AN ASSIGNMENT REPORT ON

AI-Powered Customer Purchase Analysis

By

SUCHIT SINGH NAGPAL

(sn3791@rit.edu)

FOR INTERVIEW AT BOOKEDBY

February 202

Table of Contents

Table of Illustrations

Introduction

The project/assignment is authored by Suchit Singh Nagpal as a part of interview process at BookedBy.

Problem Statement

You are a software developer at a retail company that wants to enhance its customer experience by leveraging data. The company has provided you with anonymized customer purchase data and is interested in identifying customer purchase patterns, classifying customers, and recommending products to customers based on their purchase history.

Dataset Creation

The synthetic dataset was generated using a Python notebook to simulate realistic customer purchase behavior in a retail environment. The resulting dataset meets the assignment requirements and includes the following fields:

- Order ID: A unique identifier assigned to each purchase order.
- Customer ID: An identifier representing one of 500 distinct customers (ranging from C1 to C500).
- Product ID: A unique identifier for each product.
- Product Category: The category of the product, chosen from Electronics, Clothing, Home & Kitchen, Books, or Sports.
- Purchase Amount: A monetary value for the purchase, calculated using realistic ranges based on the product category.
- Purchase Date: The date on which the purchase occurred, randomly assigned within the year 2023.

Code Overview

1. Library Imports:
   The script begins by importing necessary libraries:
   - csv for CSV file operations
   - random for random data selection
   - os for directory management
   - datetime and timedelta for date generation
2. Product Generation:
   The generate_products function creates 50 products distributed across 5 predefined categories. Each product is assigned a unique Product ID and a corresponding category, ensuring diversity within the product dataset.
3. Customer Generation:
   The generate_customers function generates 500 customer identifiers (C1 through

C500), establishing a broad customer base for the simulation.

4. Random Date Generation:
   The random_date function produces a random date within a specified range, ensuring that purchase dates are realistically distributed over the year 2023.
5. Purchase Record Generation:
   The generate_purchases function simulates purchase records by:
   - Randomly selecting a customer for each order.
   - Assigning a random order date.
   - Determining a random number of items per order (between 1 and 6).
   - For each item, selecting a random product and calculating a realistic purchase amount based on the product's category.
     Each order is assigned a unique Order ID to maintain traceability.
6. Saving Data to CSV:
   The save_to_csv function writes the generated purchase records to a CSV file. This file includes a header row corresponding to the required attributes, facilitating subsequent data analysis.
7. Execution Flow:
   The main function coordinates the data generation process by invoking the product and customer generation functions, setting the purchase date range for 2023, creating the necessary directory structure, generating purchase records, and finally saving the complete dataset to a CSV file named customer_purchases3.csv within a designated data folder.

This approach ensures that the dataset not only conforms to the required structure but also closely mimics real-world retail purchase data, providing a robust foundation for further analysis, customer segmentation, and product recommendation tasks.

DATA ANALYSIS

1. Top-Selling Categories

Aggregates purchase data by product category to determine which categories generate the highest total revenue and purchase frequency. These visualizations provide insights into the primary revenue drivers and the popularity of each category among customers.

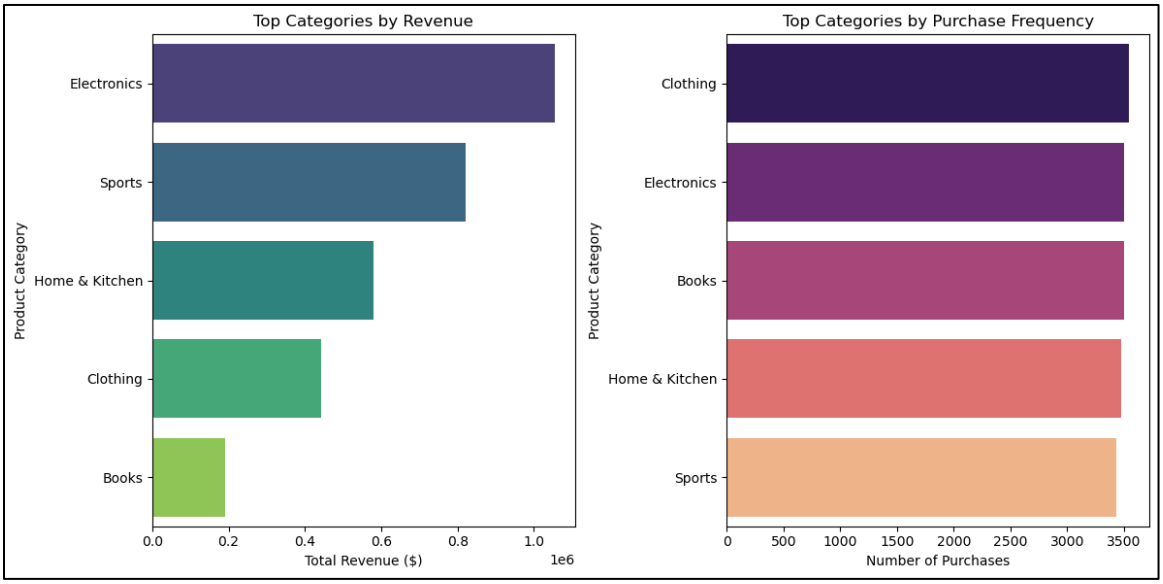

Figure 1: Top Selling Categories

2. Top-Selling Products

The top 10 products are identified based on both total revenue and purchase frequency. The data is grouped by product ID to calculate the revenue each product has generated, as well as how frequently each product is purchased.

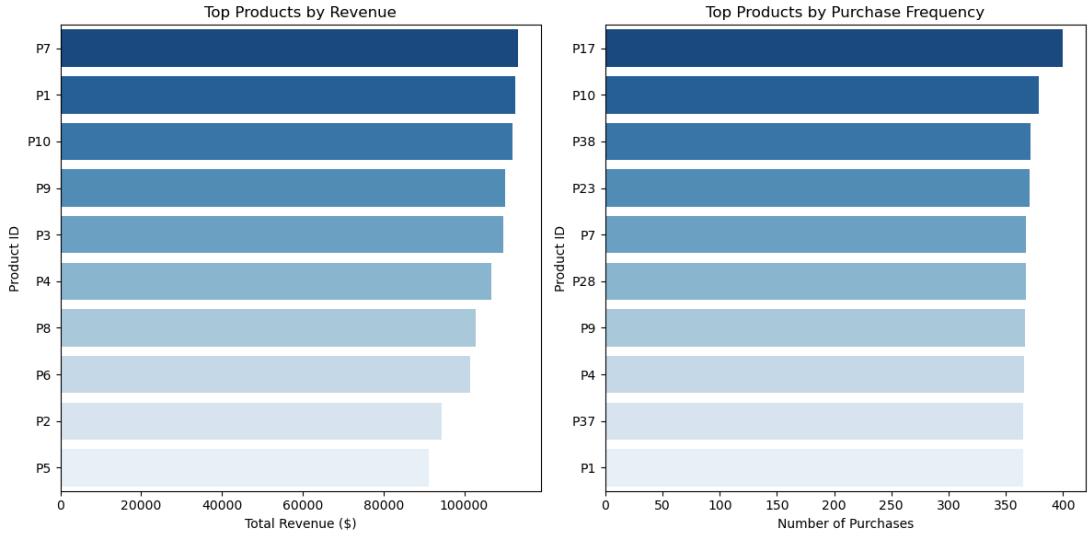

Figure 2: Top Products

3. Average Customer Spending per Order

Average spending per order for each customer by first aggregating the purchase amounts at the order level and then computing the mean spending per customer.

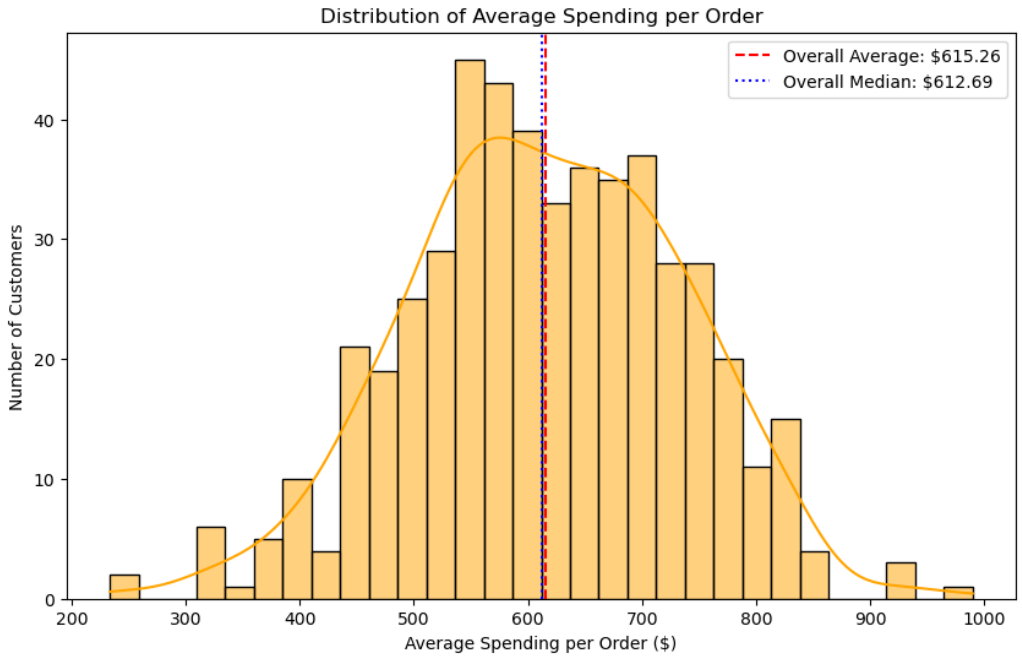

Figure 3: Average Customer Spending Per Order

4. Top Customers by Average Spending per Order

Customers are ranked based on their average spending per order to identify the top 10 customers contributing the highest revenue on a per-order basis.

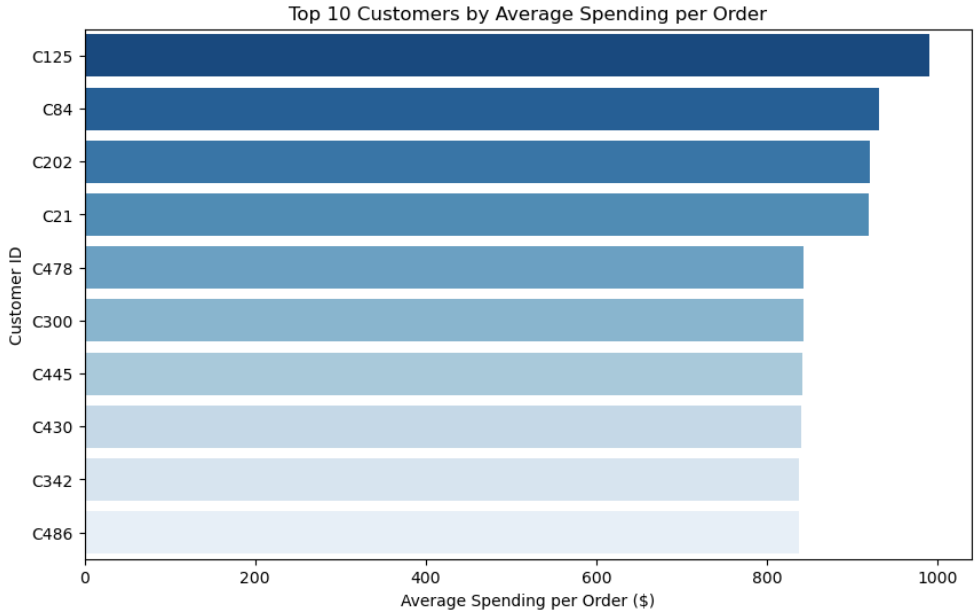

Figure 4: Top Customers by Avg Spending per Order

5. Correlation Between Number of Orders and Average Spending per Order

This section examines the relationship between the number of orders and the average spending per order for each customer. By grouping the data accordingly, customers are categorized into quadrants based on the median values of order count and average spending.

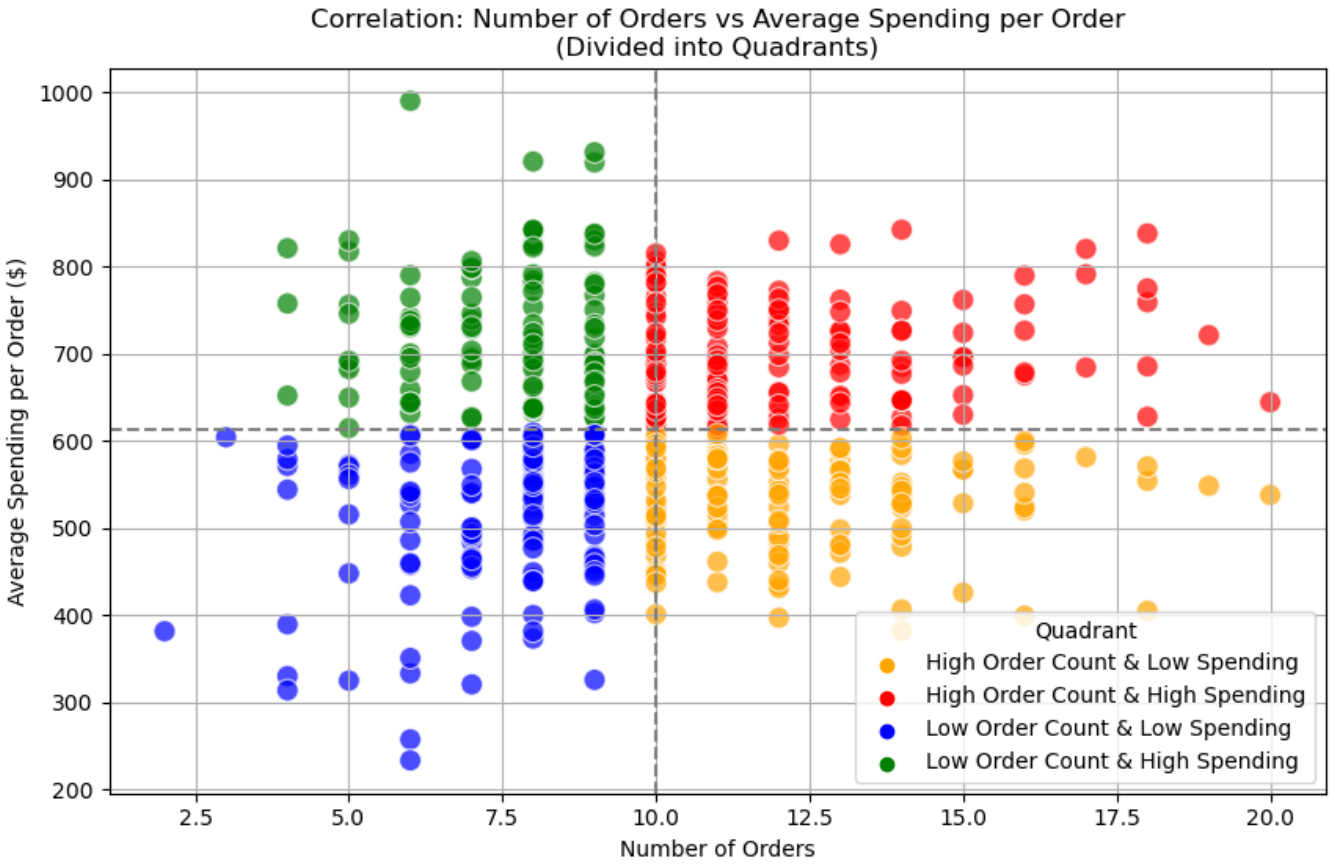

Figure 5: Correlation b/w no. of orders and avg. spending per order

6. Average Spending per Order by Category

The analysis investigates variations in average spending per order across different product categories. Group the data by product category and calculating the mean purchase amount.

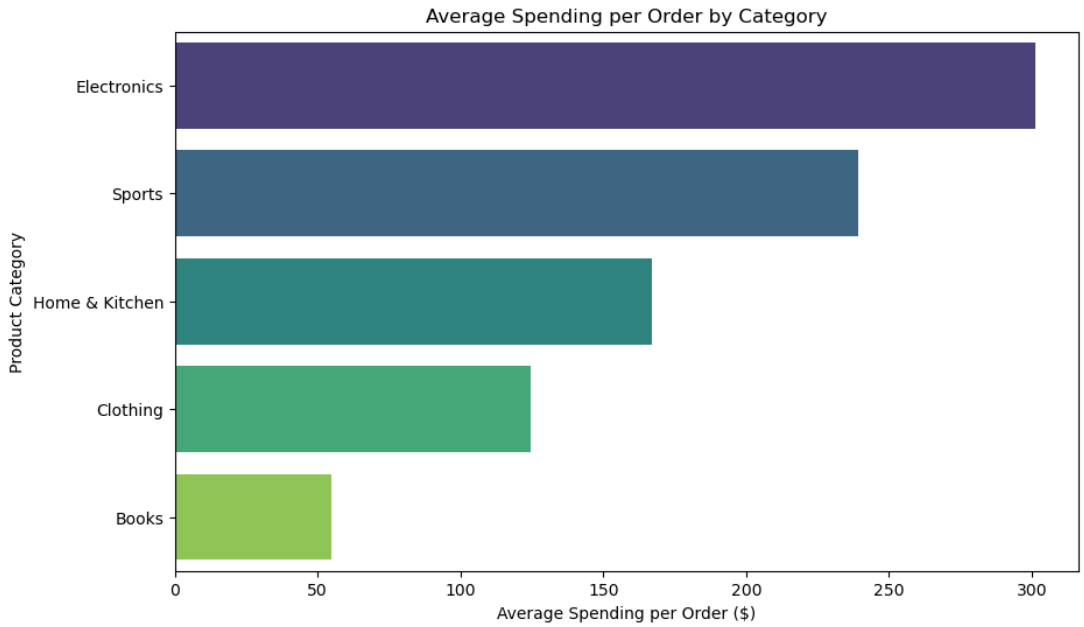

Figure 6: Average Spending per Order by Category

7. Average Order Value (AOV)

Average Order Value (AOV) is computed by summing the purchase amounts for each order and then calculating the mean and median order values across all orders.

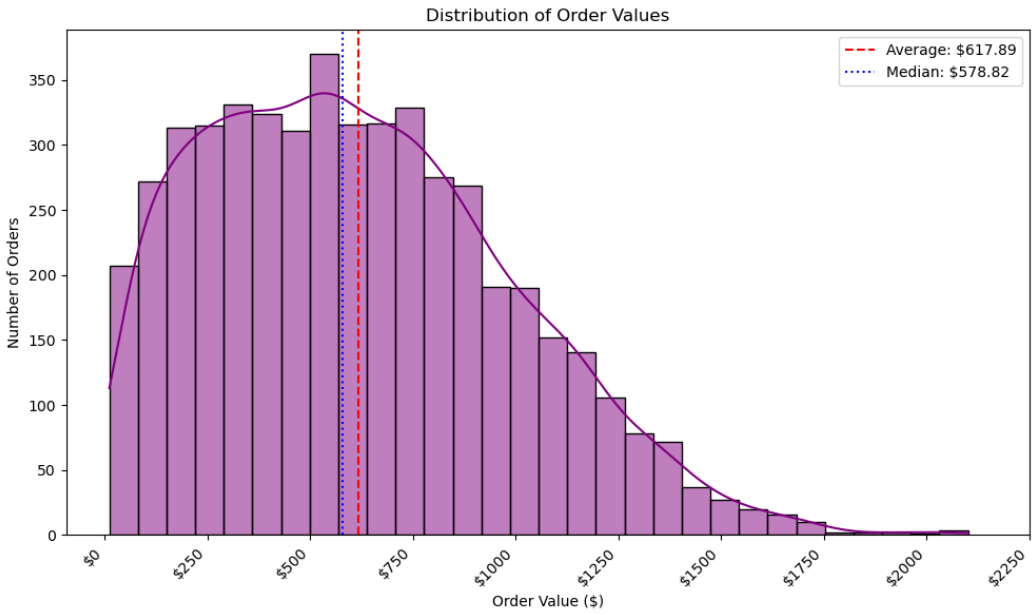

Figure 7: Average Order Value

8. Average Spending per Customer

Lifetime spending per customer is determined by aggregating all purchase amounts per customer over the course of 2023 (Lifetime).

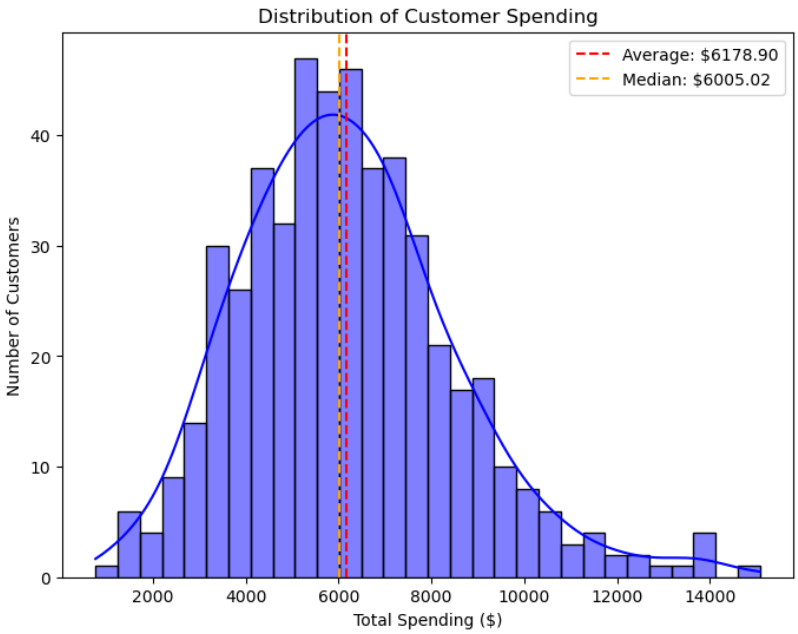

Figure 8: Lifetime Average Spending per Customer

9. Top Customers by Spending

The top 10 customers are identified based on their total spending, or lifetime value, by ranking customers according to their aggregated purchase amounts.

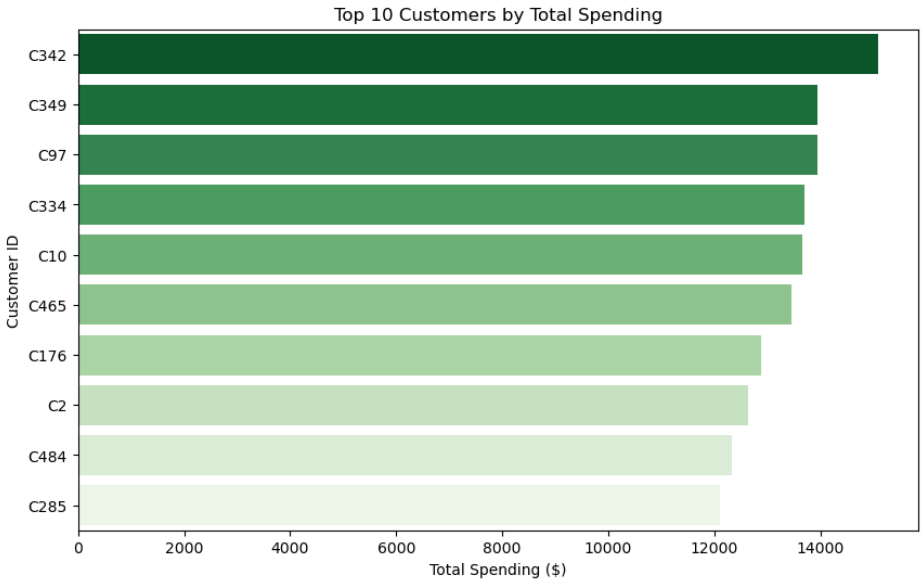

Figure 9: Top Customers by Lifetime Spending

10. Monthly Revenue Trends

Revenue trends are analyzed by aggregating purchase data monthly.

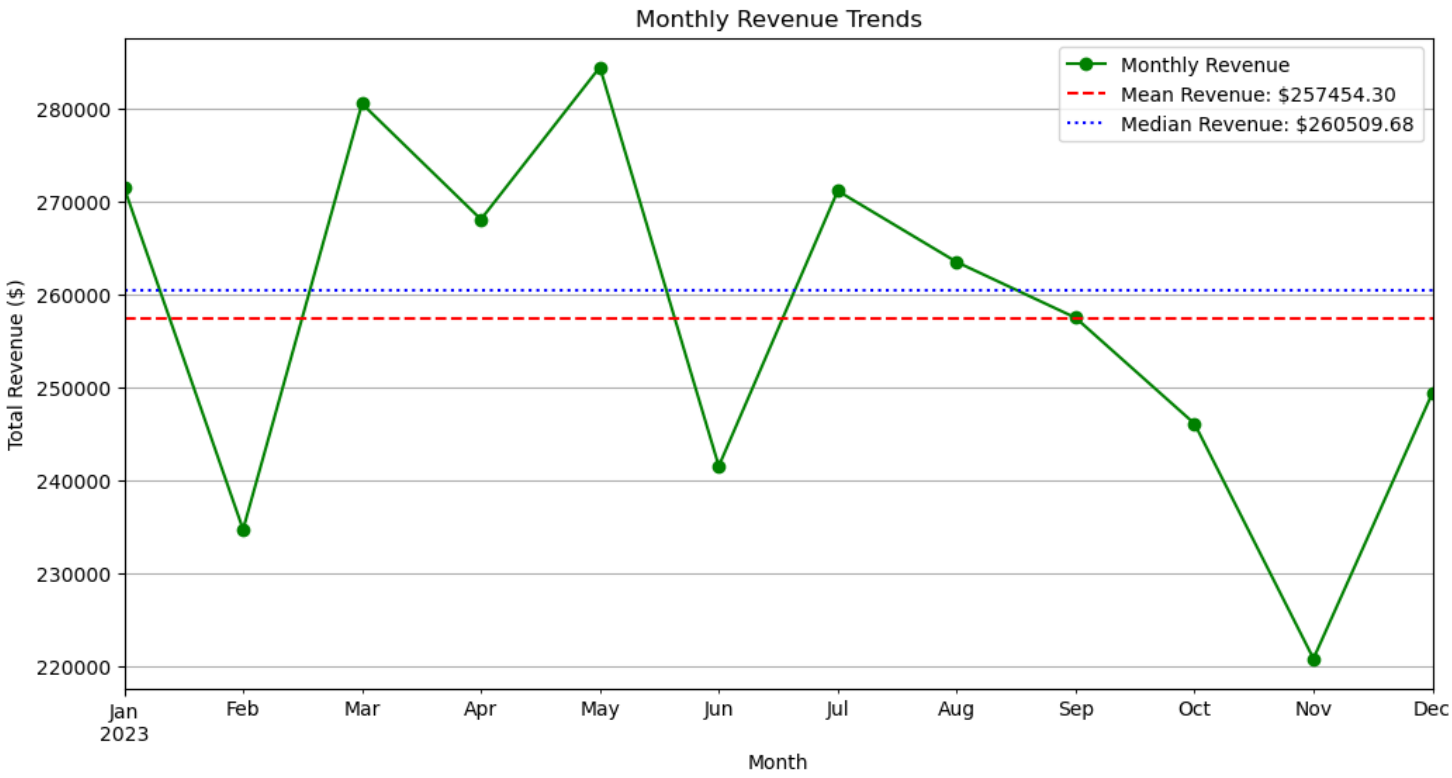

Figure 10: Monthly Revenue Trend

11. Customer Cohort Analysis

Customer cohort analysis is conducted by determining the first purchase month (cohort month) for each customer and tracking retention over subsequent months. A retention matrix is created by calculating the number of unique customers making repeat purchases for each cohort, and a heatmap visualization is generated to display these retention patterns. This analysis offers valuable insights into customer retention and the longevity of customer relationships.

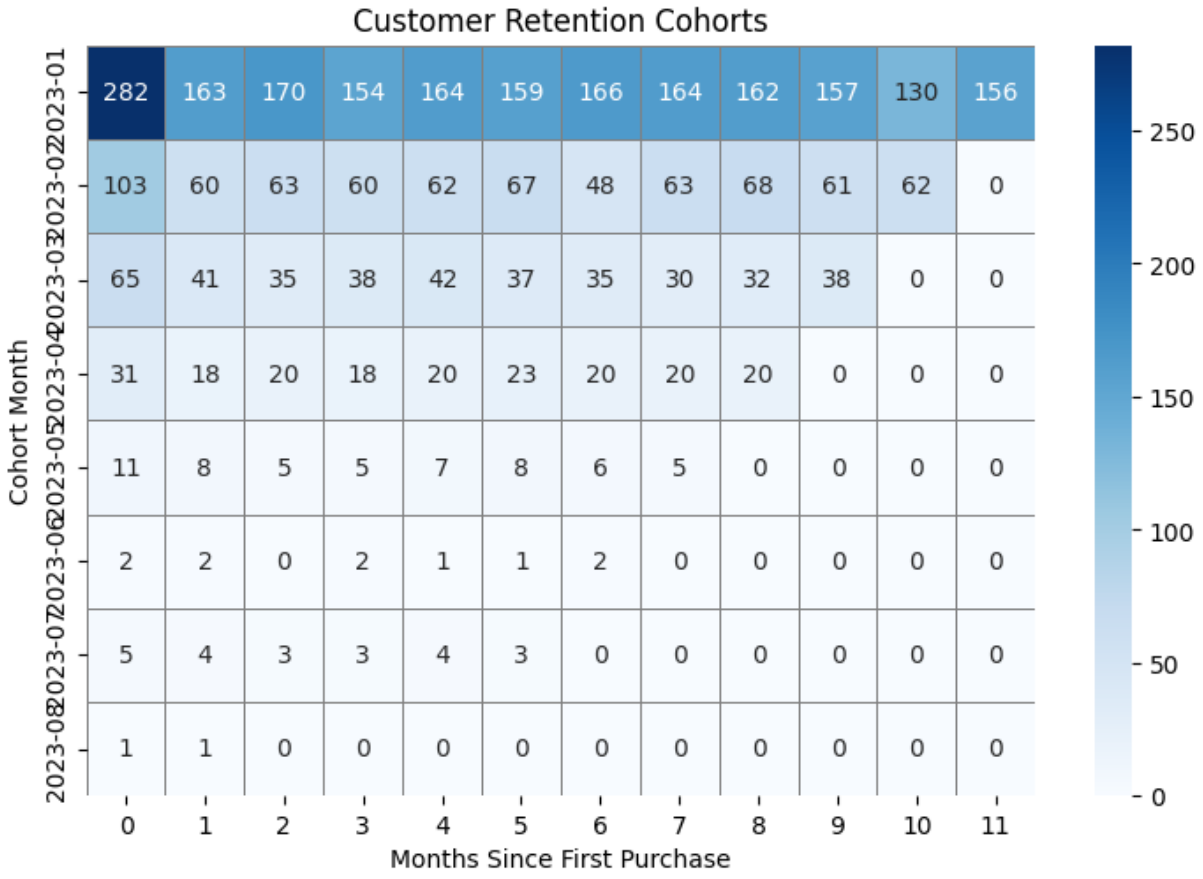

Figure 11: Customer Retention Cohort

Classification: Clustering

The classification section employs various clustering techniques to segment customers and products based on their purchasing behavior. The objective is to derive actionable insights by grouping similar entities, which can inform targeted marketing strategies, inventory management, and customer engagement initiatives. The following subsections outline the methodologies and insights derived from the clustering analyses.

1. Customer Segmentation via K-Means Clustering

Objective:
Segment customers by their purchasing behavior to identify distinct groups based on order frequency, total spending, and category preferences.
Methodology:
Feature Engineering:
Frequency: Calculated as the number of unique orders per customer.
Total Spending: Aggregated purchase amounts for each customer.
Category Preferences: Computed as the percentage of spending in each product category, offering insights into individual customer inclinations.

Elbow Method:
An inertia plot is generated for varying cluster counts (1 to 10) to determine the optimal number of clusters. This method helps identify the point at which adding additional clusters offers diminishing returns in reducing variance.

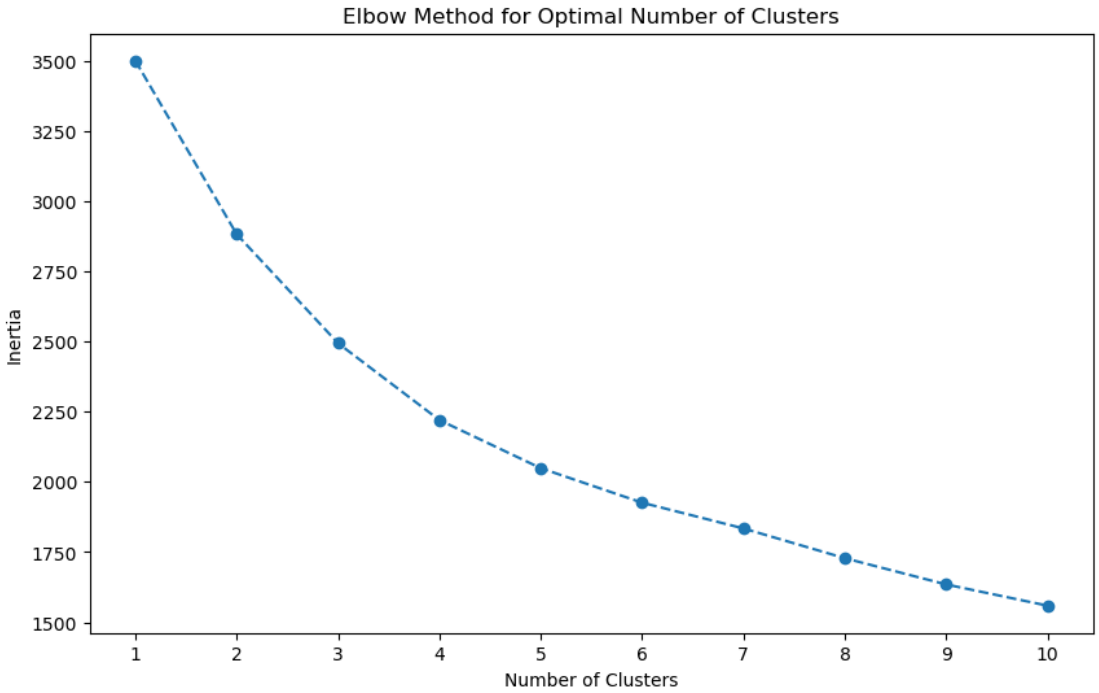

Figure 12: Plot for Elbow Method for Customer Segments

Clustering Implementation:
K-Means clustering is applied with the optimal number of clusters (4 in this case derived from elbow method above).

Cluster Labeling and Visualization:
Intuitive labels (e.g., "Budget Shoppers," "Frequent Buyers," "High Spenders," "Occasional Shoppers") are assigned based on the observed behavior.

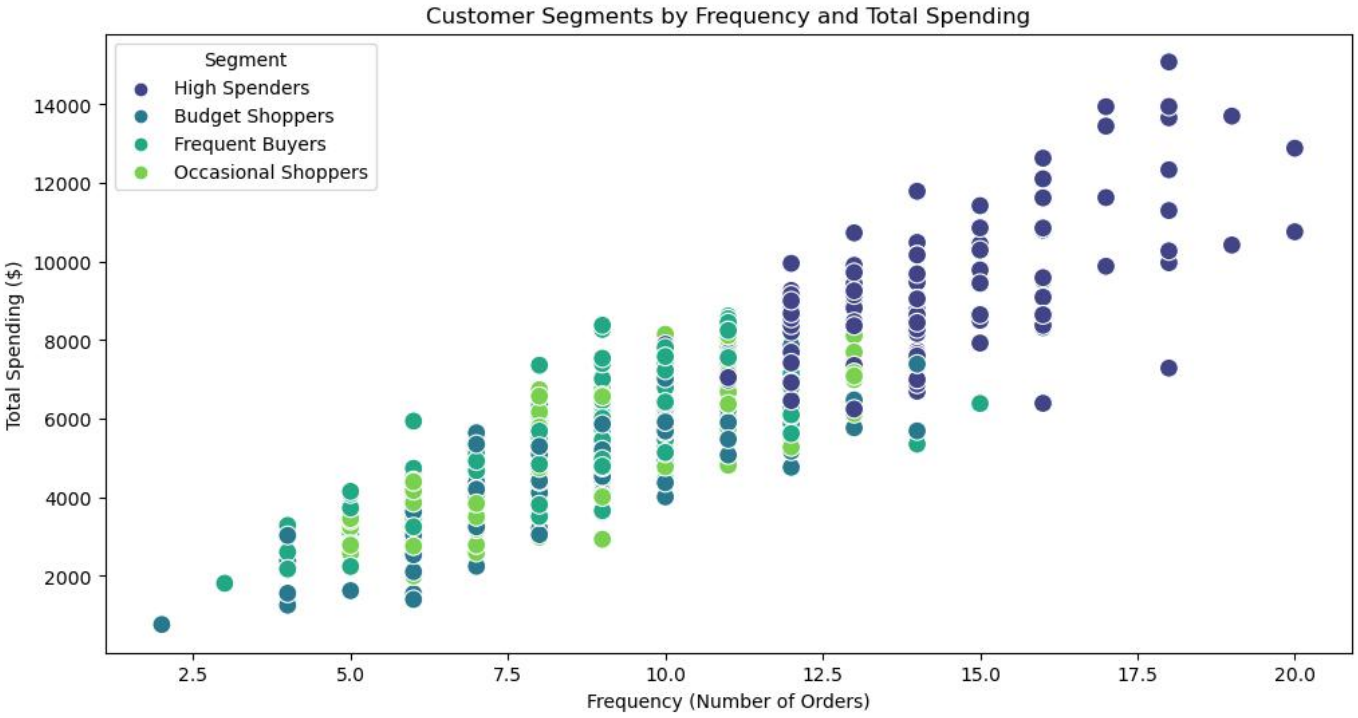

Figure 13: K-Means clustering for Customer Segments by Frequence and Total Spending

Insights:
This analysis identifies distinct customer segments, enabling targeted strategies for high-value or frequent customers, while also highlighting potential areas for encouraging increased spending among lower-value segments.

As speculated, the nature of the cluster plot for these attributes is linear, since there exists a strong correlation between frequency and spending. The data is almost uniform, with each order value being normalized and having less variance.

2. 3D Clustering Exploration

Objective:
Examine customer segmentation in a multidimensional context by incorporating an additional behavioral attribute (i.e., preference for a specific product category).

Feature Selection:
A subset of features—frequency, total spending, and a chosen category metric (e.g., Sports or Books)—is used to create a three-dimensional feature space.

Clustering Implementation:
K-Means clustering is applied with three clusters, and intuitive labels (e.g., "Active Bargain Hunters," "Elite Category Enthusiasts," "Casual Shoppers") are assigned.
Visualization:

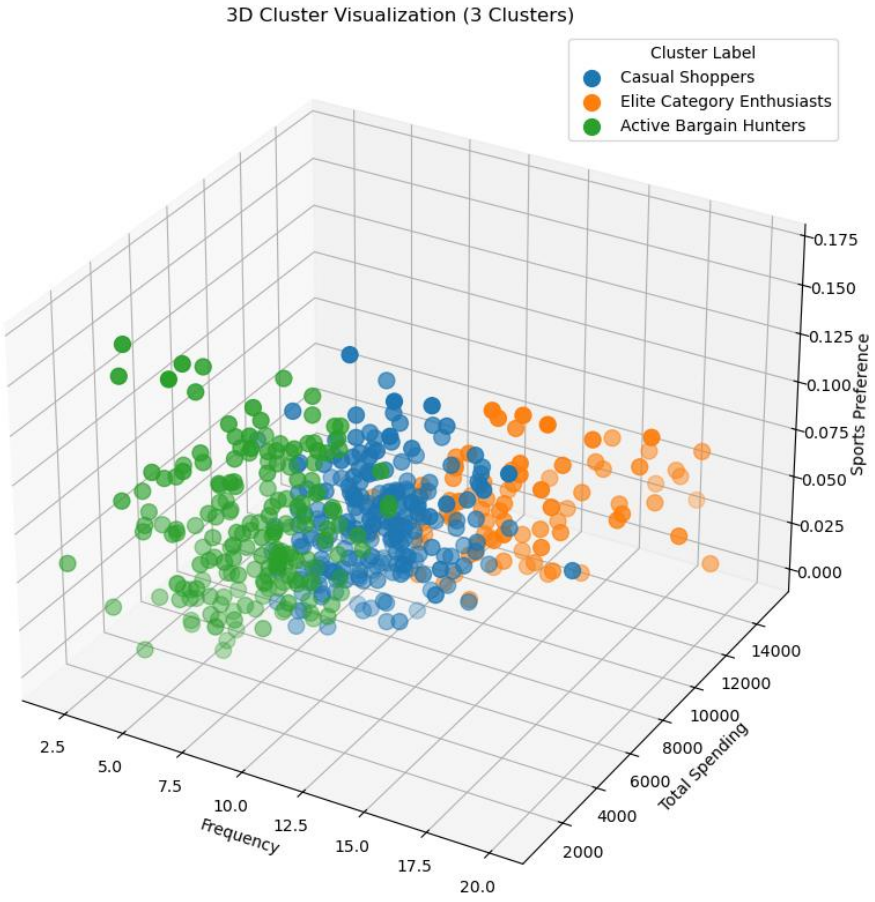

Figure 14: 3D Cluster (Scatter Plot) for Product preference

Insights:
The 3D clustering analysis provides a nuanced perspective of customer segments, facilitating the identification of specialized groups with distinct category preferences and enabling more personalized marketing strategies.

3. Product Clustering via Agglomerative Clustering

Objective:
Group products based on their sales performance to gain insights into product-level demand and value.

Feature Aggregation:
Products are analyzed by aggregating total purchase amounts and order counts.

Clustering Implementation:
Agglomerative Clustering (using Ward's linkage) is applied to the aggregated product data, and clusters are derived based on sales performance.
Cluster Labeling and Visualization:
Each product cluster is assigned an intuitive label such as "High-Value, High-Demand Products," "High-Value, Low-Demand Products," "Low-Value, High-Demand Products," and "Low-Value, Low-Demand Products."

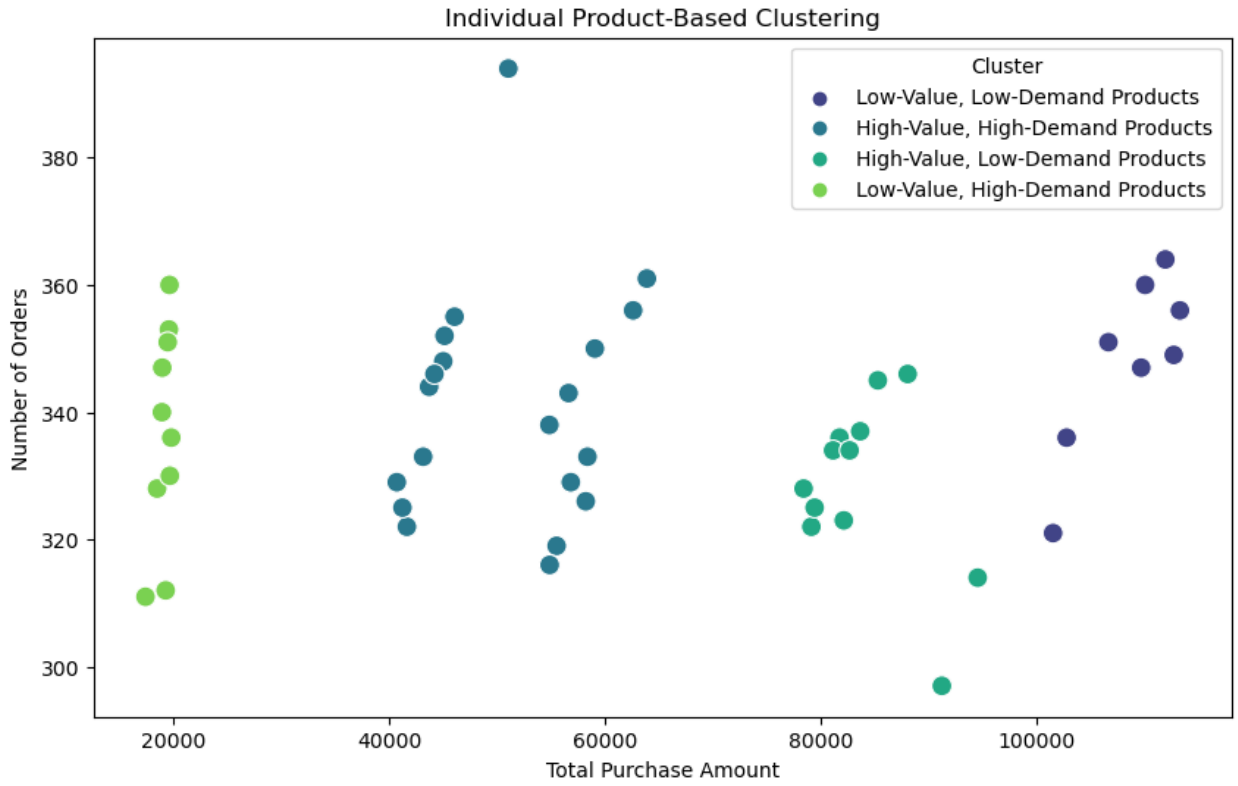

Figure 15: Product Based Agglomerative Clustering

Insights:
This product clustering enables identification of key products that drive revenue, as well as those that may require promotional efforts or inventory adjustments based on demand patterns. Product – Cluster data can be seen in the ipynb file.

4. RFM (Recency, Frequency, Monetary) Analysis

Objective:
Segment customers using the RFM model to assess their engagement and value based on recent activity, purchase frequency, and monetary contributions.

RFM Calculation:

- *Recency*: Measured as the number of days since the customer's last purchase.
- *Frequency*: Determined by the number of unique orders per customer.
- *Monetary*: Total spending by the customer.

Elbow Analysis for RFM Segmentation:
An inertia plot is generated for a range of cluster numbers to identify the optimal segmentation solution.

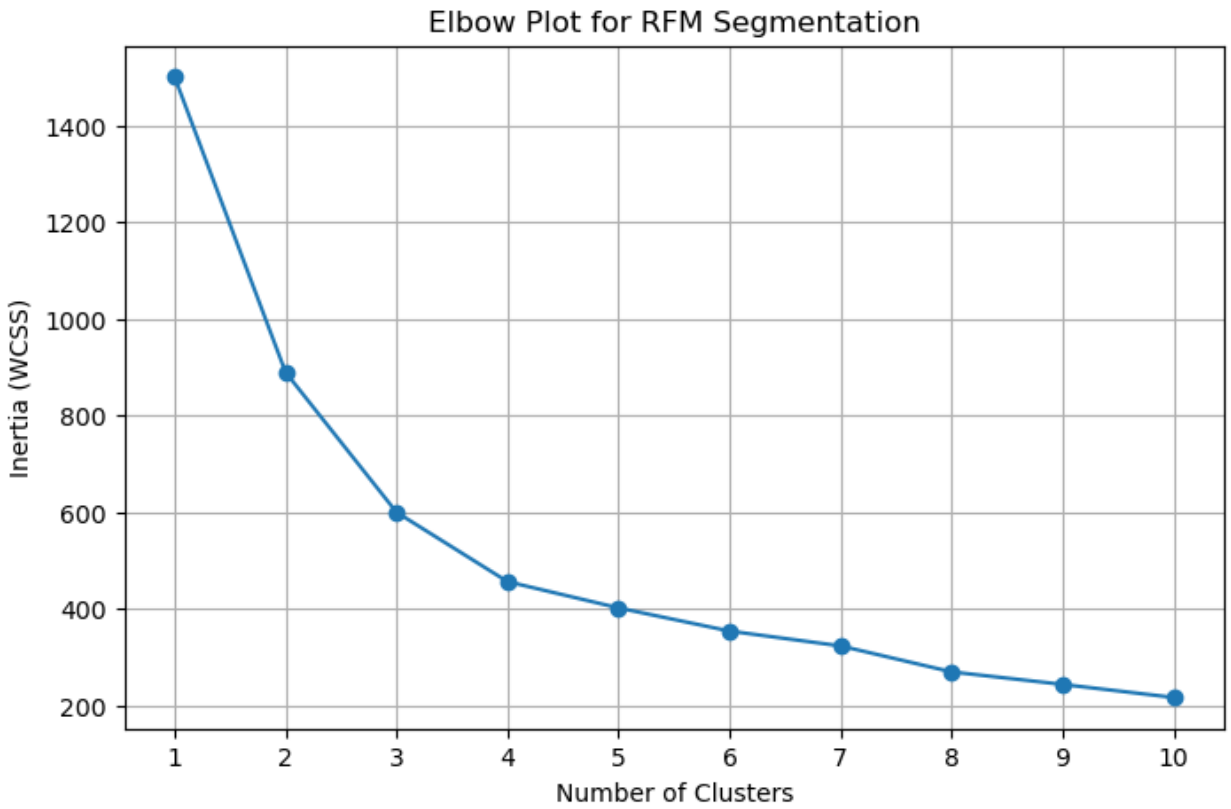

Figure 16: Elbow Plot for RFM Segmentation

Clustering Implementation:
K-Means clustering is applied to the standardized RFM data (with 3 clusters selected based on the elbow method). Clusters are then labeled intuitively (e.g., "Budget Regulars," "Frequent High Rollers," "At Risk Customers").

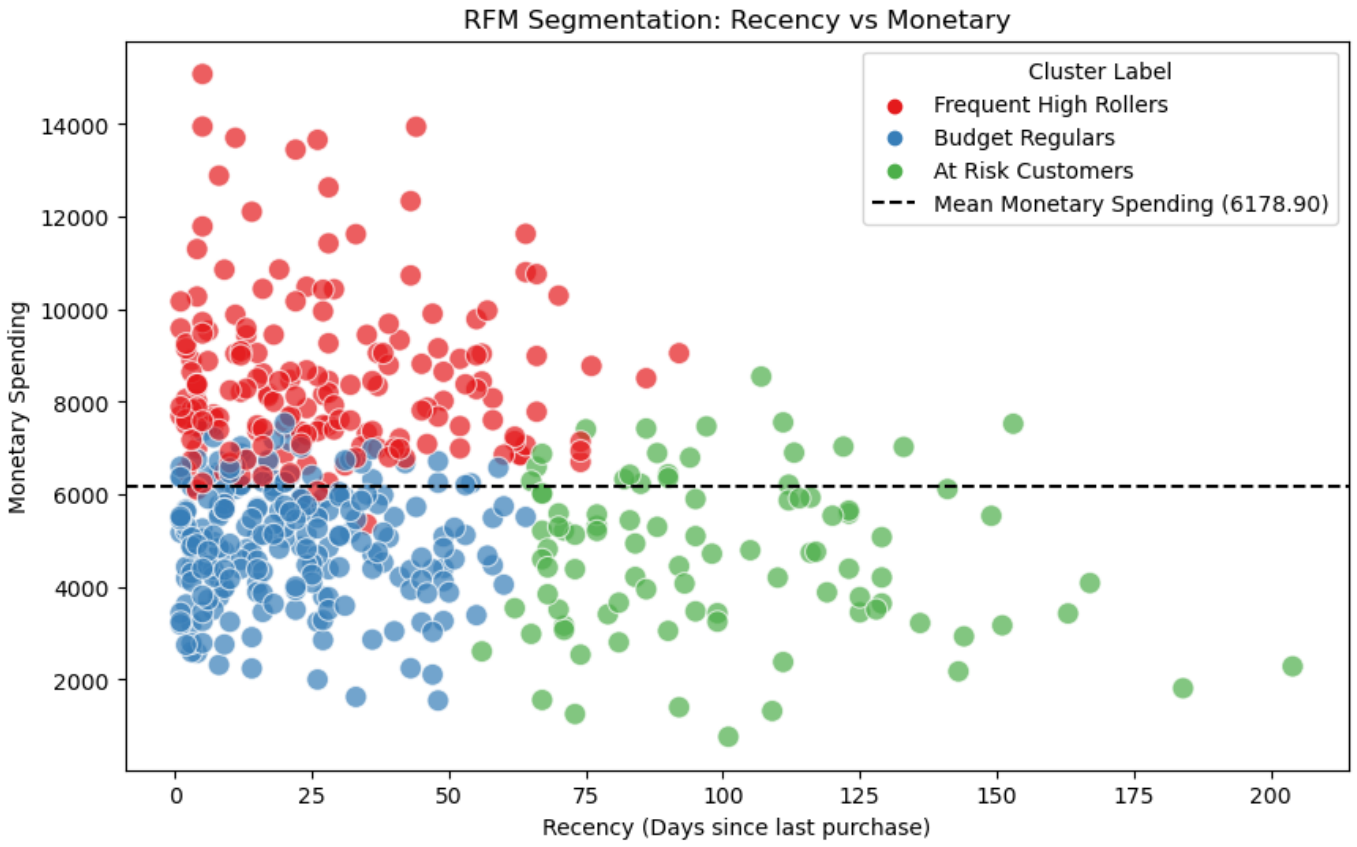

Figure 17: RFM Clustering

RFM Clusters:

	Customer ID	Recency	Frequency	Monetary	Cluster	Cluster Label
0	C1	28	13	7491.83	1	Frequent High Rollers
1	C10	26	18	13659.33	1	Frequent High Rollers
2	C100	11	12	6500.60	1	Frequent High Rollers
3	C101	35	14	5353.91	1	Frequent High Rollers
4	C102	55	15	9785.58	1	Frequent High Rollers
..	...	...	...	...	...	...
495	C95	28	15	11422.79	1	Frequent High Rollers
496	C96	105	9	4793.56	2	At Risk Customers
497	C97	44	17	13941.50	1	Frequent High Rollers
498	C98	5	8	3813.56	0	Budget Regulars
499	C99	16	11	7047.33	1	Frequent High Rollers

[500 rows x 6 columns]

Figure 18: RFM Cluster Data

Insights:
The RFM analysis provides a comprehensive view of customer engagement, identifying high-value customers as well as those at risk of churn. This segmentation supports tailored retention strategies and targeted marketing efforts to optimize customer lifetime value.

Each clustering approach described herein offers unique insights into both customer and product behaviors, facilitating data-driven decision-making that can enhance overall business performance.

Recommendation: Product Recommendation Strategies

The recommendation section implements four methods to suggest products based on customer purchase data. These methods leverage different aspects of customer behavior and product characteristics to generate personalized suggestions. The following subsections describe each recommendation strategy, including its operational mechanism, potential use cases, and advantages.

1. Collaborative Filtering

Overview:
Collaborative filtering relies on the assumption that customers with similar purchasing behavior will be interested in similar products. This method computes similarities between customers based on their purchase histories and uses these similarities to recommend products that similar customers have purchased.

Working:
For a given customer, the method identifies similar users and aggregates their purchase data (weighted by similarity) to generate a ranked list of recommended products that the target customer has not yet purchased.

Visualization:
Heatmap of customer similarity for 10 customers.

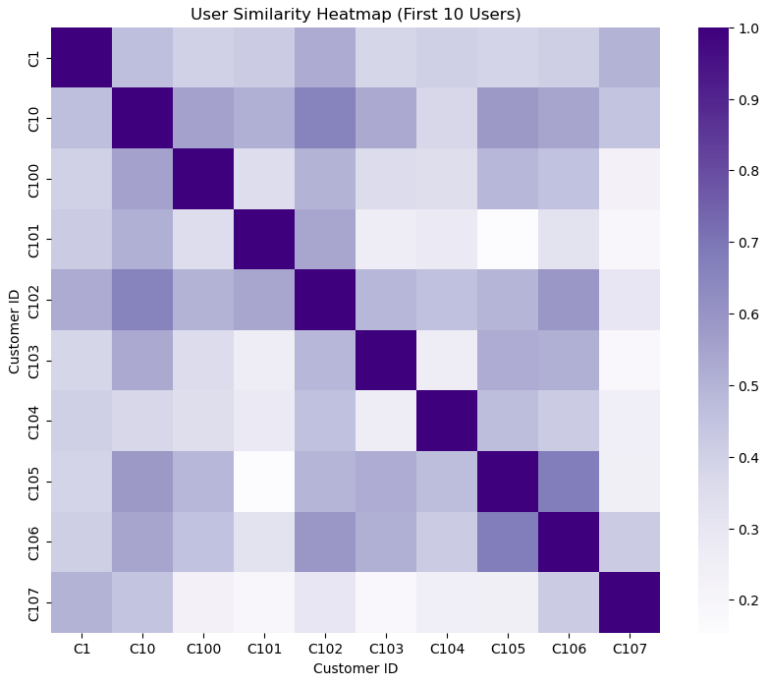

Figure 19: Customer Similarity Heatmap

Sample Usage:

```
    # Example recommendation for a sample customer
    sample_customer = user_item_matrix.index[38]
    print(f"Collaborative Recommendations for Customer {sample_customer}:")
    print(recommend_products_collaborative(sample_customer))
```

✓ 0.0s

```
Collaborative Recommendations for Customer C133:
Product ID
P9     38794.969494
P3     38231.392848
P4     36471.309049
P5     30952.823533
P43    28962.017826
```

Figure 20: Sample usage for Collaborative Filtering Recommendation

Use Cases:

- When sufficient historical purchase data is available.
- In scenarios where customer behaviors are similar, such as frequently purchased items.
- Particularly effective in environments with diverse customer bases and varied purchasing patterns.

Pros:

- Leverages collective behavior to make recommendations.
- Can capture complex patterns that are not apparent from product attributes alone.
- Adaptive to changes in customer behavior over time.

2. Content-Based Filtering

Overview:
Content-based filtering recommends products by comparing product attributes rather than relying solely on customer behavior. This approach is particularly useful when product information (e.g., category, description) is rich.

Working:

- A **product content DataFrame** is created, using product attributes such as the product category.
- **Text-based vectorization** (using CountVectorizer) is applied to encode these attributes.
- **Cosine similarity** is computed between products based on their attribute vectors.
- For a given product, the system identifies and ranks similar products by comparing their content features.

Visualization:

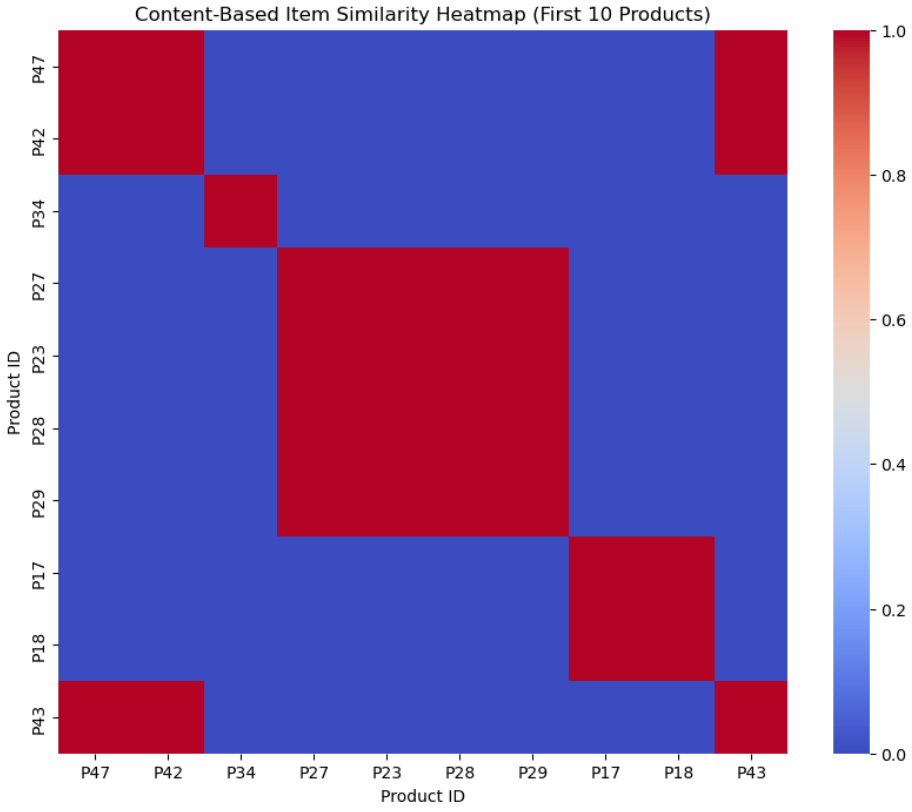

Figure 21: Content Based Item Similarity Heatmap

Observation:
Since our dataset isn't very rich in product data, it doesn't have fields like product rating, description or reviews, this recommendation isn't the best fit for our case.

Sample Recommendation:

```
# Example recommendation for a sample product
sample_product = product_content.index[17]
print(f"Content-Based Recommendations for Product {sample_product}:")
print(recommend_products_content(sample_product))
```

✓ 0.0s

```
Content-Based Recommendations for Product P45:
Product ID
P47    1.0
P43    1.0
P42    1.0
P50    1.0
P48    1.0
P44    1.0
P49    1.0
```

Note: One hot encoding used for similarity

Use Cases:

- Useful for recommending niche or specialized products where user interaction data may be sparse.
- Effective in scenarios where products have well-defined attributes (e.g., category, description, specifications).
- Can be used to suggest similar items during product searches or when a user is viewing a particular product.

Pros:

- Independent of user interactions; recommendations can be generated even with limited user data.
- Can highlight product similarities that may not be evident through collaborative patterns.
- Provides transparency in the recommendation process based on clear product attributes.

3. Item-Based Collaborative Filtering

Overview:
Item-based collaborative filtering derives recommendations by analyzing the relationships between items rather than users. It is based on the idea that products purchased together or similarly across multiple customers share commonalities.

Working:
An **item-user matrix** is created (the transpose of the user-item matrix).
Cosine similarity is calculated between items based on customer purchase patterns.
For a given product, the method identifies similar products that are frequently bought together, using the computed item similarity matrix.

Visualization:
Similarity Heatmap for 10 products

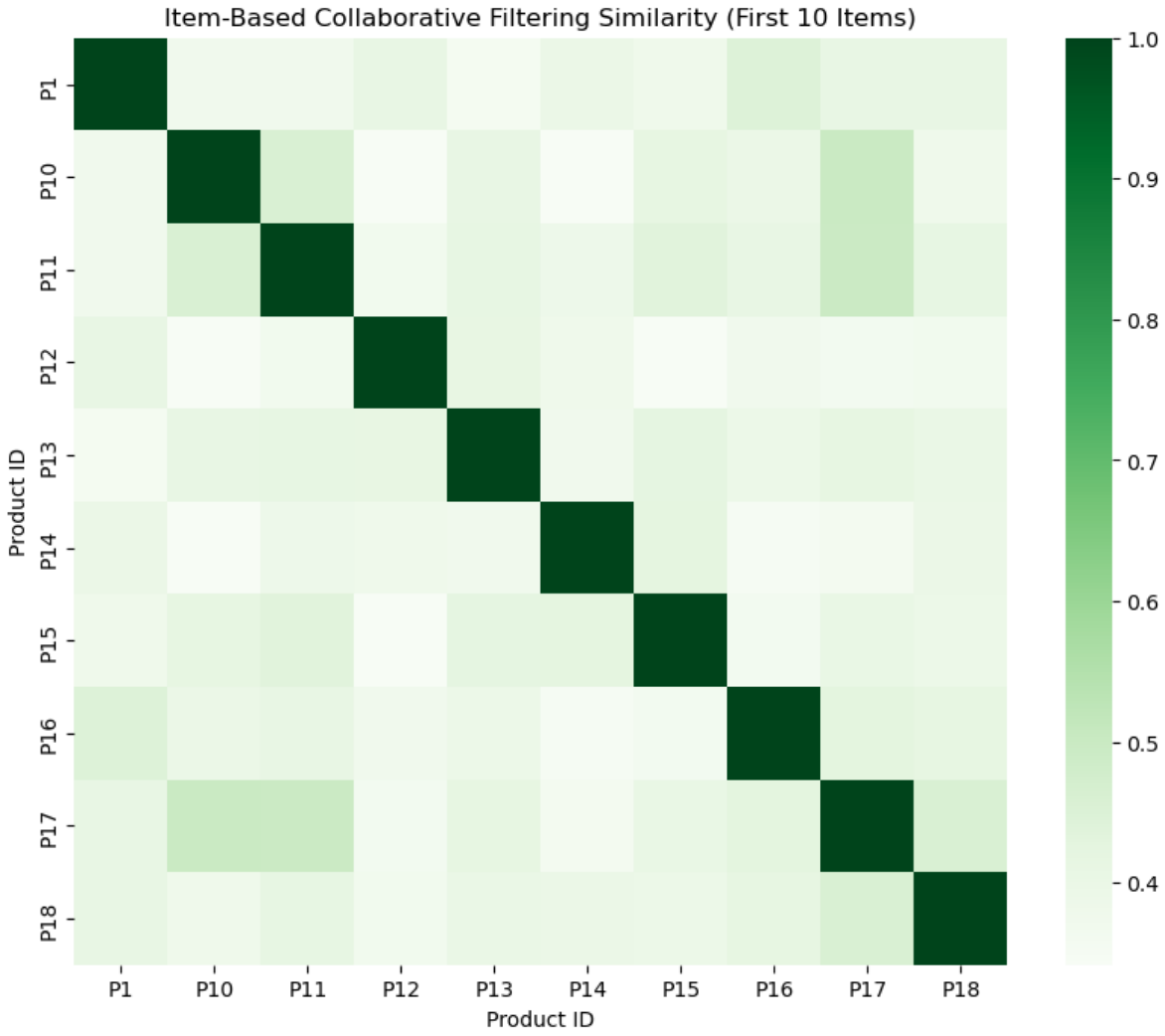

Figure 22: Item-Based Collaborative Filtering Similarity Heatmap

Sample Usage:

```
# Example recommendation for a sample product using item-based filtering
print(f"Item-Based Recommendations for Product {sample_product}:")
print(recommend_products_item_based(sample_product))
```

✓ 0.0s

```
Item-Based Recommendations for Product P45:
Product ID
P32    0.471827
P17    0.451141
P3     0.445878
P9     0.441234
P50    0.433998
Name: P45, dtype: float64
```

Figure 23: Sample Usage of Item Based Filtering

Use Cases:

- Start of an order (empty cart) or while checkout (you forgot your essentials)
- Suitable for providing recommendations when customers are initiating an order or during the checkout process.
- Effective for suggesting essential or complementary products that are commonly purchased together.
- Useful in cross-selling and upselling strategies by identifying items that pair well.

Pros:

- Focuses on product-to-product relationships, making it robust in environments with many overlapping purchases.
- Can quickly adapt to changing product trends and seasonal variations.
- Reduces the impact of sparse data on individual customer profiles by leveraging overall item popularity.

4. Hybrid Filtering

Overview:
Hybrid filtering combines multiple recommendation strategies (typically collaborative and content-based) to produce more robust and holistic product suggestions. By blending insights from both customer behavior and product attributes, hybrid systems aim to overcome the limitations of individual approaches.

Working:

- **Collaborative scores** are generated based on similar customers' purchase histories.
- **Content scores** are computed as the average similarity between candidate products and the products previously purchased by the customer.
- A **weighted sum** of the collaborative and content scores is calculated (controlled by a parameter, alpha), producing a final ranking of product recommendations.

Sample Usage:

```
print(f"Hybrid Recommendations for Customer {sample_customer}:")
print(recommend_products_hybrid(sample_customer))
```

✓ 0.0s

```
Hybrid Recommendations for Customer C133:
Product ID
P9     19397.604747
P3     19115.816424
P4     18235.774524
P5     15476.531767
P43    14481.088913
dtype: float64
```

Figure 24: Sample Usage for Hybrid Filtering

Use Cases:

- Particularly valuable when both rich customer behavior data and detailed product information are available. More holistic recommendations.
- Can be applied to provide more personalized and accurate recommendations.
- Useful in scenarios where single-method approaches may yield suboptimal results due to data sparsity or overspecialization.

Pros:

- Mitigates the limitations inherent in purely collaborative or content-based methods.
- Enhances recommendation accuracy by incorporating multiple data perspectives.
- Provides flexibility to adjust the influence of each component (customer behavior vs. product attributes) based on context.

5. Comparison of Recommendation Methods

A comparative analysis of the four methods—collaborative filtering, content-based filtering, item-based collaborative filtering, and hybrid filtering—allows for the identification of the most suitable approach under different circumstances.

Sample Run for comparative results of all 4 recommendation techniques:

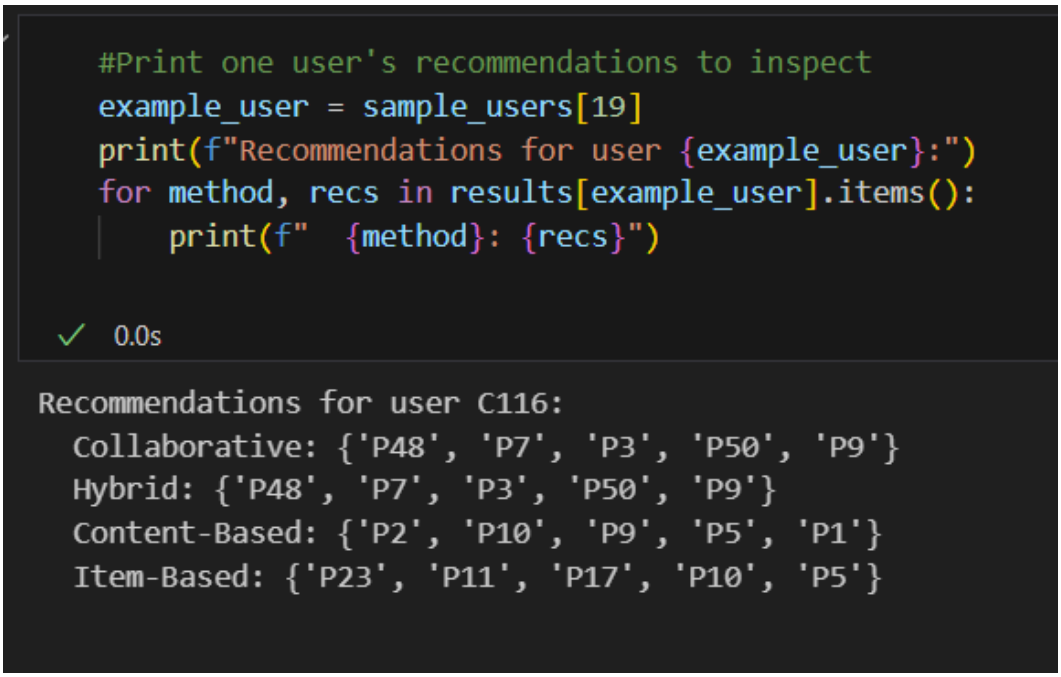

Figure 25: Sample Run for comparative results of all 4 recommendation techniques

Observation:
Overlapping elements between the four recommendations.
Intersection of these sets reflect the strongest co-relations, and best recommendations.

These strategies, when applied appropriately, can significantly enhance the customer experience by delivering personalized and relevant product recommendations.

Bonus Section

Apriori Associative Rule Mining

This analysis employs the Apriori algorithm to identify co-occurrences of product categories in customer orders. The objective is to detect strong correlations between categories that are frequently purchased together. While many of the discovered rules involve only two categories, these are deemed less insightful. Therefore, the focus was shifted to identifying three-element associations. In the current dataset, six three-category rules exceeded the set threshold (6 out of 10), indicating robust associations. In a more realistic, diverse, and non-uniform dataset, this approach could be extended to individual products to uncover even deeper insights.

Graph for 2-element rules:

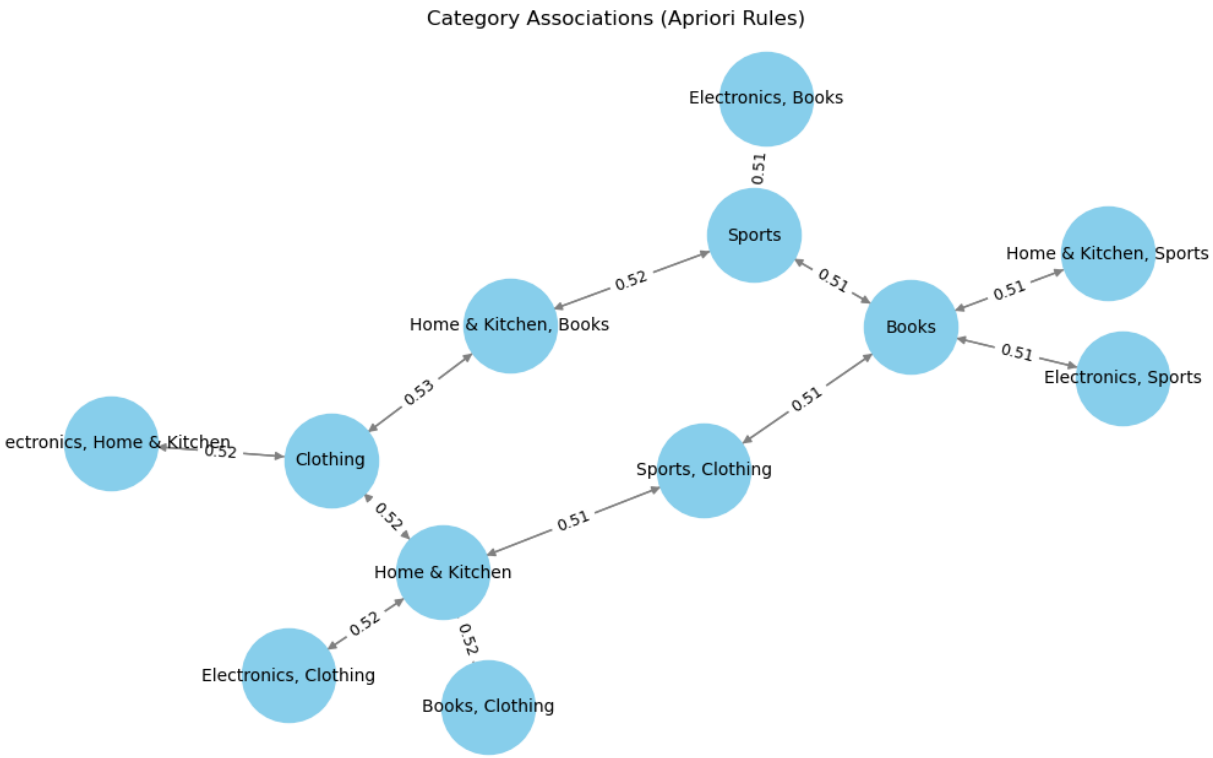

Figure 26: Apriori 2-element Rules

3-element rules output:

['Electronics', 'Sports', 'Books'],
['Electronics', 'Home & Kitchen', 'Clothing']
['Home & Kitchen', 'Books', 'Clothing']
['Home & Kitchen', 'Sports', 'Books']
['Books', 'Sports', 'Clothing']
['Home & Kitchen', 'Sports', 'Clothing']

Conclusion

Thank you for the opportunity to present this work. Your time and attention in reviewing this report are greatly appreciated. It is hoped that the methodologies and insights provided here will contribute to an enhanced understanding of customer behavior and support the development of targeted strategies at BookedBy. Your consideration is highly valued, and any feedback you provide will be instrumental in further refining these approaches.