Practical Application 3

Rafael Sojo García

rafael.sojo@alumnos.upm.es

December 2022

1. Introduction

In Machine Learning, we will not allays want to classify or make predictions over our data, sometimes we just want to understand its behavior when certain inputs are grouped together. This groups might not be clear when talking about real world scenarios, and here is where clustering techniques take place.

For this task, the clustering techniques from **Table 1** were used to find which are the main clusters in a marketing campaign dataset.

Methods	
Hierarchical	Single
Hierarchical	Complete
Hierarchical	Ward
Hierarchical	Average
Partitional	Kmeans
Probabilistic	EM

Table 1. Clustering Methods

2. Problem Description

As mentioned in the last paragraph, a marketing campaign dataset was used. This is a dataset from Kaggle that contains customer's information from a certain business. It contains 2240 instances and 29 variables. With the dataset, the following variable description is given [1].

People

- *ID*: Customer's unique identifier
- *Year_Birth*: Customer's birth year
- *Education*: Customer's education level
- *Marital_Status*: Customer's marital status
- *Income*: Customer's yearly household income
- *Kidhome*: Number of children in customer's household
- *Teenhome*: Number of teenagers in customer's household
- *Dt_Customer*: Date of customer's enrollment with the company
- *Recency*: Number of days since customer's last purchase
- *Complain*: 1 if the customer complained in the last 2 years, 0 otherwise

Products

- *MntWines*: Amount spent on wine in last 2 years
- *MntFruits*: Amount spent on fruits in last 2 years
- *MntMeatProducts*: Amount spent on meat in last 2 years
- *MntFishProducts*: Amount spent on fish in last 2 years

- *MntSweetProducts*: Amount spent on sweets in last 2 years
- *MntGoldProds*: Amount spent on gold in last 2 years

Promotion

- *NumDealsPurchases*: Number of purchases made with a discount
- *AcceptedCmp1*: 1 if customer accepted the offer in the 1st campaign, 0 otherwise
- *AcceptedCmp2*: 1 if customer accepted the offer in the 2nd campaign, 0 otherwise
- *AcceptedCmp3*: 1 if customer accepted the offer in the 3rd campaign, 0 otherwise
- *AcceptedCmp4*: 1 if customer accepted the offer in the 4th campaign, 0 otherwise
- *AcceptedCmp5*: 1 if customer accepted the offer in the 5th campaign, 0 otherwise
- *Response*: 1 if customer accepted the offer in the last campaign, 0 otherwise

Place

- *NumWebPurchases*: Number of purchases made through the company's website
- *NumCatalogPurchases*: Number of purchases made using a catalogue
- *NumStorePurchases*: Number of purchases made directly in stores
- *NumWebVisitsMonth*: Number of visits to company's website in the last month

Other

- *Z_CostContact*
- Z_*Revenue*

The main goal for this project, is to understand the different clustering techniques, considering the characteristics of our data, while extracting some information in relation to the interaction of the customers with the business.

3. Methodology

This project was done in python. Here, the preprocessing steps were done using Pandas [2] and Numpy [3], whereas for the application of the different methods and indexes, Yellowbrick [4], Scipy [5] and Scikit-Learn [6]. All the plots were obtained with Matplotlib [7].

For the preprocessing, the following steps were done.

First, there are not null values within the dataset except for the *Income* variable, for which the median of the column was used to fill them.

Second, the variables *Education* and *Marital_Status* are categorical and must label encoded. The first variable, *Education*, contains 5 different categories that were summarized in just 3, **undergraduate, graduate, and postgraduate**, while *Marital_Status* contained 8 that were summarized in 2, **together** if the person has a partner and **single** otherwise.

Third, the age of each person was calculated from *Year_Birth* as well as their number of children from the addition of *Kidhome* and *Teenhome*. Then, the total number of accepted promotions, the total spent, and the total number of purchases was calculated. For the case of accepted promotions, this was done with the addition of the *AcceptedCmp* variables, as for the total spent and purchases, with the addition of product-related and purchase-related variables.

Finally, the following variables were considered unnecessary, therefore deleted, for the customer analysis after the previous preprocessing steps:

- *AcceptedCmp1*, *AcceptedCmp2* and so on, since the number of accepted promotions contains this information.
- *Kidhome* and *Teenhome*, since the number of children is used to sum up this information.
- *Year_Birth* since the age of the customers is used instead.

- *Recency, Response and ID*, which are not relevant for the customer analysis.
- *Dt_Customer*, because we do not have the date of creation of this dataset for calculating the seniority of the customers.
- *Z_CostContact* and Z_*Revenue*, because there is not an explanation or a clear understanding of what are these variables about.

After these treatments, all the variables were standardized, containing the final dataset a total of 20 variables and 2240 instances.

3.1. Models

The number of clusters have been selected for each case using the indexes from **Table 2.**

Silhouette
Calinski Harabasz
Davies Bouldin

Table 2. Silhouette [5] , Calinski-Harabasz [6] and Davies Boulding [7] Indexes.

Now, the configuration of each model is explained.

3.1.1. Hierarchical Clustering

Here, the function "AgglomerativeClustering" from Scikit-Learn was used. The affinity was set to Euclidean, while the linkage parameter to **single**, **complete**, **ward** or **average**, depending on the hierarchical method used.

The Dendrogram of each hierarchy was configured in the same way using the function "dendrogram" from Scipy.

3.1.2. Partitional Clustering

Likewise, the function "KMeans" from Scikit-Learn was used. The rest of the configuration was left by default.

3.1.3. Probabilistic Clustering

With respect to the EM algorithm, the function "GaussianMixture from Scikit-Learn was used, where init_params is set to **kmeans**.

3.2. PCA

To plot the clusters, Principal Component Analysis (PCA) was performed. For which the following projections were obtained.

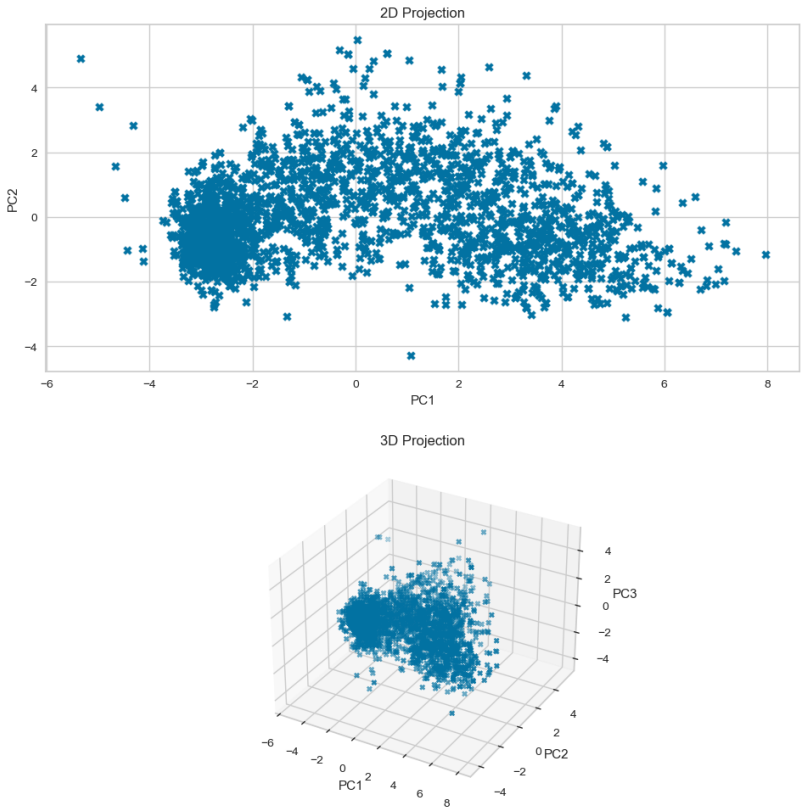


Figure 1. 2D and 3D projections of the Principal Components

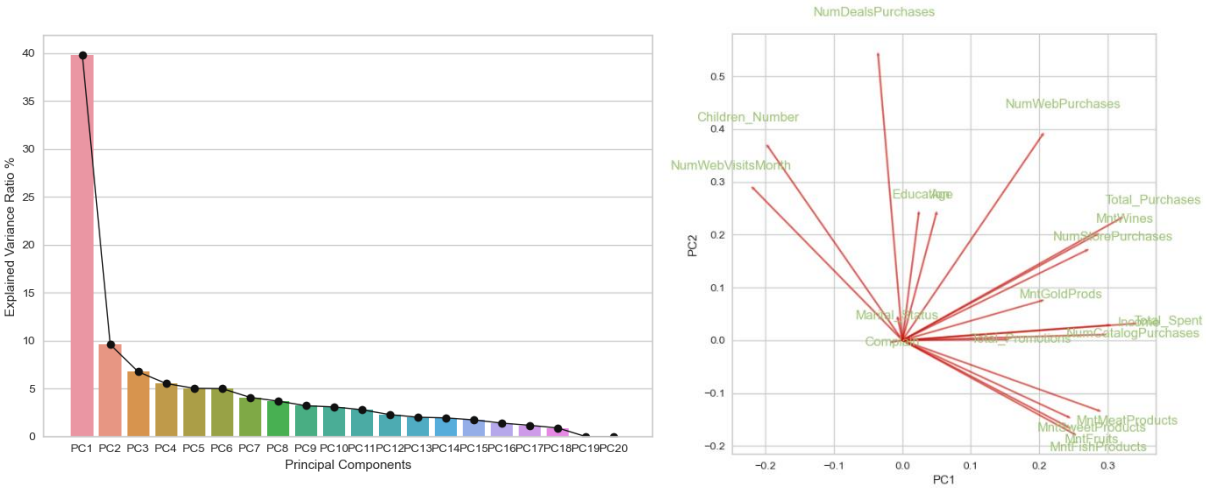
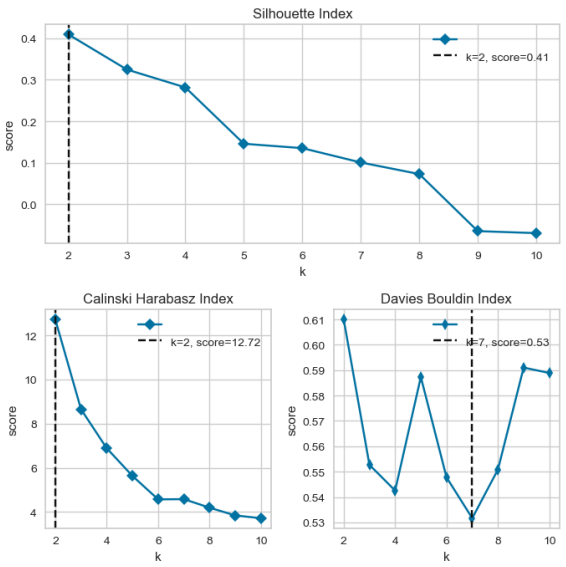


Figure 2. Explained Variance Ratio and Loadings

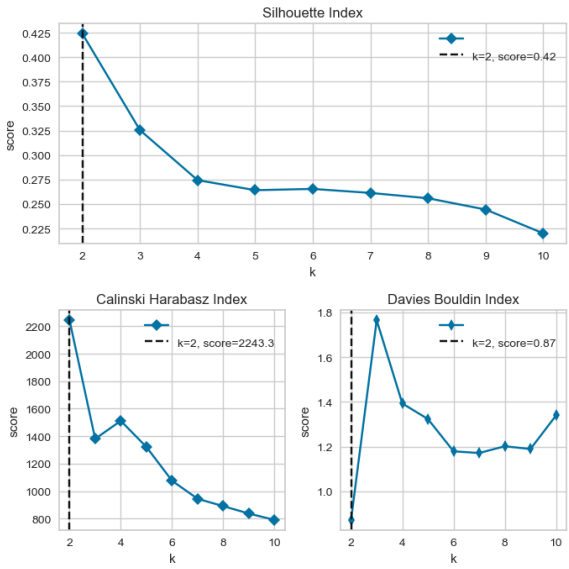
On the other hand, as it is shown in **Figure 2**, almost 50% of the variability present in the dataset is explained between PC1 and PC2, where variables like *Total_Spent* and *Income*, are high related and have an important contribution. Some others like *Marital Status* have almost no influence. Moreover, some interesting information could be extracted from *MntWines*, since it seems to be more related to *Income* and the *Total_Spent*, than the rest of products, what have sense if we consider that wines are usually expensive and more likely in important dinners.

4. Results.

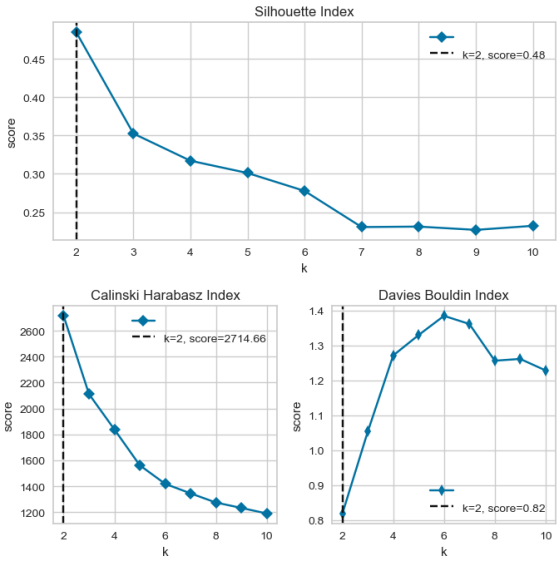
In **figure 3**, the results of the different indexes for the hierarchical methods are shown, where two clusters were the best choice in almost all the cases, except for the Average Linkage where 3 clusters were the best option



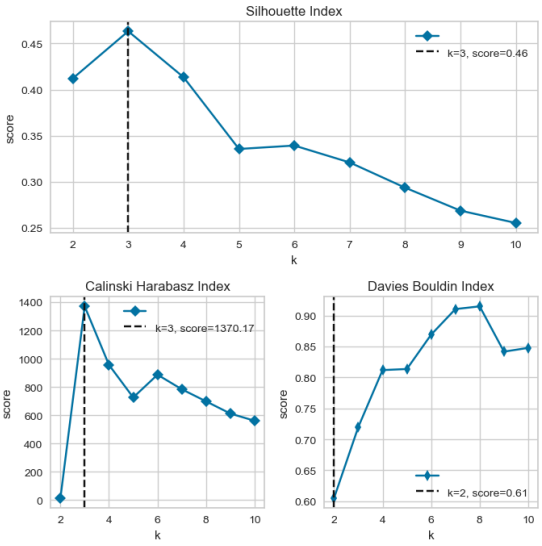
(a) Single Linkage



(b) Complete Linkage



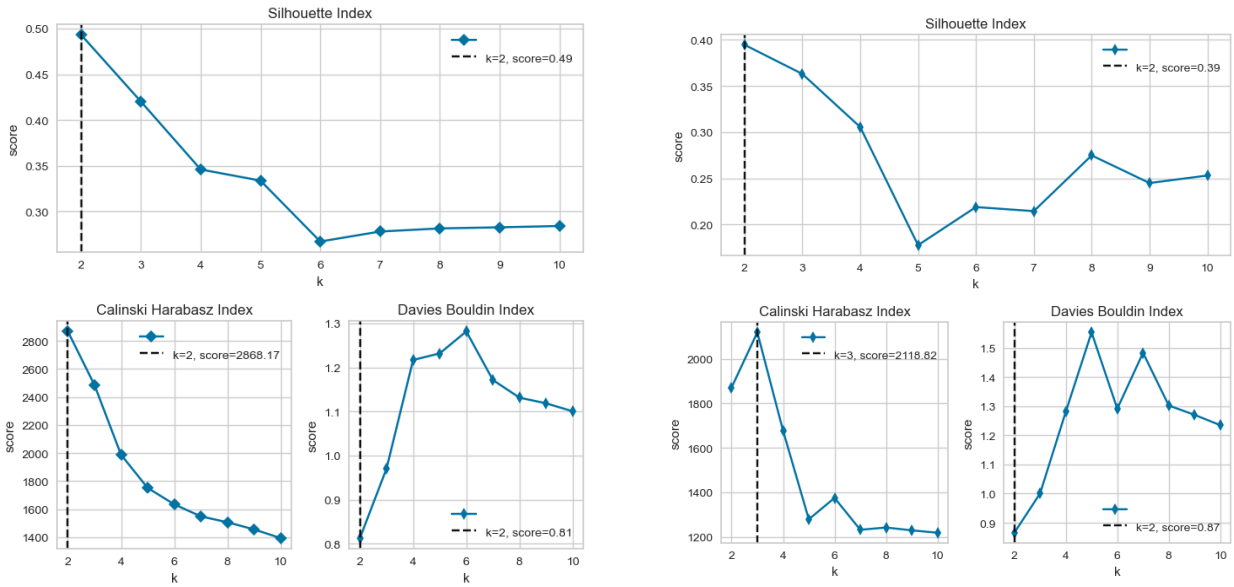
(c) Ward Linkage



(d) Average Linkage

Figure 3. Index Results for all Linkages methods.

Likewise, in **Figure 4,** the results of the different indexes for KMeans and Gaussian Mixture are shown respectively, where two clusters were also the best choice in both.



(a) KMeans (b) Gaussian Mixture

Figure 4. Index Results for KMeans and Gaussian Mixture

5. Discussion

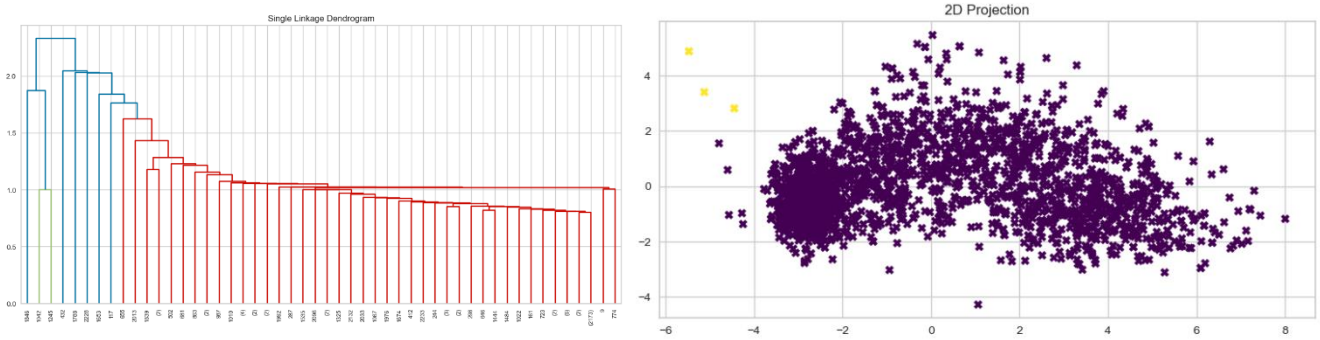


Figure 5. Dendrogram and clusters for Hierarchical Single Linkage

In **Figure 5**, it can be seen the Dendrogram as well as the clusters formed for the Hierarchical Clustering with Single Linkage, where, although from the Dendrogram we can trace a line on the top that divides the two clusters, there is one significantly larger than the other. Besides, in the picture of the right it is even clearer, the yellow cluster has just a few points while the purple occupies the whole rest

This behavior in the cluster formation has a lot to do with the noise. In **Figure 1**, this noise was shown more clearly considering that, neither in the 2D projection nor in the 3D, there is a separation between clusters. In fact, the lack of space between clusters when using Single Linkage is problematic, as this method tend to produce elongated clusters that cannot be properly separated in presence of noise between them, thus, making a chaining effect. Therefore, not being a good choice for this kind of data.

On the contrary, Complete Linkage works better in presence of noise. It produces compact clusters, however, this compactness in the formation of clusters can break down big clusters into

smaller globular ones. For instance, in **Figure 6**, this tendency can be seen, as there are not only 2 clear clusters, but also 4 if we go down in the hierarchy.

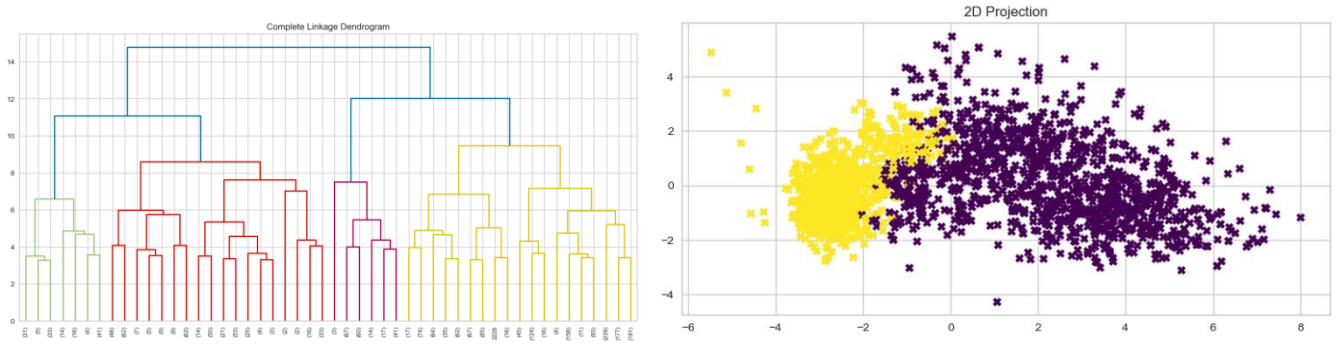


Figure 6. Dendrogram and clusters for Hierarchical Complete Linkage

Nevertheless, this clusters have more sense here than for the single linkage, since the yellow cluster has higher density of points although being smaller in size, whereas the purple cluster is more disperse.

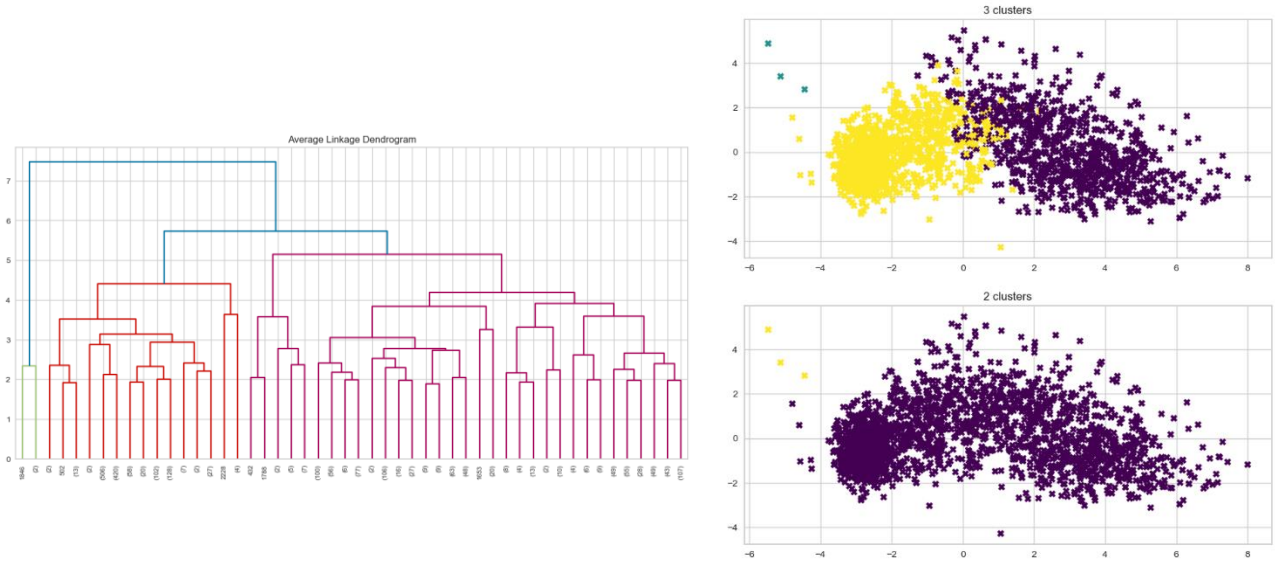


Figure 7. Dendrogram and clusters for Hierarchical Average Linkage

On the other hand, in the Average Linkage case we had 3 clusters as the best option. This method is more robust against noise than Single Linkage but has made a worst job with the outliers than the Complete Linkage. However, it has some sense considering how far are the outliers from the rest. Notice how is big difference in the evaluation of the clusters in **Figure 7** when we change from 3 to 2 clusters. Moreover, this fact can also be seen in the Calinski Harabasz graph from **Figure 3,** since the score for k=2 is almost 0. This problem could be avoided treating the outliers.

With respect to the Ward method, it has achieved the higher scores in the evaluation of the clusters with Hierarchical techniques as it shown in **Figure 3**. Something, that is reflected in the division of the Dendrogram and how are distributed in the plots of **Figure 8**.

Nevertheless, it perhaps has considered as purple cluster more points beyond x=0 that it should have if we look to their density at the 2D plot. Take in consideration that "Ward's method only performs well if an equal number of objects is drawn from each population and that it has difficulties with clusters of unequal diameters" [11]

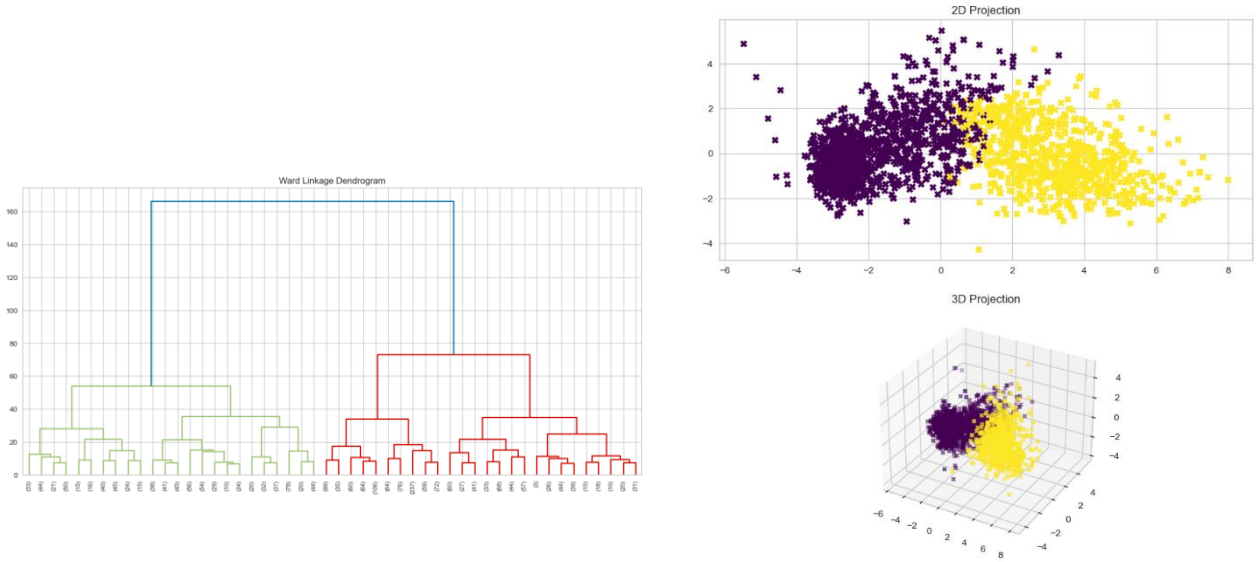


Figure 8. Dendrogram and clusters for Hierarchical Ward Linkage

Now, for the case of Partitional Clustering, it is shown in **Figure 4** that the best Silhouette score among all the models was achieved by KMeans, being just a little higher than Hierarchical Ward Linkage. Here, a vertical slightly inquinated line was traced at x=1 approximately, see **Figure 9.** There is not much else to comment about, the centroids are at (-1.5,1) and (3,0) approximately.

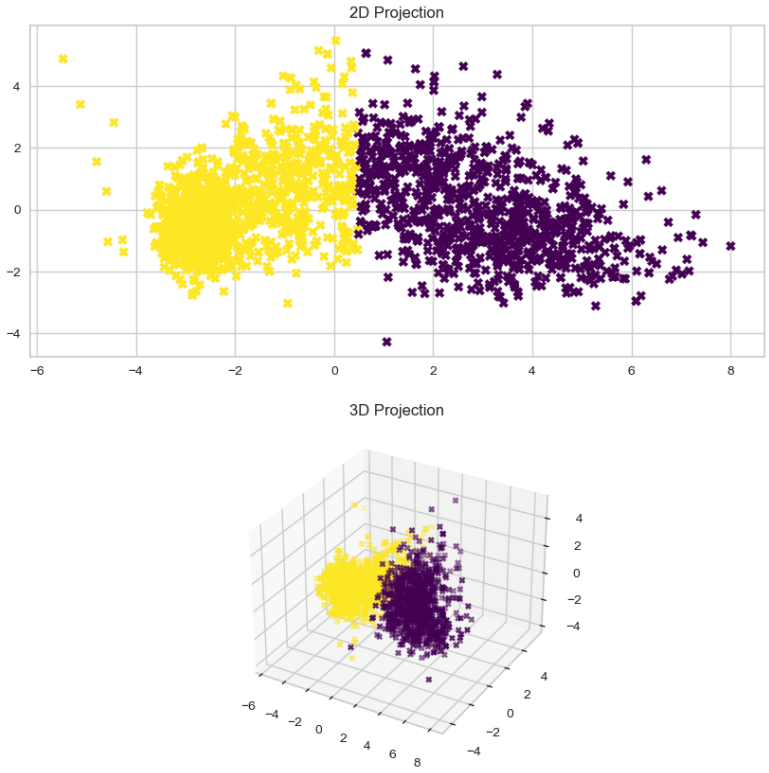


Figure 9. 2D and 3D plots for 2 KMeans clusters

Finally, **Figure 10** shows the clusters for the Probabilistic Clustering using EM technique. Here, we can notice the importance of the high-density area of the left mentioned in the previous paragraphs. Thus, having an enormous weight in the distribution of the clusters as for instance, in the Hierarchical Complete Linkage example.

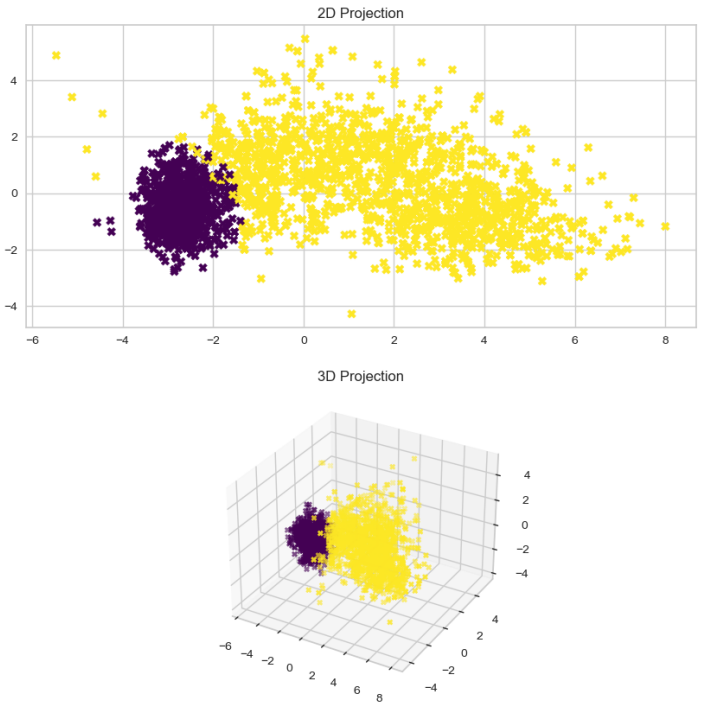


Figure 10. 2D and 3D plots for 2 Gaussian Mixture clusters

In order to "maximize the likelihood of the observed data (distribution) … the EM algorithm attempts to approximate the observed distributions of values based on mixtures of different distributions in different clusters" [12]. Thus, since we have specified 2 clusters, it is reasonable that one of the final clusters had been just the purple area.

Look at what happens in **Figure 11** when we add one more cluster, this purple (green) area stills almost the same and is the yellow one which experiments a division considering its distribution.

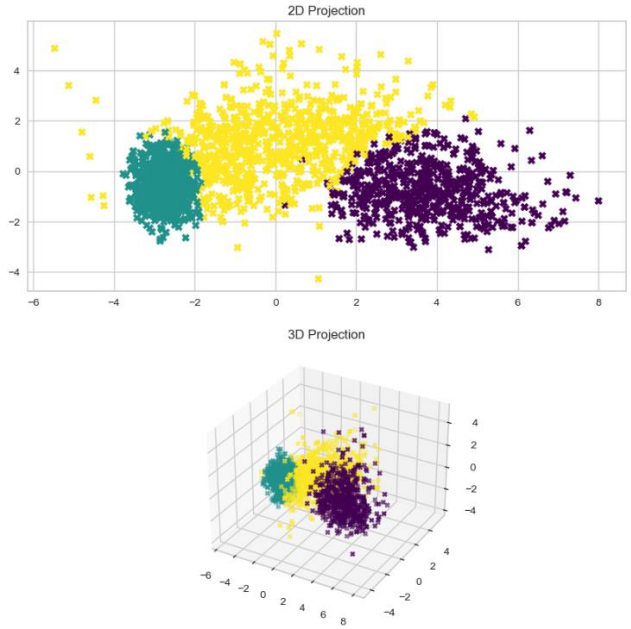


Figure 11. 2D and 3D plots for 3 Gaussian Mixture clusters

6. Conclusion

To conclude this project, we can extract some general knowledge applicable to all the methods. First, it must be mentioned the importance of the noise and the distribution of our data as it could lead to very bad performances on selecting the clusters. Look for instance the case of the Hierarchical Single Linkage. Also, the treatment of the outliers is important since there are some methods like Average Linkage or Complete Linkage, which are more likely to misclassify some of the inputs when are present. In fact, all these considerations are part of the preprocessing steps and data analysis, which, as I have mentioned in all the previous Assignments, are fundamental and sometimes even more important than the configuration of the model itself.

On the other hand, there some are interesting techniques like the EM for the Probabilistic Clustering, which might be very helpful for finding the hidden distributions of the clusters in presence of noise. However, sometimes just the simpler techniques are the ones that performs better, and this time it was the case for KMeans.

References

[1] «Kaggle,» [En línea]. Available: https://www.kaggle.com/datasets/imakash3011/customer-personality-analysis.

[2] «Pandas,» [En línea]. Available: https://pandas.pydata.org/.

[3] «Numpy,» [En línea]. Available: https://numpy.org/.

[4] «Scikit-yb,» [En línea]. Available: https://www.scikit-yb.org/en/latest/index.html.

[5] «Scipy,» [En línea]. Available: https://scipy.org/.

[6] «Scikit-Learn,» [En línea]. Available: https://scikit-learn.org/stable/index.html.

[7] «Matplotlib,» [En línea]. Available: https://matplotlib.org/.

[8] «IEEEXPLORE,» [En línea]. Available: https://ieeexplore.ieee.org/document/9260048.

[9] C. Bielza y P. Larrañaga, «Data-driven Computational Neuroscience Machine Learning and Statistical Models,» p. 458.

[10] «ResearchGate,» [En línea]. Available: https://www.researchgate.net/publication/224377470_A_Cluster_Separation_Measure.

[11] «Finding groups in data: An introduction to cluster analysis.,» 2005, p. 242.

[12] «Rapidminer,» [En línea]. Available: https://docs.rapidminer.com/latest/studio/operators/modeling/segmentation/expectation_maximization_clustering.html.