Machine Learning Engineer Nanodegree
Capstone Project Proposal

CUSTOMER SEGMENTATION FOR MARKETING CAMPAIGN OPTIMIZATION

Autor: Carlos Barros

Project Overview

It is no secret that we live in an increasingly hyper-connected technological era that has changed the way we relate and interact and has come to impact every aspect of our lives. The Internet has broken the communication barrier between cultures, it is possible to connect with a person in another part of the world without having to move to that place.

Business is no stranger to this, as companies can sell their products and/or services in different latitudes. However, this is a double-edged sword because in this market dynamic, competition increases, citing Forbes magazine [1]:

> “[…] In many technology-related industries, competition is intense and can often be the reason why a startup is not able to be profitable. However, competition cuts across the board and can contribute to potential business failure, regardless of the sector."

Because of the above, only half of the small businesses survive beyond five years, between 45.4% and 51%, depending on the year of creation. Beyond that, only one in three small businesses reaches the 10-year mark and lives to tell the tale [1].

Faced with this scenario, marketing has a crucial task and that is to understand how to meet each of the needs of its customers, bearing in mind that they have different tastes, customs, traditions, economic conditions, among other factors. Artificial Intelligence, thanks to the algorithms that have been developed in recent years, has contributed to apply this knowledge to marketing, discovering behavioral patterns, and detecting different homogeneous groups in a large set of customers. In this way, companies can take full advantage of it and anticipate the needs of their customers and thus be able to make them loyal to their brand, which will eventually become a greater economic benefit.

Problem statement

The main objective of this project takes as a starting point those customers who completed the offer by performing a segmentation that allows discovering patterns of purchasing behavior and thus determine what type of offer is the most optimal according to the different groups of segmented customers. This will help Starbucks to focus its marketing campaigns and increase its economic benefit.

Metrics

Since different unsupervised clustering models will be implemented the metrics to be used are:

1. **Elbow method:** used to determine the optimal number of clusters in k-means clusters by plotting the value of the cost function produced by different values of k.
2. **Silhouette value:** Measures how similar a point is to its own cluster (cohesion) compared to other clusters (separation).

   It is important to mention that it is composed of two different elements [3]:

   - The mean distance between a sample and all other points in the same class (a).
   - The average distance between a sample and all other points in the nearest cluster (b).

   The formula for this coefficient s is defined as follows:

$$s = \frac{b - a}{\max\left(a, b\right)}$$

Source O'Reilly [3]

3. **Davies Bouldin metric:** Average similarity measure of each cluster to its most similar cluster, where similarity is the ratio of distances within the cluster to distances between clusters. The minimum score is zero, and lower values indicate better clustering.

$$D_{ij} = \frac{(\bar{d}_i + \bar{d}_j)}{d_{ij}},$$

Source G. Drakos [4]

Where:

- D_{ij} is the "inter-group distance ratio" for groups i and j.
- $\bar{d}_{\iota}$ is the average distance between each data point in group i and its centroid, similar for $\bar{d}_{j}$. D_{ij} is the Euclidean distance between the centroids of the two groups.

If two clusters are close together (d_{ij} small) but have a large dispersion ($\bar{d}_{\iota}$ + $\bar{d}_{j}$large), then this ratio will be large, indicating that these clusters are not very distinct [4].

4. **Hopkins Statistics:** Tests the spatial randomness of the data and indicates the clustering tendency or how well the data may be clustered.
    - **0.5 or less**: data are uniformly distributed.
    - **0.77 - 0.99:** high clustering tendency.

ANALYSIS

EDA (Exploratory Data Analysis)

The data is contained in three files:

1. **portfolio.json** - containing offer ids and meta data about each offer (duration, type, etc.)
2. **profile.json -** demographic data for each customer
3. **transcript.json** - records for transactions, offers received, offers viewed, and offers completed

Here is the schema and explanation of each variable in the files:

portfolio.json

- id (string) - offer id
- offer_type (string) - type of offer ie BOGO, discount, informational
- difficulty (int) - minimum required spend to complete an offer.
- reward (int) - reward given for completing an offer.
- duration (int) - time for offer to be open, in days.
- channels (list of strings)

profile.json

- age (int) - age of the customer
- became_member_on (int) - date when customer created an app account.
- gender (str) - gender of the customer (note some entries contain 'O' for other rather than M or F)
- id (str) - customer id
- income (float) - customer's income

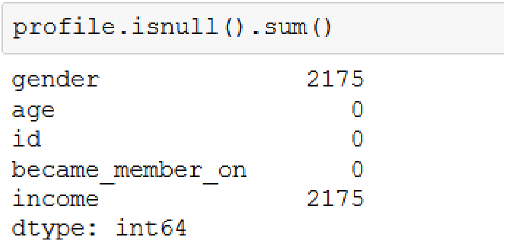
transcript.json

- event (str) - record description (ie transaction, offer received, offer viewed, etc.)
- person (str) - customer id
- time (int) - time in hours since start of test. The data begins at time t=0
- value - (dict of strings) - either an offer id or transaction amount depending on the record

However, it is important to specify that the size of the data sets are as follows:

- Dataset ***portfolio*** have 10 rows and 6 columns.
- Dataset ***profile*** have 17000 rows and 5 columns.
- Dataset ***transcript*** have 306534 rows and 4 columns.

In addition to the above, it was verified that the profile dataset had a total of 2175 null values in the gender and income columns, as shown below:



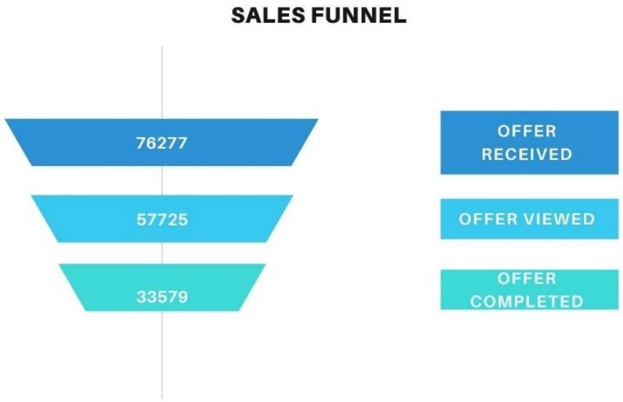
```
profile.isnull().sum()
```

```
gender              2175
age                    0
id                     0
became_member_on       0
income              2175
dtype: int64
```

Source: Own image

The following image shows the funnel of sales conversions of the marketing campaign. In the upper part are the offers received with a total of **76277**, in the middle part the offers viewed with **57725**, representing **76%** of the total offers received.

Finally, the conversion of offers completed was **33579** which is **44%** of the offers sent.

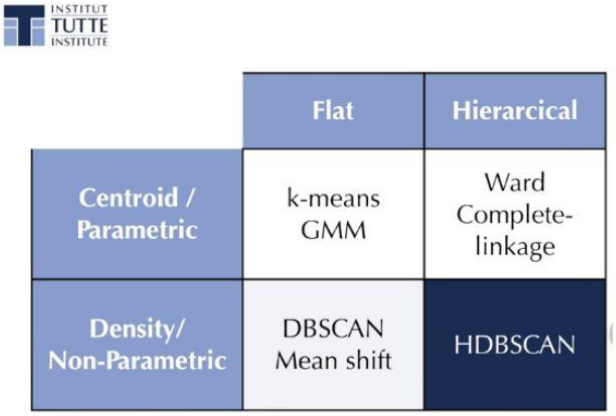


Source: Own image

Algorithms and Techniques

Since the final dataset to implement the model had several columns, the **PCA** application was implemented to explain at least 80% of the variance resulting in the construction of 5 principal components, which was trained in the AWS environment and stored in an S3 bucket.

Before going into details about the algorithms, it is important to mention what is meant by cluster and it is nothing more than a way of defining a group that is usually consistent with our intuitive notion of groups is: very dense regions separated by sparse regions [3]. Now, based on the research of *Tutte Institute for Mathematics and Computing* and the development of Leland McInnes as one of the developers of the **HDBSCAN** library. It can be summarized in the following image:


INSTITUT
TUTTE
INSTITUTE

	Flat	Hierarcical
Centroid / Parametric	k-means GMM	Ward Complete-linkage
Density/ Non-Parametric	DBSCAN Mean shift	HDBSCAN

Source: J. Healy [5]

In this way and according to [6], it can be stated that **K-means** works best when the clusters are:

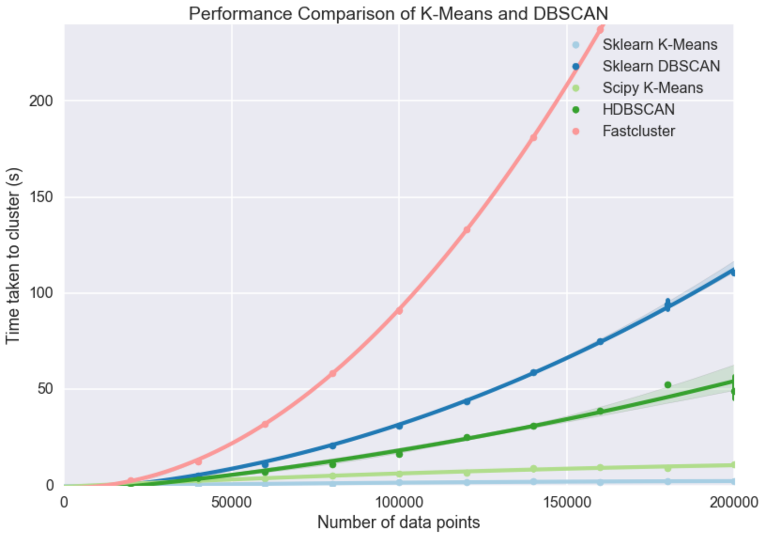
- "round" or spherical
- of equal size
- equally dense
- denser in the center of the sphere
- not contaminated by noise / outliers

Further elaborating on the above, the general characteristic of the data is:

- Clusters with arbitrary shapes
- Clusters of different sizes
- Clusters with different densities
- Some noise and maybe some outliers

The data set of the present project does not present this characteristic so **K-means** would not be a good choice. Therefore, the model was performed with **HDBSCAN** because as observed in the image it uses a density-based approach that makes few implicit assumptions about the clusters. It is a non-parametric method that looks for a hierarchy of clusters formed by the multivariate modes of the underlying distribution. Instead of looking for clusters with a particular shape, it looks for regions of data that are denser than the surrounding space [6].

Thus, another advantage observed **HDBSCAN** is its speed in processing the data which is summarized in the following graph [4]:



Source: hdbscan documentation [7]

NOTE: For more information see HDBSCAN's Github repository.

Therefore, the algorithm chosen to perform the training was **HDBSCAN**, however, it was also performed with **K-means** to check the differences that in the first place the optimal number of clusters applying the elbow metric was 9 over the 3 clusters generated by **HDBSCAN**.

Benchmark model

The benchmark model for the present project was developed by several researchers from the Sardar Patel Institute of Technology [2] in which they propose a systematic approach that combines clustering and recommender systems offering an improved and more novel approach to customer segmentation that will be of greater benefit to companies.

It is important to note that the authors use different metrics to those proposed for this project in choosing the optimal number of clusters such as Akaike and Bayes Information Criterion, Renyi and

Shannon Entropies, so these factors will be considered when making a comparison between the different models developed in this project.

METHODOLOGY

Data Preprocessing

The first process implemented was the normalization of the 3 datasets provided. From there, 19 variables were organized that were considered important and were taken into account in the implementation of the algorithm which are:

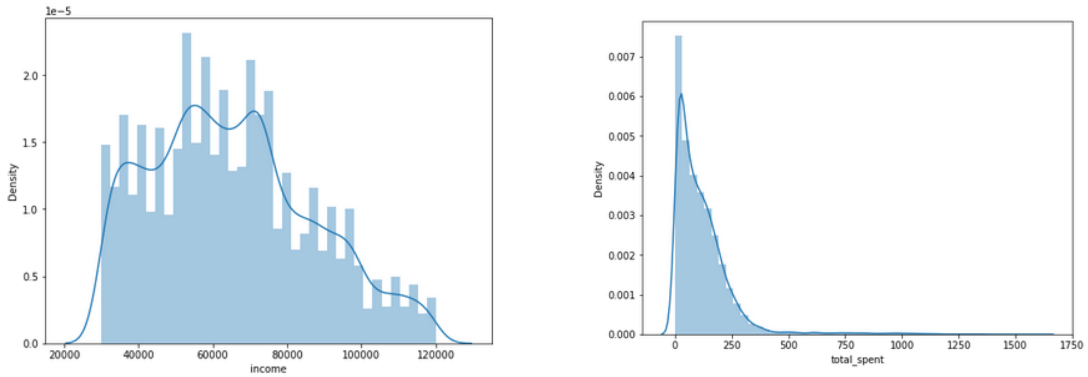
1. **user_id**: The unique customer identifier
2. **age_range**: The age range the customer falls into
3. **gender**: The gender of the customer, either male (M), female (F), or other (O)
4. **income**: How much money the customer makes each year
5. **total_completed**: The total number of offers actually completed by the customer
6. **total_received**: The total number of offers that Starbucks sent to the customer
7. **total_viewed**: The total number of offers that the customer viewed
8. **total_spent**: The total amount of money spent by the customer across all transactions
9. **avg_spent**: The mean average amount of money spent by the customer across all transactions
10. **completed_bogo**: The number of completed BOGO offers by the customer
11. **completed_discount**: The number of completed discount offers by the customer
12. **discount_percent_completed**: The ratio of how many discount offers were actually completed by the customer as compared to how many Starbucks sent them
13. **num_transactions_range:** Range the total amount of individual monetary transactions performed by the customer
14. **bogos_received_range:** Range of number of BOGOs received.
15. **discounts_received_range:** Range of number of discounts received
16. **percent_viewed_range:** Percentage range of offers viewed
17. **percent_completed_range:** Percentage range of completed offers
18. **discount_percent_completed_range**: Range the ratio of how many discount offers were actually completed by the customer as compared to how many Starbucks sent them
19. **bogo_percent_completed_range:** Range the ratio of how many BOGO offers were actually completed by the customer as compared to how many Starbucks sent them

Following the standardization process, the main conclusions are as follows:

- the majority of the population is **male** with 57.2% followed by **females** with 41.4% and finally **Others** with only 1.4% of our data set.
- 60% of the population is in the age range **40 - 69 years old**, belonging to the Baby boomers generation. On the other hand, Millenials represent only **20%** of the population.
- the total population more than **50%** receive between 4 and 5 offers, of which **28%** manage to see 3 and only complete between 2 and 3 offers. Clearly this represents a *funnel* as specified

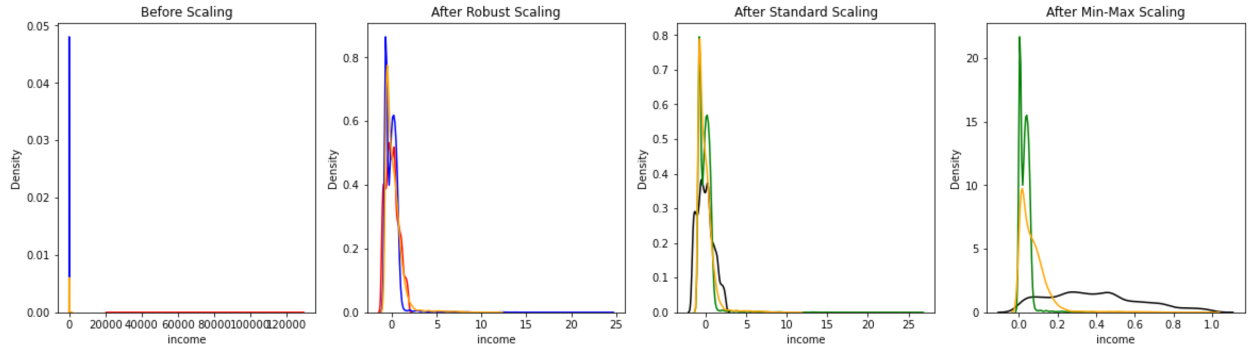
above. The company needs to send many offers in order to have a conversion rate and thus have a return on investment for its marketing campaign.

In addition to the above, we also considered continuous numerical data such as income, total expenditure and average expenditure, which, as can be seen in the following image, has a bias in its distribution and it is important to be able to manage it in a single range:



Source: Own image

Therefore, a comparative analysis was performed with different rescalers that allowed a best-case decision to be made with respect to the data set on which we worked. The three scalers are: **Robust Scaler**, **Standard Scaler** and **Min Max Scaler**. The results are shown in the following image:



Source: Own image

First to do the pre-processing we applied RobustScaler because it removes outliers from the data. However, for the columns we are interested in, not all of them are organized in the same range because *time* is in the range of -1 to 1, but *income* is in -2 and 2. What we want is that all the specified columns are in the same range, therefore, applying MinMaxScaler will achieve this since all the columns are in the range of 0 to 1.

In conclusion, first **RobustScaler** and then **MinMaxScaler** will be applied. Finally, coding of the categorical variables will be applied using **LabelEncoder**.

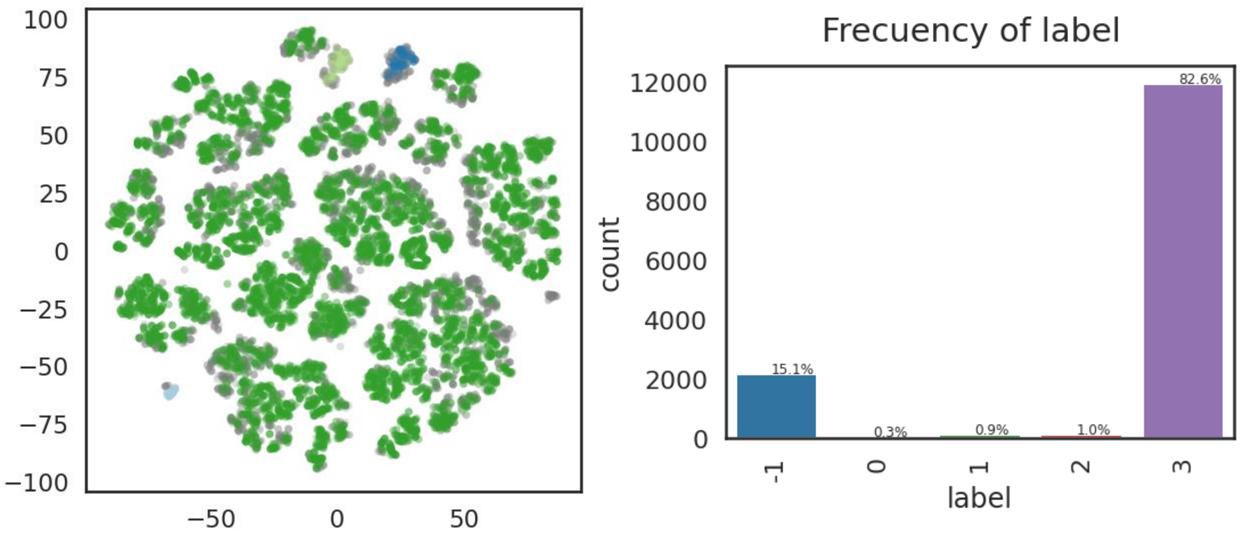
Implementation

As defined in the Algorithms and Techniques section, the algorithm chosen was HDBSCAN, it is important to mention some aspects to consider in the latter and they are its parameters. While there are many parameters in the implementation of this algorithm, there are two main ones to keep in mind: *min_samples* and *min_cluster_size*.

- **min_samples:** Provides a measure of how conservative you want your clustering to be. The larger the min_samples value you provide, the more conservative the clustering will be more points will be declared as noise, and the clusters will be restricted to progressively denser areas.[6]
- **min_cluster_size:** tells the maximum size of a "bump" before it is considered a peak. By increasing the value of min_cluster_size. In a way, it is smoothing the estimated PDF so that the true peaks of the distributions become prominent.

Thus, to define the two parameters mentioned above, the algorithm was firstly trained with the scaled data, and secondly with the data applying PCA. The results were as follows:

WITHOUT PCA



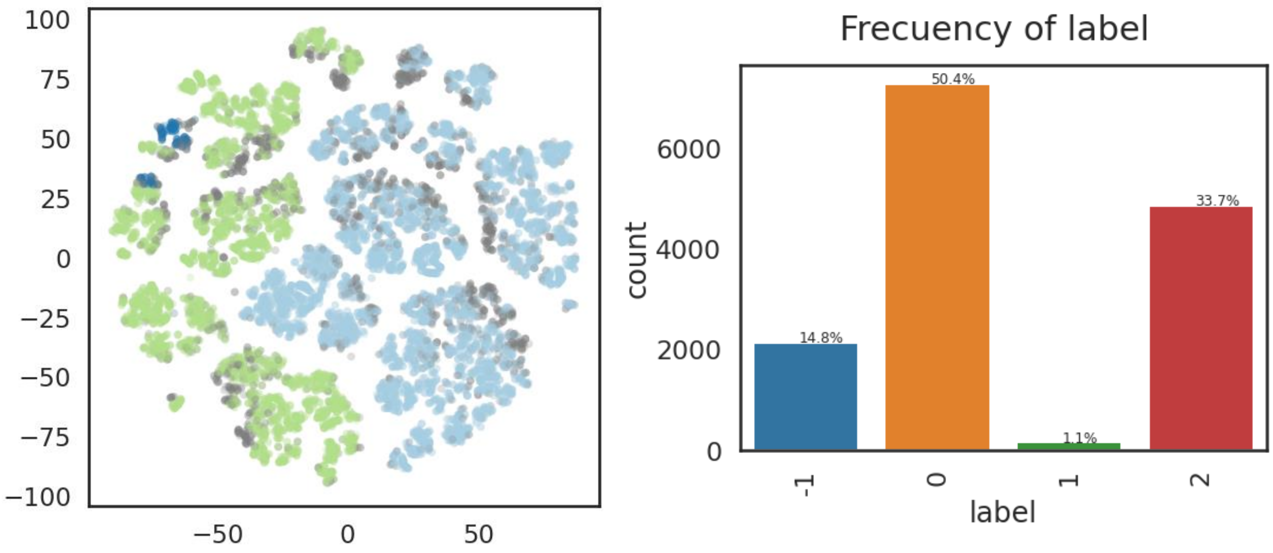
Source: Own image

It is important to mention that the black dots inside the graph represent the noise detected by the **HDBSCAN** algorithm, and that the label will be identified with the value **-1.** In the meantime, it can be observed that the green cluster represents a large percentage of the data, as shown in the following image.

As can be seen, cluster 3 has a distribution of *82.6%* and the data that generated noise was *15.1%.* Clearly the algorithm did not perform a good segmentation because the data has not been applied PCA.

NOTE: After several trials with the parameters, it was defined that the best one was min_cluster_size=35and min_samples=10 as it groups the data better and generates less noise.

WITH PCA



Source: Own image

With the previous image it can be concluded that the **HDBSCAN** algorithm performs a better segmentation with the data that have been applied *PCA* since it represents a better redistribution of the clusters. The data that generated noise will be left within the final dataset that contains each of these clusters for an analysis in the visualization of the results.

Refinement

To refine the model for better results, other parameters of the HDBSCAN algorithm were tested, such as *cluster_selection_epsilon*, *"that it depends on the given distances between your data points".* For example, set the value to 0.5 if you don't want to separate clusters that are less than 0.5 units apart. This will basically extract DBSCAN* clusters for epsilon = 0.5 from the condensed cluster tree but leave HDBSCAN* clusters that emerged at distances greater than 0.5 untouched. [7].

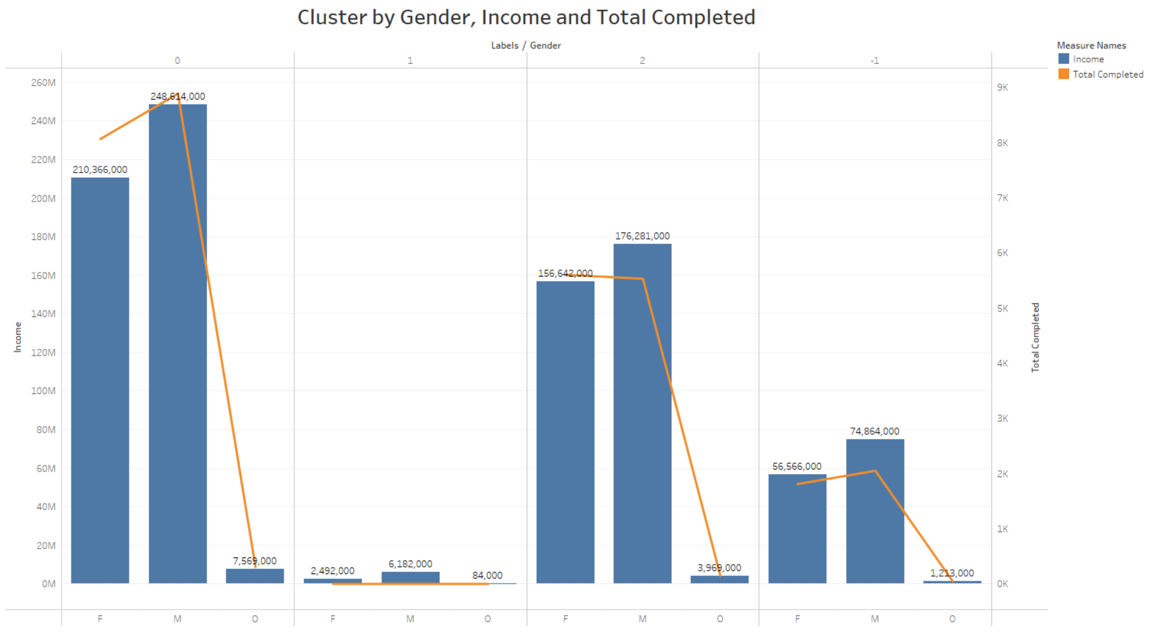
Although several ranges were tested for this parameter, no improvement was found in the segmentation performed by the algorithm. It is not contributing to the performance of the clustering; therefore, it was not placed as a parameter and was left by default.

RESULTS

Final analysis and Evaluation

The main results with their respective conclusions of the segmentation applied in this project are presented below. Keep in mind that cluster -1 represents the noise detected by the ***HDBSCAN*** algorithm, for analysis purposes the graphs will be left.

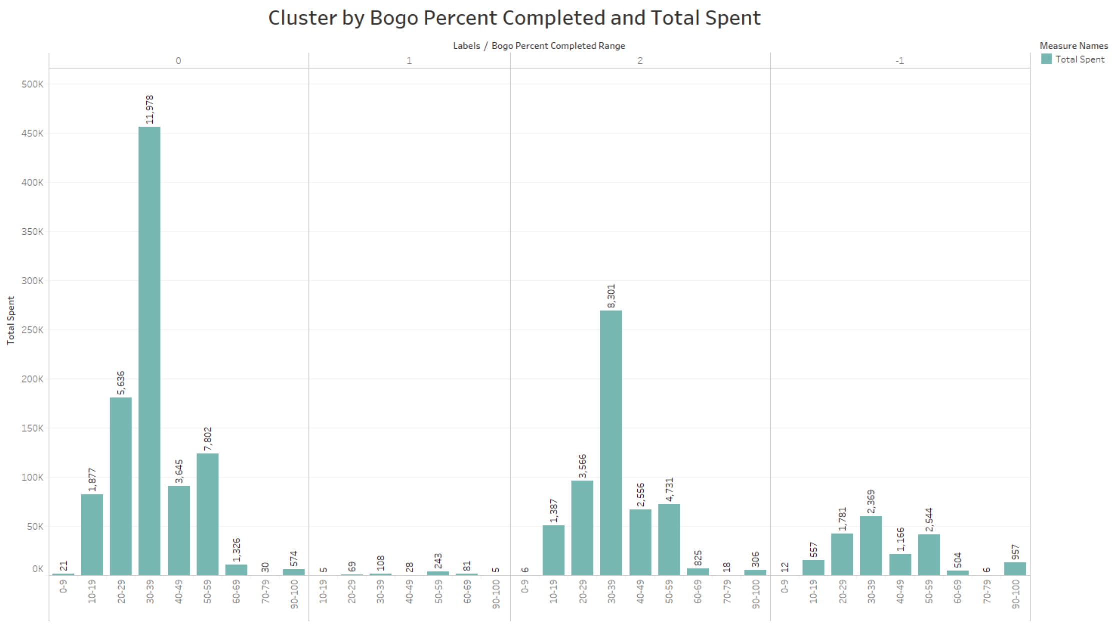
The following image shows the cluster classification by gender, income and total offers completed, where men are the ones who report more income in each segmented group. There is a linear relationship between clusters **0, 1** and -**1,** since the higher the economic income, the more offers will be completed.



Source: Own image

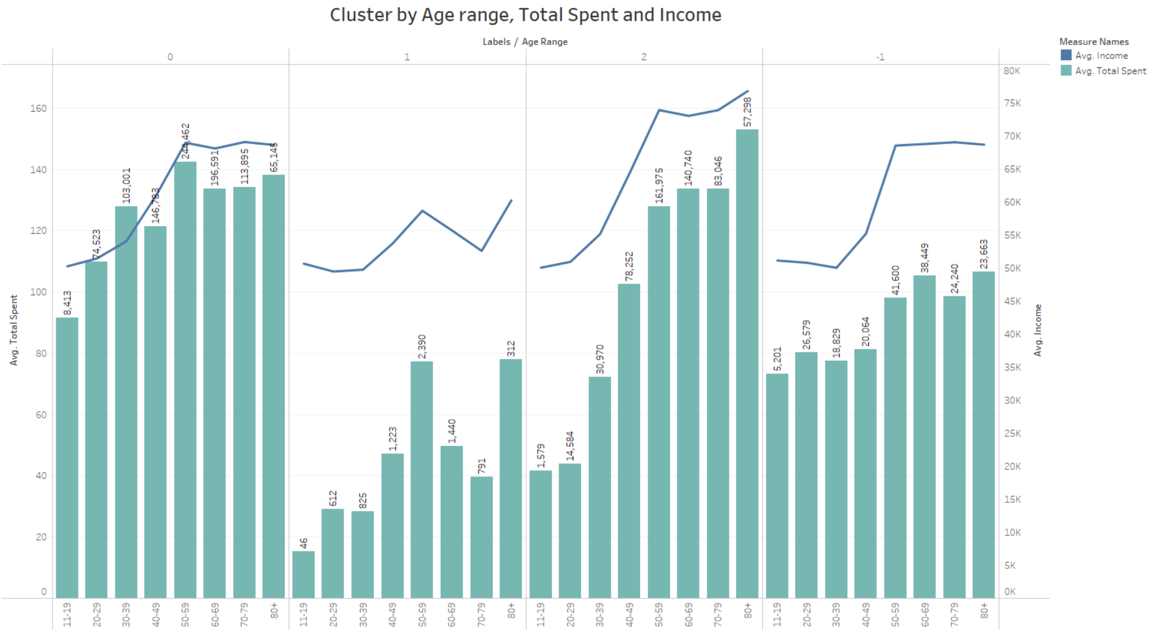
However, **cluster 2** presents a different case because although women receive less income, they are the ones who complete more offers compared to men.

Another aspect to highlight is the **BOGO offers completed** with the **Total spent** segmented by **age groups**. There is a pattern of behavior for the **30-39 age** group since, as can be seen in the following image in each cluster, they are the ones who spend the most, followed by the **20-29** age group:



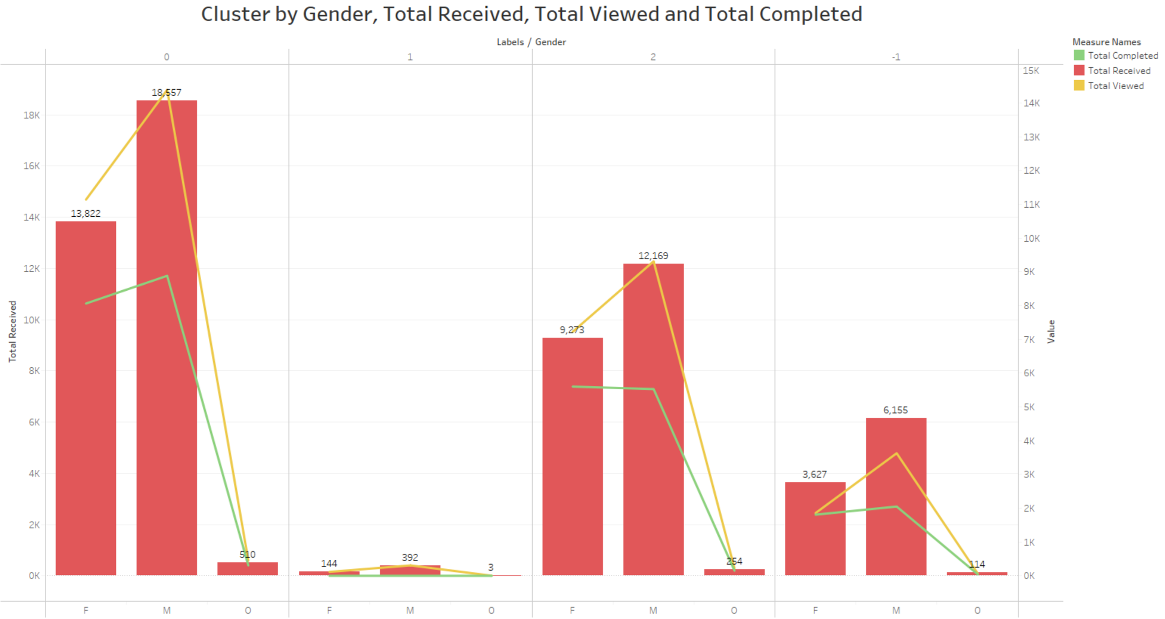
Source: Own image

Meanwhile, another factor that can be analyzed is comparing the **Total Spent** with the Income for each age group, where a growth curve in Income can be observed as the person gets older, which makes sense. For example, in ***Cluster 0*** people aged **50-59** spend on **average $244,462** versus an average income of ***$68,973*** the second largest of this group. However, the income curve is steeper in *cluster 2* as is the average income.



Source: Own image

Now, it is important to analyze the behavior of the clusters in terms of offers received versus offers seen. The first conclusion that can be made is that men are the gender that receives the most offers.



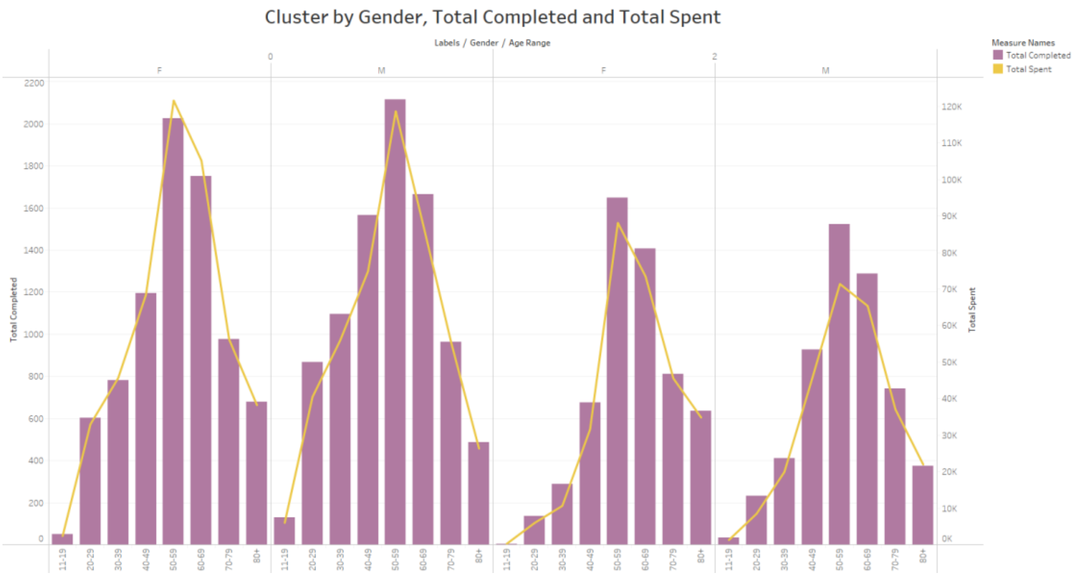
Source: Own image

The proportions in each of the clusters are presented below:

- Cluster 0
    - Women received 13,882 offers of which they viewed 11,141 and completed 8,064, i.e. a conversion rate of 58%.
    - Men received 18,577 offers, of which they viewed 14,391 completing only 8,885 with a conversion of 47.8%.
- Cluster 1
    - Although offers were sent and people received them, according to the graph, there was no conversion rate because people did not complete them.
- Cluster 2
    - Women received 9,273 offers of which they viewed 7,227 and completed 5,603, i.e. a conversion rate of 60.4%.
    - Men received 12,169 offers, of which they viewed 9,314 completing only 5,530 with a conversion of 45.4%.
- Cluster -1
    - Females received 3,627 offers of which they viewed 1,857 and completed 1,810, i.e., a conversion of 50%. Note that the number of offers viewed and completed is very close, which is not observed in the previous clusters.

- Men received 6,155 offers, of which they viewed 3,630 completing only 2,051 with a conversion rate of 33.3%.

Finally, the total number of offers completed was analyzed with the total spent by gender and age group where only women and men were filtered out, in addition, clusters 1 and -1 were not considered since they were not contributing to the analysis, the results were as follows:



Source: Own image

It can be observed that as mentioned above, the **50-59 age** group is the one that completes the most offers, and it is also important to point out that women are the ones who spend more compared to men, even though in some cases they have completed fewer offers, i.e., the greater the number of offers completed does not necessarily mean that more is spent. The latter is since there are different types of offers with different values.

CONCLUSIONS

1. If the company's objective is to strengthen BOGO offers, it should focus on the **30-39 age group**, since the results show that these are the ones who spend the most on this type of offer, followed by the **20–29-year-olds.**
2. The conversion funnel in each of the clusters for the **female** gender are the ones with the highest conversion rate in terms of offers sent and offers viewed. This means that women are good customers for the types of offers for the marketing campaign.
3. The age group where they spend the most on the offers is between **50-59 years old**, predominantly female. Even though on some occasions they present less completed offers.

REFERENCES

[1] Otar, C., 2018. *What Percentage Of Small Businesses Fail -- And How Can You Avoid Being One Of Them?*. [online] Forbes. Available at: <https://www.forbes.com/sites/forbesfinancecouncil/2018/10/25/what-percentage-of-small-businesses-fail-and-how-can-you-avoid-being-one-of-them/?sh=2cf9ecba43b5>

[2] K. Bhade, V. Gulalkari, N. Harwani and S. Dhage, "A Systematic Approach to Customer Segmentation and Buyer Targeting for Profit Maximization", *Ieeexplore.ieee.org*, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8494019.

[3] "Machine Learning for Developers", *O'Reilly Online Learning*. [Online]. Available: https://www.oreilly.com/library/view/machine-learning-for/9781786469878/5ebb37f2-b0b3-4414-9f28-1d10854e64ab.xhtml

[4] G. Drakos, "Silhouette Analysis vs Elbow Method vs Davies-Bouldin Index: Selecting the optimal number of clusters for KMeans clustering", *GDCoder*, 2020. [Online]. Available: https://gdcoder.com/silhouette-analysis-vs-elbow-method-vs-davies-bouldin-index-selecting-the-optimal-number-of-clusters-for-kmeans-clustering

[5] J. Healy, "HDBSCAN, Fast Density Based Clustering, the How and the Why - John Healy", *Youtube.com*, 2019. [Online]. Available: https://www.youtube.com/watch?v=dGsxd67IFiU&t=49s

[6] P. Berba, "Understanding HDBSCAN and Density-Based Clustering", Medium, 2020. [Online]. Available: https://towardsdatascience.com/understanding-hdbscan-and-density-based-clustering-121dbee1320e

[7] h. docs, "Benchmarking Performance and Scaling of Python Clustering Algorithms — hdbscan 0.8.1 documentation", Hdbscan.readthedocs.io. [Online]. Available: https://hdbscan.readthedocs.io/en/latest/performance_and_scalability.html