Project Data Analytics

Shop Customer Data Analysis

RAVEY Bruce & Clouet Benjamin

1. Discovery:

A Shop has collected data from its customers through membership cards and need helps to understand them. The CEO came asking “Ravey & Clouet Data Consulting company” to understand them and get any advice that we could have according to their data.

The dataset has the following variables :
- ID
- Gender
- Age
- Income
- Spending Score
- Profession
- Experience
- Family Size

The Spending Score is a metric that is determined according to predefined parameters like purchasing data and customer behaviour.

We have firstly thought about doing regression/ML supervised, to predict the spending score according to some variables. However, we finally thought it was not very useful, from a business point of view. So, we finally decide to do unsupervised machine learning to be able to create a group of "target customers" that the shop must advertise by phone/e-mail.

As "student challenge", we have decided to try to do different unsupervised machine learning that we did not see in course, to understand better each advantages / disadvantages / characteristic of each model.

We have supposed different things on the data set:

- Is the repartition between woman and men is the same?
- Is the variable normalized ?
- Is the repartition of each variable being the same between men and women?
- Is it possible to generate cluster and give advice to the shop depending on the characteristics of each cluster

2. Data Preparation:

A- Find and load the data set:

First, it is Kaggle Dataset, so we did not have to do any difficult things to extract the data (Scraping, etc.). We just parallelize on the different CPUs of the computer the “load” of the data. Indeed, in our case the dataset is not huge, but good practice is important. Indeed, parallelizing the load of the data allows to have small part of the data, which is attributed to CPU of your computer, and then merged. It should allow to do the process faster.

B- Data set information:

We first tried to get information about the data set. We had this information:

- The data frame contains 2000 rows and 8 columns.

- They are all integer columns, unless gender and profession
- There is missing value for the columns: Profession.
- There is some variable as age which are equals to 0 (and little bit more) and so impossible.

C- NA treatment:

We had problems for the treatment of NA and aberrative value. Indeed, our data set is already very small, so it was difficult to remove them.

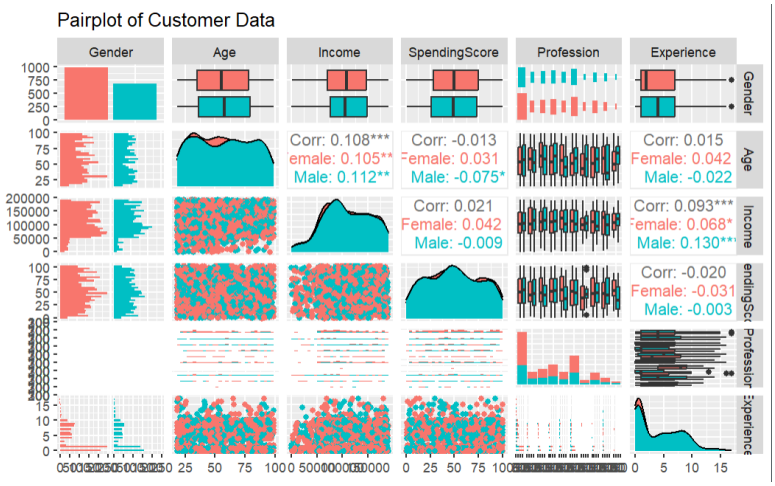
About the NA of the column profession, we decided to label them as “unknown”.

For the treatment of the customer which are under 18 and already a job we decided to remove them to have a data set consistent (indeed, there is no blue collar in our dataset, so we decided to remove before 18, but it could be 22-23 following the job they have).

3. Model Planning:

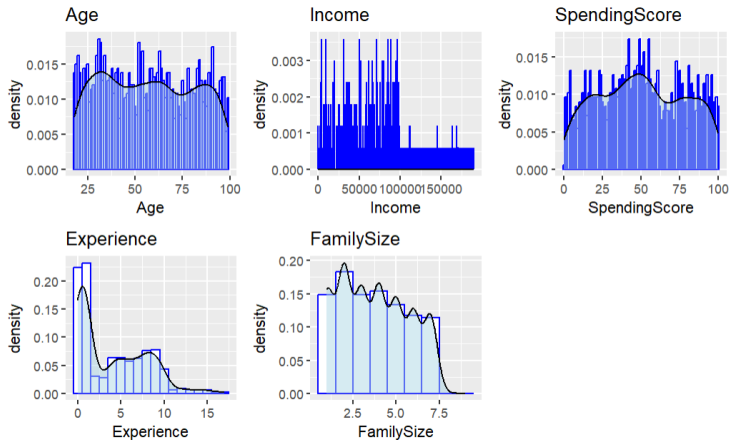
A- Exploratory Data analysis:

We first did a pair plot to look at deeper the variable. However, it does not look very well (not like Python with the seaborn package which looks amazing).



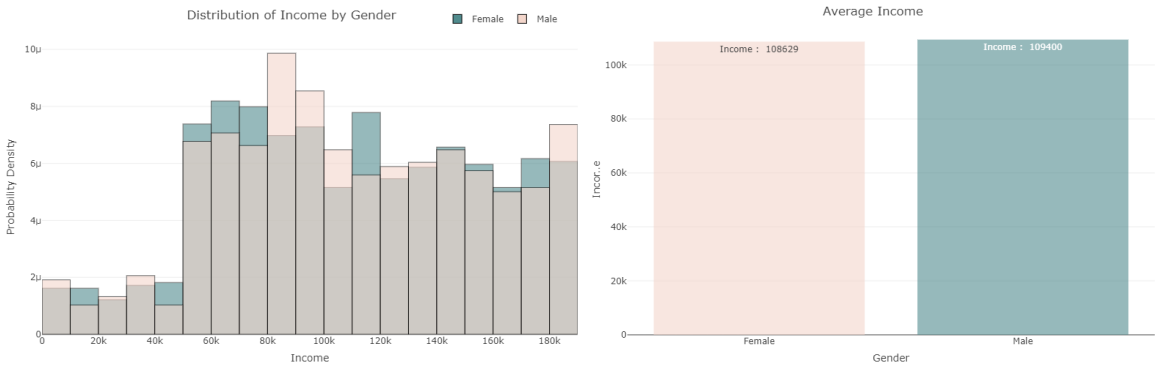
We can see from this plot that there is a lot more women than men in the shop. The density distribution of each variable is sensibly the same whatever the gender.

Then, we displayed the density histogram/curve of each variable which were integers, in another plot to see them better.

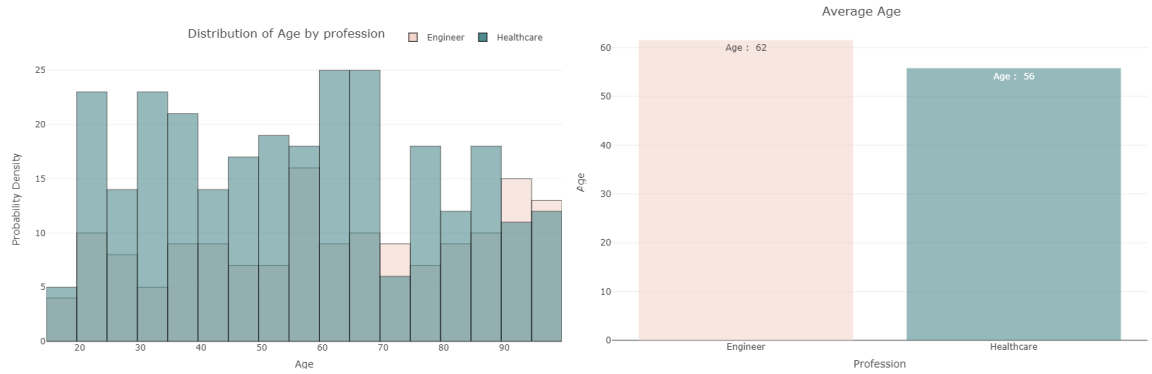


We can see from this plot that the variables are not normally distributed. ML models, unsupervised model included, generally need normalized data. We will so after normalizing them.

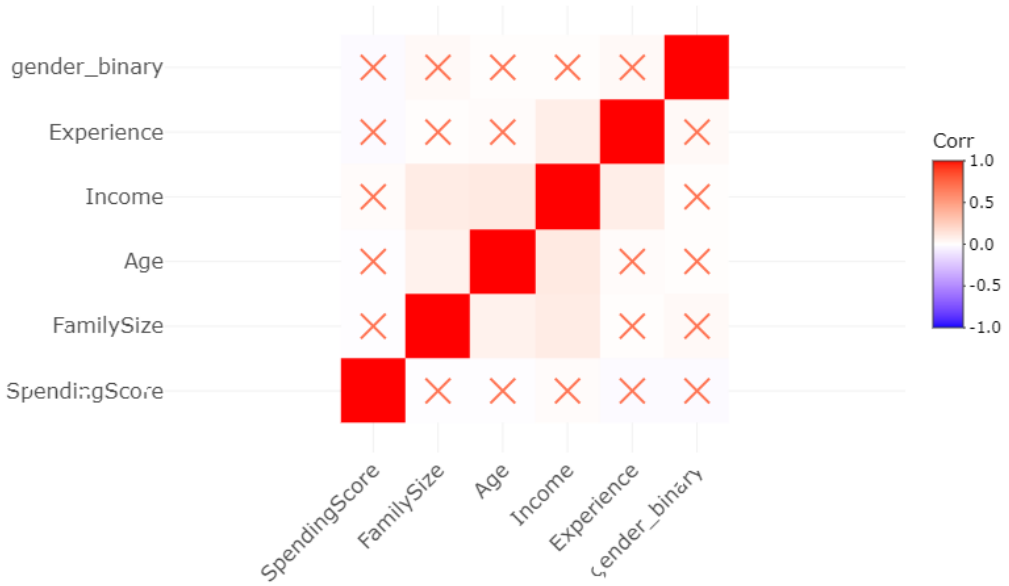
We also plot some other graphs to look at the distribution of variable and the average per gender. We will show you some here, but you can play with our RshinyApp to check more.



We also plot average of the variables depending on the profession:



Finally, we have plotted a correlation matrix to see the correlation between the variables (It is a plotly matrix, so it does not display the number, unless you put your mouse on it. Check on the Markdown)



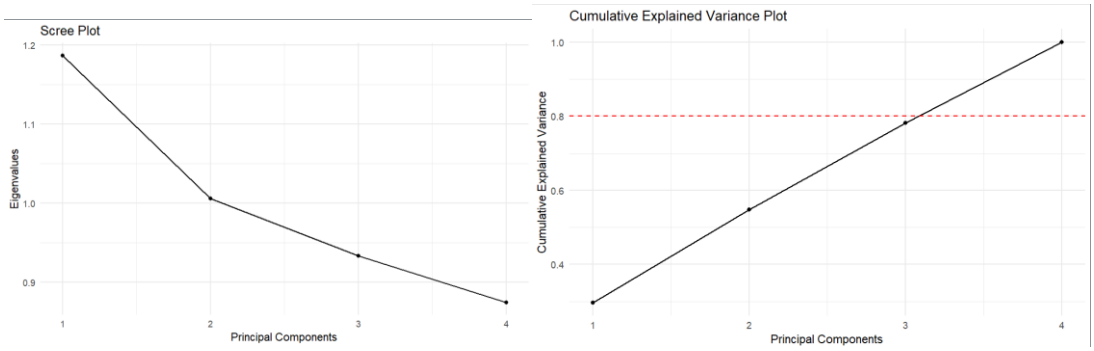
We can see that Spending Score is correlated with every variable; gender less but also correlated with all of them. Seeing, that some variable, we thought it could be interesting to remove some variable.

B- Choice of variable:

Indeed, removing some variable before doing our clustering is we think a good idea. Indeed, after understanding what exactly unsupervised model do, we were asking if using binary variable (gender, each profession) was not a mistake. Indeed, our first thought was that it would double each for each cluster by binary variable. Indeed, after reading this website, it looked strange to do that unless we did PCA. Because we did not find any marketing aim to clusterize by profession, we decided to remove it. Then, about the gender binary, because of the current situation of gender (LGBT) in our society, we also decided to remove it.

C- Principal Component Analysis:

Principal Component Analysis (PCA) is a widely used technique in machine learning for dimensionality reduction and feature extraction. It works by transforming the original high-dimensional dataset into a lower-dimensional space while preserving as much of the variance in the data as possible. PCA accomplishes this by identifying the orthogonal axes, called principal components, that capture the most significant patterns of variation in the data. These principal components are linear combinations of the original features and are ordered by the amount of explained variance, with the first component accounting for the highest variance, the second component for the next highest, and so on. By selecting a smaller subset of principal components, PCA allows for a more manageable and interpretable representation of the data, often leading to improved model performance and reduced computational complexity. Additionally, PCA can help mitigate issues related to multicollinearity, noise, and overfitting in machine learning models.



The first graph is scree plot. A scree plot is a graphical representation of the eigenvalues associated with each principal component in a principal component analysis (PCA). The scree plot allows you to visualize the amount of variance in the data explained by each principal component. The elbow point is at 2, so we should use this point. However, after looking at the ”Cumulative explained variance plot”, at 2 PCA it is less than 0.5 which is explained. Even 3, PCA is lower than 80%, so we decided to do not use PCA.

D- Characteristics/Advantages/Disadvantages of the models:

We will use different model of clustering: K-means, hierarchical clustering, and DBSCAN. It is a challenge because we have never seen/studied some of these models. We will then compare the results to propose to the CEO of the shop, the best advice for its advertisement/promotion campaign.

K-means is a partitional clustering algorithm that requires specifying the number of clusters (K) beforehand. It works by minimizing the within-cluster sum of squares. One of the main advantages of K-means is its simplicity, making it easy to implement and understand. It also has an efficient time complexity of O (n * I * K), which allows it to work well with large datasets. In cases where the clusters are well-separated and globular, K-means often produces good results.

However, the K-means algorithm has its limitations. It assumes that clusters are spherical(convex) and have similar densities, which may not always be true for real-world datasets. The algorithm is sensitive to the initial placement of centroids and may converge to local optima instead of global optima. Additionally, K-means requires a priori knowledge of the number of clusters (K), which may not always be available. Lastly, the algorithm is sensitive to outliers, which can negatively impact the clustering results.

Hierarchical clustering is a clustering algorithm characterized by its ability to build a tree-like structure, known as a dendrogram, which represents the nested cluster hierarchy. The algorithm can either be agglomerative, using a bottom-up approach, or divisive, using a top-down approach. Unlike other clustering methods, hierarchical clustering does not require specifying the number of clusters beforehand.

The advantages of hierarchical clustering include providing a full hierarchy of clusters for different granularity levels, making it more intuitive and interpretable through dendrograms. The algorithm can work with various distance metrics and linkage criteria and does not require a priori knowledge of the number of clusters. However, there are some disadvantages to using hierarchical clustering. It has a higher time complexity compared to K-means, ranging from $O(n^2)$ to $O(n^3)$ depending on the linkage method, making it less suitable for large datasets. The algorithm is sensitive to the choice of distance metric and linkage method and does not guarantee optimality in the clustering results.

DBSCAN, or Density-Based Spatial Clustering of Applications with Noise, is a density-based clustering algorithm that identifies clusters based on high-density regions separated by low-density regions. Unlike some other clustering methods, DBSCAN does not require specifying the number of clusters beforehand and considers noise points, or outliers, during the clustering process.

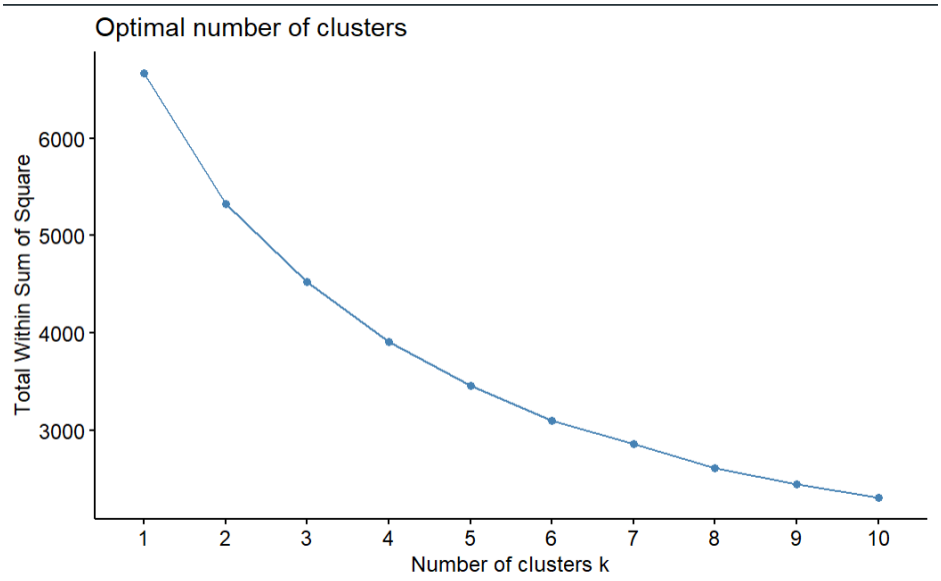
Some of the advantages of DBSCAN include its ability to find clusters of arbitrary shapes and its robustness to noise and outliers. The algorithm does not require a priori knowledge of the number of clusters and only requires tuning two parameters: the radius (Eps) and the minimum number of points (MinPts). However, DBSCAN has some disadvantages. It is not suitable for datasets with varying densities and is sensitive to the choices of Eps and MinPts parameters. The algorithm struggles with high-dimensional data due to the curse of dimensionality, leading to degraded performance as the dimensionality of the dataset increases.

4. Model Building:

A- K-Means:

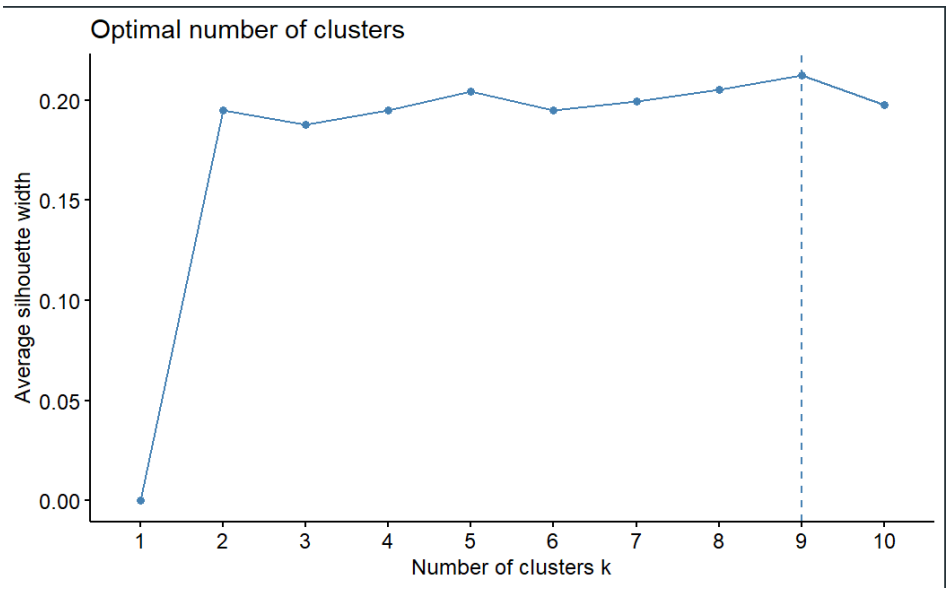
Before beginning to run our model, we need to find the number of clusters which fit the best our data. To do that, we will plot different ‘test’.

We will first begin by Within-Cluster-Sum-of-Squares (WSS). WSS is a measure used to evaluate the quality of clustering in K-means clustering algorithm. To find the optimal number of clusters, WSS calculates the sum of squared distances between each point and its assigned cluster center.



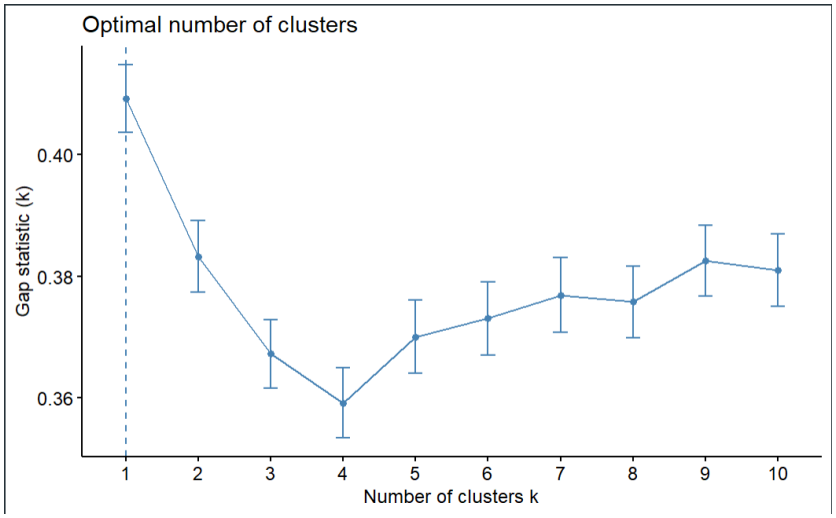
To understand this graph, we must find the last elbow of this graph. It looks like it is around 6-7.

The average silhouette width (ASW) measures how similar a point is to its own cluster compared to other clusters. A high ASW value indicates that a point is well-matched to its own cluster and poorly matched to neighbouring clusters



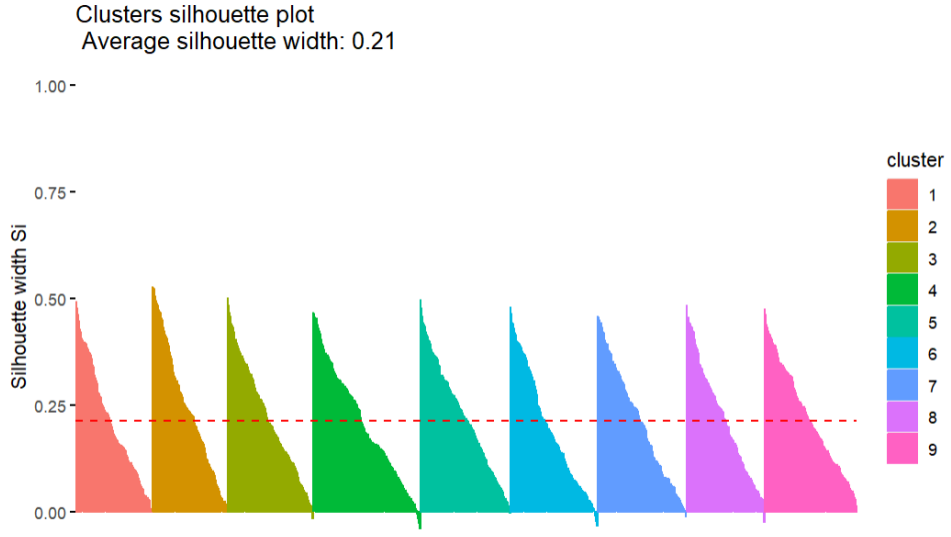
Looking at the graph, this measure is close to be the same for each number of cluster, even if 9 is the highest one.

The method compares the total within-cluster variation for different numbers of clusters with their expected values under a null reference distribution, which is generated using a Monte Carlo simulation. The optimal number of clusters is the number that maximizes the gap between the observed within-cluster variation and the expected variation. The rationale is that if the gap between the observed and expected values is large for a certain number of clusters, it indicates that the clustering is good and there is a significant structure in the data.



Following this graph; the optimal number is 2 or 9 but because we aim to target some customers, we finally chose to have more than 2 cluster.

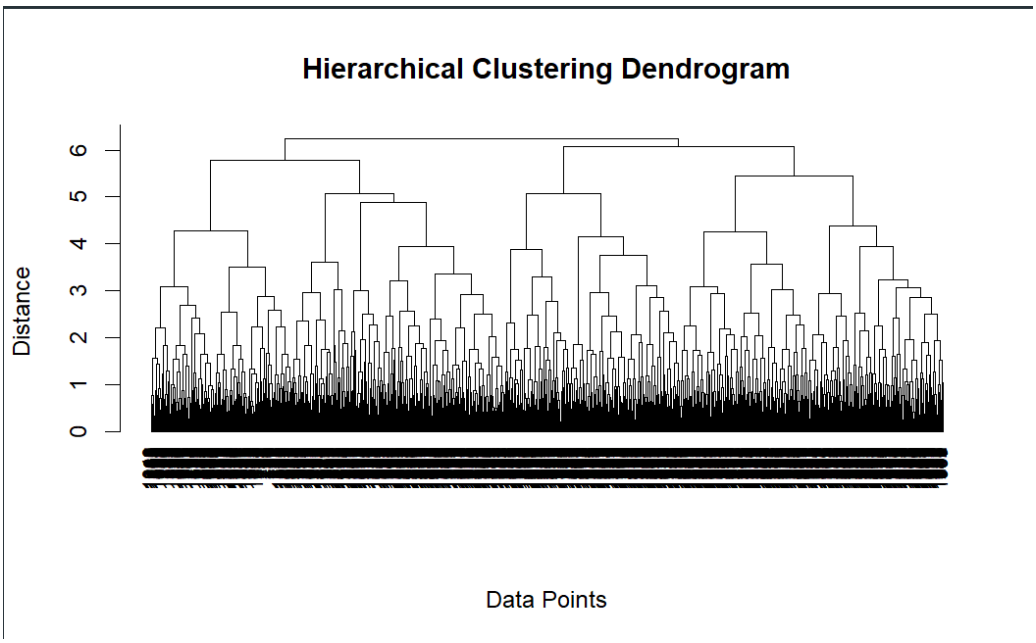
Then, we run the clustering algorithm, and we need now to evaluate our cluster. We will use the silhouette score. The silhouette score is a metric used to evaluate the quality of clustering in K-means clustering algorithm. The score measures how dense and well-separated the clusters are by considering both the intra-cluster distance and the inter-cluster distance. The score ranges from -1 to 1, with 1 indicating well-separated and dense clusters, 0 indicating overlapping clusters, and less than 0 indicating that data may be assigned to the wrong clusters. Silhouette plots can be used to identify the optimal number of clusters by examining cluster scores, fluctuations in cluster size, and the thickness of the silhouette plot. In general, a higher silhouette score indicates better clustering, and the optimal number of clusters is the one that maximizes the average silhouette score.



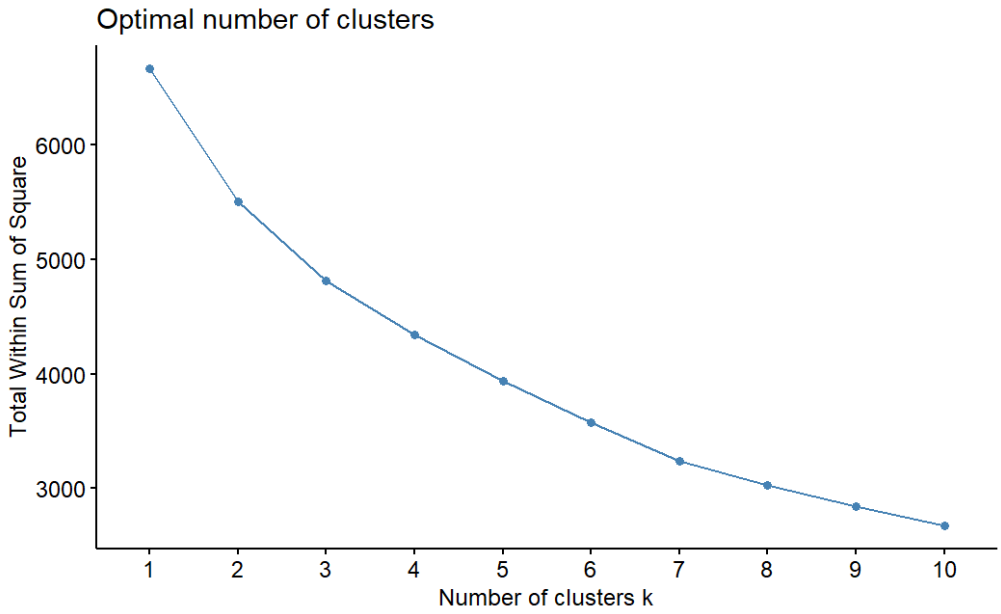
Following the size of the silhouette score, clusters are separated and well match. From the "width"of each cluster, we can know that they have roughly the same size.

B- Hierarchical clustering:

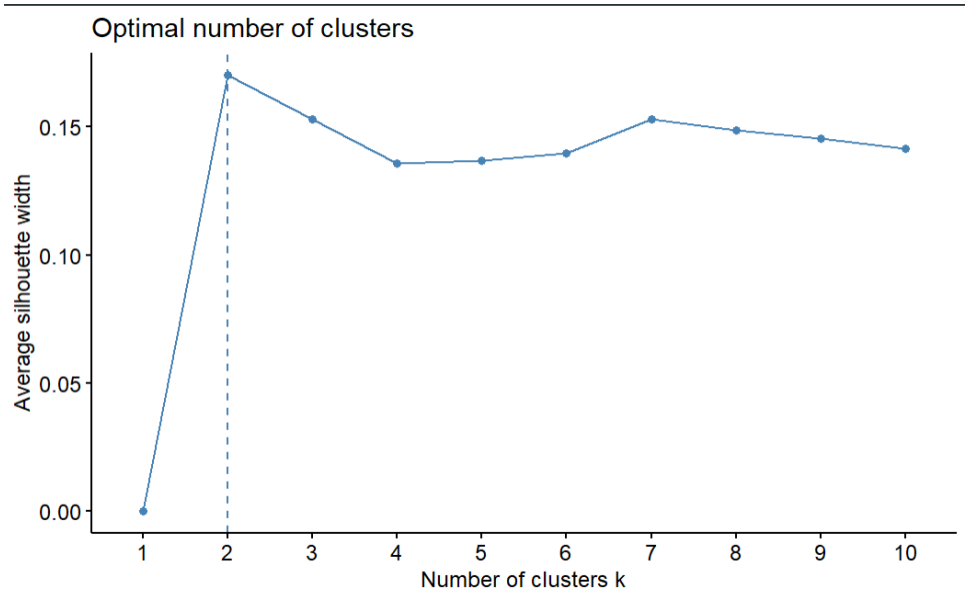
The dendrogram below displays the hierarchical clusters produced through complete linkage, which merges clusters with the smallest maximum distance between their observations. The height of the branches represents the distance between the clusters. Clusters that share greater similarity are merged at lower levels and become increasingly dissimilar as we move towards the top of the dendrogram. To determine the number of clusters, we can slice the dendrogram horizontally. For example, if we slice it at 90 heights, we obtain around six clusters, with most customers belonging to the first cluster, marked in purple. Now, let us examine the outcomes of hierarchical clustering, which is based on customer spending and income.



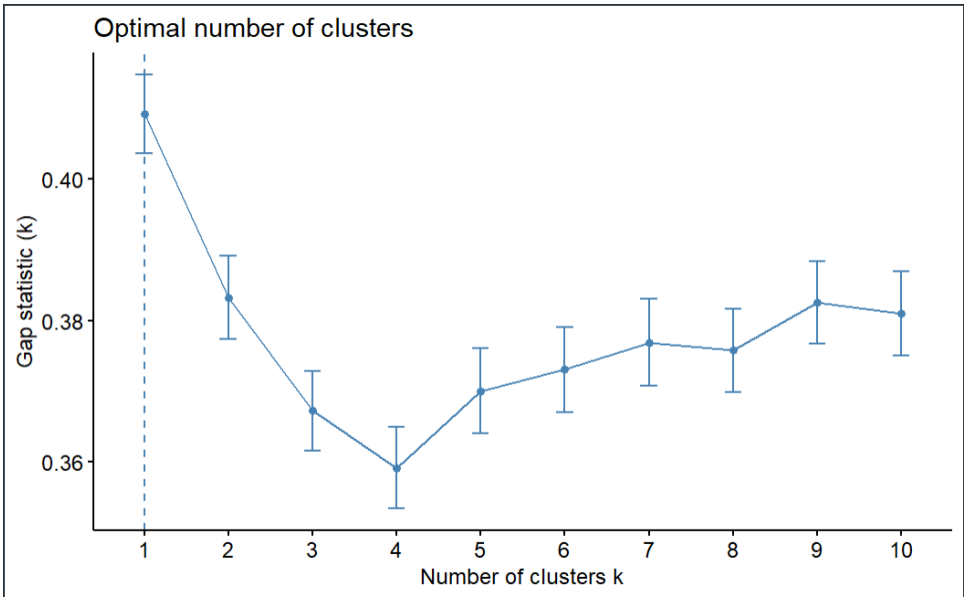
As done with the K-means we will use the same statistic plots to find how many clusters would best fit our data with Hierarchical clustering.



The last elbow looks like to be at 7.



Looking at the graph, this measure is close to be the same for each number of clusters, even if 9 is the highest one. Furthermore, we can note that the average silhouette width is lower than the K-Means one.



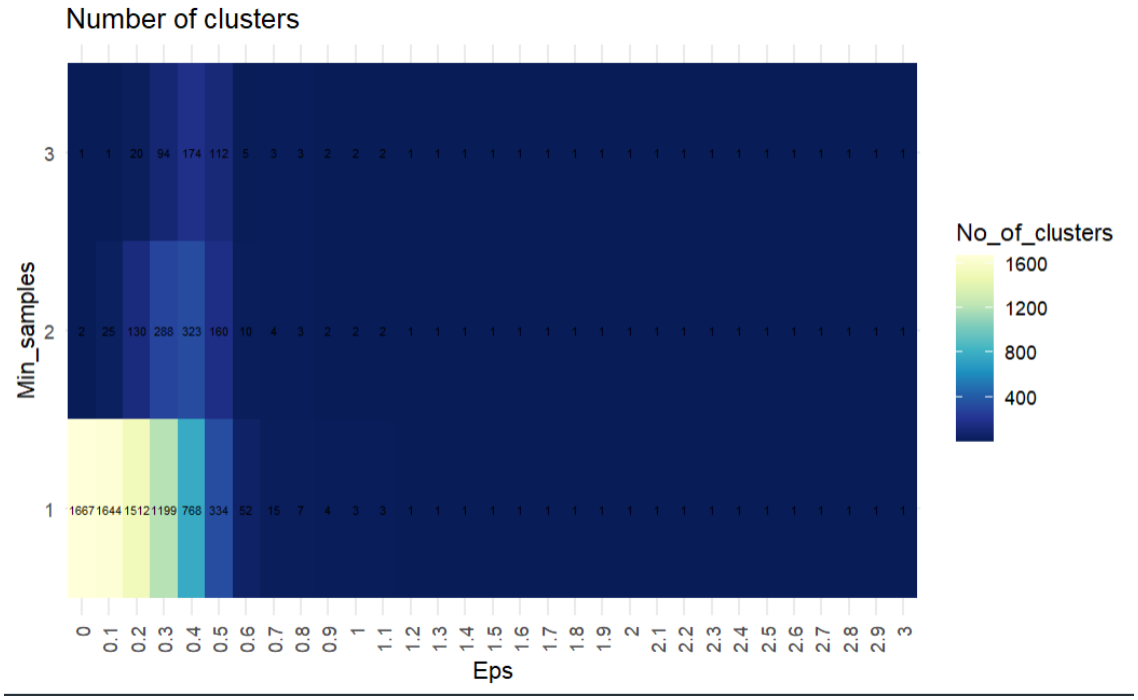
According to this graph the best number of clusters are 2,9 and 7 following this order.

To conclude the hierarchical clustering part, we can see that the K-means method has a higher silhouette average. So, we will not go further with this method.

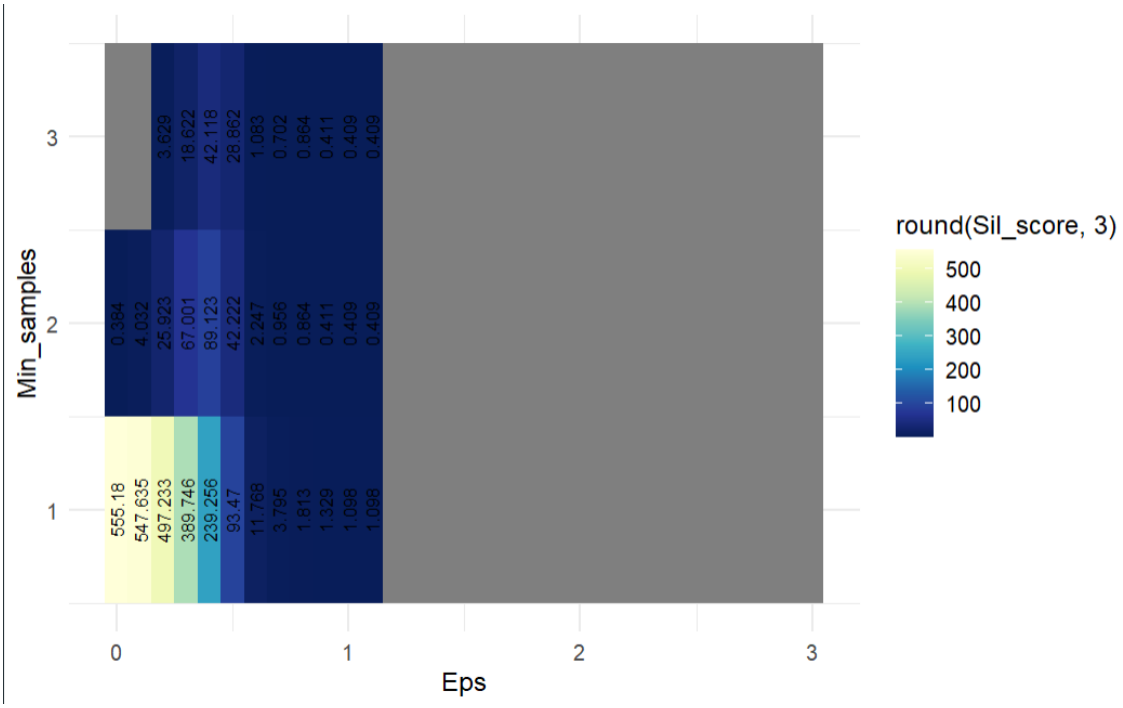
C- DBSCAN

DBSCAN is a model, where we must tune hyper-parameters to get the best clusters. What will we do to find them is to a Grid Search to find the best cluster.

The heat plot below shows how many clusters were generated by the DBSCAN algorithm for the respective parameter combinations.

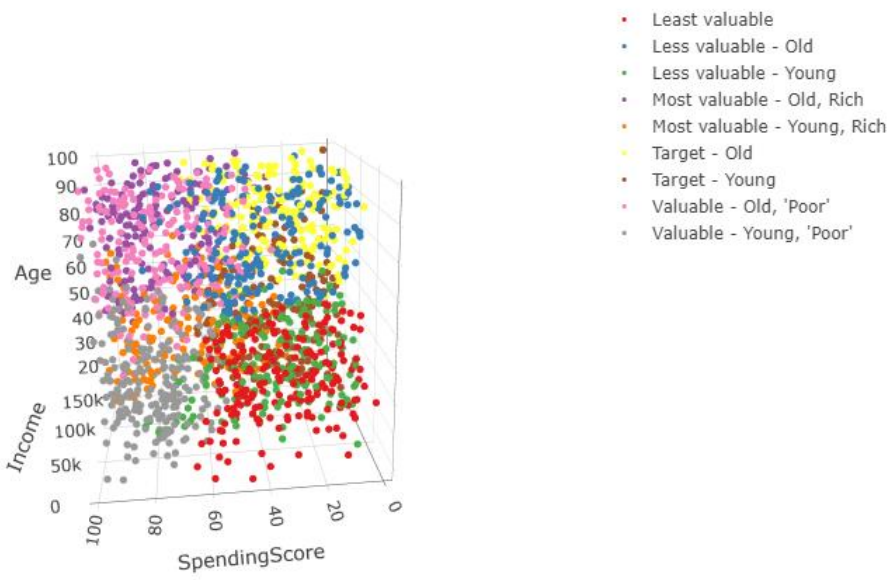


It shows some number of cluster impossible (more than 20) and only one cluster in majority. However, some value could be interesting. To choose between these number of clusters we will plot the same Grid Search but now with silhouette average score.

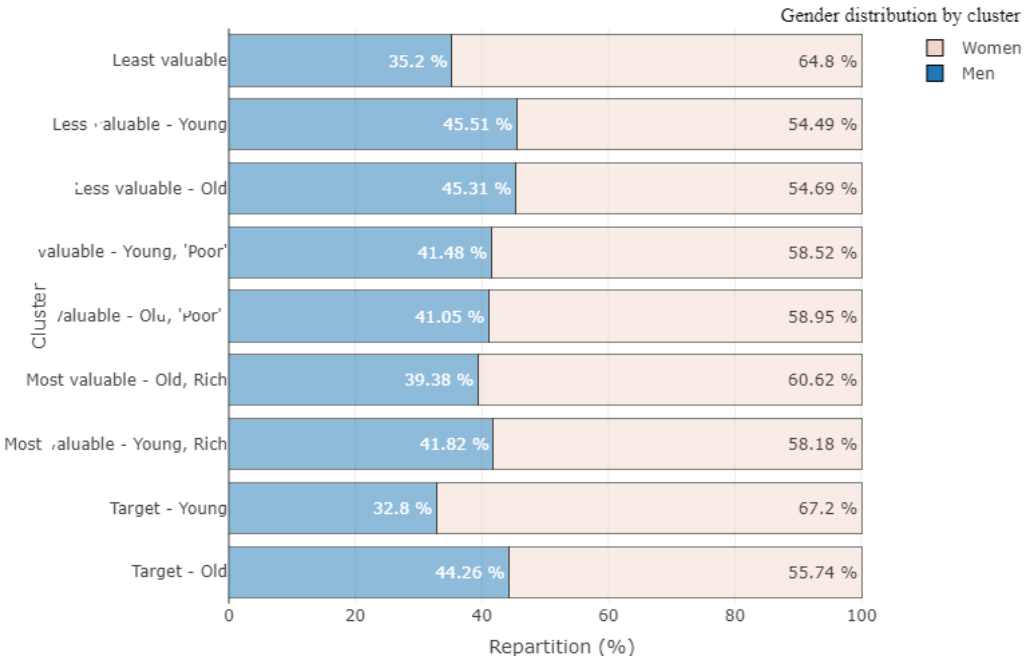


The problem with this graph is that we have average silhouette score higher than 1. As we explained in the K-means part, it should be between –1 and 1. For the number of clusters which were interesting us: 4-15, it is always number of at least 3–4-digit score. Because we did not understand why it was so huge, we will not go further with this model too.

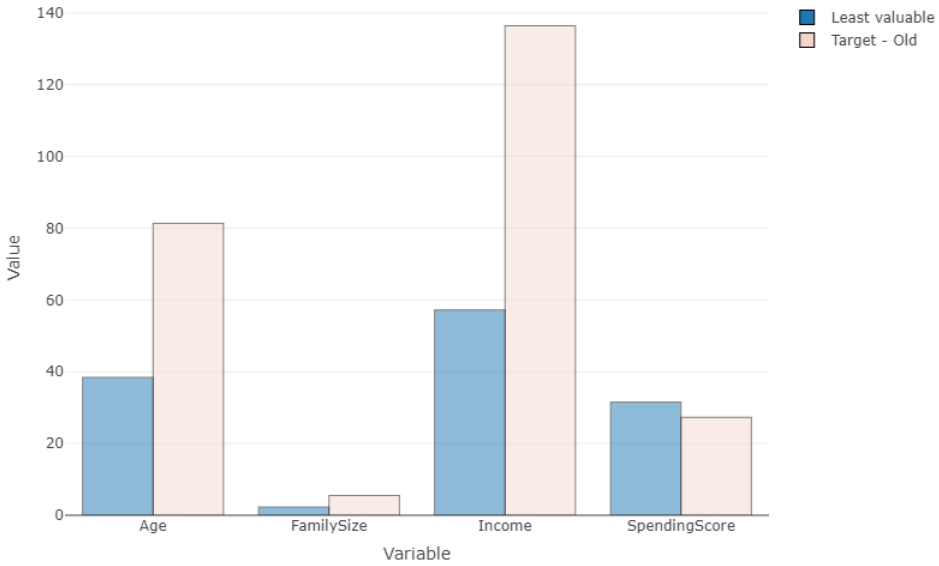
5. Results:
   A- Graphs/plots:



In this 3D plot, that you can play with in our Markdown or RshinyApp, you are able to see the repartition of the clusters depending on 3 variables: Income, Spending Score and age



On this plot, you can see the repartition of both gender in each group. They approximatively have the same repartition. Unles for Target –Young and Least valuable who has on average more women than other groups.



On this plot, and there is possibility to choose directly the clusters in our RshinyApp, you can compare the mean of each variable for clusters.

B- Succes/failure of our models:

We have chosen as metric the silhouette average score to determine which model is the best. If you look at well the 3D plot by playing with it, you can see that there are some outliers which should not belong to the cluster according to the 3 variables (probably due to the family size variable). Furthermore, we also tried to do an affinity propagation model, but we failed.

C- Stakeholders' advice:

You can find this text on the “conclusion” slide of our RShiny App. Indeed, it is the deliverable we would give to the company.

As you have been able to see, we have given names to cluster to allow you to quickly pinpoint which kind of cluster it is. We will give you some advice of advertisement.

- Least valuable, Less valuable - Young, Less valuable - old are clusters which have a low spending score. Moreover, comparatively to the other cluster their income is low, particularly the least valuable income. It is not necessary to advertise them a lot if they do not have the income, they could simply take your mail or message as spam and do not come anymore. However, it can be interesting to give voucher per mail or advertise family product to the 'Less valuable - Young' which have bigger family. Playing with the quantity sold to them, and not on the direct margin can be a good move.
- Valuable and most valuable are cluster which are doubled. Each of these clusters exist with young and old people. Each of them has already high spending score, so advertisement to have more buying is not the aim. However, it can be interesting to target the type of people they are, if you do sms/mail campaign. Indeed, Valuable - Old and Most valuable - Young have a higher number of family number. It may be not a good idea to advertise them family promotion, because they would buy anyway. Nevertheless, it can be interesting to propose them family product depending on which product you sell to keep them aware that you sell these products.

- Target - Old and Target young are the most interesting. They have a big income but a low spending score. Following, the other analytic survey we gave you 2 weeks ago in Industrial Organization of your store, you suffer a lot from competition of another store. Here, you do not have to advertise promotion/give voucher or other because they have money to spend. The goal of your SMS/mail campaign will be to emphasize the quality of your product compared to your competitor and the brand name of your shop.

Bibliography:

[1] Shop Customer Data - Kaggle - https://www.kaggle.com/datasets/datascientistanna/customers-dataset

[2] https://www.ibm.com/support/pages/clustering-binary-data-k-means-should-be-avoided