Machine Learning II
Final Project

Patrícia Bezerra: 20221907
Rita Silva: 20221920
Vasco Capão: 20221906
Group 16

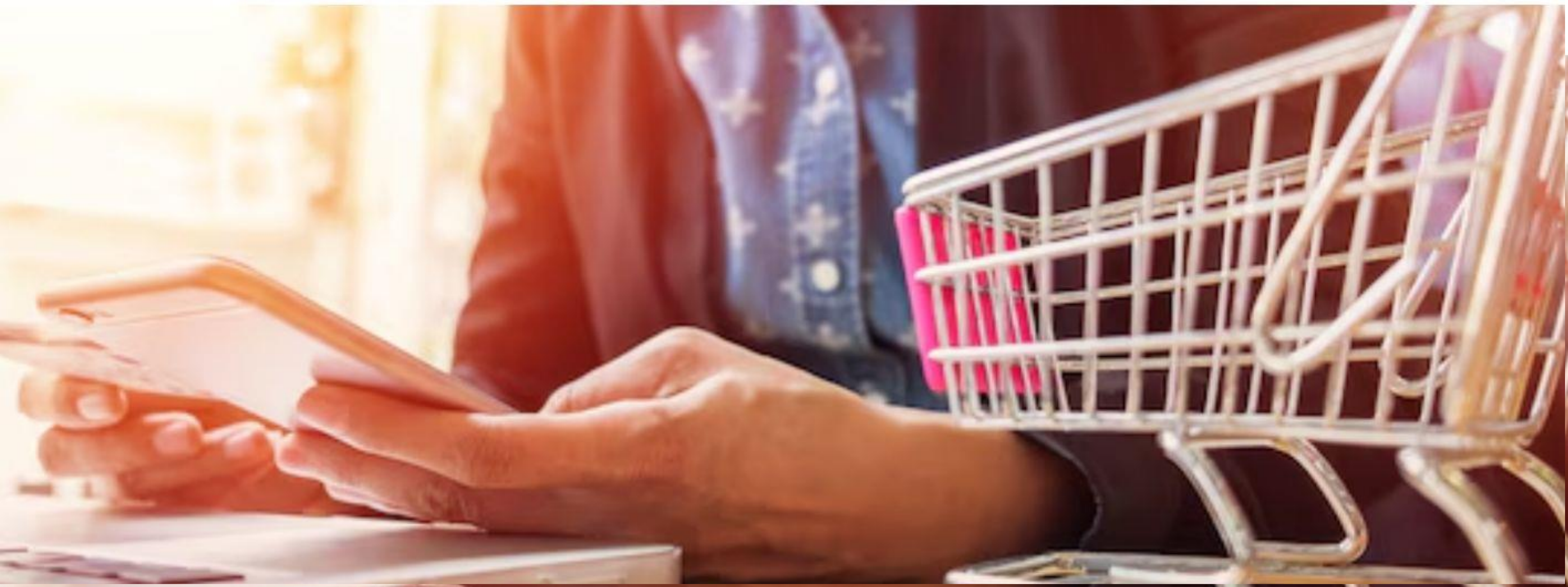


Table of contents

Executive Summary

In today's highly competitive market, businesses must understand their customers and tailor their marketing strategies to meet their specific needs. This project aims to perform customer segmentation using two datasets containing information on customer demographics, spending habits, purchasing behaviour, and historical transactions. By identifying distinct customer groups based on shared characteristics, businesses can develop targeted marketing strategies to maximize customer engagement and loyalty.

We began the project by importing the customer information from the CSV file and performing preprocessing and data cleaning. Additionally, we created new variables to improve interpretation. We then examined the correlation and distribution of the variables to understand relationships and potential multicollinearity issues, guiding our feature selection for clustering. We also visualized the geographical distribution of our customers using their latitude and longitude coordinates, identifying regional patterns in behaviour and spending.

Relating to the segmentation process, we applied several clustering algorithms such as K-Means, Hierarchical Clustering, MeanShift, and Self-Organizing Maps. Uniform Manifold Approximation Projection (UMAP) was the visualization method used to pick the best model.

A final clustering solution was reached, comprising 7 distinct groups. These segments were named based on their distinctive characteristics, such as "Diversifiers", "Occasional Discount Shoppers", "Large Families", "Vegetarians", "Pet Owners", "Young Shoppers", and "Premium Clients".

After that, a thorough analysis of every sector was conducted, taking demographics, product preferences, and spending trends into account. Targeted marketing tactics were created in response to these discoveries, taking into account the distinct tastes and habits of each group.

The company may increase consumer loyalty, personalize experiences, and improve marketing efforts by putting these data-driven ideas into practice. This will ultimately lead to corporate growth and success in the competitive market.

Exploratory Data Analysis

In the exploratory data analysis, we have done 4 phases, the data pre-processing phase, the data visualization phase, treated outliers and performed a correlation analysis.

Data Pre-Processing

For an initial phase in the preprocessing stage for the *customer_info* dataset, we started by checking if there were duplicate rows in the *customer_id* column. We verified that they did not exist and continued. Next, the data types and missing values were checked. Relating to this, only *customer_name*, *customer_gender* and *customer_birthdate* were objects, and all the other columns were classified as being a float. The variables, *kids_home*, *teens_home*, *number_complaints*, *distinct_stores_visited*, *typical_hour*, *lifetime_spend_vegetables*, *lifetime_spend_fish* and *loyalty_card_number* were the only ones who have missing values. The last variable mentioned was treated differently. We then decided to count the number of missing values that existed per row, and if there were more than 1, we chose to delete those rows from the dataset. To resolve the problem of having missing values, we have used the KNN Imputer. The KNN Imputer is a technique used to fill in missing values in datasets, utilizing the K-Nearest Neighbors (KNN) algorithm. For each missing value, the KNN Imputer finds the k nearest neighbors that have present values and calculates the mean of these neighbors to fill in the missing value. After executing this we verified that there were no missing values, except in the *loyalty_card_number*, as previously explained.

Some variables were changed and created, such as, the *customer_birthdate* variable was replaced with *customer_age.* The *customer_age* column will contain the customers' ages in years instead of their birth dates. The *customer_gender* variable was changed into a binary variable, where 1 represents male and 0 represents female. *loyalty_card_number* was also changed into a binary variable that indicates whether each customer has a loyalty card, 1 or not, 0.

Relating to the created variables, *total_lifetime_spend* was the first created variable and represents the total spent by that customer across all categories of *lifetime_spend*. We also created new columns to calculate the percentage that each spending category represents in relation to the total lifetime spending for each customer. Each new column is named starting with *percentage_* followed by the name of the corresponding spending category. *avg_spend_per_store* calculates the average spending per store for each customer by dividing the *total_lifetime_spend* by the number of distinct stores visited per customer, *distinct_stores_visited*. The variable *loyalty_years*, calculates the number of loyalty years for each customer by subtracting the year of their first transaction from 2024. Finally, a column named *degree* was created. In this column, we first created a function, *map_degree,* that checks for the presence of the titles Phd., Msc., and Bsc. If a name contains Phd., it returns 3, if it contains Msc., it returns 2, and if it contains Bsc., it returns 1. In the newly created column, these values will be inserted according to each customer's academic degree. We also created a function that removes the same academic degrees mentioned above, *remove_degree,* changing the variable *customer_name,* by applying this function.

We decided to perform a scatter mapbox, for us to be able to visualize customer distribution and academic degrees, Figure 1. The following image shows how customers are geographically spread and highlights variations in their academic qualifications using color differentiation. Each point represents a customer, with color encoding indicating the academic degree of the customer, dark purple indicating lower degrees and yellow, higher degrees.

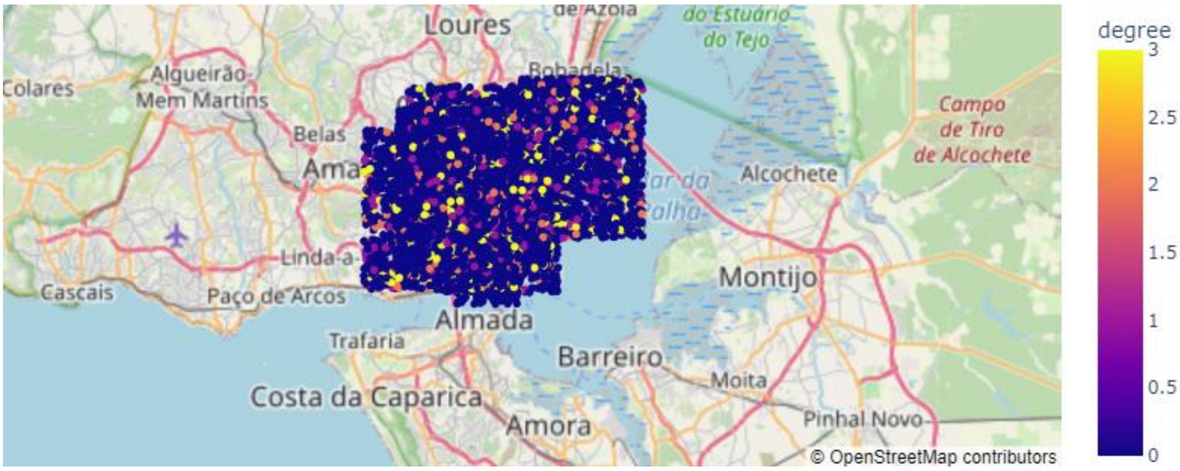


Figure 1 - Customers based on their location

Data Visualization

After doing the preprocessing phase in the dataset, we have performed some visualizations, to extract some conclusions.

We decided to perform histograms, within all the columns with numeric values, which shows the distribution of values in that column.

Starting to look at the number of male and other gender customers, we can verify that they are almost equal, 21681 male customers and 21488 other gender customers. Again, looking at the age of our customers, they are very uniform ages, so we almost always have the same number of customers per 5 year old range, around 3400 customers per range. From 20 to 24 we can see that there are fewer. Relating to the number of kids and teens at home, they are very identical, and we can see that most of our clients do not have many children, and the majority only have 1. The distribution of the number of complaints among customers reveals that the majority have between 0 and 2 complaints, with a significant concentration of 1 complaint. The histogram about the distribution of distinct stores visited by customers indicates that most customers visit only one store, with a sharp decline in frequency as the number of stores visited increases. About these 2 graphics explained previously, starting from the histogram about the distribution of *number_complaints*, we decided to delete the rows that have the number of complaints more than 2. We believe that being the number of complaints very low would not be very important for our project and could even ruin our clusters. Also, rows with number complaints that are not integer numbers will be deleted. It is not possible to have half of a complaint. Similar to this, when distinct_stores_visited is not an integer value, those rows will be deleted too. *typical_hour* also gives us some interesting insights. The histogram of this variable shows the frequency distribution of occurrences throughout the hours of the day, starting at 6am and ending at 12am. The hour with the highest frequency is 5pm, with just over 5000 occurrences. There is a gradual increase in frequency from 6am to the peak at 5pm, followed by a decline until midnight. The majority of customers have purchased between 0 and 500 unique products throughout their lifetime, with the highest frequency occurring in the range of 100-200 products.

Most customers buy around 10-30% of their products on promotion, with the peak frequency occurring around 20%. Few customers buy more than 60% of their products with promotional offers. Regarding the variable *year_first_transaction*, through the histogram, we can observe that it reveals an increase in first transactions from 2005 to 2012, with the peak of first transactions being in 2010, followed by a gradual decline until 2024. The distribution of *loyalty_card_number* suggests that the loyalty card program is popular among customers, but there's still a segment that hasn't enrolled. Most customers have spent between 0-10k in their lifetime, with a decreasing number of customers spending more. Regarding the average spending per store, the majority of customers exhibit a moderate spending behaviour per store, while a small fraction of customers have a considerably higher spending average. The histogram of the years of loyalty that customers have to the company shows that the company has been successful in retaining customers for a significant period, with most customers remaining loyal for 10-15 years. About the academic degree, we checked that the majority of our customers do not have one. We also did histograms for the lifetime spent in each category in absolute values and also in percentage.

Also, we decided to perform some radar visualizations. Firstly, we started to calculate the total amount spent within a supermarket chain across various product categories. To analyze the total expenditure per area in a supermarket, we used the radar graphic in figure 2. We can see that pet food is the product category where the customers spend the most money. On the other hand, non-alcoholic drinks are the category less bought by the customers. In the same way, we also created a radar visualization but this time for the total percentage spent by supermarket area, figure 3. Again, pet food is the category with the highest percentage expenditure, with around 20% of the total.

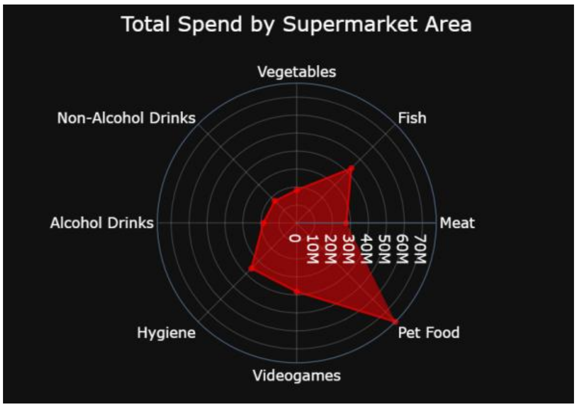


Figure 2

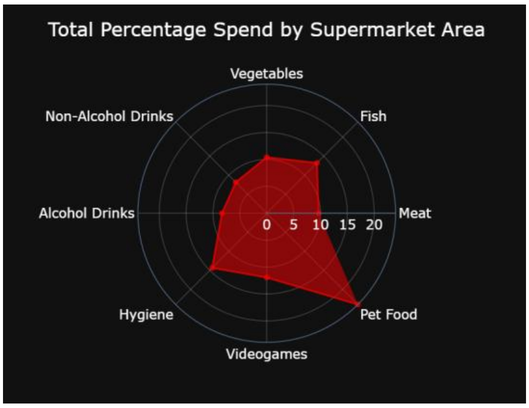


Figure 3

Outliers

Outliers are extreme values that deviate significantly from the rest of the data and can potentially skew analyses or modelling results. Because of this, they need to be identified and removed, if necessary. By making use of the histograms, explained previously, we defined some limits and those values that exceed those limits, will be considered outliers, and so will be deleted from the dataset. However, those outliers will be saved in a csv file, for a later analysis.

Correlation Analysis

After all these steps explained previously, we checked some correlations and plotted them using a pairplot and heatmap and with them, a comprehensive analysis of the relationships between spending categories in our customer data is provided.

Through the following heatmap, figure 4, we can check what are the variables with a strong positive relationship, and those that have negative correlations. We can say that fish and meat have a strong positive correlation, which means that people who spend a lot on fish tend to also spend a lot on meat. Another example is video Games and fish/meat. On the other hand, vegetables have a negative correlation with all the other categories, this suggests that vegetable consumption patterns are independent of other spending habits. The same occurs with pet food and the other categories.

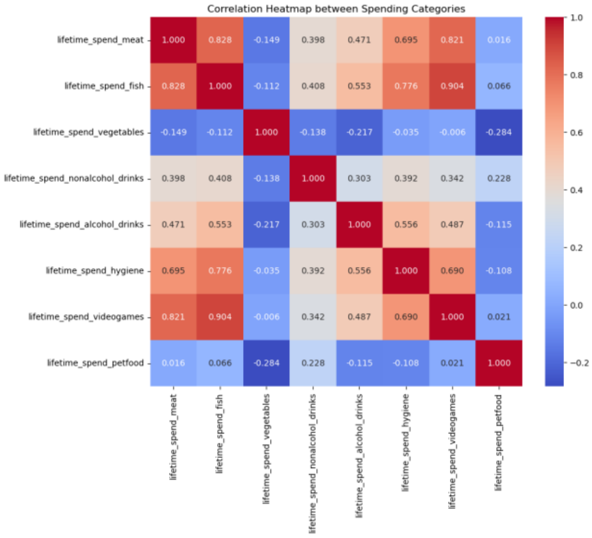


Figure 4 - Heatmap showing the correlations between the variables

Principal Component Analysis

Principal Component Analysis, also known as PCA, is a statistical method utilized in machine learning and data analysis for dimensionality reduction in a dataset while retaining as much variation as possible. The primary objective of PCA is to convert the original variables into a new set of uncorrelated variables, known as principal components, which capture the variance present in the original data.

To perform the PCA, we selected the dataset that included both the original and new variables. We then plotted the explained variance ratio for each principal component. To simplify the decision of how many principal components to use, we plotted the explained variance as a percentage of the cumulative variance, as shown in Figure 5. Based on this, we chose to work with the first 9 components, which accounted for approximately 82% of the variance in our data.

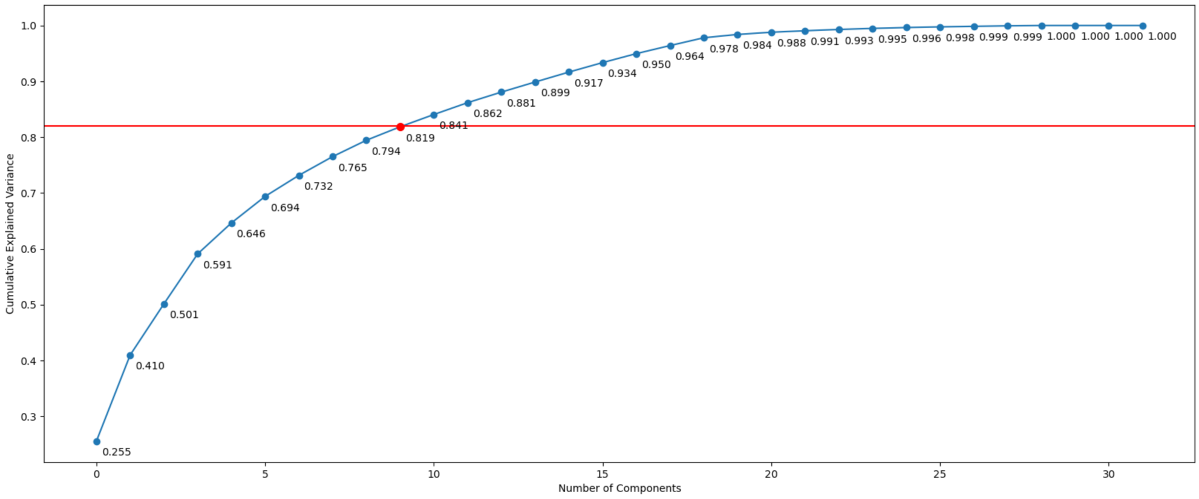


Figure 5 - Explained variance as a percentage of the cumulative variance

Customer Segmentation

The models we utilized to segment our consumer base, the steps that lead to our ultimate solution, and the significance of each of our clusters are all covered in this part.

Segmentation Approach

In this section of our project, we grouped the customers in different clusters according to their similarity. We used various algorithms like K-Means, Hierarchical Clustering, MeanShift, and Self-Organizing Maps to find the best way to group them. We also looked at the DBScan algorithm, but it did not give good results, and considering the delivery time, we focused on the better-performing algorithms.

For k-means and hierarchical clustering, we used two sets of data. One set had the original variables plus the variables we created early, while the other set only had the original information. Additionally, we tried three different ways of scaling the data for each set: ‘StandardScaler’, ‘RobustScaler’, and ‘MinMaxScaler’.

In addition, we used the K-Means algorithm on a dataset that resulted from Principal Component Analysis. However, we only used StandardScaler and RobustScaler because the clusters were becoming unsatisfactory.

With the MeanShift algorithm, we tried all the scaling methods on both datasets. Even though we could have tweaked the settings to improve results, we found that changing the parameters didn't make much difference. So, we moved on to fine-tune the other algorithms that were working better.

We only used the ‘StandardScaler’ with both datasets. However, just like with MeanShift, we chose to focus on adjusting the parameters of the other algorithms instead of spending more time on Self-Organizing maps.

To pick the best model, we used a visualization method called Uniform Manifold Approximation Projection (UMAP). It maintains the structure of the data while reducing its complexity. We also considered T-Distributed Stochastic Neighbor Embedding (*t*-SNE), which is another way to reduce complexity for visualization. However, as UMAP has some advantages over *t*-SNE, such as not needing pairwise similarities, running faster, and having better parameterization options, we only used it for the analysis. Furthermore, to achieve improved visualization results, we calculated the Silhouette Score to fine-tune the parameter 'n_neighbors'. We determined that the optimal value for this parameter was '500', as shown in Figure 6.

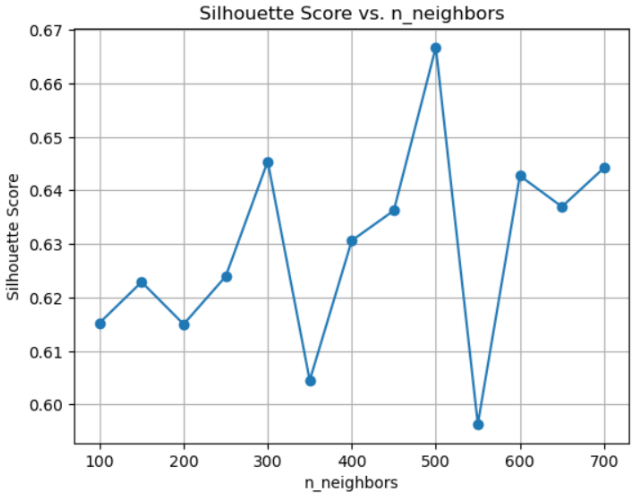


Figure 6 - Silhouette Score for the 'n_neighbors' parameter in UMAP

Hierarchical Clustering Algorithm

Hierarchical clustering is an unsupervised machine learning algorithm that groups similar objects into clusters. It creates a tree-like structure, called dendrogram, illustrating the relationships and hierarchy between clusters. Through the dendrogram, we can have an idea of the appropriate number of clusters. There are two main types: Agglomerative, merging the most similar clusters iteratively, and Divisive, recursively splitting clusters. Both rely on a distance metric to measure similarity and a linkage criterion to decide which clusters to merge or split. Common criteria include single, complete, average, and ward's linkage.

In our project, we chose to use the agglomerative approach. Based on what we learned in practical classes, we found that the ward's linkage method is the most effective, being the one we used.

Several different scenarios were analyzed, with distinct scalers and datasets. Of these analyzed cases, the one that showed significant promise when compared with the others was the Hierarchical Algorithm using the 'RobustScaler' and the dataset with both original and new variables. The dendrogram presented in Figure 7 suggests the implementation of 7 clusters.

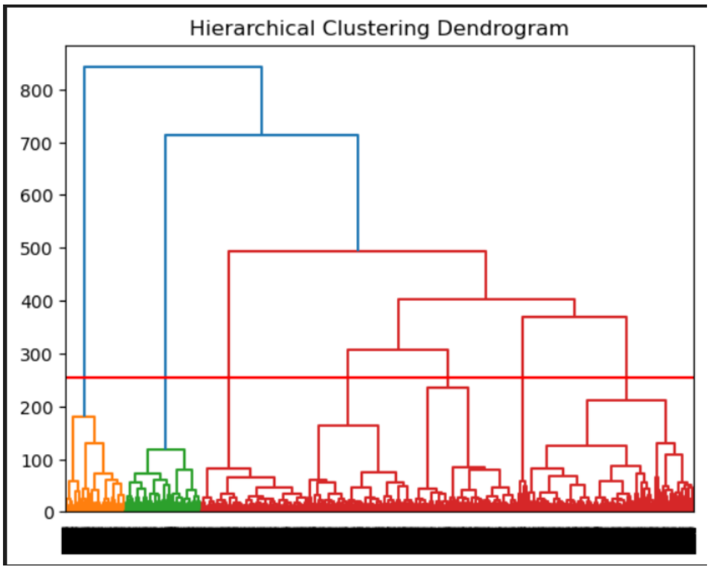


Figure 7 - Hierarchical Clustering Dendrogram

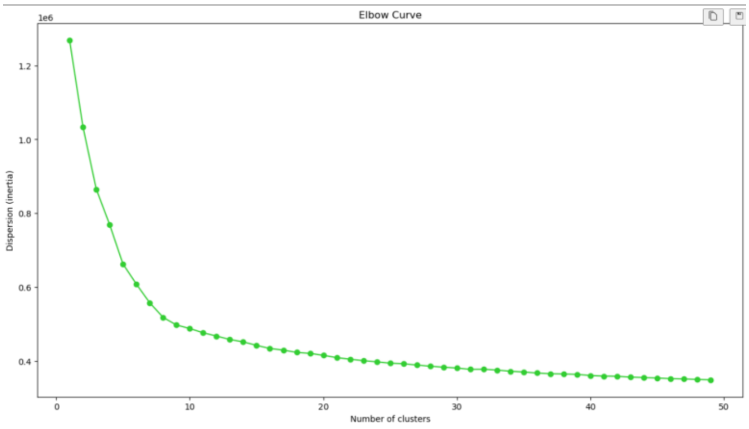
K-means Algorithm

K-Means is an unsupervised machine learning algorithm that partitions data into K distinct clusters. It iteratively assigns each data point to the nearest cluster centroid and recalculates the centroids until convergence. While simple and scalable, K-means requires predefining the number of clusters and assumes spherical cluster shapes. It is very sensitive to outliers, but remains widely used for tasks like customer segmentation, the main purpose of our project.

In K-means, Elbow and Silhouette Methods were performed to better understand which was the best value for K. The Elbow method helps determine the optimal number of clusters in K-means by plotting the within-cluster sum of squares (WCSS) against different cluster numbers and identifying the "elbow" point where the decrease in WCSS slows down significantly. The Silhouette Method assesses clustering quality by measuring how similar each data point is to its own cluster compared to others. Higher Silhouette Coefficient values indicate better clustering results.

Similar to what was done in the Hierarchical Clustering Algorithm, we looked at all the UMAPs to see which combination worked best. The conclusion was that using K-Means with the 'StandardScaler' on the dataset containing both original and new variables gave us the best results out of all the options we tried.

The Elbow method of this combination is presented in Figure 8. The graphic containing the silhouette scores in Figure 9 suggests that the highest score corresponds to having 9 clusters with a score of approximately 0.2944.



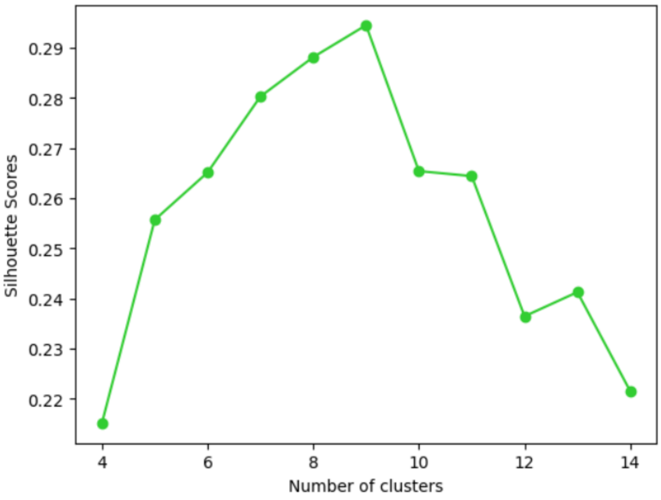


Figure 8 - Elbow Curve

Figure 9 - Silhouette Scores

Mean Shift Algorithm

Mean Shift is also an unsupervised machine learning algorithm that identifies clusters in data by locating the modes, high-density regions, of the underlying probability distribution. It iteratively shifts data points towards their nearest node, gradually converging to the cluster centers. Unlike K-means, Mean Shift doesn't require specifying the number of clusters beforehand and can discover clusters of arbitrary shapes. However, it is computationally more expensive and its performance depends on the choice of bandwidth parameter.

Although we experimented with various combinations in this algorithm, the outcomes were consistently disappointing. This could be attributed to improper utilization of parameters, particularly the 'bandwidth' parameter when defining the 'MeanShift' function. We relied on the result obtained by computing the 'estimate_bandwith' function, similar to what was demonstrated in practical classes. The team discovered the issue only after performing the association rules analysis, which showed that the values needed to be lower to form more than one cluster. This was happening because of the value chosen. Consequently, we decided to exclude this algorithm and did not include its code in our project. In future projects, we might fine-tune this parameter to achieve better results.

Self-Organizing Maps (SOM)

Self-Organizing Maps (SOMs) are a type of artificial neural network used for visualizing high-dimensional data by projecting it onto a lower-dimensional grid. SOMs work by iteratively adjusting the weights of neurons on the grid to better match the input data. Each neuron represents a prototype or cluster center, and similar data points are mapped to nearby neurons on the grid. This allows for the identification of patterns and relationships in the data that may not be easily discernible in the original high-dimensional space.

Initially, we generated a quantization errors plot, presented in Figure 10, to assess how accurately the neurons depict the real data. This analysis guides us in deciding the number of iterations needed for our SOM. Following this evaluation, we opted for 1000 epochs, although 700 epochs would have also been a viable choice.

Later, we plotted the individual points on the grid, Figure 11, to assess their grouping. However, as observed, the outcome was unsatisfactory. Consequently, our analysis of the Self-Organizing Map remained superficial. Upon realizing its limited effectiveness in our scenario, we decided not to invest further time in it.

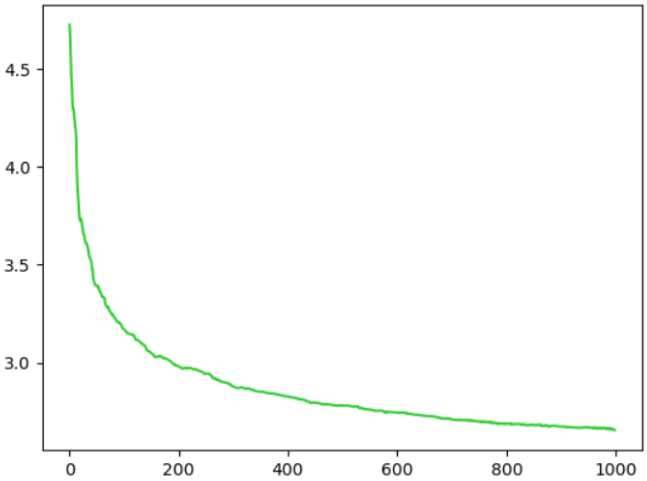


Figure 10 - Quantization errors' plot

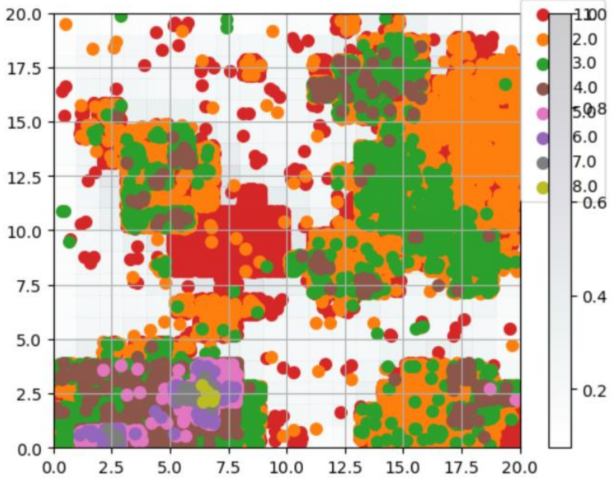


Figure 11 - Individual points plotted in the grid

Remark: Figures 10 and 11 depict the results of the Self-Organizing Map applied to the dataset containing both the original and new variables, using the 'StandardScaler' method.

Choosing Final Algorithm

Upon analyzing all UMAP visualizations of our choices, we determined that the Hierarchical Algorithm yielded the best performance, specifically, when utilizing the 'RobustScaler' method on the dataset comprising both original and new variables, as illustrated in Figure 12. This ultimate choice facilitated our progression to the cluster specification phase.

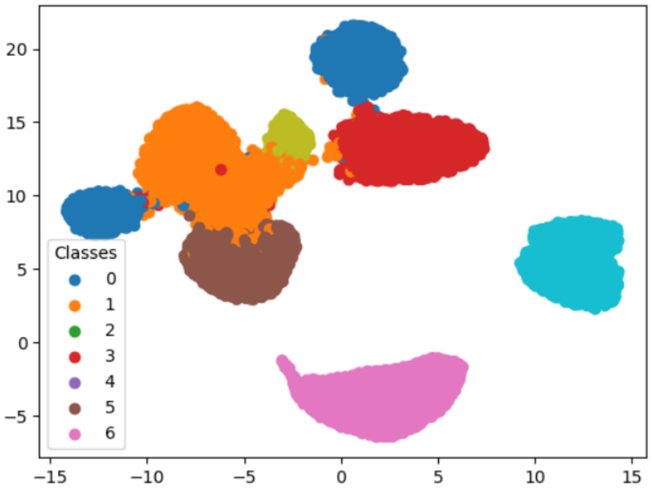


Figure 12 - UMAP of the final algorithm

Clusters

After developing the clustering model, we calculated the average values for each variable within each cluster. This analysis resulted in the identification of seven distinct customer clusters, each exhibiting unique characteristics and behaviours. These clusters were named based on their predominant traits, which were derived from the most relevant variables.

Cluster 0: Diversifiers

The Diversifiers are named for their high value in the column lifetime_total_distinct_products, which indicates that they purchase a wide variety of different products. This suggests that these customers have diverse shopping habits, preferring to try and buy a range of items rather than sticking to a limited selection.

Cluster 1: Occasional Discount Shoppers

Occasional Discount Shoppers are characterized by a high percentage of products bought on promotion, as indicated by the percentage_of_products_bought_promotion column. This means they are particularly sensitive to discounts and promotions, and they make purchasing decisions based on the availability of sales and special offers. This cluster represents customers who are thrifty and strategic in their shopping, making them ideal targets for promotional campaigns and discount-driven marketing strategies.

Cluster 2: Large Families

Large Families have a high number of children and teenagers at home, reflected in the columns kids_home and teens_home. This cluster has the highest values in these variables, indicating that these households have multiple dependents. As a result, their purchasing patterns are influenced by the need to cater to many individuals, leading to higher overall spending on groceries and essentials.

Cluster 3: Vegetarians

Vegetarians are identified by their high spending on vegetables, as shown in the lifetime_spend_vegetables column. This cluster has the highest expenditure in this category, and correspondingly low spending on meat and fish products. These customers prefer plant-based diets, which can be inferred from their minimal spending on animal products.

Cluster 4: Pet Owners

Pet Owners are distinguished by their significant spending on pet food, highlighted by the lifetime_spend_petfood column. This group allocates a large portion of their budget to pet-related products, indicating that they have one or more pets at home.

Cluster 5: Young Shoppers

Young Shoppers are characterized by their low average age, as indicated by the customer_age column. This cluster is the youngest among all the groups, with an average age significantly lower than the others.

Cluster 6: Premium Clients

Premium Clients stand out due to their high total lifetime spend, noted in the total_lifetime_spend column. This group has the highest overall expenditure across various categories, indicating a higher purchasing power and preference for premium products. They are likely to be less price-sensitive and more focused on quality and brand prestige.

Targeted Promotions

This section refers to the establishment of association rules, which led to the design of targeted campaigns for the various client categories, after the previously described phases of exploratory data analysis and customer segmentation.

Association rules are statements that uncover relationships between items in large datasets. They identify patterns where the presence of one item in a transaction implies the presence of another item with a certain probability. This information is valuable for our project to understand customer behaviour and develop effective marketing strategies.

Starting with the algorithm that we have used, The Apriori Algorithm. The term "apriori algorithm" describes the algorithm that determines the principles of association between items. It denotes the relationship between two or more objects. There are three major components of the Apriori algorithm, support, confidence and lift. Support calculates the average popularity of each product or data item in the data set. The total number of transactions involving that product must be divided by the total number of transactions. Confidence is the probability that item Y will be purchased if item X is purchased. It is computed by dividing the total number of transactions that contain X by the total number of transactions that contain Y. Lastly, lift indicates the likelihood that item Y will be bought when item X is bought, accounting for item Y's level of popularity. If item X is purchased, item Y is likely to be purchased if the lift value is larger than 1, but a value less than 1 indicates that item Y is unlikely to be purchased if item X is purchased. In our project, our clusters have lift values higher than 1, which means a noteworthy correlation between the objects in every cluster.

Following are the promotions that we make based on each association rule.

Diversifiers

In this segment, it is possible to analyse that the products 'cake', 'cooking oil', and 'oil' are frequently purchased together. When 'cake' is acquired, there's a tendency for the simultaneous purchase of 'cooking oil' and 'oil', and vice versa. The associations between these products are moderate, with confidences around 40-50% and lifts of approximately 1.10.

Regarding this, the following promotions were proposed:

1. Buy Cake, Cooking Oil, and get a 10% discount on a loyalty card
2. Buy Cake and get Cooking Oil at 20% Off
3. Buy Cooking Oil and Cake and get a Free Recipe Book
4. Buy 2 Cakes and get a Candy Bar Free
5. Buy Cake and Candy Bars, save 5€ on the loyalty card
6. Buy Muffin and Cooking Oil and get 15% Off on Baking Accessories
7. Buy Cooking Oil and Muffin and get a discount on Healthier Baking Options

Occasional Discount Shoppers

The analysis of occasional discount shoppers reveals patterns in purchasing behaviour, particularly regarding the items 'oil', 'cooking oil', and 'candy bars'. These shoppers often show a preference for specific combinations of these products, such as 'oil' leading to the purchase of 'cooking oil' or 'candy bars'. Confidence levels are relatively high, ranging from 57.97% to 78.60%, indicating a strong likelihood of these associations occurring. The lift values, which range around 1.41 to 1.42, suggest that these associations are more frequent than random chance.

Regarding this, the following promotions were proposed:

1. Buy Any Cooking Oil and get a discount on other Oil products
2. Buy Cooking Oil and Vegetable Oil and save 20% on the loyalty card
3. Buy Cooking Oil and get 50% Off on second Oil purchase
4. Buy Candy Bars and Any Oil and get 10% Off in both products
5. Buy 2 Oils and get a Pack of Candy Bars Free
6. Buy Cooking Oil and get one Free Cake

Large Families

The association analysis for large families, reveals significant patterns in purchasing behaviour, particularly concerning the items 'cake', 'cooking oil', and 'babies food'. These items frequently appear together in shopping transactions, indicating potential preferences or needs among large families. For instance, 'cake' is often purchased alongside 'babies food' and 'cooking oil', and vice versa. Confidence levels range from 45.94% to 85.08%, suggesting strong associations between these items. Lift values, which range from 1.04 to 1.10, indicate that these associations are more likely than random chance. However, while statistically significant, the practical impact of these associations on consumer behaviour may vary.

Regarding this, the following promotions were purposed:

1. Buy Cake and Babies' Food and save 15% on the loyalty card
2. Buy any Cooking Oil and Babies' Food and get 10% Off on the loyalty card
3. Buy Cake, Babies´ Food and Cooking Oil and get free Baking Utensil Set
4. Buy 2 Candy Bars and get Cooking Oil at 50% Off
5. Buy Cooking Oil and Cake and get a free Cake Decorating Kit

Vegetarians

The association analysis about vegetarians reveals purchasing patterns among items commonly consumed as expected by vegetarians, such as 'tomatoes', 'asparagus', 'mashed potato', and 'carrots'. There are frequent associations between these items within transactions, indicating potential dietary preferences or meal compositions among vegetarians. Confidence levels range between 36.93% and 85.11%, suggesting moderate to strong associations between these items. Lift values, ranging from 1.04 to 1.11, further support the significance of these associations.

Regarding this, the following promotions were proposed:

1. Buy tomatoes, asparagus, and mashed potatoes together, and get 20% off the total price
2. Buy any tomatoes and get asparagus for half-price
3. All mashed potatoes 30% off this week
4. Buying mashed potatoes and asparagus together offers a 20% discount on tomatoes
5. Give a 10% discount on tomatoes and carrots when purchased with asparagus

Pet Owners

Notably, between pet owners there are strong associations between certain types of fish products such as 'canned tuna', 'fresh tuna', 'catfish', and other related items. These associations are characterized by high confidence levels of 100% and lift values of 4, indicating a strong likelihood of purchasing these items together. Additionally, there are associations between 'grated cheese', 'shrimp', 'shower gel', and 'olive oil', suggesting potential purchasing patterns among pet owners.

Regarding this, the following promotions were proposed:

1. Buy fresh tuna and catfish, and get a 15% discount on canned tuna
2. Buy grated cheese and get 20% off on shrimp and shower gel, available only for loyalty card members
3. Buy grated cheese and get 20% off on shrimp and shower gel
4. Buy 3 packs of shrimp and get 1 bottle of olive oil and 1 pack of salmon for free

Young Shoppers

The association analysis suggests that young shoppers exhibit interesting preferences in alcoholic beverages. Specifically, there are significant associations between 'beer' and 'white wine', as well as between 'white wine' and 'dessert wine'. These associations indicate that young shoppers are likely to purchase beer and white wine together, as well as white wine and dessert wine together. Additionally, there is a notable association between 'cider' and 'white wine', suggesting that young shoppers may also favor cider along with white wine. These associations are characterized by moderate to high confidence levels and lift values, indicating a likelihood of these items being purchased together by young shoppers.

Regarding this, the following promotions were proposed:

1. Offer a 10% discount on white wine when customers purchase beer
2. Buy white wine and get a 15% discount on beer
3. Offer a 25% discount on white wine when customers buy dessert wine
4. Buy cider and get 50% discount on white wine
5. 10% discount on white wine exclusively for loyalty card members
6. Buy white wine and get 15% discount on dessert wine, only available for loyalty card members

Premium Clients

For premium clients, analysis reveals interesting trends. There is a moderate link between purchasing champagne and laptops, with around 27% confidence and a slight uplift. Similarly, Samsung Galaxy 10 purchasers tend to buy Bluetooth headphones, and vice versa, with about 41% confidence. Additionally, there's an association between champagne and Bluetooth headphones, spaghetti, and Airpods, with moderate confidence levels. However, these associations may not significantly influence purchase decisions due to low lift values.

Regarding this, the following promotions were proposed:

1. Buy a laptop and get 20% off on champagne
2. Buy a Samsung Galaxy 10 and receive a free pair of Bluetooth headphones
3. Buy champagne with spaghetti and save 15% discount on the loyalty card.
4. Get 10% off on champagne to celebrate when you buy a Samsung Galaxy 10

Note: By running promotions for loyalty card members, it encourages customers to create the same card, and thus the company has more and more data about its customers.

Final Analysis

As previously said, we have saved a CSV file with the outliers, for a later analysis. So firstly, we decided to create a new column named *cluster_ward*, in the outliers dataframe.

We concatenated the outliers dataframe with the dataframe with clusters, so we could obtain a final solution with the columns,*customer_id, customer_name, customer_gender, cluster_ward, latitude* and *longitude*.

Finally, we have performed a map, figure 13, and by analyzing this, it is evident that outside of Lisbon, the majority of customers fall into the outlier category. While further in-depth studies to analyze the specifics and characteristics of these outliers may not be conducted, their prevalence highlights potential areas of interest for future research and strategic analysis.

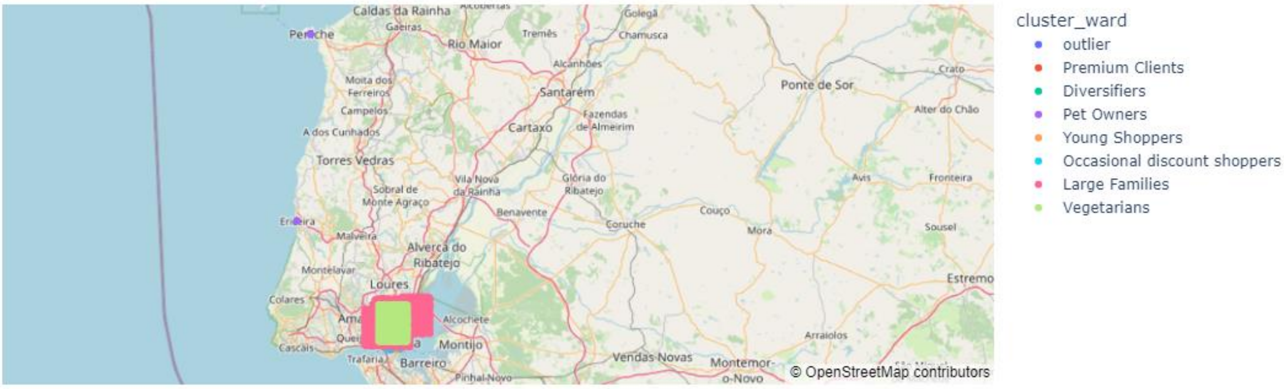


Figure 13 - Geographic Distribution of Customers and their Clusters

Conclusion

This project demonstrated the power of customer segmentation and association rule analysis to develop targeted promotions in a supermarket. By analyzing demographics, purchasing habits, and transactional data, we identified distinct customer segments such as "Diversifiers," "Occasional Discount Shoppers," "Large Families," "Vegetarians," "Pet Owners," and "Young Shoppers."

For each segment, association rules revealed purchasing patterns and product affinities, allowing for the creation of personalized promotions. By implementing these targeted promotions, the supermarket can expect increased customer engagement, higher sales, and improved loyalty. Additionally, the analysis provided valuable insights into customer behavior, such as the popularity of the loyalty card program and the preferences of different demographic groups.

The project employed various clustering algorithms, including K-means, Hierarchical Clustering, and Self-Organizing Maps (SOM), to identify the optimal segmentation solution. After evaluating the performance of different algorithms and parameter settings, the Hierarchical Clustering algorithm with RobustScaler on the combined dataset of original and new variables was selected as the final model, resulting in the identification of distinct customer segments.

In conclusion, this project highlights the importance of understanding customers on a granular level and tailoring marketing strategies accordingly. By doing so, supermarkets can not only boost sales but also build stronger relationships with their customers, leading to sustainable growth and success in the future.

References

1. *A demo of the mean-shift clustering algorithm*. scikit. (n.d.-a). https://scikit-learn.org/stable/auto_examples/cluster/plot_mean_shift.html

2. *Agglomerativeclustering*. scikit. (n.d.). https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AgglomerativeClustering.html

3. *Apriori algorithm - javatpoint*. www.javatpoint.com. (n.d.). https://www.javatpoint.com/apriori-algorithm

4. JustGlowing. (n.d.). *JustGlowing/minisom: :red_circle: Minisom is a minimalistic implementation of the self organizing maps*. GitHub. https://github.com/JustGlowing/minisom

5. *Scipy.cluster.hierarchy.dendrogram#*. scipy.cluster.hierarchy.dendrogram - SciPy v1.13.1 Manual. (n.d.). https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.dendrogram.html

6. Tanner, G. (n.d.). *Mean shift*. Machine Learning Explained. https://ml-explained.com/blog/mean-shift-explained