**Clustering Analysis Report: K-Means and DBSCAN**

**1. Introduction**

Clustering is an essential unsupervised learning technique used to group similar data points together. In this analysis, we employed two clustering algorithms: **K-Means** and **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)** to segment customers based on various attributes. The study also includes data preprocessing, outlier removal, and a comparative analysis of both methods.

**2. Data Preprocessing**

**2.1 Loading and Handling Missing Values**

A dataset containing customer attributes such as age, income, spending score, and others was loaded into Python. Missing values were handled appropriately by either imputing them using mean/mode or removing them to ensure data integrity.

**3. Determining the Optimal Number of Clusters (K)**

**3.1 Elbow Method**

The **Elbow Method** was applied to determine the optimal number of clusters for **K-Means**. The plot of inertia vs. K displayed a significant bend at a particular point, suggesting the optimal cluster count.

**3.2 Silhouette Score**

The **Silhouette Score** was used as an additional validation metric. The score indicated how well-separated the clusters were, confirming the optimal number of clusters chosen from the elbow method.

**4. K-Means Clustering Implementation**

After determining the optimal value of **K**, K-Means clustering was applied to the preprocessed dataset. The clustering results were visualized using scatter plots for relevant features.

**Findings:**

* The dataset was successfully segmented into **7 clusters**, as determined by the optimal K value.
* Each cluster exhibited unique characteristics based on spending habits, income levels, and purchasing behaviors.

**5. Cluster Interpretation**

Clusters were analyzed by computing the average values of each feature per cluster. The following insights were obtained:

* Some clusters represented **high-income, high-spending customers**.
* Others contained **low-income, low-spending customers**.
* Certain clusters showed **frequent website visitors with high product purchases**, while others had lower visit rates.

These insights can help businesses target different customer segments with personalized marketing strategies.

**6. Handling Outliers**

**6.1 Outlier Detection and Removal**

* **Z-score method** was used to detect outliers, leading to the removal of **10 outliers** from the dataset (original dataset: 500 rows → after removal: 490 rows).
* Clustering was re-applied after outlier removal, showing an improvement in cluster compactness and accuracy.

**7. Comparison with DBSCAN**

**7.1 DBSCAN Clustering Implementation**

DBSCAN was applied to the dataset, identifying **dense clusters** while marking noise points (outliers). Unlike K-Means, DBSCAN does not require predefining the number of clusters.

**7.2 Findings from DBSCAN**

* Many points were categorized as **noise (-1 label)**, meaning they did not belong to any cluster.
* Several small, meaningful clusters were detected (Cluster labels: **-1, 0, 1, 2, ..., 23**), indicating DBSCAN's sensitivity to density variations.
* Some customer segments were captured better compared to K-Means, especially when considering **irregularly shaped clusters**.

**8. Comparison of K-Means and DBSCAN**

| **Feature** | **K-Means** | **DBSCAN** |
| --- | --- | --- |
| **Cluster Shape** | Spherical Clusters | Arbitrary-Shaped Clusters |
| **Noise Handling** | Does not identify noise | Effectively detects noise |
| **Performance** | Works well on large datasets | Struggles with high-dimensional data |
| **Number of Clusters** | Must be predefined | Determined dynamically |
| **Best For** | Well-separated, balanced clusters | Densely packed, irregularly shaped clusters |

**When to Use K-Means vs. DBSCAN?**

* **K-Means** is preferable when **clusters are well-separated and compact**.
* **DBSCAN** is better when the dataset contains **varying density regions** and outliers.

**9. Conclusion**

* **K-Means successfully grouped customers into well-defined segments**, which can be used for targeted marketing strategies.
* **DBSCAN helped in detecting noise and capturing irregular clusters**, useful for identifying distinct behavioral patterns.
* Outlier removal improved clustering results.
* A combination of both algorithms can provide deeper insights into customer segmentation.