Customer Segmentation Report

Team Optiminders

1. Introduction

1.1 Background

Customer segmentation is a crucial task for e-commerce platforms to understand different customer behaviors and optimize marketing strategies. This project aims to segment customers based on their shopping patterns using clustering techniques.

1.2 Objectives

- To analyze customer behavior using the given dataset.
- To identify three customer segments: **Bargain Hunters, High Spenders, and Window Shoppers**.
- To apply clustering techniques and evaluate their performance.
- To visualize the identified clusters for better understanding.

1.3 Dataset Overview

The dataset consists of six features,

- **customer_id**: Unique identifier for each customer.
- **total_purchases**: Total number of purchases.
- **avg_cart_value**: Average value of items in the cart.
- **total_time_spent**: Total time spent on the platform (in minutes).
- **product_click**: Number of products viewed.
- **discount_count**: Number of times a discount was used.

2. Exploratory Data Analysis (EDA)

2.1 Data Cleaning

- Checked for **missing values** and handled them appropriately.

While checking the missing values, identified that there were NULL values for *total_purchases, avg_cart_value* and *product_click.* For those 20 entries, all 3 of these columns were empty. So no purpose of keeping those entries. Totally data set had 999 entries, so 20 will be a percentage of 0.02. Decided to drop those entries and handled missing values.

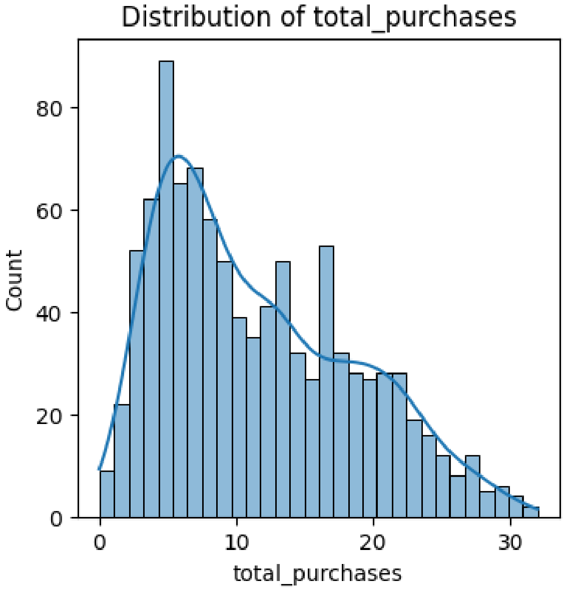
- Checked for **duplicate rows**.

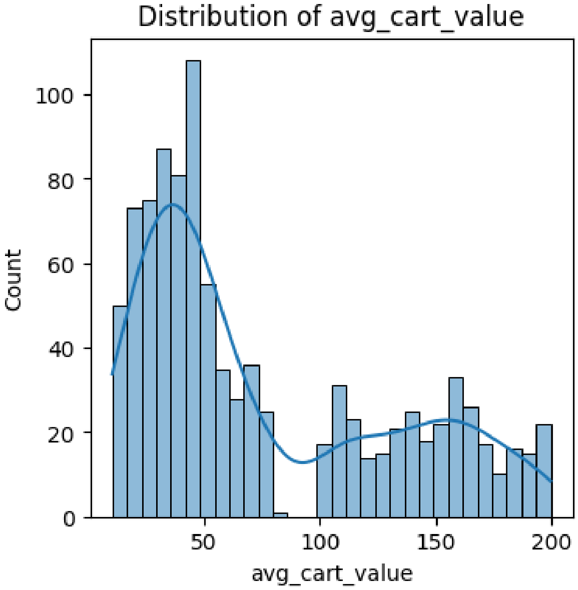
No duplicates were found by customer_ID.

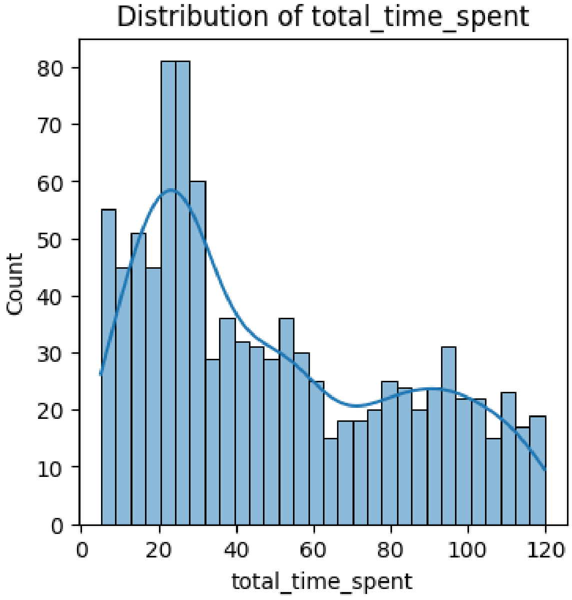
- Dropped **customer_id** as it is not required for clustering.

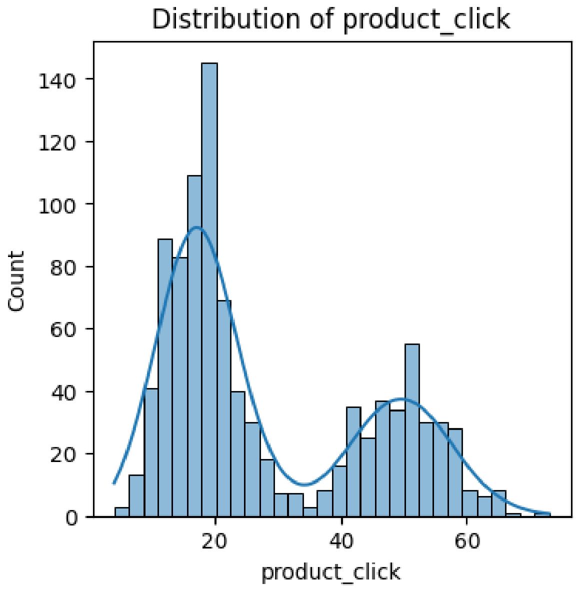
2.2 Understand the Feature Distribution

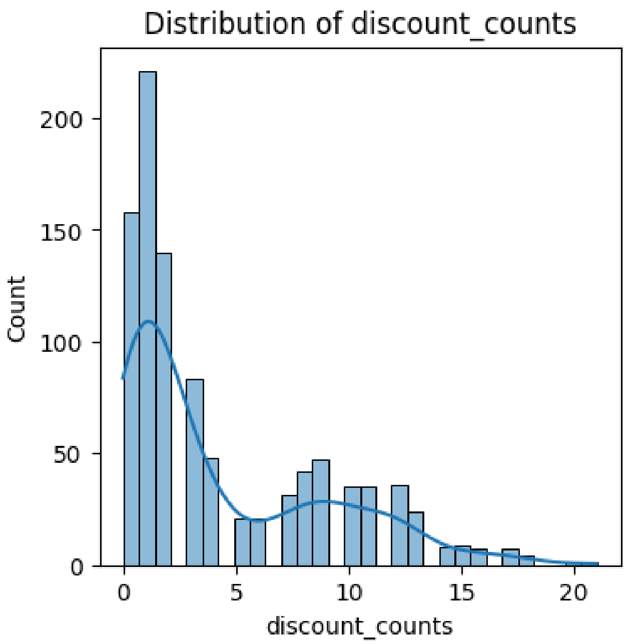
Visualizing the distributions of the features using the histograms. I used *matplotlib* and *seaborn* libraries for that.





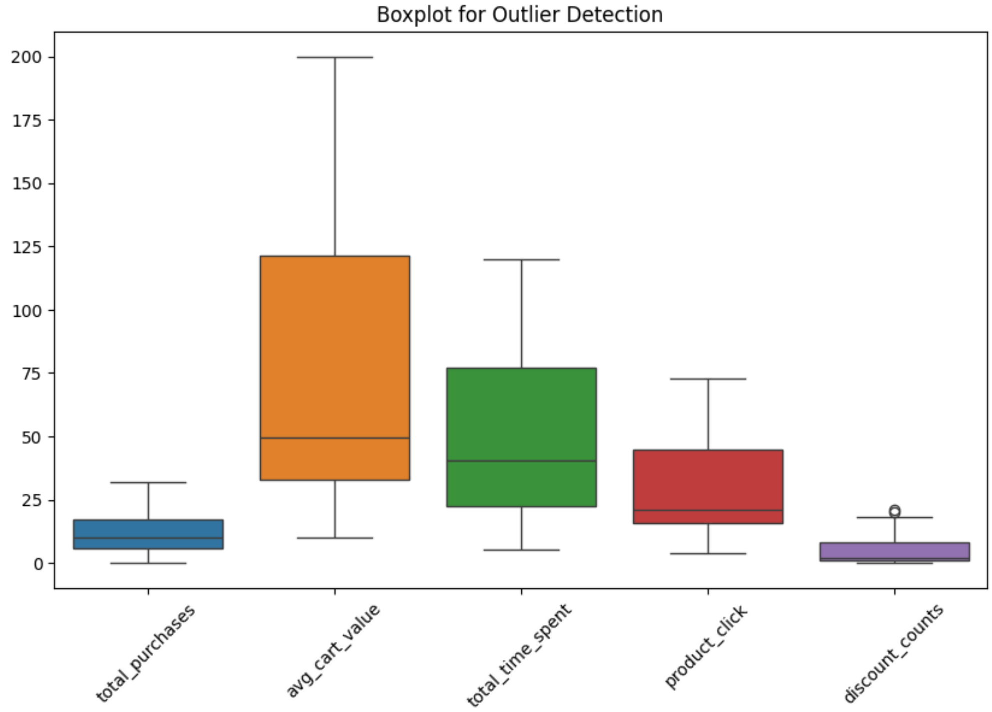






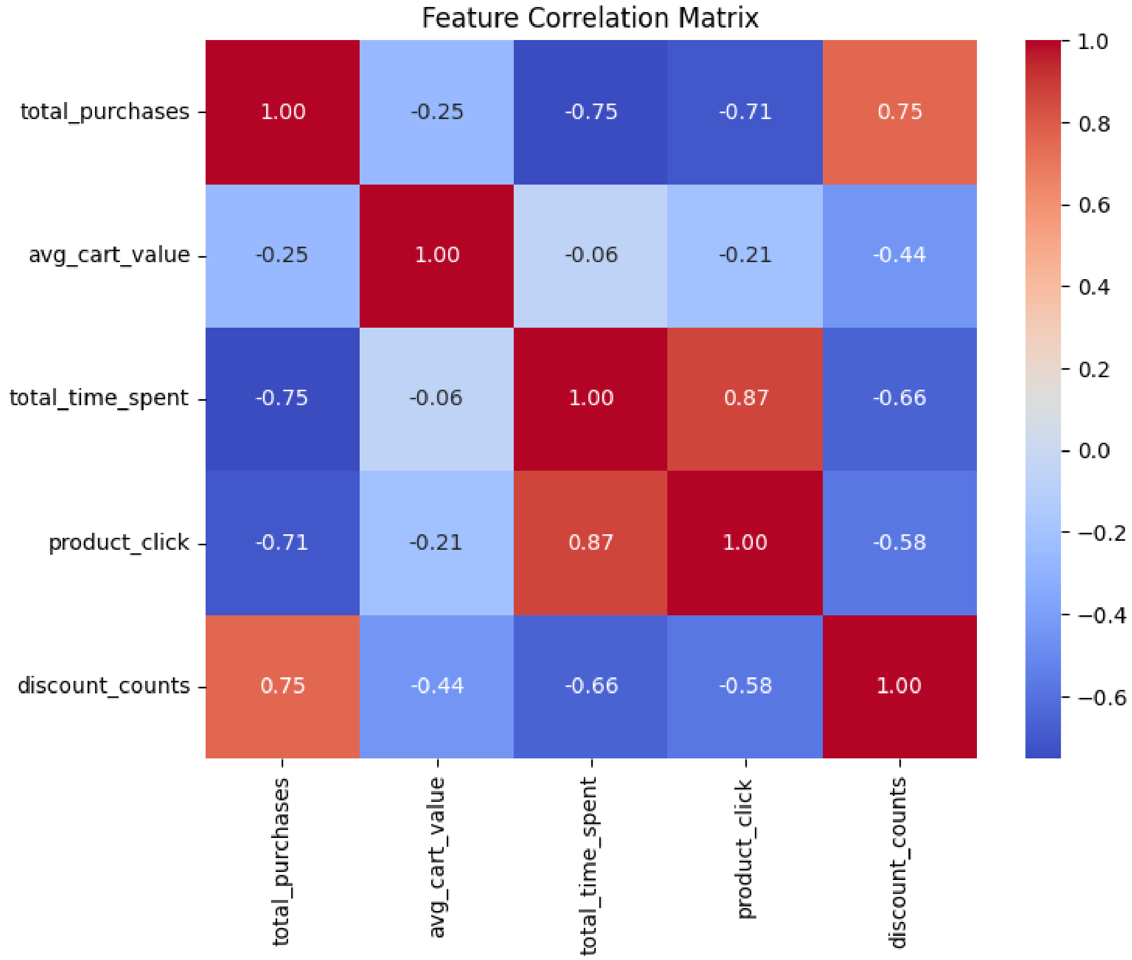
2.3 Outlier Detection

For the outlier detection, used box plot. There weren’t any.



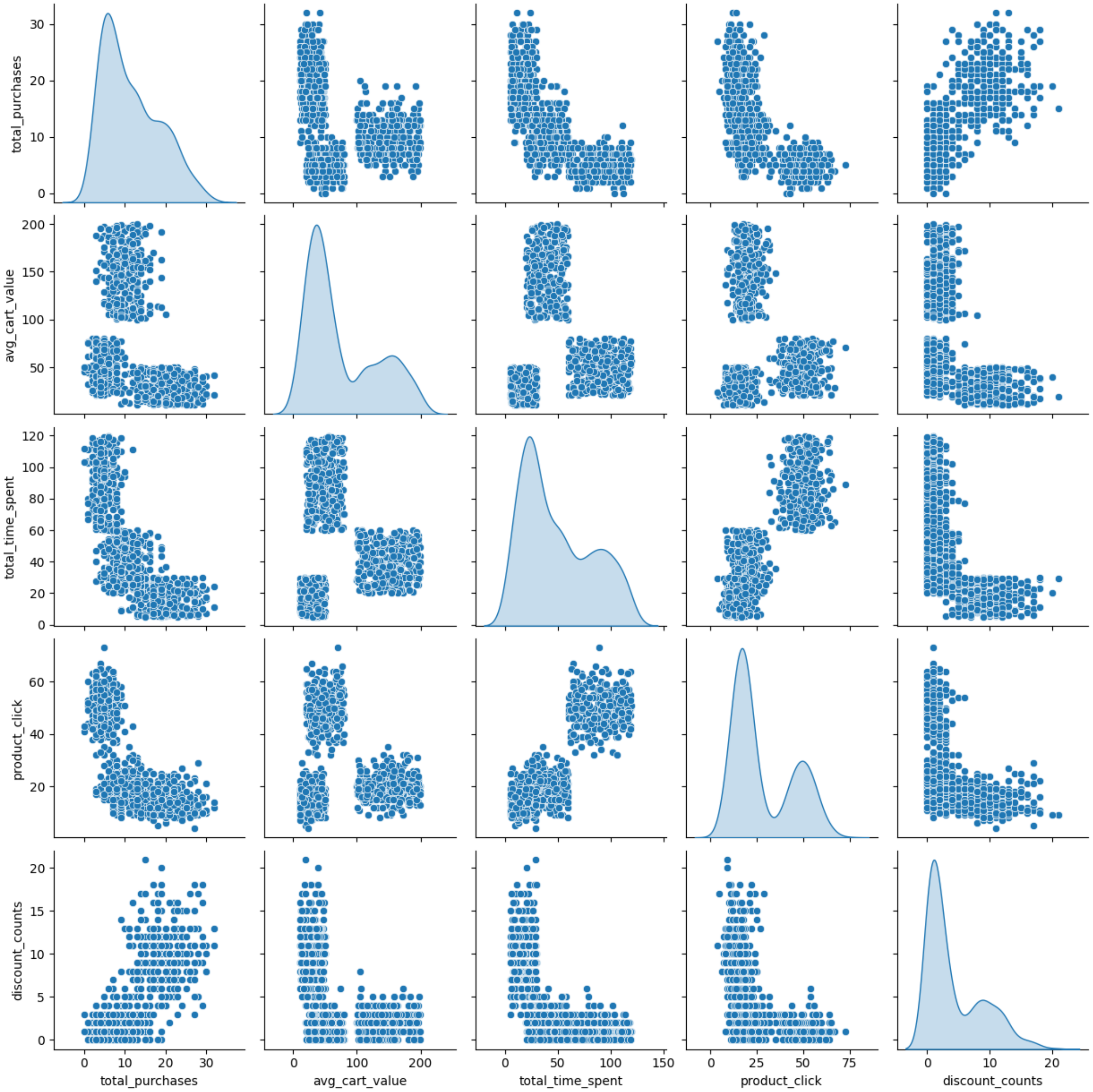
2.4 Checking for correlation

By plotting the correlation matrix, Identified that there’s a correlation between the product_click and total_time_spent. So I dropped the product_click column and went forward.



Though I dropped the column product_click, later when evaluating the clusters, it gave better values for Silhouette Score and Davies-Bouldin Index.

Pair plots



When looking at the pair plots, kit gave a sense of 3 separated clusters.

2.3 Feature Engineering

- Using the sklearn StandardScaler, scaled the data set before forming the clusters.

3. Model Selection

3.1 Clustering Techniques Considered

- **K-Means**

Which is suitable for spherical clusters. Used K Means from skLearn to form the clusters.

Evaluation Metric values:-

Silhouette Score: 0.6260176986578468
Davies-Bouldin Index: 0.5499863872687267
Inertia: 812.4713322917868

- **Agglomerative Clustering**

Which is a hierarchical approach for flexible segmentation. Used AgglomerativeClustering from skLearn to form the clusters.

Evaluation Metric values:-

Silhouette Score: 0.6260176986578468
Davies-Bouldin Index: 0.5499863872687267

- **DBSCAN**

Which is effective for non-convex clusters and noise handling. Used DBSCAN from skLearn to form the clusters.

Evaluation Metric values:-

Silhouette Score: 0.6260176986578468
Davies-Bouldin Index: 0.5499863872687267

From all three clustering methods, same metric values were given. Main reason behind that was the data had clear, well- separated clusters making different algorithms converge to similar results. Since we already know there are 3 clusters, K-Means and Agglomerative may produce similar groupings.

- The above cluster methods were chose because the number of clusters were already given.

4. Model Evaluation

4.1 Metrics Used

- **Silhouette Score**: Measures the quality of clustering.
- **Davies-Bouldin Index**: Evaluates cluster compactness and separation.
- **Inertia (for K-Means)**: Measures within-cluster variance.

5. Identifying Clusters

- Mapped cluster labels to the predefined segments: - **Bargain Hunters, High Spenders, and Window Shoppers**.

After taking the centroid values and comparing.

Index	total_purch	avg_cart_val	total_time_spent	product_click	discount_counts
0	-0.199010	1.305770	-0.277476	-0.512192	-0.518013
1	-0.956571	-0.480165	1.243180	1.318663	-0.726757
2	1.160839	-0.818654	-0.974183	-0.816132	1.247639

Interpret the Cluster Centers

- **Cluster 0** had,
    - **High total_purchases**
    - **Low avg_cart_value**
    - **High discount_count**

Labeled the cluster as **Bargain Hunters**.

- **Cluster 2** had,
    - **Moderate total_purchases**
    - **High avg_cart_value**
    - **Low discount_count**

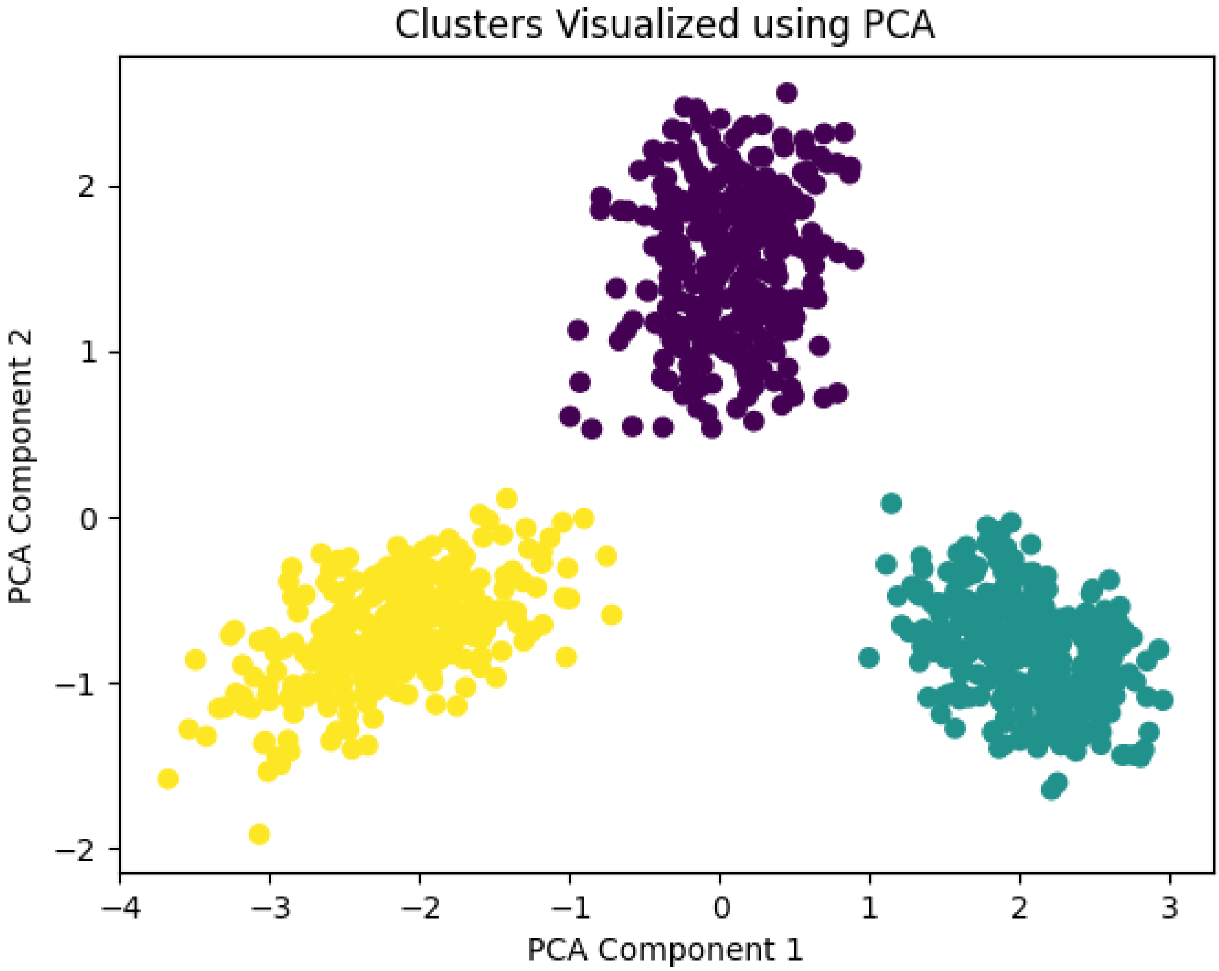
Labeled the cluster as **High Spenders**.

- **Cluster 3** had,
    - **Low total_purchases**
    - **Moderate avg_cart_value**
    - **High product_click**
    - **Low discount_count**

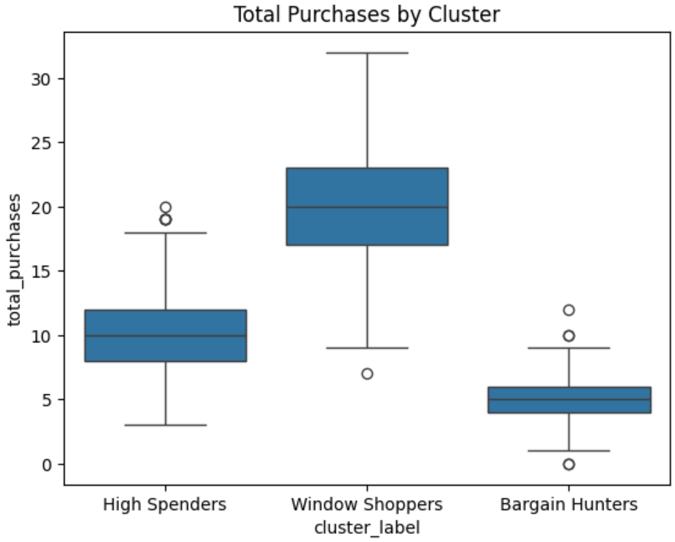
Labeled the cluster as **Window Shoppers**.

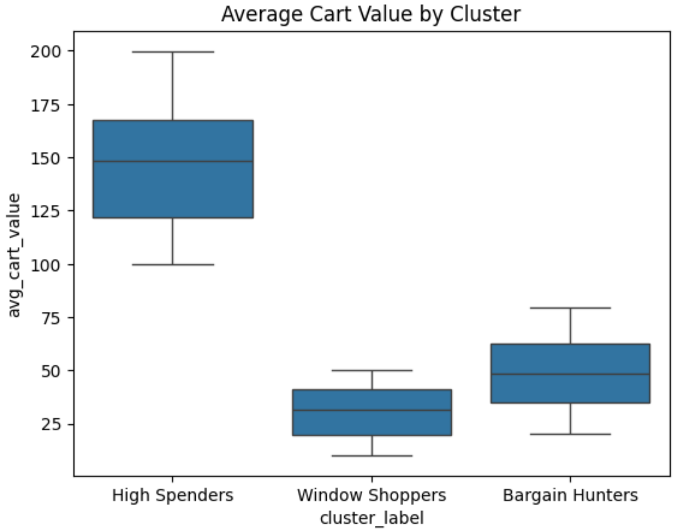
6. Visualization

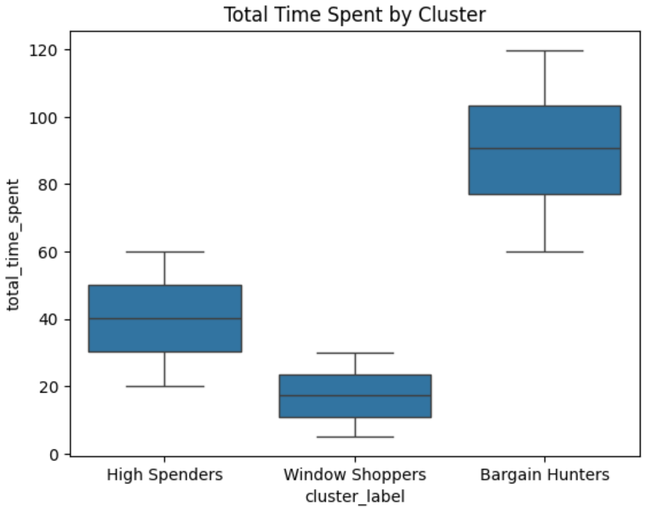
Used **PCA** for dimensionality reduction and plotted clusters.
Created scatter plots with different cluster colors.

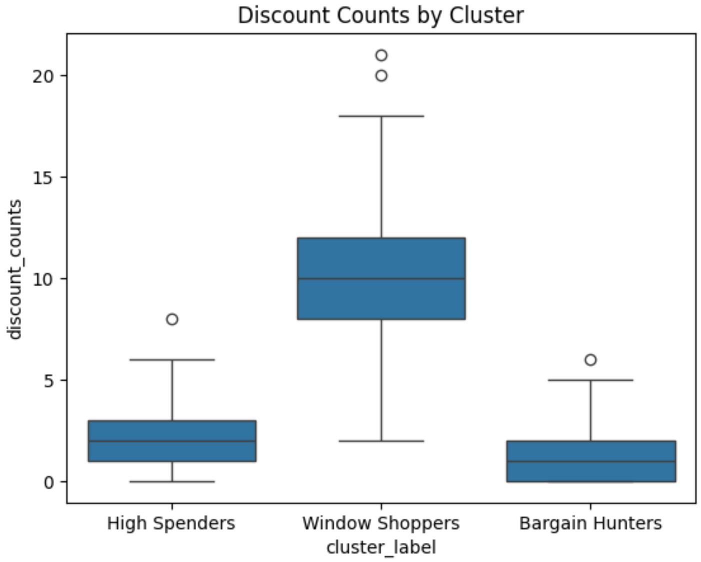


Additionally used box plots to see the clusters, across the features we used.


Total Purchases by Cluster


Average Cart Value by Cluster


Total Time Spent by Cluster



Finally divided the customers into the clusters we have figured out.