# Project Report: Customer Segmentation with K-Means Clustering

## 1. Title

Customer Segmentation with K-Means Clustering

## 2. Aim

To apply K-Means clustering to segment customers based on their purchase behavior.

## 3. Problem Statement

Businesses often seek ways to understand customer purchasing patterns to better tailor their marketing strategies and product offerings. By analyzing customer purchase data, distinct customer segments can be identified. This project aims to use K-Means clustering to classify customers into groups with similar purchasing behaviors, thereby assisting businesses in targeted marketing efforts.

## 4. Technologies Used

- Python: The programming language used for data analysis and machine learning.  
- Pandas: Library for data manipulation and analysis, used for loading and exploring data.  
- NumPy: Library for numerical operations, complementing pandas in data analysis.  
- Matplotlib & Seaborn: Libraries for data visualization, helping to create insights from clusters.  
- Scikit-Learn (Sklearn): Machine learning library in Python, particularly the KMeans function used for clustering.

## 5. Dataset

The dataset, named customer\_purchase\_data.csv, contains customer purchasing details, which provide the basis for clustering analysis.

## 6. Code Explanation

### Section 1: Importing Libraries

This section imports essential libraries:  
- Pandas and NumPy: For data manipulation and numerical operations.  
- Matplotlib and Seaborn: To create visualizations that help in analyzing and interpreting clusters.  
- KMeans from sklearn.cluster: For performing K-Means clustering.  
- StandardScaler from sklearn.preprocessing: For normalizing features, ensuring fair treatment in clustering.

### Section 2: Loading and Exploring the Dataset

This section loads and inspects the dataset:  
- pd.read\_csv: Loads the data into a DataFrame for further processing.  
- head(): Displays the first few rows, providing a quick view of the data structure.  
- info(): Displays details about columns, data types, and missing values.  
- describe(): Provides descriptive statistics, helping in identifying the range, mean, and spread of each feature.

### Section 3: Data Preprocessing

This section prepares data for clustering:  
- Handling Missing Values: Missing values are dropped to avoid clustering issues.  
- Scaling Data: The data is scaled using StandardScaler, normalizing all features to a similar range, as K-Means clustering is sensitive to feature scale.

### Section 4: Determining the Optimal Number of Clusters (k)

This section determines the optimal number of clusters, k:  
- Inertia Calculation: Runs K-Means with different values of k (1 to 10) and appends each model's inertia (sum of squared distances from each point to its cluster center) to a list.  
- Elbow Plot: Plots inertia values against k. The “elbow” point (where inertia begins to decrease more slowly) suggests the optimal number of clusters.

### Section 5: Applying K-Means Clustering

This section clusters customers based on purchase behavior:  
- Applying K-Means: With k=3 (chosen from the elbow method), the K-Means algorithm is applied to the scaled data.  
- Adding Cluster Labels: Each customer is assigned a cluster label, saved in a new column named Cluster, categorizing each customer into a specific segment.

### Section 6: Analyzing Cluster Characteristics

This section examines the characteristics of each cluster by calculating the mean feature values for each:  
- Cluster Centers: Retrieves the centers of each cluster in the original feature scale.  
- Cluster Summary: A DataFrame is created to represent the average feature values per cluster, providing insight into the distinguishing characteristics of each customer segment.

### Section 7: Visualizing Clusters

This section plots customer segments for visualization:  
- Scatter Plot: Uses Seaborn to create a scatter plot of customers based on two selected features (Feature1 and Feature2), with colors indicating different clusters. This helps visualize how the customers are distributed across clusters.

### Section 8: Saving Clustered Data

This section saves the DataFrame with cluster labels as a new file, customer\_segments.csv.  
- Exporting Results: This file can be shared or further analyzed, containing each customer’s segment assignment.

## Conclusion

This project successfully applies the K-Means clustering algorithm to group customers based on purchase behavior. Key insights include understanding the number of customer segments (k), identifying segment characteristics, and visualizing clusters for further interpretation. Each customer’s segment assignment is saved, enabling businesses to focus marketing strategies on specific customer groups effectively.

Colab Link : [https://colab.research.google.com/drive/1MlMAAu8etqMxS9wQXcRDllbI\_VmxqDKM?authuser=1#scrollTo=f5-NjT15HArc](https://colab.research.google.com/drive/1MlMAAu8etqMxS9wQXcRDllbI_VmxqDKM?authuser=1%23scrollTo=f5-NjT15HArc)