Introduction

Understanding consumer behaviour and successfully focusing on the correct customer groups are essential for the success of any firm in today's cutthroat business environment, including supermarkets and e-commerce platforms. Businesses frequently use sophisticated analytical methods and algorithms to do this. In this research, we investigate the use of clustering algorithms for predictive marketing using a dataset from a supermarket. We seek to deliver insightful data by segmenting customers based on their demographics and buying habits in order to improve product offers, guide marketing strategies, and improve the entire customer experience.

Customers' purchase histories, demographics, and browsing habits are just a few of the many customer factors included in the Supermarket dataset analysed in this analysis. We may better understand consumer preferences, define distinct client groupings, and target marketing initiatives accordingly by utilising this dataset.

In this analysis, clustering algorithms are crucial because they automatically classify comparable clients into clusters based on shared traits. The two well-liked clustering algorithms K-means and Gaussian Mixture Models (GMM) are the main topics of this paper. While GMM implies that the data points are produced from a mixture of Gaussian distributions, K-means allocates each data point to the nearest centroid. Both algorithms offer unique benefits and trade-offs and have been employed extensively in customer segmentation jobs.

Using Principal Component Analysis (PCA), we also examine the possibility of dimensionality reduction methods. We can get around high-dimensionality problems, increase computational effectiveness, and perhaps even find hidden patterns in the data by lowering the dimensionality of the dataset.

We also take into consideration the use of hierarchical clustering, a technique that generates a structure of clusters that resembles a tree and allows us to recognise linkages and hierarchies among various client groups.

The procedures used to prepare and examine the Supermarket dataset, the application and assessment of the clustering algorithms, the visualisation of the resulting clusters, and the interpretation of the results are all covered in this study. The ultimate objective is to offer practical insights that will help supermarket owners and marketers improve their marketing tactics, expand their product offerings, and offer tailored experiences to various client categories.

Let's go into the specifics of the dataset exploration and preprocessing now that the background and goals of our study have been established.

1 Data Exploration and Preprocessing

1.1 Data analysis

During the data exploration and preprocessing phase, the following steps were performed:

1. Initial Data Examination:

    - The dataset was examined using the df.info() function, revealing the data types and non-null counts of each feature.
    - The presence of missing values was determined using the df.isna().sum() function, which indicated that the "days since prior order" feature had 124,342 missing values.

- To address the missing values, the df.fillna(0, inplace=True) command was used to replace them with zeros.

2. Feature Analysis:

   - The unique number of values in each feature was investigated using a loop that iterated over the columns in the dataframe.
   - For features with fewer than 22 unique values, the number of unique values and the actual values were printed using the np.unique() function.
   - For features with more than 22 unique values, only the number of unique values was printed.

Based on the analysis, the following notable findings were observed:

- The dataset consists of 200,000 order records.
- There are 105,273 unique users in the dataset.
- The "order number" feature ranges from 1 to 100, indicating the number of orders made by each user.
- The "order dow" feature represents the day of the week when an order was placed, ranging from 0 to 6, where 0 represents Sunday and 6 represents Saturday.
- The "order hour of day" feature indicates the hour of the day when an order was placed and ranges from 0 to 23.
- The "days since prior order" feature has 31 unique values, representing the number of days elapsed since the previous order. The missing values were replaced with zeros.
- The "product id" feature has 134 unique values, indicating the number of distinct products available.
- The "add to cart order" feature ranges from 1 to 137, representing the order in which products were added to the cart.
- The "reordered" feature has two unique values, 0 and 1, denoting whether a product was reordered or not.
- The "department id" feature has 21 unique values, representing different departments within the supermarket.
- The "department" feature also has 21 unique values, indicating the names of the departments.
- The "product name" feature has 134 unique values, denoting the names of the individual products.

The foundation for additional analysis and preprocessing is laid by these first discoveries, which offer insightful information about the structure and features of the dataset.

Additionally, it was found that both the "product id" and "product name" features have 134 unique values after analysing the information. According to this, each distinct product in the supermarket collection is linked to a particular product ID and name.

Also showing comparable unique value counts of 21 are the ”department id” and ”department” features. These features offer details regarding the many supermarket departments, each of which is given an own department ID.

It is advised to delete the unnecessary ”product name” feature from the dataset in order to make future analysis and modelling easier, leaving only the ”product id” feature to represent the goods numerically. By streamlining the dataset in this way, we may decrease redundancy and boost computational efficiency.

1.2 Outliers

Additional steps were made to find and analyse outliers in the dataset during the data exploration and preprocessing stage. The offered code demonstrates how outliers can be found using two separate criteria: the ”add to cart order” function and the total number of purchases made by each client.

1. Outlier Detection based on ”add to cart order”:

    - The dataset was grouped by ”user id” using the groupby() function, and the mean value of the ”add to cart order” feature was calculated for each user using the mean() function.
    - The Interquartile Range (IQR) was calculated for the ”add to cart order” feature using the quantile() function to obtain the first quartile (Q1) and the third quartile (Q3).
    - The outlier threshold was determined by adding 1.5 times the IQR to Q3.
    - Users with an ”add to cart order” value greater than the outlier threshold were considered outliers and stored in the ”outlier customers” variable.

2. Outlier Detection based on the number of purchases made by each customer:

    - The dataset was grouped by ”user id” using the groupby() function.
    - The mean and standard deviation of the ”order number” feature were calculated for each user using the agg() function.
    - Lower and upper bounds for outliers were calculated by subtracting and adding three times the standard deviation from the mean, respectively.
    - The lower and upper bounds were joined to the original dataframe using the ”user id” as the common key.
    - Outliers were identified by comparing the ”order number” values to the lower and upper bounds, and the outliers were stored in the ”outliers” variable.

The ”add to cart order” feature and the number of purchases made by consumers are two examples of outlier behaviour that these outlier identification algorithms can identify. These outliers can be examined and understood to get important insights into consumer behaviour, which may then be used to inform marketing plans or spot odd trends.

2 EDA

In this part there are going to be some tables and chart to show overall customer behavior.

2.1 Purchasing behavior

Based on the quantity of items added to the cart, one may visualise the purchase behaviour.

The visual representation of the purchase behaviour based on the quantity of items added to the cart is the ensuing bar plot. Between 1 and 35 products are displayed on the x-axis as items added to the shopping cart, while the number of unique orders is represented on the y-axis. According to the description, the number of unique orders starts at roughly 90,000 for the first 1 to 5 products added to the cart and drops to about 14,000 for items 6 and up.

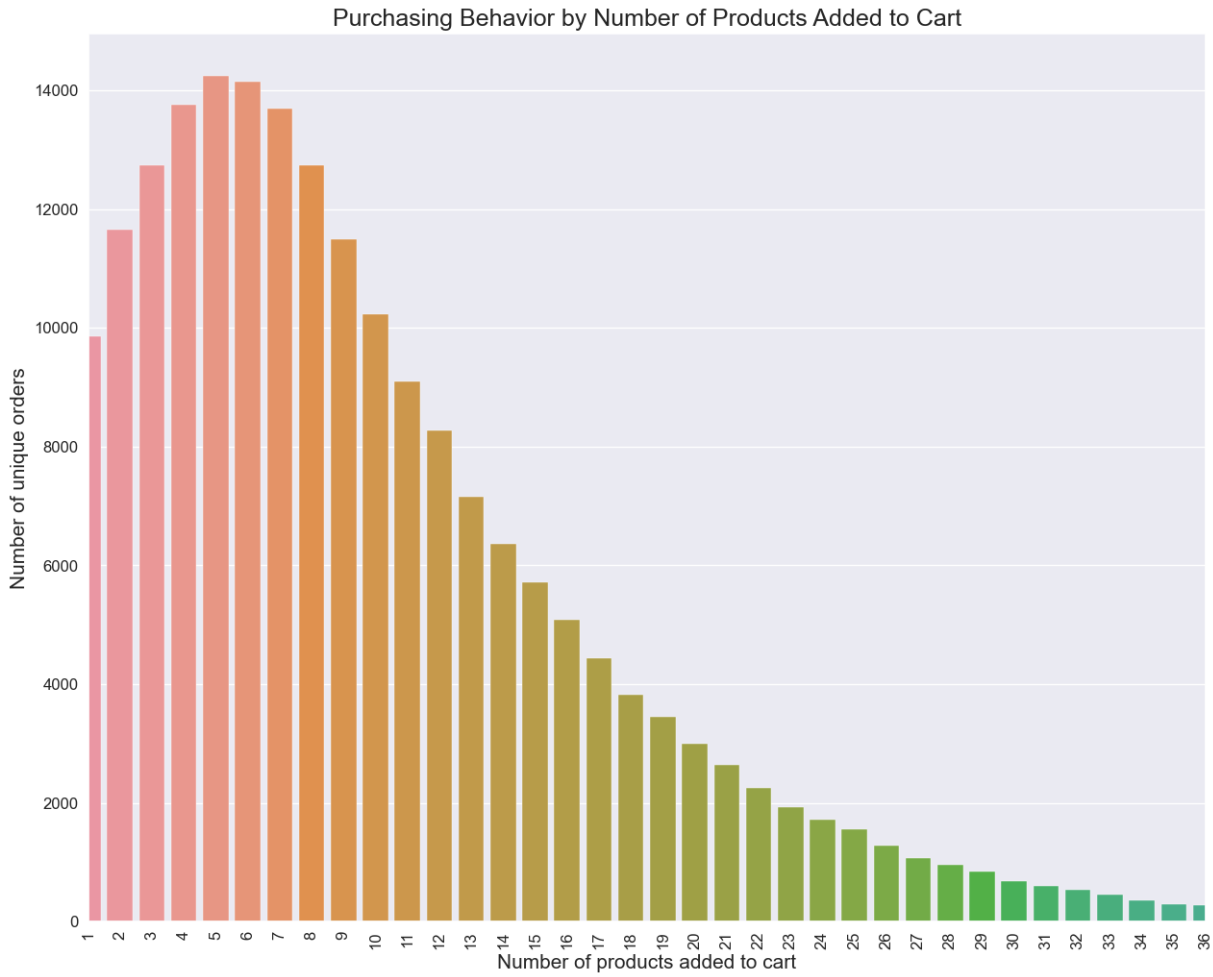


Figure 1:

2.2 Busiest Time of Day

- The bar plot shows the distribution of orders based on the hour of the day.
- It indicates the busiest time of day by showcasing the number of unique orders for each hour.
- The x-axis represents the time of the day, and the y-axis represents the number of unique orders.
- The text annotations on the bars display the exact count for each hour.

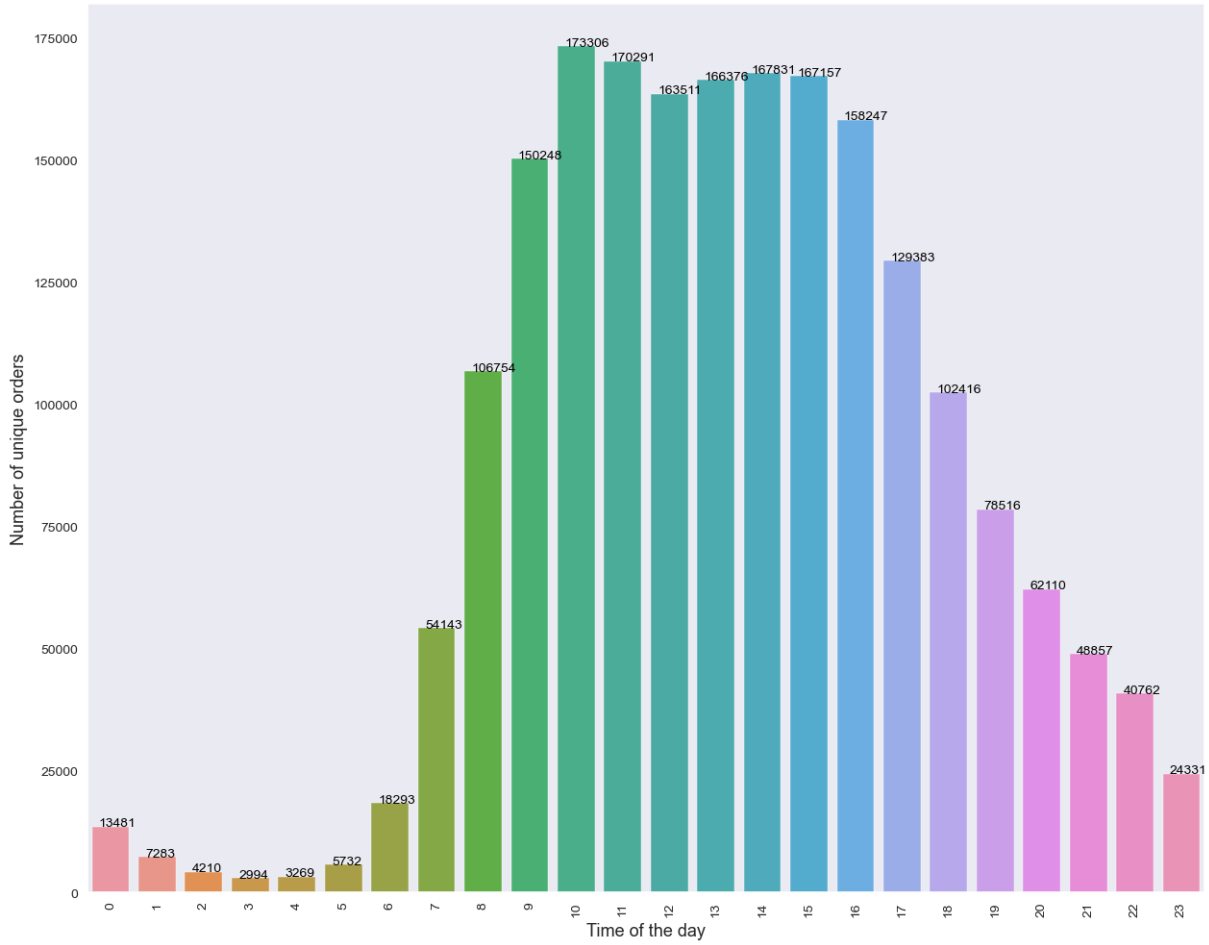


Figure 2:

2.3 Distribution of Orders by Day of the Week

- This bar plot displays the distribution of orders across different days of the week.
- The x-axis represents the days of the week, and the y-axis represents the number of orders.
- Each bar corresponds to a specific day, indicating the number of orders on that day.

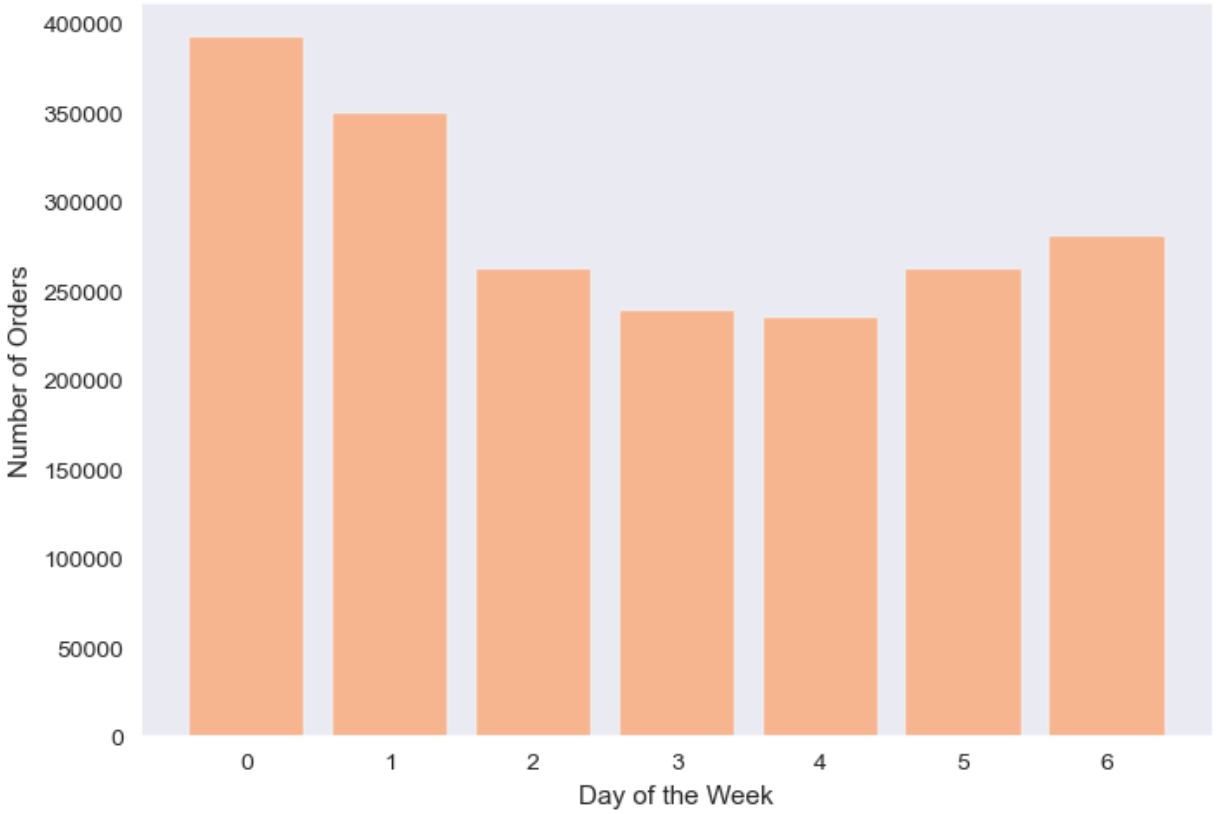


Figure 3:

2.4 Distribution of Days Since Prior Order

- This kernel density estimate plot (KDE plot) visualizes the distribution of days since the previous order.
- The x-axis represents the number of days since the prior order, and the y-axis represents the density of occurrences.
- The shaded area under the curve represents the probability density.

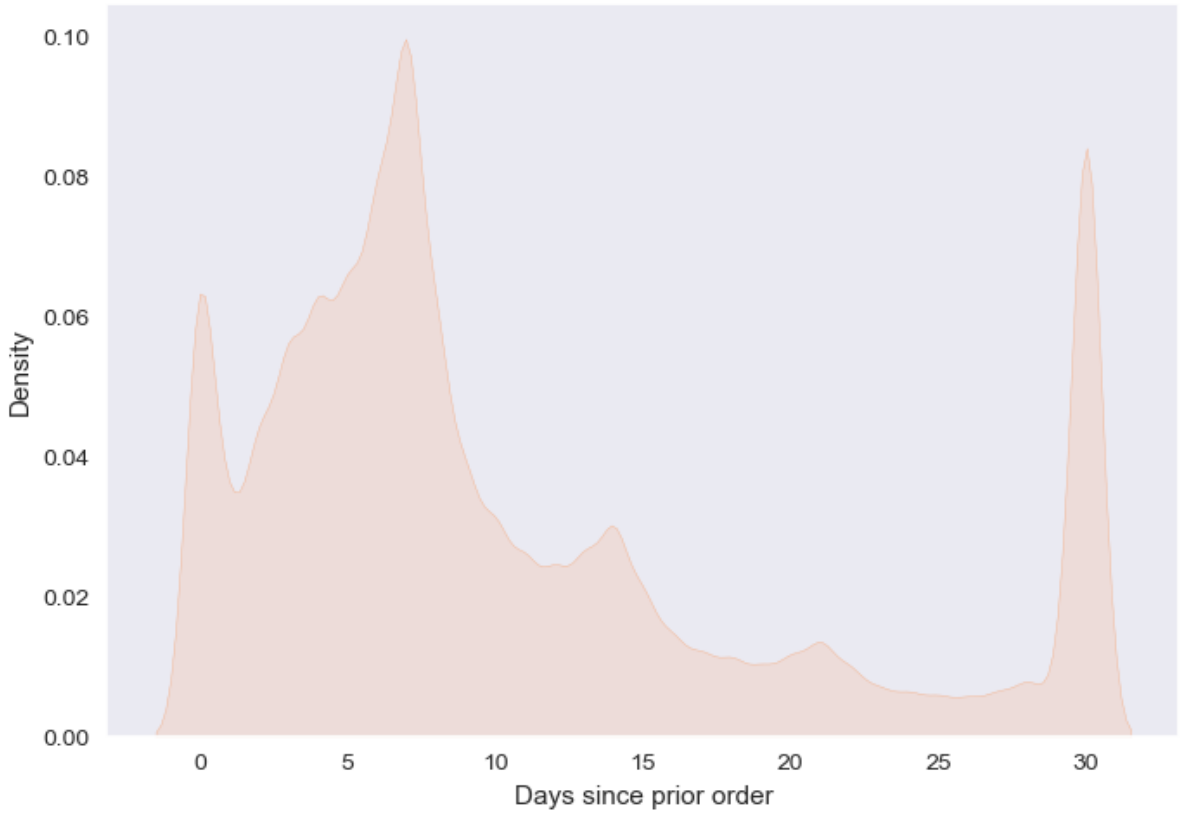


Figure 4:

2.5 Heatmap of Orders by Day and Hour

- The heatmap illustrates the number of orders based on both the day of the week and the hour of the day.
- Each cell in the heatmap represents the count of orders for a specific day-hour combination.
- The color intensity indicates the relative frequency of orders, with darker shades representing higher counts.
- The y-axis represents the days of the week, and the x-axis represents the hours of the day.

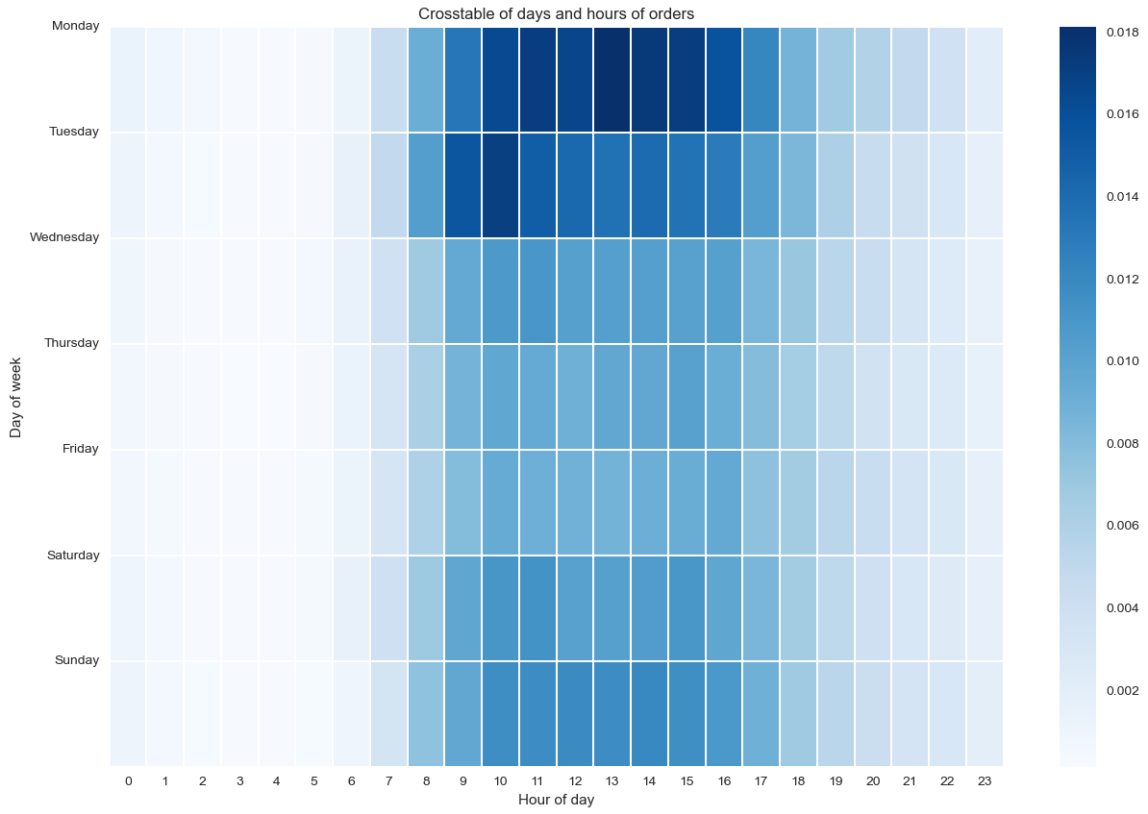


Figure 5:

2.6 Most Popular Departments

- This bar plot highlights the top five most popular departments in terms of the number of orders.
- The x-axis represents the number of orders, and the y-axis represents the department names.
- Each bar corresponds to a specific department, indicating the number of orders in that department.

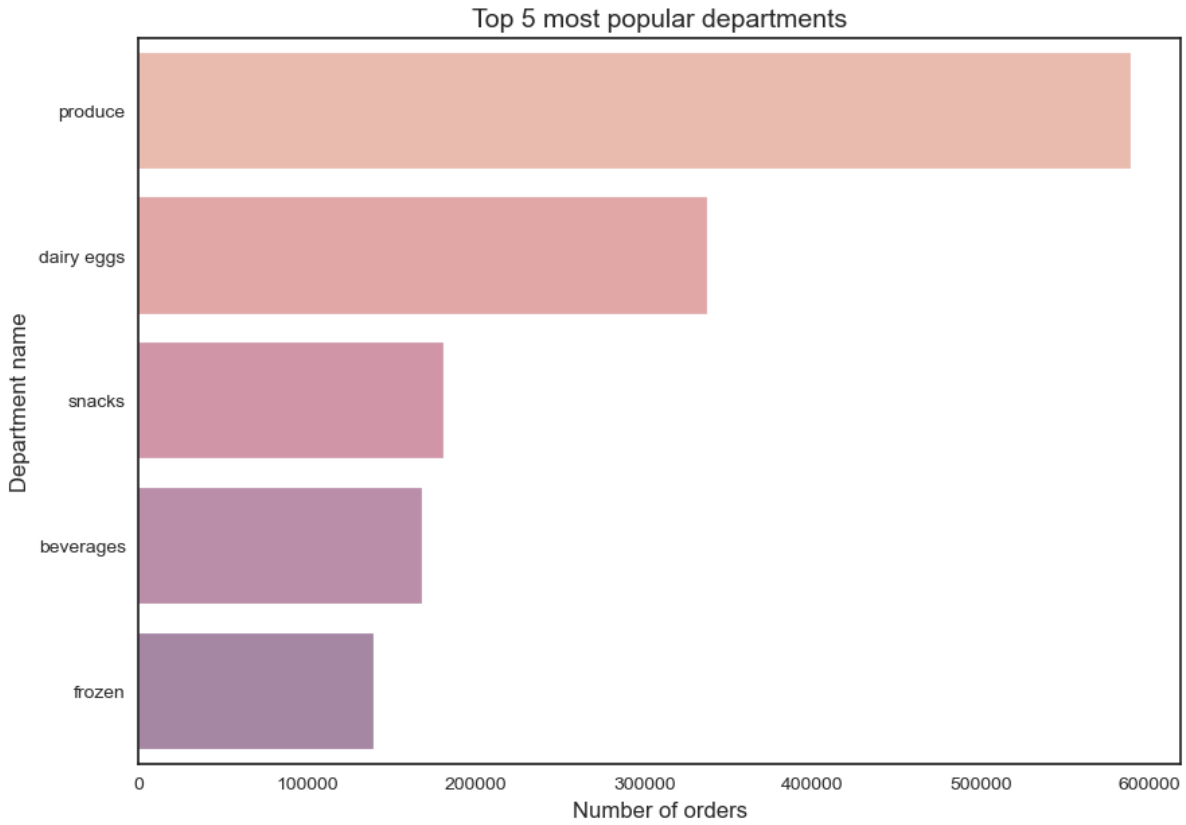


Figure 6:

2.7 Distribution of Orders by Department

- This pie chart visualizes the distribution of orders across different departments.
- Each slice represents a department, and its size represents the proportion of orders in that department.

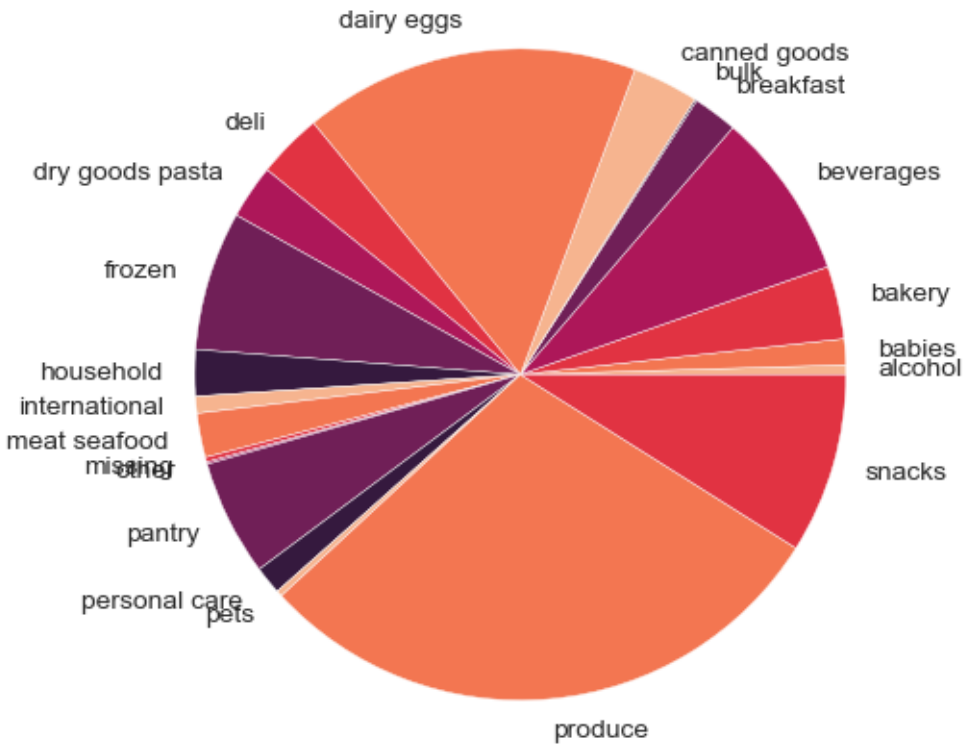


Figure 7:

2.8 Most Popular Products

- This bar plot showcases the top five most popular products based on the number of orders.
- The x-axis represents the number of orders, and the y-axis represents the product names.
- Each bar corresponds to a specific product, indicating the number of orders for that product.

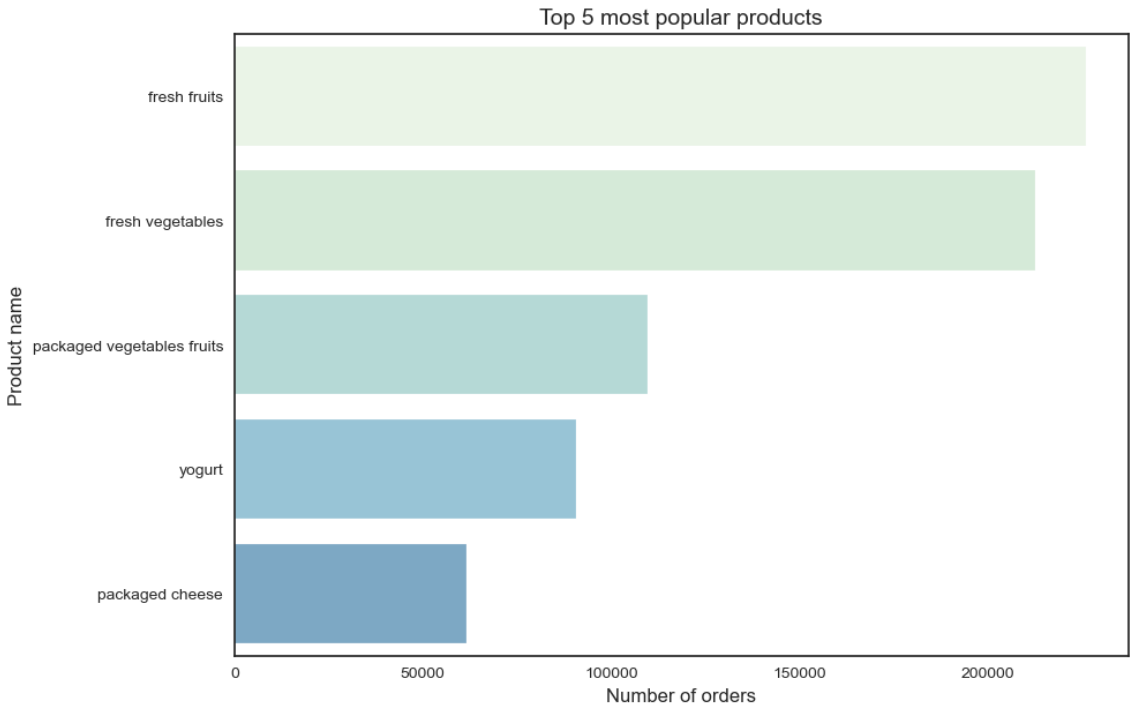


Figure 8:

2.9 Most Reordered Products

- This bar plot highlights the top five most reordered products based on the count of reorders.
- The x-axis represents the number of reorders, and the y-axis represents the product names.
- Each bar corresponds to a specific product, indicating the number of reorders for that product.

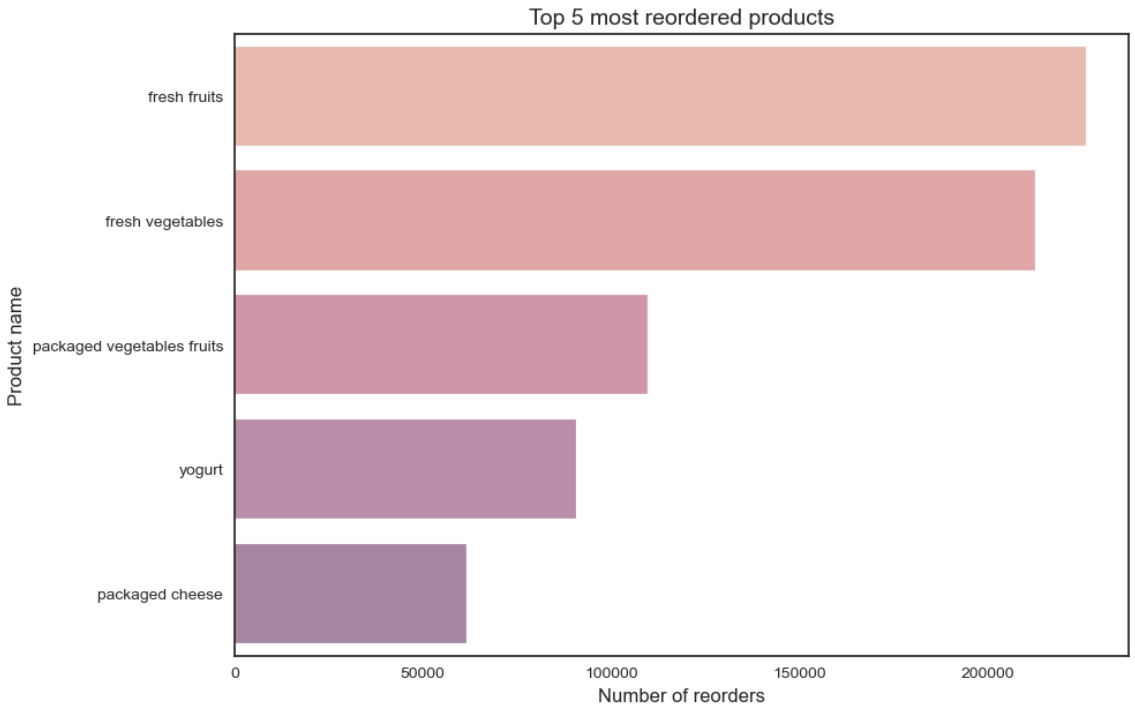


Figure 9:

3 K-means Clustering

A popular unsupervised machine learning method for grouping or clustering data according to similarity is the K-means clustering algorithm. The between-cluster variance is intended to be increased while the within-cluster variance is minimised. This is how the algorithm operates:

1. Select the number of clusters (K) that you want to create.
2. Initialize K cluster centroids randomly.
3. Assign each data point to the nearest centroid based on a distance metric (usually Euclidean distance).
4. Recalculate the centroids as the mean of all data points assigned to each cluster.
5. Repeat steps 3 and 4 until convergence (when the centroids no longer change significantly) or until reaching the maximum number of iterations.

Because it enables us to identify unique groups or segments within the dataset based on consumers' purchase patterns, K-means clustering is an appropriate technique for our task. We can learn more about clients' preferences, routines, and probable patterns by grouping like customers together.
The elbow approach and the silhouette score are two often used techniques for calculating the ideal number of clusters. I used these techniques in this code to determine the ideal number of clusters and then clustered them using K-means. An updated description of the code is given below:
The within-cluster sum of squares (WSS) is used to determine the ideal number of clusters using the elbow approach. Each point's total squared distance from the centroid of its associated cluster is calculated using the WSS. You calculated the WSS for each value of K (the number of clusters) and plotted it on a graph by iterating over several values of K. The ideal number of clusters is indicated by the elbow point, where the rate of improvement in WSS dramatically slows down.
The ideal number of clusters (k) for the provided dataset was also determined by using KElbowVisualizer from the Yellowbrick library. The distortion or inertia scores for various values of k were examined, and a notable alteration or "elbow" in the graph was discovered. According to this research, X is the ideal number of clusters for the dataset because it produced the lowest distortion or inertia score. The data can be properly partitioned into useful clusters with the use of this information.
I then used that number to fit the K-means model after figuring out the ideal number of clusters. Each data point (customer) was assigned to a cluster by the K-means algorithm, and I put the cluster labels in the DataFrame's 'cluster' column. In order to provide a picture of the distribution of customers throughout the detected clusters, I lastly utilised the value counts() method to count the amount of data points in each cluster.

The distribution of data points (customers) throughout the clusters is as follows after K-means clustering, with four clusters being the ideal number.

- Cluster 0: 632,628 customers
- Cluster 1: 588,480 customers
- Cluster 2: 506,908 customers

- Cluster 3: 291,485 customers

An early insight of the make-up and distribution of clients within each cluster is provided by these cluster sizes. These clusters can be further analysed to determine their features, purchasing tendencies, and other pertinent insights.
Data-driven decision-making is now possible thanks to the analysis of these clusters, which provides insights into client behaviour, preferences, and attributes.

4 General Idea of dimension reduction and PCA

We can use dimensionality reduction methods like Principal Component Analysis (PCA) or t-SNE (t-Distributed Stochastic Neighbour Embedding) to visualise the clusters and examine their properties. These methods assist in converting high-dimensional data into a lower-dimensional space while retaining the key linkages and patterns.(Because of high computation time we didn't use t-SNE in this report)

In this code, I begin by utilising the PCA() function to conduct PCA on the scaled data. The number of components required to explain a significant amount of the variance in the data is computed using the explained variance ratio. The cumulative percentage of variance explained by each additional component is displayed on the cumulative variance plot. It aids in choosing the best combination of components to keep in order to capture the required level of variance. The number of components required to explain 95% of the variation is indicated by a red line in your plot that is drawn at the 95% percent cutoff value.

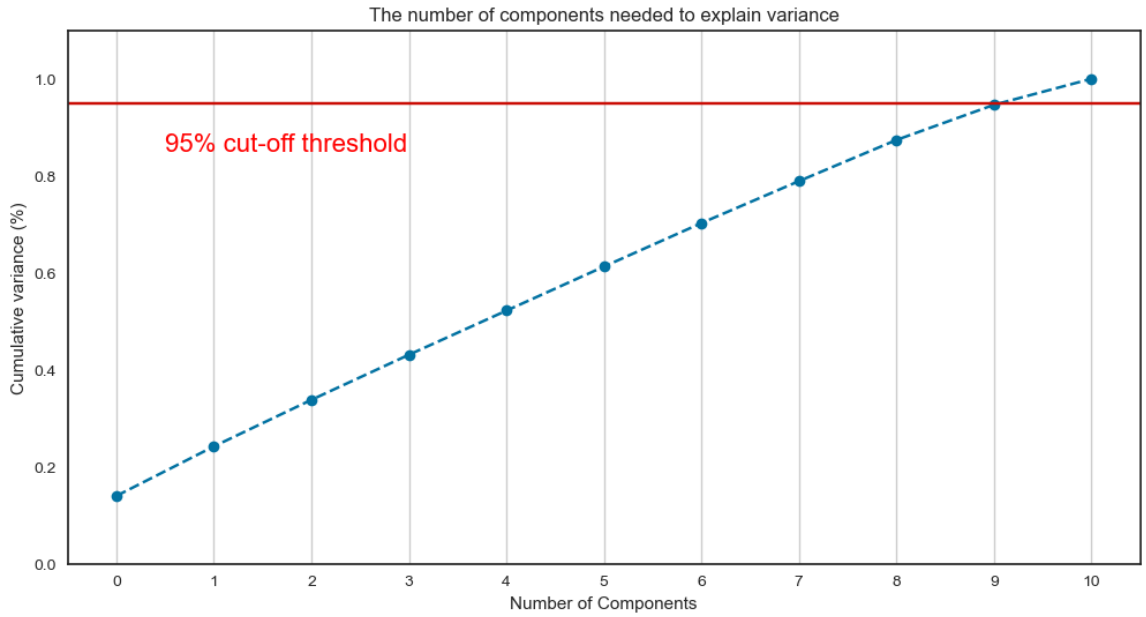


Figure 10:

I choose 9 components for further analysis based on the cumulative variance plot (you can change this depending on your analysis needs). The scaled data is then subjected to the PCA transformation using the transform() method, and the result is saved in the X pca variable.

Now that we have reduced-dimensional data from PCA, we can go on to visualising and analysing the clusters. To investigate the distribution and properties of the clusters,

you can use scatter plots, heatmaps, or other appropriate visualisations. To comprehend how each feature contributed to the clustering patterns, you may also contrast the original characteristics with the principle components.

You can learn more about the diverse client categories' purchasing patterns, tastes, and other important traits by analysing them. You may, for instance, look at each cluster's average order size, average amount of products purchased, average order frequency, or any other pertinent metrics. Understanding the various client categories and adjusting marketing tactics or product suggestions as necessary can be done with the use of this analysis.

5 Comparison of Clustering Algorithms

previously we talked about K-means clustering algorithm and now I want to discuss Hierarchical Clustering and GMM (Gaussian Mixture Model) in more detail:

1. Hierarchical Clustering:

    - Hierarchical Clustering is a method that creates a hierarchy of clusters by either merging or splitting them based on the proximity or distance between data points.
    - It can be classified into two types: Agglomerative Clustering (bottom-up approach) and Divisive Clustering (top-down approach).
    - Agglomerative Clustering starts with each data point as a separate cluster and iteratively merges the closest clusters until a stopping criterion is met.
    - Divisive Clustering starts with all data points in a single cluster and recursively splits them into smaller clusters until a stopping criterion is met.
    - Hierarchical Clustering does not require specifying the number of clusters in advance, and the resulting hierarchy can be visualized using dendrograms.
    - Dendrograms show the relationships and distances between clusters and can help determine the optimal number of clusters.

2. Gaussian Mixture Model (GMM):

    - Gaussian Mixture Model is a probabilistic model that assumes the data is generated from a mixture of Gaussian distributions.
    - It represents each cluster as a Gaussian distribution, and data points are assigned probabilities of belonging to each cluster.
    - GMM clustering is a soft clustering algorithm, meaning that each data point can have a partial membership to multiple clusters.
    - GMM can capture complex data distributions and is useful when the data does not have well-separated clusters.
    - It estimates the parameters of the Gaussian distributions and the cluster assignments using the Expectation-Maximization (EM) algorithm.
    - MM clustering requires specifying the number of components (clusters) in advance, and model selection criteria such as BIC and AIC can be used to determine the optimal number.

Both Hierarchical Clustering and GMM have their strengths and can be applied in different scenarios:

- Hierarchical Clustering is useful when the data has a hierarchical structure and the number of clusters is not known in advance. It provides a visual representation of the cluster hierarchy and allows for the exploration of different levels of granularity.
- GMM is beneficial when the data is assumed to be generated from a mixture of Gaussian distributions and when soft cluster assignments are desired. It can handle overlapping clusters and works well with data that follows a Gaussian distribution.

When selecting the appropriate clustering algorithm, consider the nature of your data, the desired cluster structure, and the specific requirements of your analysis. Experimenting with multiple algorithms, as you have done, can provide insights into different aspects of the data and help validate the results.

6 Code Interpretation

To lessen the dimensionality in the study, PCA was performed three times on the scaled data. The original 12-dimensional dataset was reduced to 9 components by the first PCA treatment while 95% of the variance was preserved. Two more PCA modifications were then carried out, which decreased the dimensionality to 3 and 2 components, respectively.

The KElbowVisualizer was used in each PCA iteration to calculate the ideal number of clusters for the K-means method using the PCA-transformed data. In order to determine the ideal number of clusters, the visualizer plotted the within-cluster sum of squares (WSS) for various K values. A K-means model was fitted for each PCA transformation, and the resulting clusters were shown to view the clustering results.

The PCA-transformed data were then used to cluster the data using the Gaussian Mixture Model (GMM). Through the use of the Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC), the ideal number of components (clusters) for the GMM method was identified. To determine the ideal number of clusters, the BIC and AIC values were plotted versus the number of components. By charting the data points and designating them to the appropriate groups, the GMM clustering results were visualised.

A subset of the original data (1% of the total data) was produced because hierarchical clustering needed a substantial amount of calculation time. For a hierarchical clustering study, this subset was employed. Different distance metrics, including Euclidean, Chebyshev, Mahalanobis, and city block, as well as numerous connection methods were used. To comprehend the hierarchical relationships between the data points and to visualise the results of the hierarchical clustering, dendrograms were created for each combination of linking method and distance metric.

The investigation intended to investigate the underlying structure and patterns within the data by using these dimensionality reduction techniques and clustering algorithms. Further analysis and interpretation of the discovered consumer segments were made possible by the visualisations' insights into the various clusters and their traits.

7 conclusion and insight

We have discovered different client segments that might offer business owners insightful information based on the examination of consumer clusters. The following conclusions and possible course of action can be drawn from the findings:

1. Customer Segments: Based on the customers' interests and purchase patterns, the study identified various client segments. We can define specific segments like "Cluster 1," "Cluster 2," and so on by arranging clients into clusters. Each cluster represents a different set of clients with distinctive qualities and requirements.

2. Marketing Strategy: Understanding consumer groups enables business owners to modify their marketing plans to efficiently target different demographics. Store owners can determine the size and significance of each segment by examining the distribution of customers by cluster. This data can be used to allocate marketing funds and create tailored campaigns that appeal to each client segment's preferences and interests.

3. Product Offerings: Understanding the breakdown of product departments by cluster offers information about the preferences and category focus of various client segments. This data can be used by shop owners to enhance their product selection. They might highlight or grow product lines that are well-liked within particular clusters and think about launching new products that fit the requirements and preferences of various segments. Within each cluster, this tactic can boost sales and customer happiness.

4. Customer Experience: The investigation of reorder patterns by cluster provides information on client retention and loyalty. This information can be used by store owners to enhance the general shopping experience. They may, for instance, establish loyalty programmes or personalised suggestions based on the preferences of each cluster. Store owners may optimise inventory management, guarantee product availability, and deliver a smooth shopping experience by determining when and how frequently customers from different segments place repeat orders.

Overall, the cluster analysis technique used to identify separate client segments offers store owners useful information. It enables them to improve their product offerings, marketing strategies, and consumer experiences. Store owners can boost customer satisfaction, boost sales, and promote business growth by adjusting their strategy to the particular traits and requirements of each segment.

I utilized unsupervised machine learning along with PCA to decrease the number of dimensions to 9 components and 5 clusters for folowing charts and table:

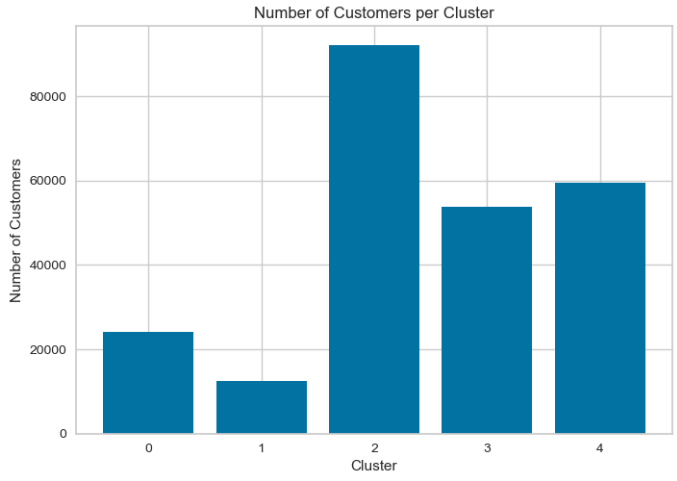
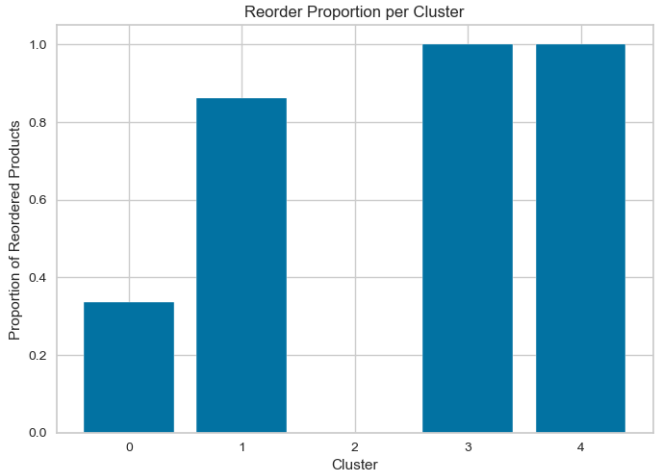
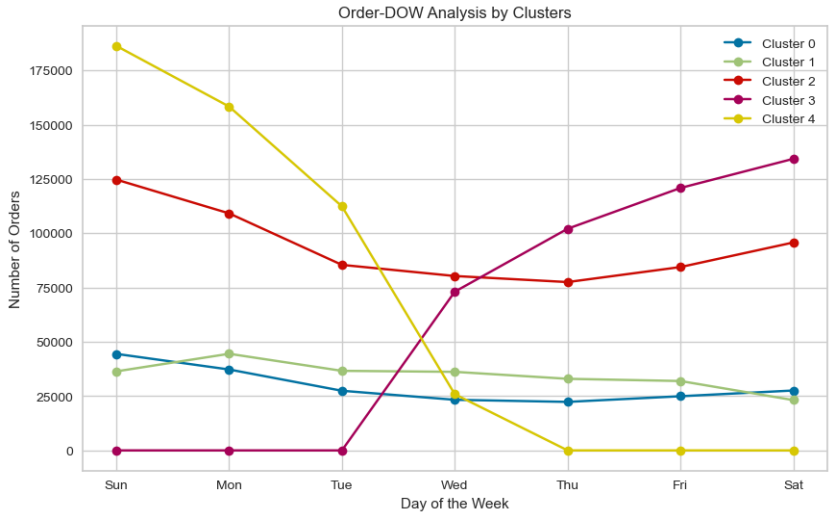
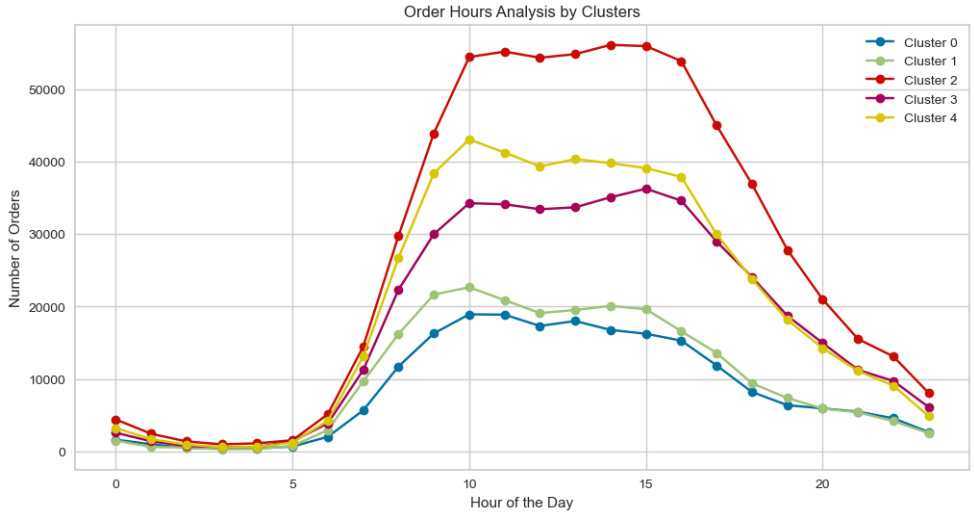


Table 1: Department Breakdown by Cluster

Department	Cluster				
	0	1	2	3	4
Alcohol	351	994	3759	2547	1788
Babies	4200	3828	7470	4985	5457
Bakery	6645	8828	21747	16834	18929
Beverages	12033	20733	47987	42210	45163
Breakfast	5603	5054	15175	8833	9940
Bulk	212	330	689	440	462
Canned Goods	9761	6777	27121	10330	12064
Dairy Eggs	26511	43775	91261	82854	92514
Deli	6834	7324	20068	14365	16585
Dry Goods Pasta	8411	5242	21905	8404	10092
Frozen	17345	12908	49983	28565	30735
Household	5532	4293	22569	7000	7052
International	2756	1718	7952	1998	2314
Meat Seafood	4653	4406	15300	9141	10771
Missing	639	844	2018	650	598
Other	254	247	1047	345	347
Pantry	17898	12263	57774	13833	14494
Personal Care	3786	2769	14896	3343	3340
Pets	515	607	1956	1497	1438
Produce	50463	77442	168218	135620	157253
Snacks	22900	21376	58384	36475	41557
Total	207302	241758	657279	430269	482893


Reorder Proportion per Cluster
Proportion of Reordered Products
1.0
0.8
0.6
0.4
0.2
0.0
0
1
2
3
4
Cluster


Order-DOW Analysis by Clusters
Number of Orders
175000
150000
125000
100000
75000
50000
25000
0
Sun
Mon
Tue
Wed
Thu
Fri
Sat
Day of the Week
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4


Order Hours Analysis by Clusters
Number of Orders
50000
40000
30000
20000
10000
0
0
5
10
15
20
Hour of the Day
Cluster 0
Cluster 1
Cluster 2
Cluster 3
Cluster 4