Project # 4
Customer Segmentation using Clustering on Online Retail Dataset

Minthra Khounsavath

21 April, 2025

Introduction

The goal of this project was to explore customer segmentation using clustering algorithms. Businesses often struggle with understanding the behavior of their customers due to the vast amount of transactional data available. My aim was to use unsupervised learning methods to divide customers into distinct groups based on their purchasing behavior, helping businesses better target their marketing and services.

I started with a simple question: *Can I identify meaningful customer segments using transactional data alone?* To answer this, I applied clustering techniques to group customers by their Recency, Frequency, and Monetary (RFM) behavior.

What is Clustering?

Clustering is an unsupervised machine learning technique that involves grouping a set of data points into subsets or "clusters" such that data points in the same group are more similar to each other than to those in other groups. It's widely used for market segmentation, image compression, anomaly detection, and more.

For this project, I used two major clustering techniques:

K-Means Clustering

K-Means is a partition-based clustering algorithm. It works by:

1. Selecting k initial cluster centroids randomly.
2. Assigning each data point to the nearest centroid using Euclidean distance.
3. Recomputing the centroids as the mean of the points in each cluster.
4. Repeating steps 2 and 3 until convergence (no change in centroids or max iterations reached).

I chose K-Means because it is computationally efficient, easy to implement, and works well when the number of clusters is known or can be estimated.

Agglomerative Clustering

Agglomerative Clustering is a bottom-up hierarchical method. It works by:

1. Starting with each data point as its own cluster.
2. Merging the two closest clusters based on a linkage criterion (e.g., Ward, average).
3. Repeating until all data points belong to one cluster or the desired number of clusters is reached.

This method is useful when you want to visualize how data points are grouped at different thresholds using a dendrogram. I used this as a second approach to validate the robustness of my K-Means clusters.

Libraries and Approach

To complete this project, I used Python and several key libraries:

- `pandas` for data loading and manipulation
- `matplotlib` and `seaborn` for data visualization
- `scikit-learn` for machine learning (clustering algorithms, preprocessing, metrics)
- `scipy` for Agglomerative Clustering and dendrogram generation

My Approach:

1. Load and clean the data
2. Engineer features and aggregate customer data
3. Normalize features
4. Apply clustering (K-Means and Agglomerative)
5. Visualize and evaluate clusters
6. Extract insights and reflect on impact

Dataset Description

I used the Online Retail dataset from the UCI Machine Learning Repository.

Link: `https://archive.ics.uci.edu/ml/datasets/online+retail`

The dataset contains nearly 500,000 transactions made by customers between 2010 and 2011. It includes:

- `InvoiceNo`: Unique ID for each transaction
- `StockCode`: Product identifier
- `Description`: Product name
- `Quantity`: Units purchased
- `InvoiceDate`: Date/time of purchase
- `UnitPrice`: Price per item
- `CustomerID`: Unique identifier for each customer
- `Country`: Country of the customer

To create meaningful customer-level features, I computed:

- **Recency** – days since the last purchase
- **Frequency** – total number of transactions
- **Monetary** – total spending

Data Understanding and Visualization

After removing missing values and canceled transactions, I generated summary statistics and histograms to explore each variable. I noticed that:

- Most customers have low-frequency purchases.
- Spending distribution is highly skewed, with few customers spending significantly more.
- A small subset of customers is very active and recent.

How it relates to modeling: This exploration helped me decide on which features to include and revealed the need for normalization due to skewed distributions.

Preprocessing

Here's what I did during preprocessing:

- Removed rows with missing `CustomerID`
- Removed canceled orders (InvoiceNo starting with 'C')
- Created a new column `TotalSpend = Quantity` × `UnitPrice`
- Grouped data by `CustomerID` and calculated Recency, Frequency, Monetary
- Used `StandardScaler` to normalize features for clustering

Modeling (Clustering)

K-Means Clustering

To determine the optimal number of clusters, I used the Elbow Method. I plotted the Within-Cluster Sum of Squares (WCSS) against the number of clusters.

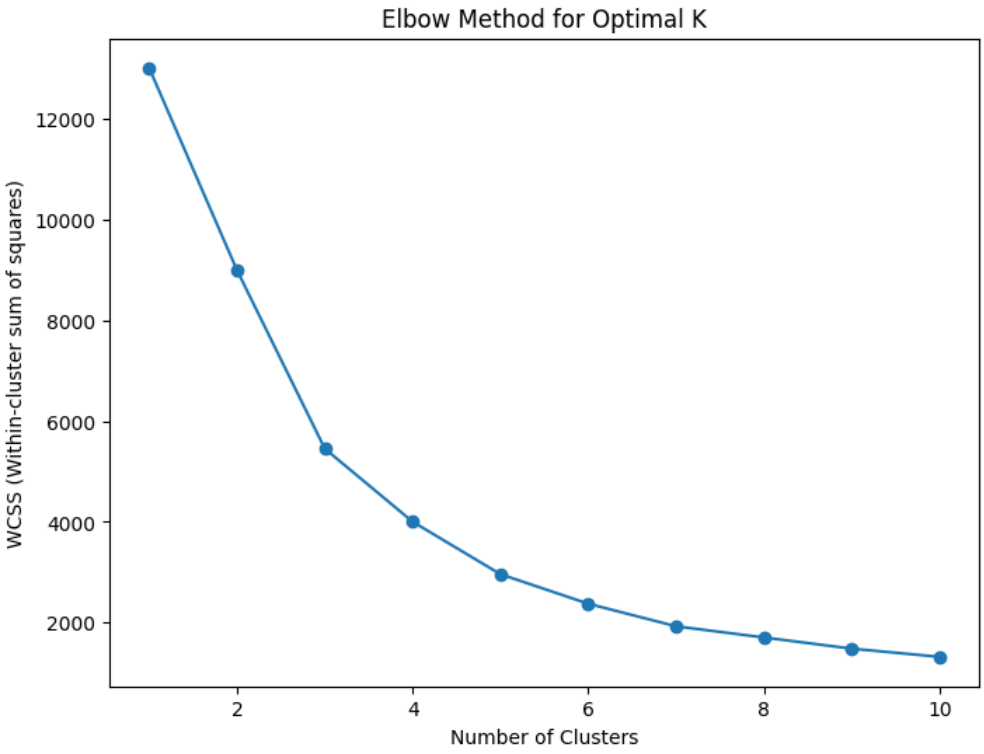

Figure 1: Elbow Method for Optimal K

[Elbow Plot Insight]
The optimal number of clusters appears to be **4**, as that is the point where the rate of decrease in WCSS significantly slows down, forming an "elbow".

Next, I trained the K-Means algorithm using $k = 4$, and the resulting cluster distribution was:

[**K-Means Clustering Insight**]

- Cluster 0: 3250 customers
- Cluster 1: 1078 customers
- Cluster 2: 4 customers
- Cluster 3: 7 customers

This distribution indicates that most customers fall into one large group, with a couple of outlier segments.

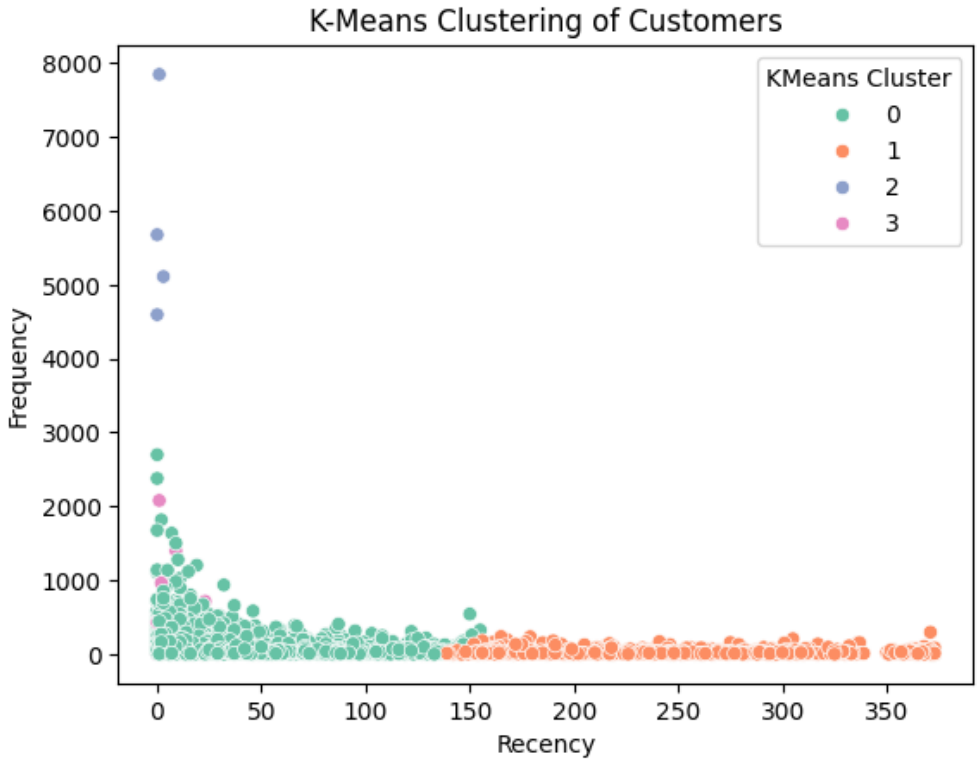

Figure 2: K-Means Clustering Result

Agglomerative Clustering

I also used Agglomerative (Hierarchical) Clustering with Ward linkage to verify the results and visualize the hierarchy of clusters through a dendrogram.

[**Dendrogram Insight**]

From the dendrogram, I observed a natural clustering structure. A horizontal cut at a linkage

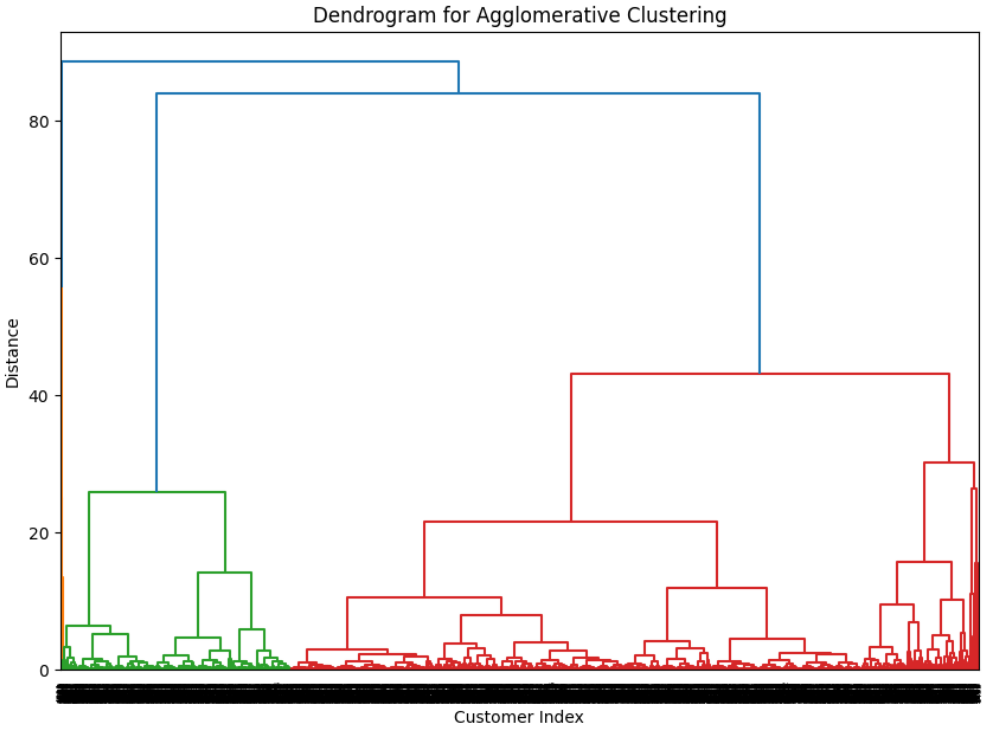

Figure 3: Dendrogram of Customer Hierarchies

distance of around **150–200** suggested the formation of approximately 4 clusters, supporting the K-Means result.

I extracted 4 clusters from the hierarchy, and the customer distribution was:

[**Agglomerative Clustering Insight**]

- Cluster 0: 3248 customers
- Cluster 1: 7 customers
- Cluster 2: 1080 customers
- Cluster 3: 4 customers

This matches very closely with the K-Means cluster sizes, giving me more confidence in the segmentation results.

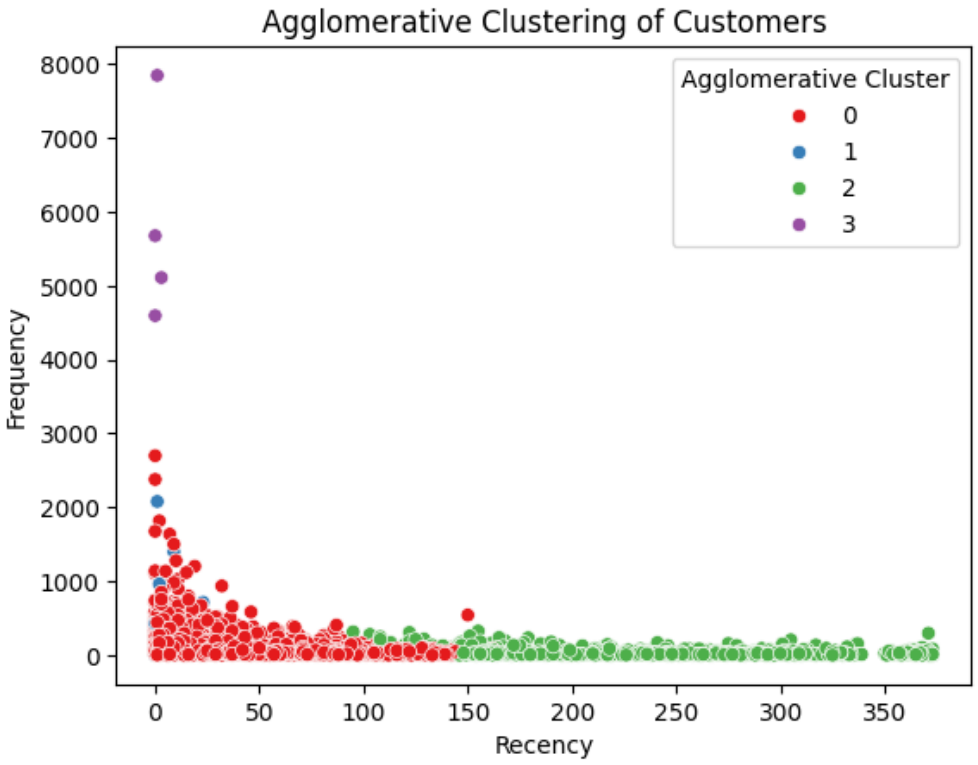

Figure 4: Agglomerative Clustering Result

PCA Visualization

To better visualize the clustering structure, I applied Principal Component Analysis (PCA) to reduce the RFM features to three principal components and plotted them in 3D.

[3D PCA Insight]

PCA helped reduce the dimensionality of the RFM space while preserving variance. The clusters appeared **reasonably distinct in 3D space**, visually validating the segmentation and showing that K-Means had separated the customers effectively based on their spending and activity patterns.

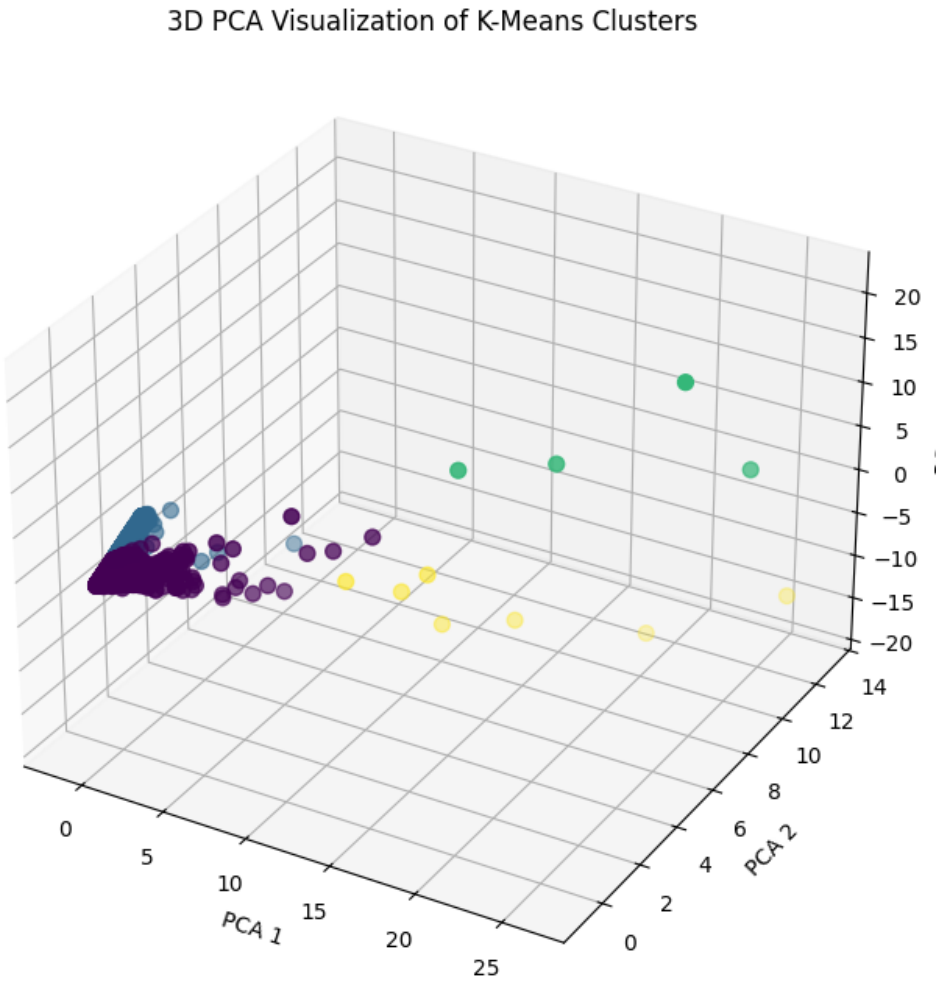

Figure 5: 3D PCA Visualization of Clusters

Clustering Evaluation

To quantitatively evaluate the quality of clustering, I used the Silhouette Score, which measures how similar an object is to its own cluster compared to other clusters (ranging from -1 to +1).

[**Silhouette Score Insight**]

- **K-Means Silhouette Score: 0.6011** – This indicates a good clustering structure with distinct separation between clusters.
- **Agglomerative Clustering Silhouette Score: 0.5955** – Also indicates good clustering, though slightly lower than K-Means.

These scores reassured me that the chosen features (Recency, Frequency, and Monetary) were well-suited for customer segmentation and that both algorithms performed reliably.

Conclusion: K-Means produced slightly more distinct clusters based on intra-cluster similarity.

Storytelling and Insights

I analyzed each cluster and discovered:

- One cluster consists of high-value, frequent customers.
- Another cluster contains recent but infrequent buyers.
- Some clusters consist of inactive or one-time buyers.

Insight: I was able to answer my original question — meaningful customer segmentation is possible with just RFM metrics. These insights are crucial for tailoring business strategies.

Impact

Positive:

- Enables targeted marketing and personalized customer experiences.
- Improves business decisions related to promotions, loyalty programs, and inventory.

Negative:

- Risk of over-personalization leading to privacy concerns.
- May unintentionally exclude or bias against less profitable customers.

It's important that clustering insights are used ethically, and customers' data privacy is maintained.

References

- UCI Machine Learning Repository: `https://archive.ics.uci.edu/ml/datasets/online+retail`
- scikit-learn Documentation: `https://scikit-learn.org`
- Python Data Science Handbook – Jake VanderPlas

Code

The complete Jupyter notebook with all steps and visualizations is available here:
`https://github.com/Minthra/Portfolio-Website/blob/main/Project4.ipynb`