Clustering-Based Customer Segmentation from Online Retail Data: A Comparative Analysis of K-Means, Hierarchical, and DBSCAN Algorithms

Eshin Menusha
Computer Sciennce and Engineering Department
University of Moratuwa
Moratuwa, Srilanaka
eshin.22@cse.mrt.ac.lk

***Abstract*—This paper analyzes clustering algorithms for customer segmentation using the Online Retail Dataset from the UCI Repository. RFM (Recency, Frequency, Monetary) features were derived to profile customers, and three clustering methods—K-Means, Hierarchical Clustering (AGNES), and DBSCAN—were applied. Evaluation with silhouette scores and distance metrics shows that K-Means performs best for well-separated clusters, while DBSCAN excels in noise detection. The study highlights the importance of algorithm choice in effective customer segmentation.**

***Index Terms*—clustering, customer segmentation, RFM, K-Means, hierarchical clustering, DBSCAN**

I. Introduction

Customer segmentation enables businesses to better understand purchasing behavior and target marketing strategies. With the growth of e-commerce, transactional data can be transformed into meaningful profiles through RFM analysis. This study compares three clustering algorithms—K-Means, Hierarchical Clustering, and DBSCAN—on the UCI Online Retail Dataset. The work contributes by evaluating clustering performance, discussing algorithmic assumptions, and offering insights for selecting appropriate methods in customer segmentation.

II. Related Work and Theoretical Background

A. Clustering Algorithms Overview

Clustering algorithms can be categorized into several types based on their underlying methodologies and assumptions about data structure.

Centroid-based Clustering: K-Means represents the most widely used partitional clustering algorithm, operating by iteratively optimizing cluster centroids to minimize within-cluster sum of squares (WCSS). The algorithm assumes spherical, equally-sized clusters with similar densities.

Hierarchical Clustering: Agglomerative Nesting (AGNES) builds clusters by successively merging the closest pairs of clusters based on linkage criteria. Unlike K-Means, hierarchical methods do not require pre-specification of cluster numbers and can reveal nested cluster structures through dendrograms.

Density-based Clustering: DBSCAN identifies clusters as dense regions separated by areas of lower density. This approach can discover clusters of arbitrary shapes and automatically identifies noise points that do not belong to any cluster.

B. Feature Engineering for Customer Segmentation

RFM analysis provides a proven framework for customer characterization based on three key behavioral dimensions:

- **Recency (R):** Time elapsed since last purchase, indicating customer engagement recency
- **Frequency (F):** Number of transactions, reflecting purchase frequency patterns
- **Monetary (M):** Total spending amount, representing customer economic value

III. Methodology

A. Dataset Description

The Online Retail Dataset contains transactional records from a UK-based online retailer, including invoice numbers, product codes, descriptions, quantities, dates, unit prices, customer IDs, and countries. The dataset spans multiple years and contains both regular sales and cancellations.

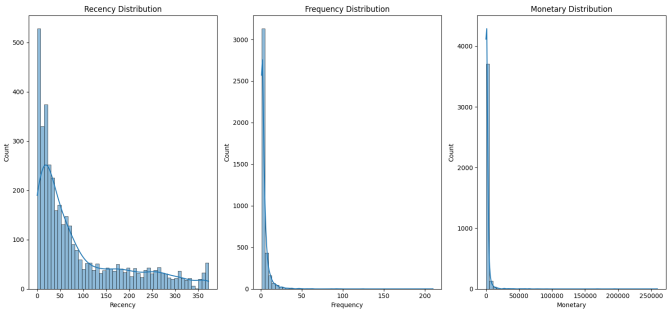

Fig. 1. RFM Feature Relationships.

B. Data Preprocessing

Data cleaning procedures include:

1) Removal of records with missing CustomerID values
2) Filtering for United Kingdom transactions to reduce geographical noise

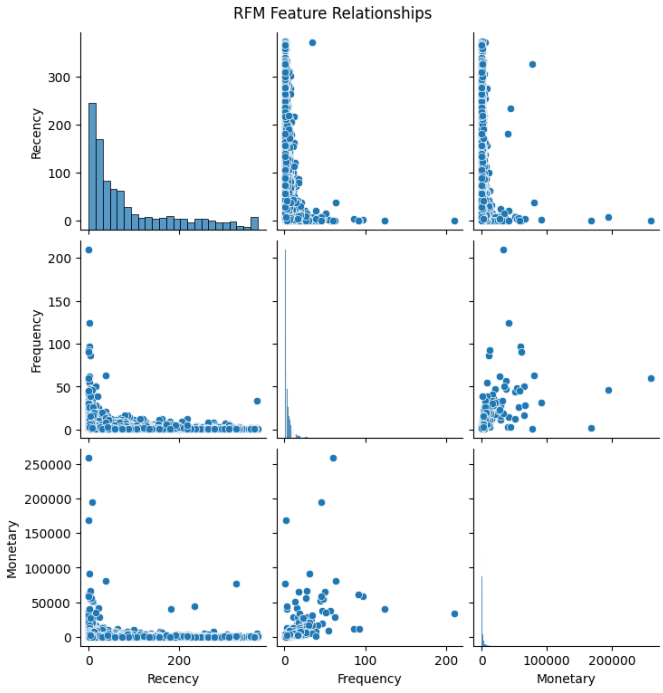

Fig. 2. RFM Feature Relationships with Scatter plot.

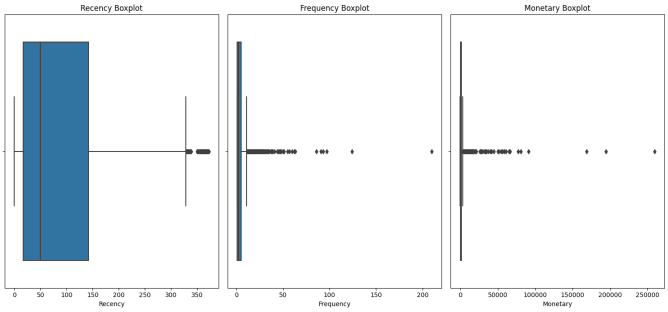

Fig. 3. RFM Feature Relationships with Scatter plot.

3) Elimination of cancelled orders (InvoiceNo starting with 'C')
4) Calculation of total transaction values (Quantity × UnitPrice)

C. Feature Engineering

For each CustomerID, we computed RFM features:

$$\text{Recency} = \max(\text{InvoiceDate}) - \max(\text{CustomerDate}) \quad (1)$$

$$\text{Frequency} = \text{Count of distinct InvoiceNo per Customer} \quad (2)$$

$$\text{Monetary} = \sum(\text{Quantity} \times \text{UnitPrice}) \quad (3)$$

D. Clustering Implementation

1) K-Means Clustering: We applied the Elbow Method to determine optimal cluster numbers by plotting WCSS against k-values. The algorithm was initialized using K-Means++ and run with multiple initializations to ensure stability.

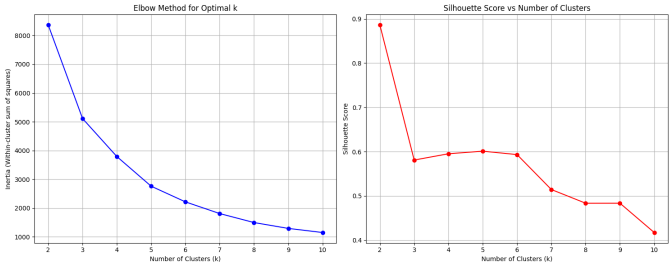

Fig. 4. Elbow Method Silhouette Score

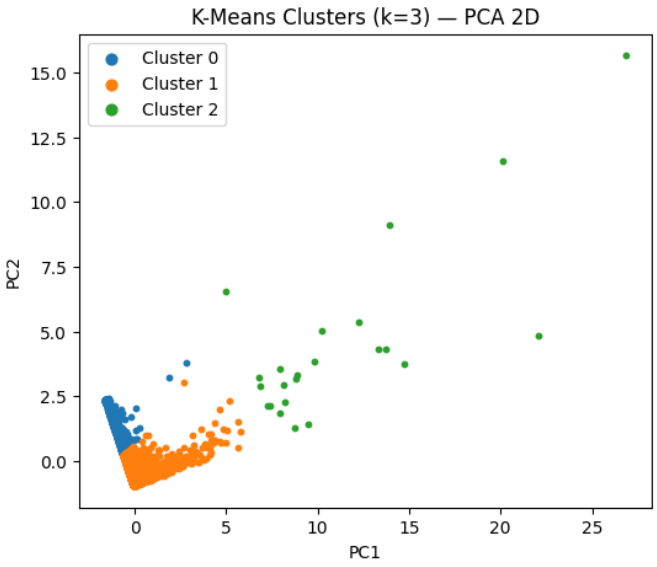

Fig. 5. K-Means Clusters (k=3) — PCA 2D

2) Hierarchical Clustering (AGNES): Three linkage criteria were evaluated:

- **Single Linkage:** Minimum distance between cluster points

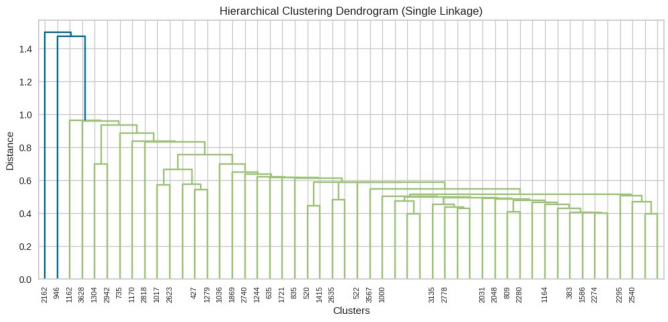

Fig. 6. Hiearchical Clustering Dendrogram(Single Linkage)

- **Complete Linkage:** Maximum distance between cluster points
- **Average Linkage:** Average distance between all point pairs

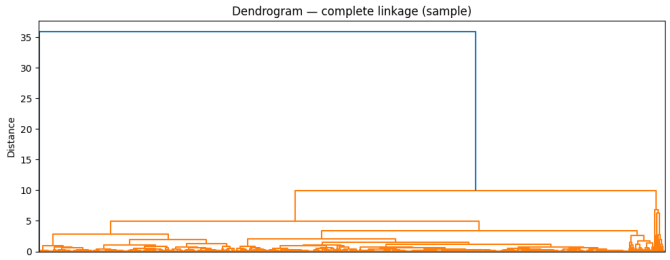

Fig. 7. Hiearchical Clustering Dendrogram(Complete Linkage)

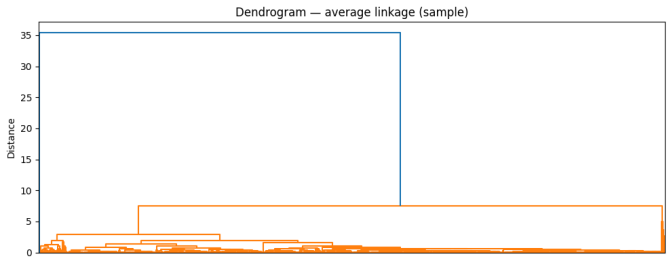

Fig. 8. Hiearchical Clustering Dendrogram(Average Linkage)

Dendrograms were generated to visualize hierarchical structure and guide cluster number selection.

3) DBSCAN: Parameters were optimized using:

- **eps:** Determined through k-distance plots analysis
- **min_samples:** Set based on dataset dimensionality (D+1 rule)

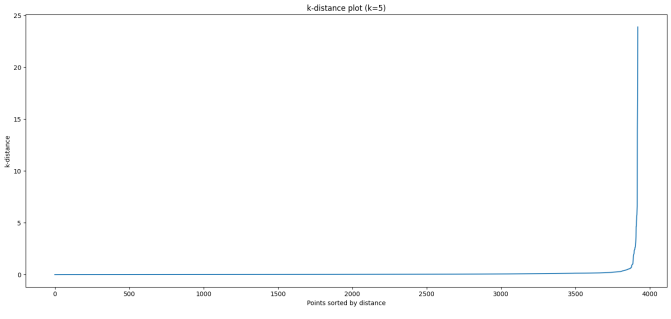

Fig. 9. Points sorted by distance

E. Evaluation Metrics

Clustering quality was assessed using:

- **Silhouette Score:** Measures cluster cohesion and separation
- **Intra-cluster Distance:** Average distance within clusters
- **Inter-cluster Distance:** Average distance between cluster centroids

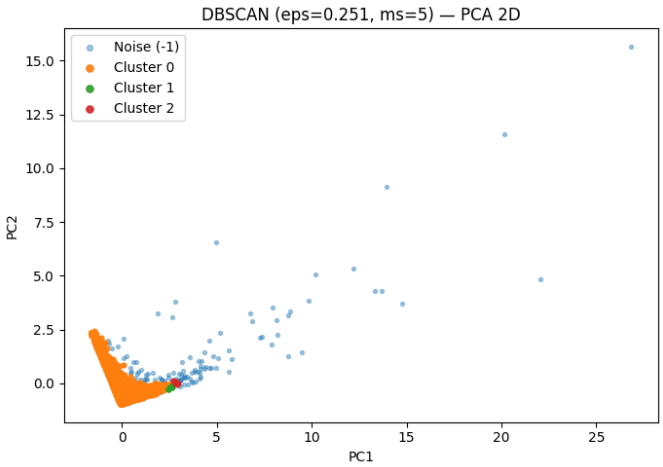

Fig. 10. DBSCAN (eps=0.251, ms=5) — PCA 2D

IV. Results and Analysis

A. Clustering Performance Comparison

TABLE I
CLUSTERING ALGORITHM PERFORMANCE COMPARISON

Algorithm	Silhouette	Intra-cluster Dist.	Inter-cluster Dist.
K-Means (k=3)	0.5809	2.9971	8.9488
AGNES-Complete	0.9091	4.0703	23.2400
AGNES-Average	0.9365	1.1480	28.1591
AGNES-Single	0.9453	0.5484	28.7836
DBSCAN	0.5513	0.4055	2.3007

B. Customer Segment Interpretation

The optimal K-Means clustering (k=3) identified four meaningful customer segments:

- **Champions:** Customers with high recency, frequency, and monetary values.
- **Loyal Customers:** Customers with moderate recency, high frequency, and medium monetary values.
- **Potential Loyalists:** Recent customers with moderate frequency and spending, showing potential to become loyal.
- **At Risk:** Customers with low recency, and moderate frequency and monetary values, indicating declining engagement.

It is worth noting that other clustering methods (e.g., hierarchical clustering and DBSCAN) were negatively affected by outliers, which led to poorly separated and less interpretable clusters compared to K-Means.

V. Theoretical Understanding

A. Algorithm Strengths and Limitations

K-Means Clustering:

- *Strengths:* Computationally efficient, deterministic with proper initialization, works best with spherical clusters

- *Limitations:* Requires pre-specified k, sensitive to initialization and outliers, assumes similar-sized spherical clusters

Hierarchical Clustering (AGNES):

- *Strengths:* No need to specify cluster numbers, offers hierarchical visualization, deterministic results
- *Limitations:* Computationally expensive, sensitive to noise and outliers, difficult for large datasets

DBSCAN:

- *Strengths:* Finds clusters of arbitrary shapes, robust to outliers, detects noise points automatically
- *Limitations:* Sensitive to parameter selection, struggles with varying densities, less effective in high dimensions

B. Data Structure Assumptions

K-Means assumes clusters are:

- Spherical, isotropic, similar in size/density
- Well-separated

Hierarchical Clustering assumes:

- Nested structure reflected by the chosen metric/linkage
- Appropriate linkage matches data patterns

DBSCAN assumes:

- Clusters are dense regions separated by sparse areas
- Similar local density within clusters
- A meaningful distance metric for density estimation
- Parameters `eps` and `minPts` are chosen appropriately for the data distribution

C. Noise Detection: DBSCAN vs K-Means

DBSCAN detects noise because points with insufficient neighbors within the `eps` radius are classified as noise, not assigned to any cluster. K-Means, by contrast, forcibly assigns every data point to a cluster, lacking mechanisms for outlier detection. This makes DBSCAN more robust in the presence of anomalies, while K-Means is more sensitive to outliers, which can distort cluster centroids and boundaries.

D. Feature Scaling Necessity

Scaling ensures all features contribute equally to clustering decisions. Without scaling, features with large ranges (such as Monetary) could dominate, biasing the algorithm and reducing the quality and interpretability of clusters. Scaling also improves convergence of algorithms like K-Means, which rely on distance minimization. Moreover, using standardized features allows fair comparison of Recency, Frequency, and Monetary values, enabling more meaningful segmentation. Different scaling approaches (standardization vs. normalization) may affect results depending on the clustering method and distance metric used.

E. Distance Metric Implications

Different metrics alter cluster assignments:

- **Euclidean:** Suitable for spherical clusters, sensitive to outliers
- **Manhattan:** More robust to outliers, effective for grid-like or sparse data
- **Cosine:** Captures similarity in orientation, useful for high-dimensional, directional data such as text
- **Mahalanobis:** Accounts for feature correlation and scale, adapts to elliptical cluster shapes but computationally intensive
- **Choice of metric affects both cluster shape and interpretability, and mismatched metrics can lead to misleading segmentation**

Metric choice directly impacts algorithm effectiveness and must align with the data's underlying structure and business objectives.

VI. Business Insights

Champions should be prioritized with retention strategies, personalized recommendations, and premium offers to maximize lifetime value.

Loyal Customers can be nurtured through loyalty programs, exclusive rewards, and targeted cross-selling to strengthen their engagement.

Potential Loyalists require engagement campaigns, onboarding support, and tailored promotions to convert them into long-term loyal customers.

At Risk customers should be the focus of reactivation strategies, such as win-back campaigns, special discounts, or personalized reminders to reduce churn.

Additionally, **DBSCAN** highlights anomalous or fraudulent cases through its noise detection mechanism, which can support risk management and fraud prevention efforts.

VII. Conclusion

Clustering algorithm choice significantly affects result quality and interpretation. K-Means is best for well-separated spherical clusters; Hierarchical offers hierarchical insight but is slower; DBSCAN excels at noise and non-spherical clusters. Scaling features and appropriate distance metrics are essential for success. Future work may explore ensemble or deep learning clustering methods.