PHASE 2 : Innovation

Project Title: Customer Segmentation with Data Science

Problem Statement: Implement data science techniques to segment customers based on their behavior, preferences, and demographic attributes, enabling businesses to personalize marketing strategies and enhance customer satisfaction.

Problem Explanation: The problem at hand is to leverage data science techniques to effectively segment customers based on various aspects such as their behavior, preferences, and demographic attributes. The ultimate goal of this project is to empower businesses with the ability to tailor their marketing strategies in a personalized manner, ultimately leading to improved customer satisfaction and potentially increased sales.

You own the mall and want to understand the customers, like those who can easily converge [Target Customers], so that sense can be given to the marketing team and the strategy can be planned accordingly.

Information about the dataset:

Link: https://www.kaggle.com/datasets/akram24/mall-customers

Columns used for the Customer Segmentation Dataset:

- Customer ID:

  A unique identifier for each customer.

- Gender:

  Gender helps identify the customer's biological sex.

- Age:

  It indicates the customer's age group.

- Annual income:

  It represents the customer's yearly earnings.

- Spending Score:

  Spending score reflects the customer's tendency to spend money.

Libraries used:

- Pandas:

  For data manipulation and analysis.

  ```
  !pip install pandas
  import pandas pd
  ```

- Numpy:

  Essential for numerical operations and array manipulations.

  ```
  !pip install numpy
  import numpy as np
  ```

- Matplotlib and Seaborn:

  For data visualization,which is crucial in understanding customer patterns.

  ```
  !pip install matplotlib
  import matplotlib.pyplot as plt
  !pip install seaborn
  import seaborn as sns
  ```

- Os:

  Provides a way to interact with the operating system , allowing us to perform various file and directory operations.

  ```
  import os
  ```

TRAINING:

- Feed the training data (features and labels) into the chosen model.
- The model uses this data to adjust its internal parameters or coefficients to make predictions as close as possible to the actual target values.
- This process is often referred to as "fitting" the model to the data.
- To ensure the model's generalization ability and minimize overfitting, you can use techniques like k-fold cross-validation.
- Once the model is trained and evaluated to your satisfaction, you can deploy it to make predictions on new, unseen data.

TESTING:

- Testing a dataset typically refers to the process of evaluating the performance of a trained machine learning model on a dataset that it has not seen during the training process. This dataset is often called a "test set" or "validation set," and it's used to assess how well the model generalizes to new, unseen data.
- Similar to the training phase, you need to prepare the test dataset by splitting it into features (inputs) and labels (ground truth or target values).
- Load the machine learning model that you previously trained on the training dataset. This model should be ready to make predictions.
- Feed the test dataset (features) into the trained model.
- The model will produce predictions or outputs based on the test data.
- Compare the model's predictions to the actual target values (labels) in the test dataset to assess its performance.

METRICS USED FOR ACCURACY CHECK :

- **Rand Index:** The Rand index is used for comparing the similarity between true class labels and cluster assignments. It ranges from 0 to 1, with a higher score indicating better clustering.
- **Normalized Mutual Information (NMI):** NMI measures the mutual information between true class labels and cluster assignments, normalized to a value between 0 and 1. Higher NMI values indicate better clustering.
- **Precision, Recall, and F1-Score:** These classification metrics can be used in situations where the ground truth labels are known. You can compare the predicted cluster assignments to actual labels.
- **Visual Inspection:** In some cases, visual inspection of the clusters using scatter plots or other visualization techniques can be a valuable way to assess the quality of segmentation.
- **Inertia (Within-Cluster Sum of Squares)**: Inertia measures the sum of squared distances from each point to its assigned cluster center. Lower inertia indicates better clustering. However, this metric may not be ideal for all cases, as it tends to decrease with the number of clusters.
- **Dunn Index**: The Dunn index compares the minimum inter-cluster distance to the maximum intra-cluster distance. A higher Dunn index suggests better separation of clusters.
- **Calinski-Harabasz Index (Variance Ratio Criterion):** This index calculates the ratio of the between-cluster variance to within-cluster variance. Higher values are better, suggesting more distinct and well-separated clusters.