CLUSTERING

Overview

Clustering involves dividing the customer base into groups based on certain similar feature characteristics. it enables businesses to optimize and strategize services and products.

In this project clustering involved segmenting the customers based on features referred and derived from the Products.csv, Customers.csv and Transactions.csv.

Data Overview

1. Datasets Used:
   (a) Customers.csv: Contains demographic and profile information such as customer ID, age, and location.
   (b) Transactions.csv: Includes transactional data, such as purchase recency, frequency, and monetary value (RFM metrics).
2. Preprocessing Steps:
   (a) Data cleaning and merging based on CustomerID.
   (b) Feature engineering to calculate:
      i. Recency: Time since last purchase.
      ii. Frequency: Number of transactions.
      iii. Monetary Value: Total spending.
   (c) Standardization of features to ensure uniform scaling for clustering algorithms.

Clustering Approaches

1. Algorithms Applied:
   (a) K-Means Clustering:
      i. A centroid-based clustering algorithm effective for compact and spherical clusters.
      ii. Optimal cluster number determined using the Elbow Method (minimizing within-cluster sum of squares, WCSS).
   (b) DBSCAN (Density-Based Spatial Clustering):
      i. A density-based algorithm capable of handling noise and irregularly shaped clusters.
      ii. Parameters tuned: eps (radius of neighborhood) and min_samples (minimum points per cluster).
   (c) Hierarchical Clustering:
      i. Utilized to explore cluster hierarchies and relationships between clusters.
2. Cluster Numbers Tested:

   A range of 2 to 10 clusters was evaluated to find the optimal segmentation strategy.

Evaluation Metrics

1. Davies-Bouldin Index (DB Index):
   (a) Measures intra-cluster compactness and inter-cluster separation.
   (b) Lower DB Index values indicate better clustering quality.
2. Silhouette Score:
   (a) Evaluates how similar points are within a cluster compared to other clusters.
   (b) Higher scores indicate well-separated, cohesive clusters.
3. WCSS:
   (a) Assesses the compactness of clusters by measuring within-cluster distances.
   (b) Used in the Elbow Method to determine the optimal number of clusters for K-Means.

Content

Dataset Used

The dataset used is from the files Customers.csv, Products.csv and Transactions.csv. The datasets are merged using inner join on ProductID and CustomerID. The dataset was structured as:

1. TransactionID 1000 non-null object
2. CustomerID 1000 non-null object
3. ProductID 1000 non-null object
4. TransactionDate 1000 non-null object
5. Quantity 1000 non-null int64
6. TotalValue 1000 non-null float64
7. Price 1000 non-null float64
8. CustomerName 1000 non-null object
9. Region 1000 non-null object
10. SignupDate 1000 non-null object

from this dataset few more features were derived:

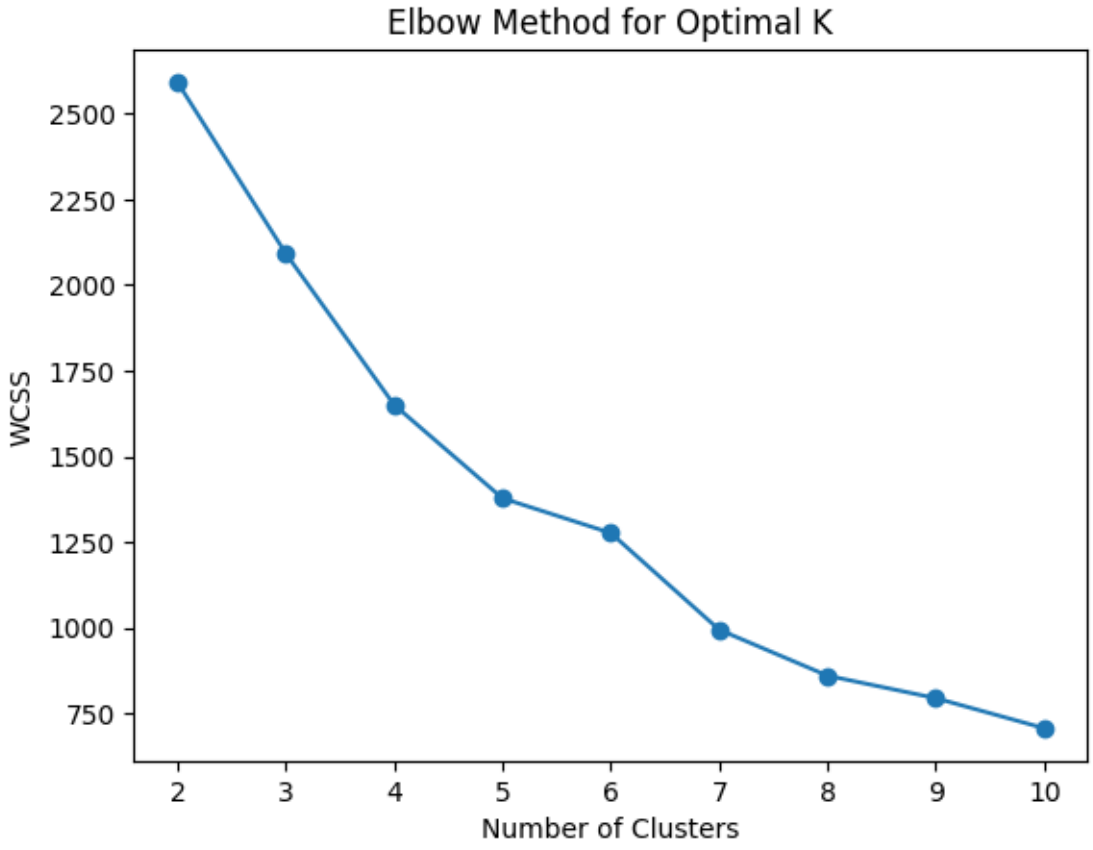
1. TransactionID 1000 non-null object
2. CustomerID 1000 non-null object
3. ProductID 1000 non-null object
4. TransactionDate 1000 non-null datetime64[ns]
5. Quantity 1000 non-null int64
6. TotalValue 1000 non-null float64
7. Price 1000 non-null float64
8. CustomerName 1000 non-null object

9. Region 1000 non-null object
10. SignupDate 1000 non-null datetime64[ns]
11. CustomerTenure 1000 non-null int64
12. TransactionFrequency 1000 non-null int64
13. Recency 1000 non-null int64

The key features considered in the clustering algorithm are Quantity, TotalValue, Price, CustomerTenure, TransactionFrequency, Recency. These features were passed through the StandardScaler.fit_transform() to standardize the data.

K-Means Clustering

1. Elbow Method Plot



The elbow method is a technique used to determine the optimal number of clusters for a clustering algorithm.The plot typically shows a decreasing trend as the number of clusters increases because adding more clusters reduces the distance within each cluster. The

”elbow” point on the plot is where the rate of decrease sharply changes, forming an angle. This point indicates the optimal number of clusters.

2. Calculating DB Index and Silhouette Score

   - The Davies-Bouldin Index (DB Index) and Silhouette Score are both metrics used to evaluate the quality of clustering results.
   - The DB Index measures the average similarity ratio of each cluster with its most similar cluster. A value of 0 implies perfect clustering, while higher values indicate poor clustering quality.
   - The Silhouette Score measures how similar an object is to its own cluster compared to other clusters. The score ranges from -1 to 1. Negative values indicate assignment of wrong cluster. Higher value indicates better clustering.
   - The metrics were calculated for clusters of range 2...10:
     - K-Means - Clusters: 2, DB Index: 1.256, Silhouette Score: 0.316
     - K-Means - Clusters: 3, DB Index: 1.379, Silhouette Score: 0.286
     - K-Means - Clusters: 4, DB Index: 1.099, Silhouette Score: 0.305
     - K-Means - Clusters: 5, DB Index: 1.120, Silhouette Score: 0.301
     - K-Means - Clusters: 6, DB Index: 1.123, Silhouette Score: 0.277
     - K-Means - Clusters: 7, DB Index: 1.036, Silhouette Score: 0.325
     - K-Means - Clusters: 8, DB Index: 1.007, Silhouette Score: 0.331
     - K-Means - Clusters: 9, DB Index: 0.976, Silhouette Score: 0.328
     - K-Means - Clusters: 10, DB Index: 0.978, Silhouette Score: 0.329

     9 clusters gives ideal DB Index which describes inter cluster distance and silhouette score that describes compactness and separation of clusters.

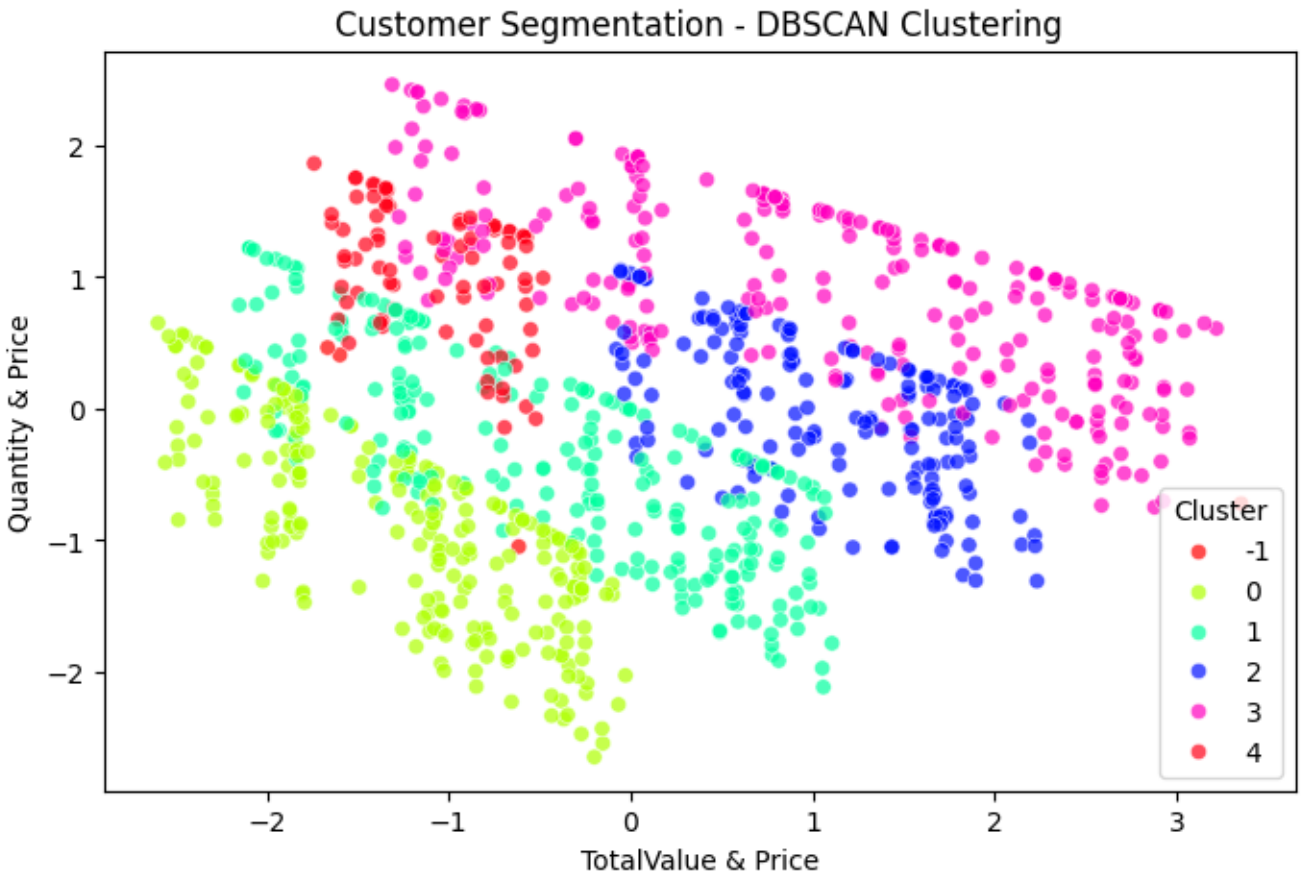
3. PCA and plotting the customer data:

   - Principal Component Analysis (PCA) is a technique used to reduce the dimensionality of large datasets, increasing interpretability while minimizing information loss.
   - The features are reduced to two features and the features that contribute the most are used to name these new features.

- A scatter plot is plotted between these features:

DBScan

1. plotting of customer data:



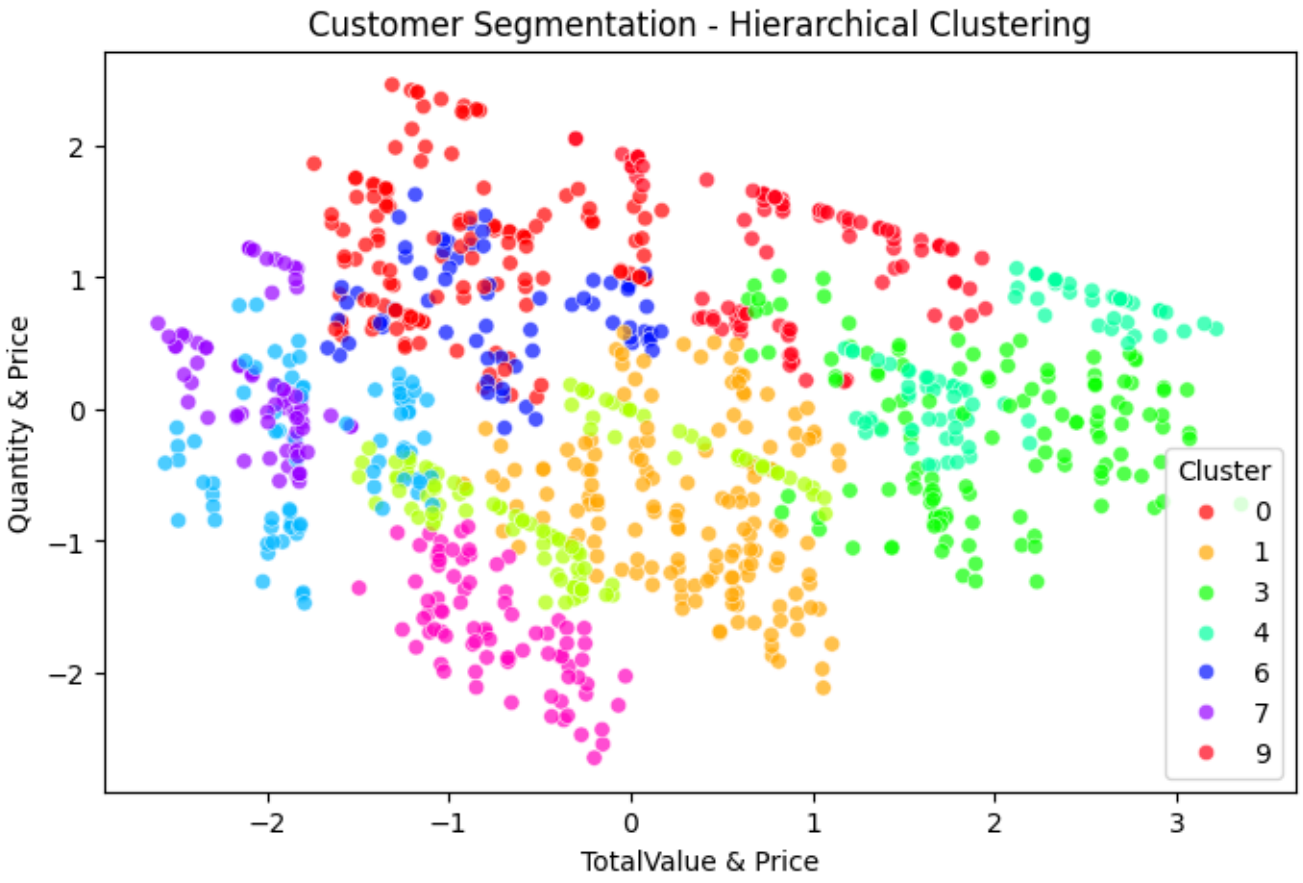
This clustering algorithm created 5 clusters along with one outlier cluster represented by -1

Hierarchical Clustering

1. Calculating DB Index and Silhouette Score

   - he metrics were calculated for clusters of range 2...10:
     - Hierarchical - Clusters: 2, DB Index: 1.235, Silhouette Score: 0.297
     - Hierarchical - Clusters: 3, DB Index: 1.289, Silhouette Score: 0.289
     - Hierarchical - Clusters: 4, DB Index: 1.211, Silhouette Score: 0.273
     - Hierarchical - Clusters: 5, DB Index: 1.248, Silhouette Score: 0.250
     - Hierarchical - Clusters: 6, DB Index: 1.221, Silhouette Score: 0.264
     - Hierarchical - Clusters: 7, DB Index: 1.103, Silhouette Score: 0.275
     - Hierarchical - Clusters: 8, DB Index: 1.024, Silhouette Score: 0.273
     - Hierarchical - Clusters: 9, DB Index: 1.018, Silhouette Score: 0.289
     - Hierarchical - Clusters: 10, DB Index: 1.020, Silhouette Score: 0.282

2. PCA and plotting the customer data:



Result

Method	DB Index	Silhouette Score
KMeans (9 Clusters)	0.976	0.328
DBSCAN (5 Clusters)	2.34	0.074
Hierarchical (9 Clusters)	1.018	0.289

Table 1: Clustering Evaluation Metrics

From the results we find that KMeans(9 Clusters) performs the best with DB Index 0.976 and Silhouette Score 0.328