REPORT ON CUSTOMER SEGMENTATION AND CLUSTERING RESULTS

1. Introduction

- Customer segmentation helps businesses categorize customers into groups based on behavior and spending patterns.
- The goal is to identify distinct customer groups to improve marketing strategies, product recommendations, and customer service.
- In this analysis, we use **K-Means clustering** on transaction and profile data to segment customers into meaningful groups.

2.Goals of the Analysis

- Identify **distinct customer groups** using clustering techniques.
- Evaluate clustering performance using **Davies-Bouldin Index** and other metrics.
- Visualize customer segments for better interpretation

3. Data Overview

2.1 Datasets Used

Customers.csv

This dataset contains customer profile information. Key features:

- CustomerID – Unique customer identifier.
- Other demographic attributes (if available).

Transactions.csv

This dataset records customer transactions. Key features:

- CustomerID – Foreign key to match customers.
- TransactionID – Unique transaction identifier.
- TotalValue – Transaction amount.

4. Data Preprocessing & Feature Engineering

We generate three key features from transaction data:

1. **Total Spend** (total_spend): Sum of all transaction values for a customer.
2. **Average Spend per Transaction** (avg_spend): Mean value per transaction.
3. **Total Transactions** (total_transactions): Count of total transactions by a customer.

REPORT ON CUSTOMER SEGMENTATION AND CLUSTERING RESULTS

Feature Scaling

Since our features have different scales, we normalize them using **MinMaxScaler**:

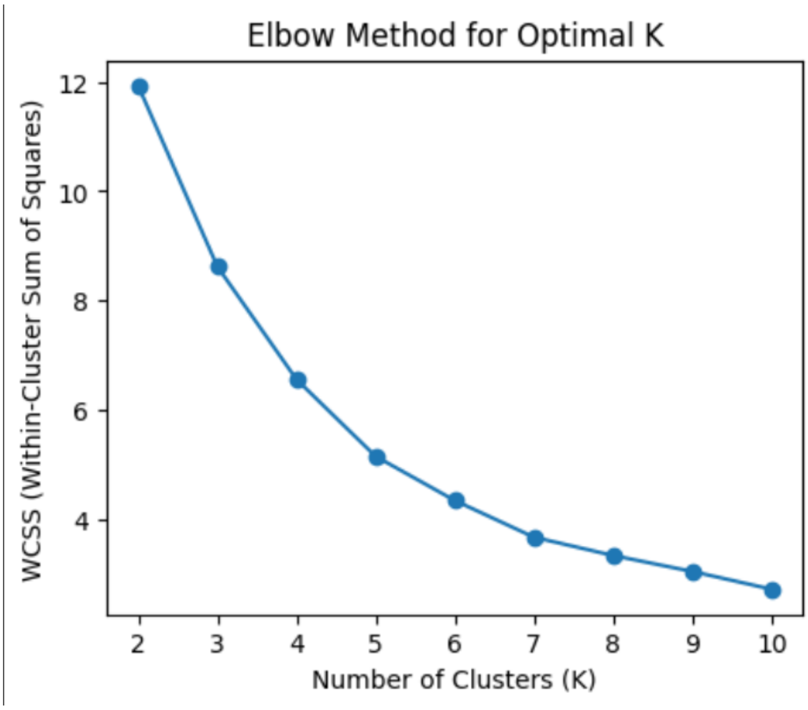
```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

customer_data_scaled[features] = scaler.fit_transform(customer_data[features])
```

5. Clustering Approach

5.1 Choosing the Optimal Number of Clusters (K)

We use the **Elbow Method** to determine the best value for K. The Within-Cluster Sum of Squares (WCSS) is calculated for different K values.

```
#K-Means Clustering (Finding Optimal K using WCSS)

wcss = []

K_range = range(2, 11)

For  k in K_range:

   kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)

   kmeans.fit(customer_data_scaled[features])

   wcss.append(kmeans.inertia_)
```



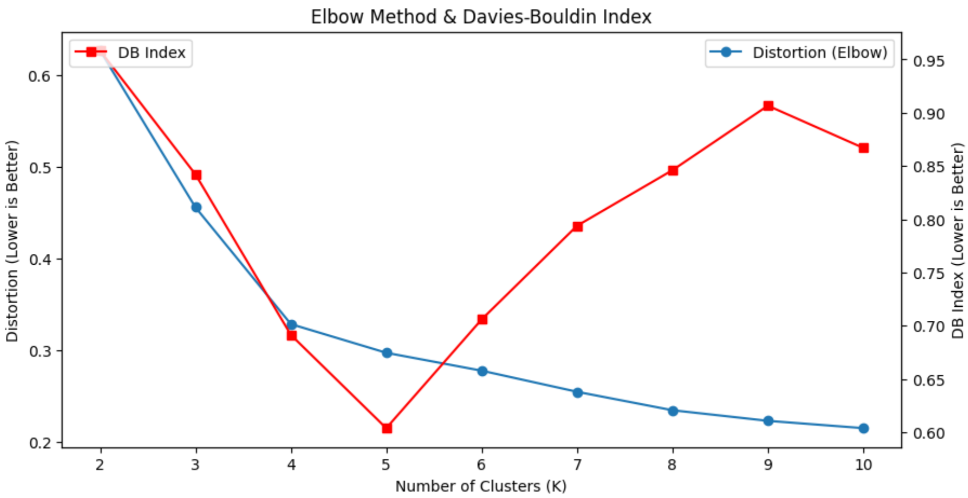
REPORT ON CUSTOMER SEGMENTATION AND CLUSTERING RESULTS

6. Evaluation of Clustering:

6.1 Davies-Bouldin Index:

- The **Davies-Bouldin Index (DBI)** evaluates the compactness and separation of clusters. A **lower DB Index** indicates better clustering.

```
distortions = []
db_indexes = []
K_range = range(2, 11)

for k in K_range:
   kmeans = KMeans(n_clusters=k, random_state=42, n_init=10)
   kmeans.fit(customer_data_scaled[features])

   distortions.append(sum(np.min(cdist(customer_data_scaled[features], kmeans.cluster_centers_, "euclidean"), axis=1)) / customer_data_scaled[features].shape[0])
   db_indexes.append(davies_bouldin_score(customer_data_scaled[features], kmeans.labels_))

fig, ax1 = plt.subplots(figsize=(10, 5))
ax2 = ax1.twinx()

ax1.plot(K_range, distortions, marker="o", label="Distortion (Elbow)")
ax2.plot(K_range, db_indexes, marker="s", color="red", label="DB Index")

ax1.set_xlabel("Number of Clusters (K)")
ax1.set_ylabel("Distortion (Lower is Better)")
ax2.set_ylabel("DB Index (Lower is Better)")
ax1.set_title("Elbow Method & Davies-Bouldin Index")

ax1.legend(loc="upper right")
ax2.legend(loc="upper left")
plt.show()
```



REPORT ON CUSTOMER SEGMENTATION AND CLUSTERING RESULTS

6.2 Silhouette Score (Additional Metric)

To further validate clustering quality, we compute the **Silhouette Score**.

The Silhouette Score is:0.3487

7. Insights & Conclusion

7.1 Key Findings

- **Number of Clusters Formed:** 4
- **Davies-Bouldin Index:** 0.6038251031069202
- **Silhouette Score:** 0.3487

7.2 Interpretation of Clusters

Cluster	Characteristics
Cluster 0	High spenders with frequent transactions
Cluster 1	Moderate spenders with occasional transactions
Cluster 2	Low spenders with rare transactions
Cluster 3	Customers with minimal activity

7.3 Business Recommendations

1. **Targeted Marketing**:
    - High spenders (Cluster 0) should receive premium offers.
    - Low spenders (Cluster 2) should receive retention campaigns.
2. **Personalized Discounts**:
    - Offer incentives to moderate spenders (Cluster 1) to increase transactions.
3. **Customer Loyalty Program**:
    - Reward frequent customers to maintain engagement.

This report provides an **end-to-end customer segmentation analysis**, from **data processing to clustering evaluation and visualization**. By using **Davies-Bouldin Index and Silhouette Score**, we ensure optimal clustering performance.