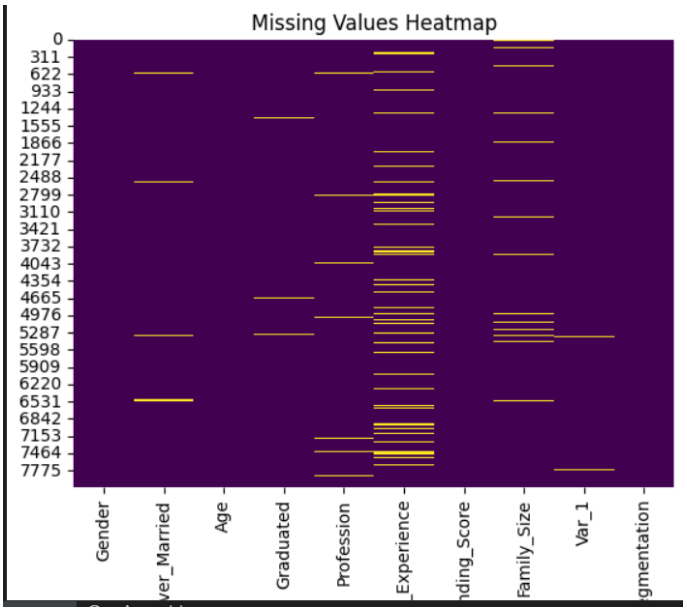
The Profession column was one-hot encoded, creating dummy variables to represent the different professions.

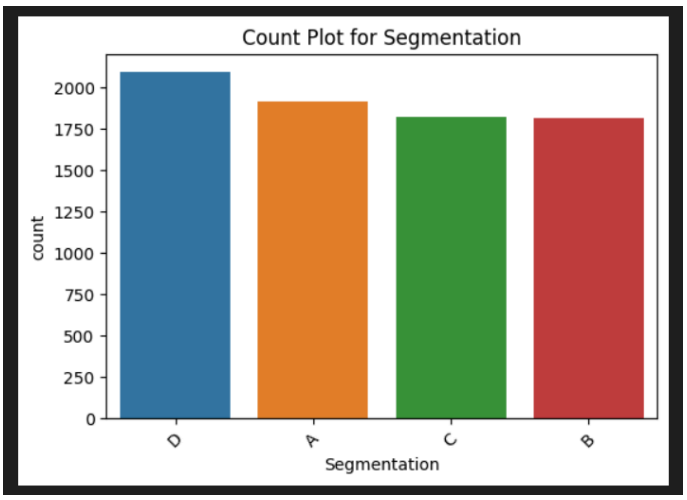
- Checking for duplicate values
- EDA:

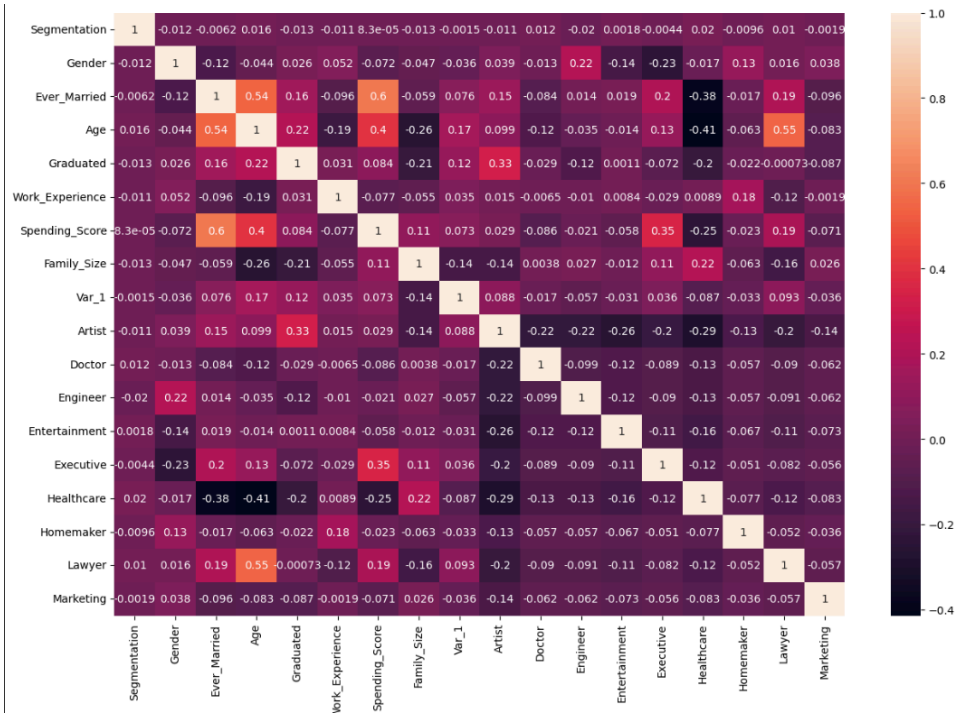
  The shape of the dataset and summary statistics were examined to understand the distribution of features. The correlation matrix was plotted to check for correlations between numerical features. This helped in identifying any strong correlations that might lead to multicollinearity in the models.But there is not much correlation between the columns. Count plots were used to visualize the distribution of categorical features, providing insights into the balance of the dataset.

  A heatmap was created to check for missing values across the dataset, which confirmed that missing values were successfully imputed.

  As shown in figures below:







- Feature Scaling: Tried both StandardScaler and minmax Scale for continuous columns.Numerical features like Age, Work_Experience, and Family_Size were scaled using MinMaxScaler, which scaled the values between 0 and 1. This ensured that all numerical features had the same scale, making the data suitable for training machine learning models.

Methodology:

Created separate datasets for training and validation to ensure the model's performance is tested on unseen data.

- Different models like Random Forest,Gradient Boosting,SVM,Logistic Regression,K-Nearest Neighbors (KNN),XGBoost and LightGBM used to train and predict on validation set. Gradient Boosting and LightGBM performed best overall, with the highest accuracy of 53% and 51%. Hypertuned for different values of parameter for each model.

- Applied clustering algorithms also like k-means in which found optimal k-3 using elbow method and DBSCAN. Got the accuracy of 39.13% and 50.87% respectively.
- A grid search with cross-validation was performed for each model to tune hyperparameters. This helped to identify the best model configurations for each classifier.
- Applied PCA (95% variance maintained) but still got accuracy in range of 48-51%
- Performed Leave-One-Out Analysis but didn't get an improved result.

Conclusion:

The Gradient Boosting Classifier was identified as the best performing model based on the evaluation metrics (53%). This model showed the highest accuracy and robustness across the evaluation.

Predictions on the Test Set:

After identifying the best model (Gradient Boosting), predictions were made on a separate test set (test2.csv). The preprocessing steps applied to the training data (such as missing value imputation and scaling) were also applied to the test data to ensure consistency.

- The predicted segmentation labels were mapped back from numeric labels to their corresponding categorical values (A, B, C, D).
- The predictions were then added as a new column (Segmentation) to the original test dataset.

The final output was saved as a new CSV file (test2_predictions.csv), which contained both the original features and the predicted segmentation labels for each customer.