Customer Segmentation Using Clustering Methods

Objective: The goal of this project is to segment customers from an online retail dataset into distinct groups based on their purchasing behavior. This segmentation is achieved by applying various unsupervised clustering algorithms to RFM (Recency, Frequency, Monetary) metrics, allowing for targeted marketing strategies and a deeper understanding of the customer base.

---

1. Data Loading and Preprocessing

1.1. Data Exploration

The dataset, `Online_retail_data.csv`, was loaded into a pandas DataFrame. An initial inspection confirmed that the data was clean, with **no missing values or duplicate rows**.

A preview of the first five rows:

	Invoice	Description	Quantity	InvoiceDate	Price	Customer ID
0	INV1	Product D	12	2023-01-01 00:00:00	63.86	C56
1	INV2	Product E	12	2023-01-01 01:00:00	75.73	C76
2	INV3	Product C	4	2023-01-01 02:00:00	17.88	C51
3	INV4	Product E	16	2023-01-01 03:00:00	41.54	C37
4	INV5	Product E	4	2023-01-01 04:00:00	42.85	C17

A descriptive statistical summary showed that the `Quantity` and `Price` columns contained only positive values, which is valid for this analysis.

1.2. Feature Engineering: RFM Analysis

To segment customers effectively, RFM metrics were calculated for each `Customer ID`:

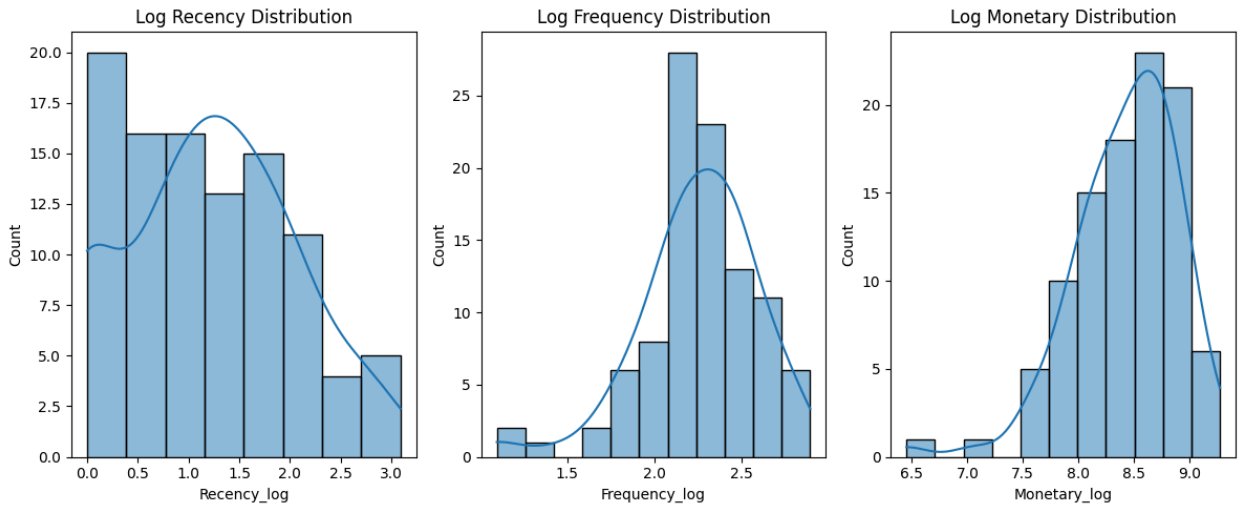
- **Recency (R):** The number of days since the customer's last purchase.
- **Frequency (F):** The total number of purchases made by the customer.
- **Monetary (M):** The total amount of money spent by the customer (`Quantity * Price`).

The resulting RFM DataFrame:

	CustomerID	Recency	Frequency	Monetary
0	C1	1	9	6319.24
1	C10	2	14	8097.72
2	C100	11	13	5465.74
3	C11	2	8	4313.14
4	C12	5	6	4148.32

1.3. Data Transformation and Scaling

The RFM features exhibited a skewed distribution. To normalize them, a **logarithmic transformation** was applied. Following this, the features were scaled using `StandardScaler` to ensure that each metric contributed equally to the clustering models. This step is crucial as clustering algorithms are often sensitive to the scale of the data.



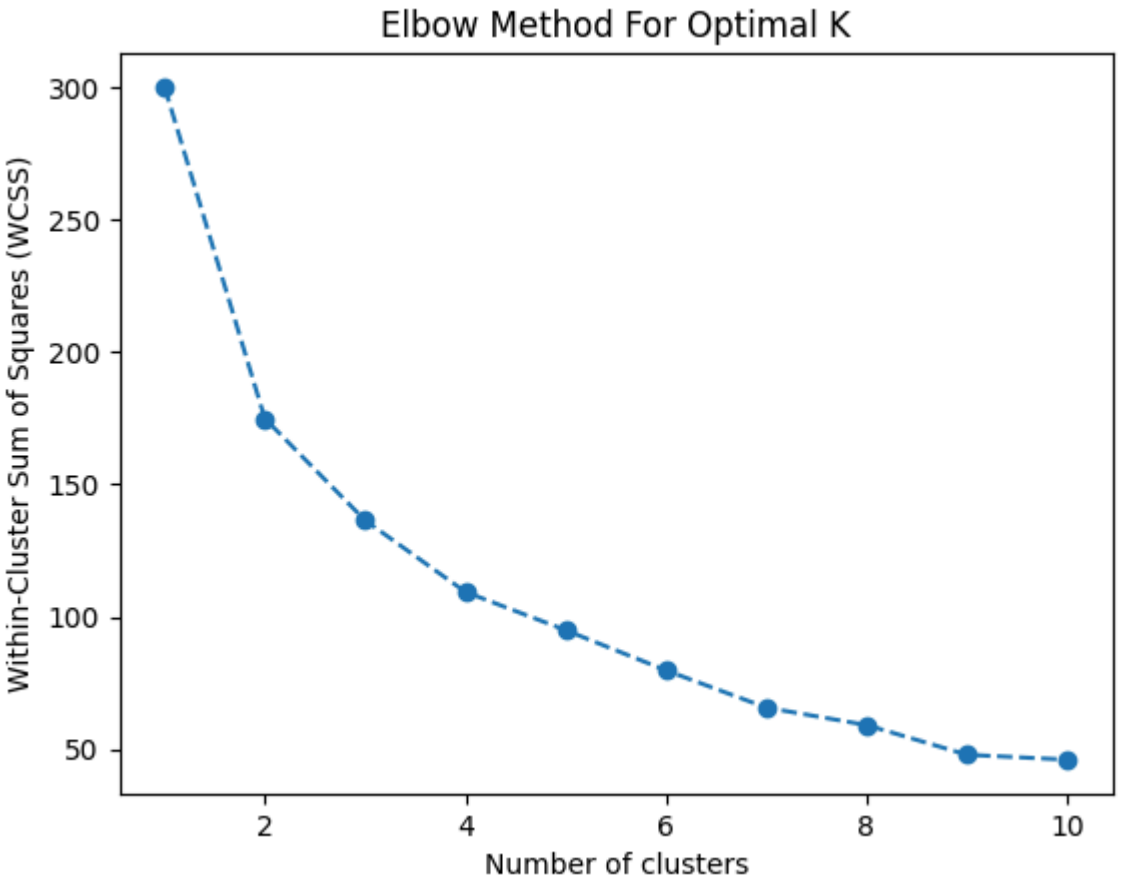
---

2. Clustering Model Implementation and Evaluation

Several clustering algorithms were applied to the scaled RFM data. The primary metric for evaluating the quality of the clusters was the **Silhouette Score**, which measures how similar an object is to its own cluster compared to other clusters. A higher score (closer to 1) indicates better-defined clusters.

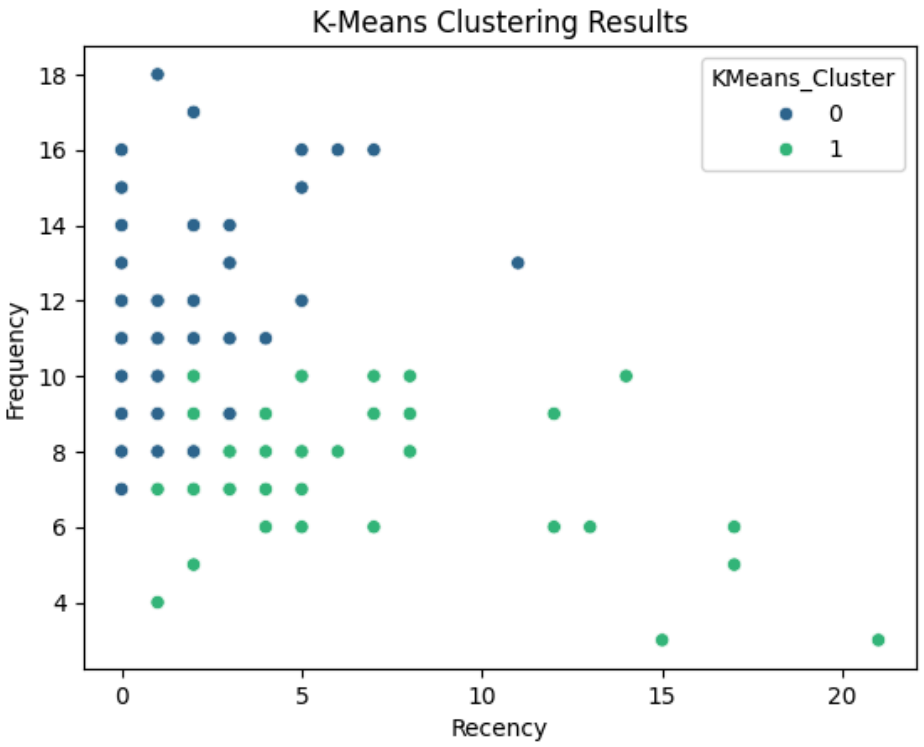
2.1. K-Means Clustering

The **Elbow Method** was used to determine the optimal number of clusters (K). By plotting the within-cluster sum of squares (WCSS) for different values of K, a clear "elbow" was observed at **K=2**, indicating that two clusters are optimal for this dataset.



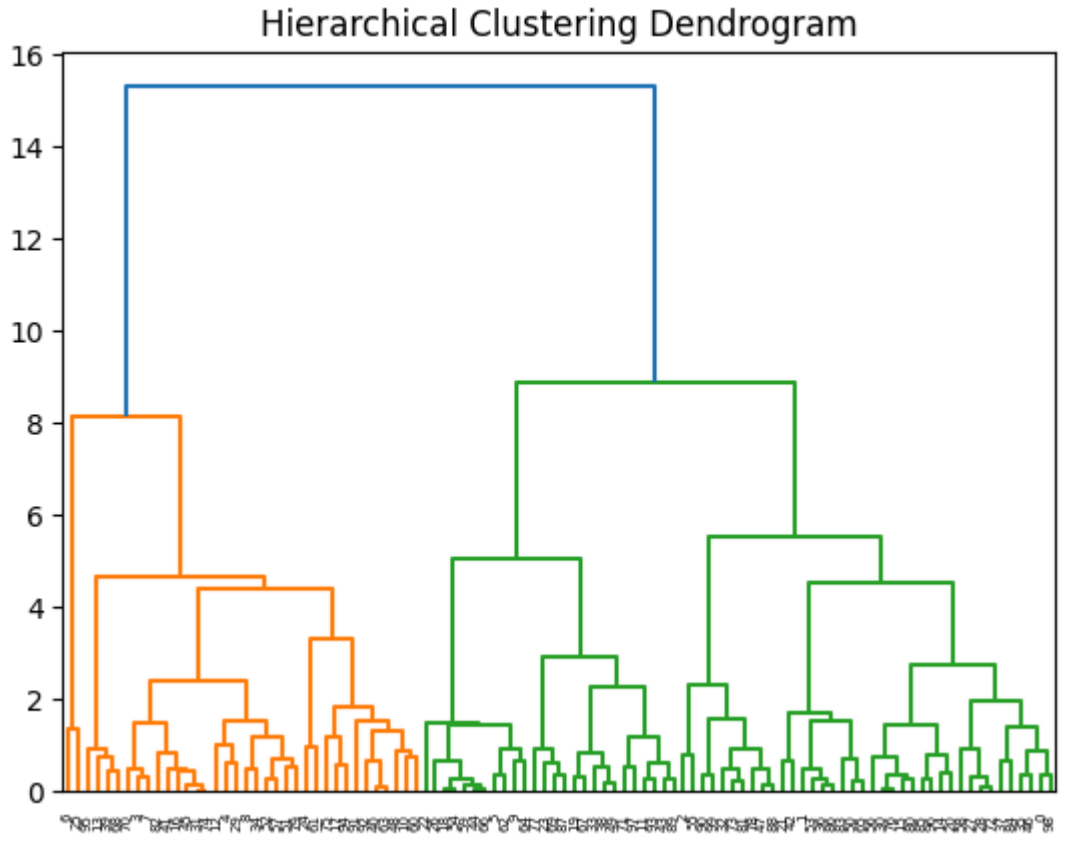
The K-Means model was trained with K=2, and the resulting clusters were visualized.

- **Silhouette Score: 0.3625**



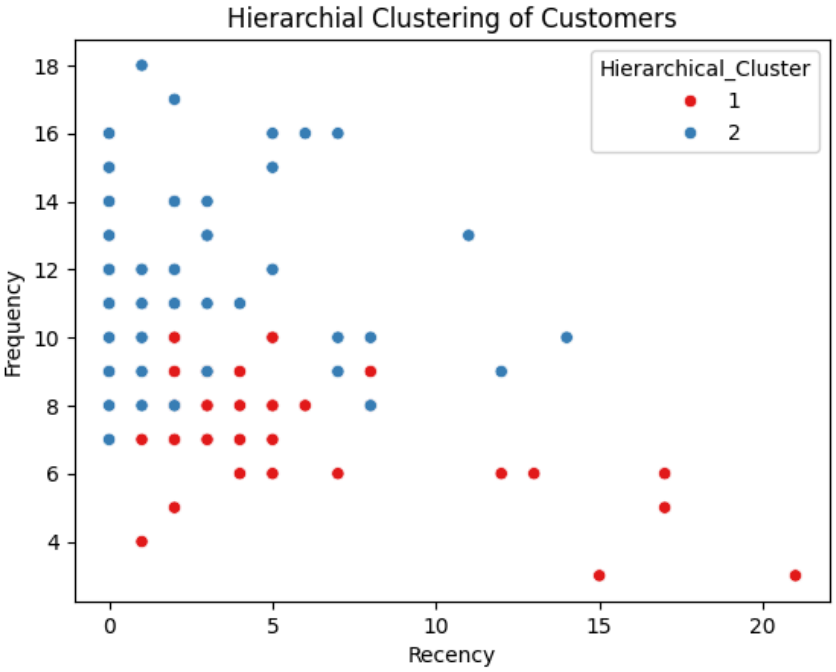
2.2. Hierarchical Clustering

Hierarchical clustering was performed using a dendrogram to visualize the cluster hierarchy. Based on the dendrogram's structure, the data was cut to form **2 clusters**.



- **Silhouette Score: 0.3416**

The results were similar to K-Means, successfully grouping customers into two primary segments.

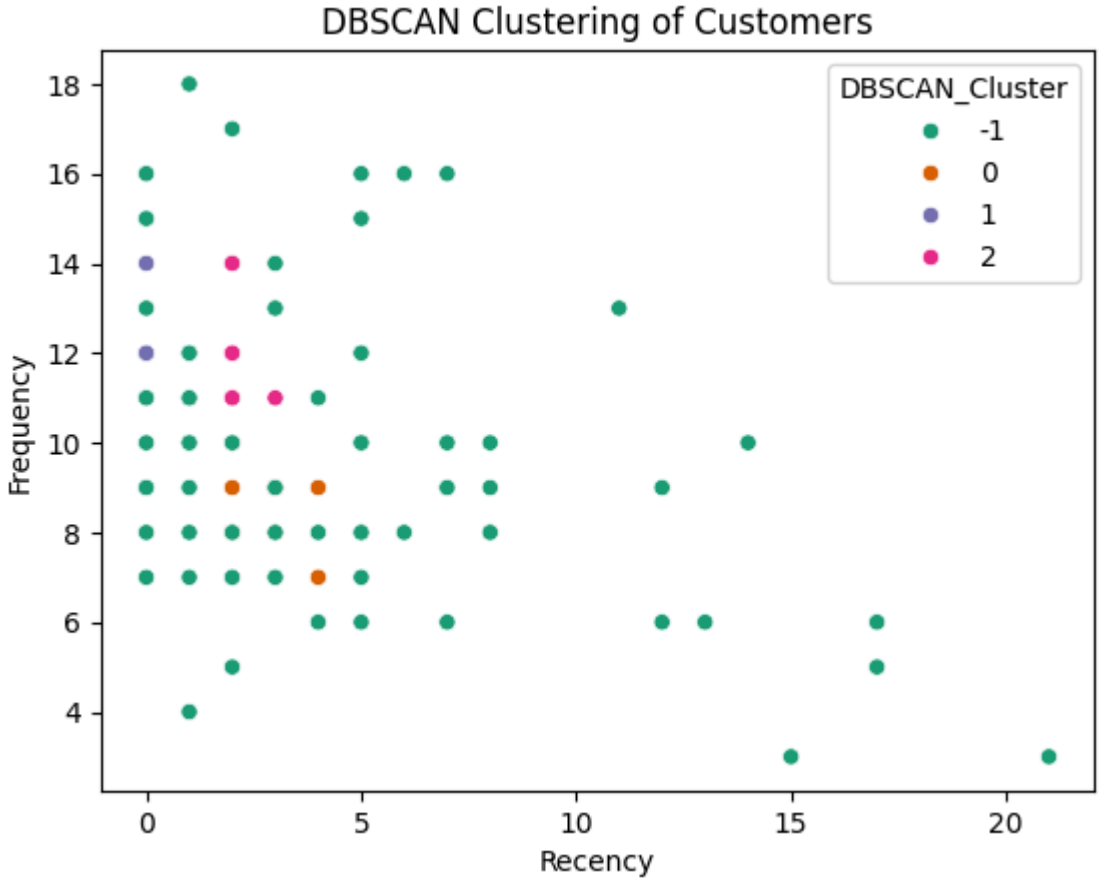


2.3. DBSCAN (Density-Based Spatial Clustering)

DBSCAN was applied with `eps=0.5` and `min_samples=6`. This method is effective at identifying clusters of varying shapes and sizes and can also detect noise points (outliers).

- **Clusters Identified:** 3 clusters and several noise points (labeled as -1).
- **Silhouette Score (excluding noise): 0.6829**

The high silhouette score indicates that DBSCAN created very dense and well-separated clusters, outperforming K-Means and Hierarchical Clustering in cluster quality.



2.4. Mean-Shift Clustering

Mean-Shift is another density-based algorithm that does not require specifying the number of clusters beforehand.

- **Silhouette Score: 0.2751**

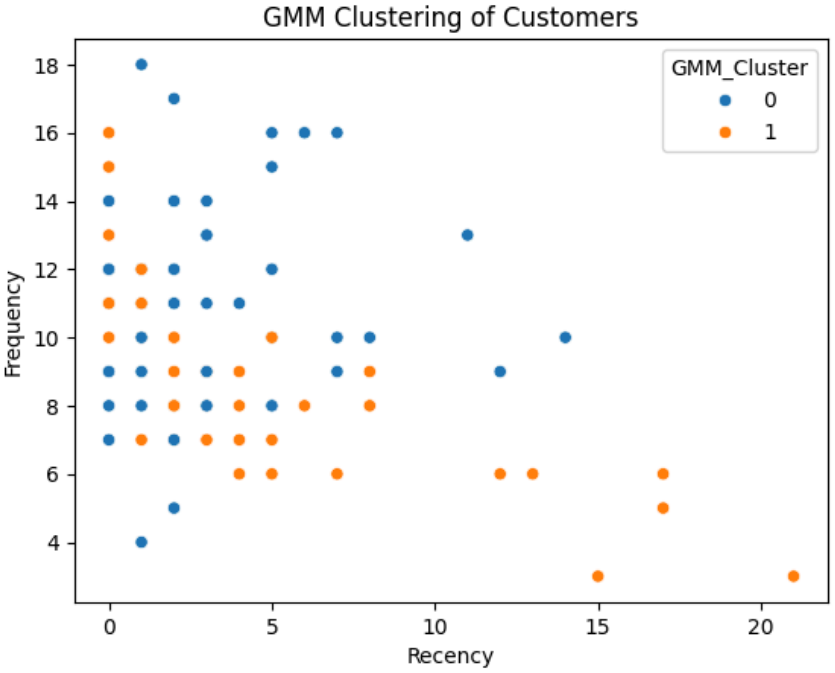
While it automatically determined the clusters, the silhouette score was lower than that of the other methods, suggesting less optimal segmentation for this dataset.

2.5. Gaussian Mixture Model (GMM)

GMM is a probabilistic model that assumes the data points are generated from a mixture of a finite number of Gaussian distributions. It was configured to find **2 clusters**.

- **Silhouette Score: 0.1601**

The GMM produced the lowest silhouette score, indicating that its soft-clustering approach was less effective for this specific RFM data compared to the other algorithms.



---

3. Conclusion and Model Comparison

The table below summarizes the performance of each clustering algorithm based on its Silhouette Score.

Clustering Algorithm	Silhouette Score	Number of Clusters
K-Means	0.3625	2
Hierarchical	0.3416	2
DBSCAN	0.6829	3 (+ noise)
Mean-Shift	0.2751	Multiple (auto-detected)
GMM	0.1601	2

Conclusion:

Based on the Silhouette Scores, **DBSCAN** provided the most effective and meaningful segmentation of the customer data. Its ability to identify dense regions and isolate outliers resulted in the highest quality clusters. While K-Means and Hierarchical clustering also provided reasonable two-cluster solutions, their lower scores indicate less distinct separation.

The results from DBSCAN suggest that the customer base can be segmented into three primary groups, with an additional set of outliers who do not fit into any specific behavioral pattern. This insight is highly valuable for developing targeted business strategies.