

**Data Mining Project**

**MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS**

**Customer Segmentation**

**

Débora Santos, number: 20200748

Diana Furtado, number: 20200590

January, 2022

INDEX

[1. Introduction iii](#_Toc92417364)

[2. Business and data understanding iii](#_Toc92417365)

[3. Data preparation iv](#_Toc92417366)

[3.1. Feature engineering iv](#_Toc92417367)

[3.2. Correlation iv](#_Toc92417368)

[3.3. Outliers and missing data v](#_Toc92417369)

[3.4. Data standardization vii](#_Toc92417370)

[4. Clustering viii](#_Toc92417371)

[4.1. K-means viii](#_Toc92417372)

[4.1.1. Evaluation and visualization ix](#_Toc92417373)

[4.2. Hierarchical clustering x](#_Toc92417374)

[4.2.1. Evaluation and visualization x](#_Toc92417375)

[5. Results xi](#_Toc92417376)

[6. Conclusions and marketing approach xii](#_Toc92417377)

[7. References xiii](#_Toc92417378)

[8. Appendix xiv](#_Toc92417379)

[8.1. Appendix I – Features description xiv](#_Toc92417380)

# Introduction

DD Consulting was hired by an insurance company, based in Portugal, to develop a Customer Segmentation to provide the Marketing Department with the best understanding of its customers’ profile, so they can better build and adjust the company’s strategy.

The main goal of this report is to provide the client with useful information on the types of clients buying the insurances, by organizing them in clusters according to their own characteristics (demographic, commercial, etc).

The present report is submitted with a Jupyter Notebook, and it goes through the whole process performed on this Customer Segmentation, showing the reasoning and decisions behind each step of the process, from data cleaning and preprocessing to the clusters’ definition and interpretation. The last section of the report shows the marketing approach recommended for each of the clusters defined.

A GitHub repository can be accessed through the following link:

<https://github.com/dianafurtado/DM_Project2021_GroupAD.git>

The following deliverables will be submitted:

* The present report
* A jupyter notebook with detailed code and some explanations of our decisions
* A pandas profiling file with the data detailed.

# Business and data understanding

This project was made following the CRISP-DM reference model (Cross Industry Standard Process for  
Data Mining). CRISP-DM is a standard process built in the end of the 90’s and it was built by more than 200 members lead by a consortium of big companies. CRISP-DM succeeds because it is soundly based on the practical, real-world experience of how people conduct data mining projects.[1].

## Background

The insurance company operates in Portugal and with the present customer segmentation the marketing department expects to get a good understanding of the different types of clients they have.

Using the knowledge on the current customers, the company would like to improve customer retention and loyalty’s strategy, and therefore they have reached DD Consulting and Services.

## Business Objectives

The customer’s objectives are understanding the following:

* The key characteristics that best distinguish the customers.
* Which and how many customer segments there are in the provided database.
* Improve the interaction with the customers by creating new marketing strategies.

## Data Understanding

Understanding the data available is a key step to avoid unexpected problems during the data preparation phase. At this stage DD Consulting had a deep-dive into the dataset provided (*a2z\_insurance.sas7bdat*), to understand its potential and limitations. Pandas profiling was used to have an overview of the dataset, its composition, features’ distribution, presence of outliers, missing and duplicated values.

The dataset contains the insurance company’s customers information, from 2016. It has 10.296 observations and 14 features, from which 4 categorical (*CustID ,EducDeg*, *GeoLivArea* and *Children*) and 10 numerical (*FirstPolYear*, *BirthYear*, *MonthSal*, *CustMonVal*, *ClaimsRate*, *PremMotor*, *PremHousehold*, *PremHealth*, *PremLife* and *PremWork*). To better understand the variables meaning and relevance in the present assessment context, we also analyzed the metadata file made available by the client.

At this stage we checked for missing values in the dataset (in which the cells filled in with an empty space were also included) and verified that 10 features have missing values, being these missing values more representative in *PremLife* variable (1,01%), followed by *PremWork* (0,84%).

There are, however, no duplicated observations in the insurance company’s customers dataset.

Still as part of the data exploration, we have looked into the categorical variables’ frequency bar plots and the numerical variables histograms, density plots and boxplots – to check for the variables’ distribution and outliers.

# Data preparation

This stage defines the quality of the result, for this reason, we have invested a big proportion of the resources in data preparation.

Firstly, we have set the *CustID* to be the Index and dropped the three categorical variables as, to run clustering solutions, only numerical variables are relevant.

We have then performed feature engineering, checked the variables’ correlation, dealt with the outliers, missing data, checked the variables’ correlation again and normalized the data, which processes we go through in the subsections below.

## Feature engineering

Two variables were created from the existing ones, taking 2016 as the reference year:

* *Seniority*: Meaning client seniority, calculated by the difference between the reference year (2016) and the *FirstPolYear*;
* *Age*: Representing the customer’s age, the difference between the reference year (2016) and the *BirthYear*.

The two existing variables (*FirstPolYear* and *BirthYear*) were replaced by the newly created ones, which we believe are more meaningful for the present analysis.

We also investigated variables with low variance (under 10%) as they would not improve the model’s performance and concluded there weren’t any.

## Correlation

To perform clustering is useless and harmful to keep variables that are either highly or lowly correlated with each other. In the first case because it leads to redundancy of the data, the latter because lowly correlated columns will have the same value among all the clusters and add no information to our cluster analysis.

At this stage, as we were working only with numerical variables, we have looked at the Pearson correlation, shown in the figure below.

A screenshot of a computer

Description automatically generated with low confidence

Figure 3.1 - Pearson correlation heatmap.

Considering the strong correlation observed between *CustMonVal* (Customer’s value, since they have contracted the insurance company’s services) and *ClaimsRate* (Amount paid by the insurance company divided by the premiums, in the last two years, counting from the reference year), and to avoid redundancy in the data, as described above, we decided to drop *ClaimsRate*.

## Outliers and missing data

The presence of missing values reduces the data available to be analyzed, compromising the statistical power of the study, and eventually the reliability of its results. In addition, it causes a significant bias in the results and degrades the efficiency of the data. Outliers significantly affect the process of estimating statistics (e.g., the average and standard deviation of a sample), resulting in overestimated or underestimated values. [2]

Before starting to work on the outliers, we have temporarily filled the missing values with the median and normalized the data.

For outliers’ analysis, we had looked at the boxplots for the 9 variables, shown in Figure 3.2.

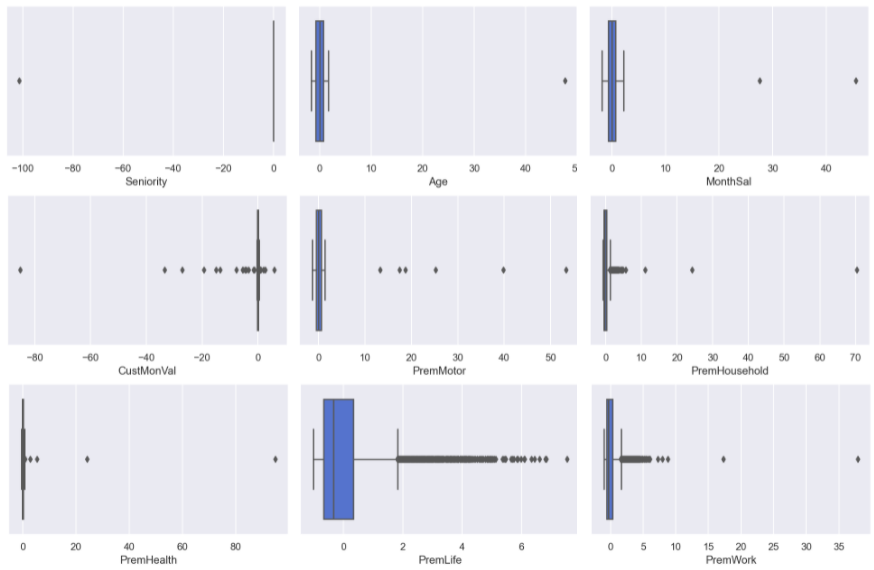
**

Figure 3.2- Boxplots.

Through the observation of the boxplots, we have concluded there are outliers. To find a reasonable way to eliminate them, we have used a combination of the following five different methods:

1. **Z-score** that checks how many standard deviations is a datapoint distant from the mean, with the threshold set to 3.0;
2. **InterQuantile Range (IQR)** that creates boundaries using the first and third quantile and the interquartile range, with the multiplier set to 1.5;
3. **Local Outlier Factor (LOF)** that detects outliers by comparing the density of the neighborhood of a point to the ones of its neighbors;
4. **Isolation Forest** that differs from the others by starting directly from outliers rather than from normal observations, so, on average, outliers will be closer to the root node (i.e. at a lower depth) than normal instances [3];
5. **One-Class SVM** tries to predict a hypersphere that separates the cluster of data points from the anomalies. The algorithm will try to find the smallest possible hypersphere and point outside of it will be considered outliers. [4] It has the particularity that the percentage of points expected to be outliers needs to be set in advance, we have used the outcome value from the LOF method.

After analyzing the observations considered outliers by each of the methods mentioned above, we have tested the removal of entries considered as outliers by at least 3 of the methods, by 4 methods and by 5 methods. However, as the boxplots did not show good results, we decided to drop this approach and deal with the outliers in the following way:

* Applying the IQR method for 5 features (*PremMotor*, *PremHealth*, *Age*, *Seniority*, *MonthSal*);
* Filtering out manually the outliers of 4 features (*CustMonVal*, *PremWork*, *PremHousehold*, *PremLife*).

The dataset was reduced to 10.059 observations, 97,7% of initial data.

In the next step, we checked the missing values. At this moment, the data missing is shown in Figure 3.3.

Graphical user interface, table

Description automatically generated

Figure 3.3- Missing values per feature

After scaling the data, we have used KNN Imputer to fill in missing values. This method utilizes the k-Nearest Neighbors method to replace the missing values in the datasets with the mean value from the parameter ‘n\_neighbors’ nearest neighbors found in the training set [5], which was set to 100.

At this stage we have also opted for removing 12 observations of customers with work insurance and ages ≤15 years old. We considered this to be an anomaly in the data as in Portugal the minimum age to work is 16 years old.

As for the records where the Seniority was higher than the Age (1.921 entries), those were replaced by missing values and the same KNN Imputer method described above was applied to fill them in.

Finally, we have checked Pearson correlation matrix to ensure there were no significant changes.

## Data standardization

The last step before proceeding to the clustering was to standardize the data using the previously mentioned *StandardScaler* followed by *fit\_transform*. Standardizing a dataset involves rescaling the distribution of values so that the mean of observed values is 0 and the standard deviation is 1. [6]

# Clustering

This report intends to present the insurance company customers segmentation, meaning we aim to create clusters of clients, which are groups of clients that in general are more similar between them than those in other groups.

We have tried two different clustering methods, K-means and Hierarchical clustering, whose results are explained in the present section.

## K-means

K-means method is mostly popular for being efficient, easy to understand and to implement. However, it has the limitation that the number of clusters need to be defined in advance and it is very sensitive to the initial seeds position.

To assist in defining the number of clusters, we plotted three different metrics:

* The Inertia plot, showing the dispersion of the points within the cluster, meaning a small inertia is the best outcome.
* The Average silhouette plot, showing how well each object lies within its cluster, being the best outcome a higher number.
* The Davies-Bouldin score plot, which index is based on a ratio between distances within the cluster and distances between clusters, the best outcome is the smallest index.

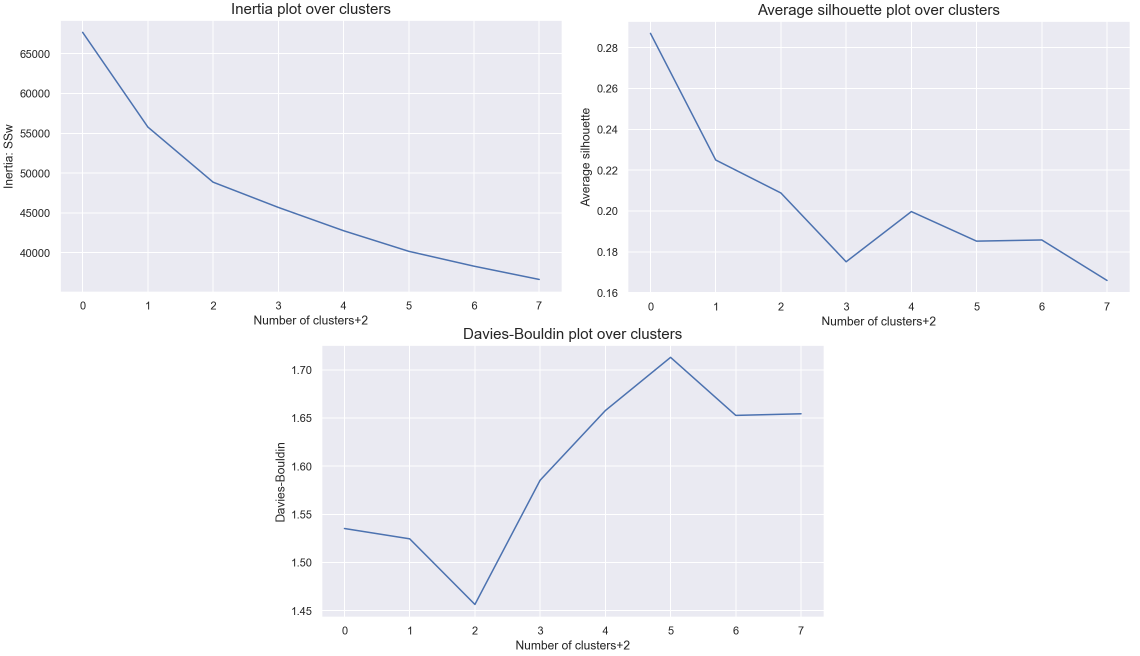


Figure 4.1 - Inertia, Average silhouette and Davies-Bouldin score plots.

We decided to run silhouette plots for 2, 3, 4 and 5 clusters to have a deeper analysis. The conclusion was clusters=5 seems to be the optimal one as it is the one having a more uniform thickness and each cluster is above average silhouette scores.

Chart, funnel chart

Description automatically generated

Figure 4.2 – Optimal silhouette plot, for clusters=5.

### Evaluation and visualization

To evaluate the quality of the clustering solution, the R2 was computed as a measure of the homogeneity.

For this purpose, we have used two different functions:

* *get\_ss*, which computes the sum of squares for all variables given a dataset;
* *adjusted\_r2*, which returns the overall adjusted R2.

The outcome was R2= 0.59.

To visualize the cluster profile, we have used the following function:

* *cluster\_profiles*, which shows the means and frequency of each of the 5 clusters defined.

Chart

Description automatically generated

Figure 4.3 - Clusters profile.

Looking at Figure 4.3 we can conclude that the output clusters are fairly consistent and well balanced. Also analysing the t-SNE scatter plot in Figure 4.4 we can see that the five clusters are well defined.

Chart, scatter chart

Description automatically generated

Figure 4.4 - t-SNE scatter plot.

## Hierarchical clustering

We also tried Hierarchical clustering whose outcome was reasonable, but as K means clustering has a higher R2, we decided to proceed the analysis with the latter method. For this reason, in this section we only show the method and its results very briefly.

### Evaluation and visualization

To evaluate the quality of the clustering we computed the R2, similarly to what was done for the K-means. For that we used the function *get\_r2\_hc*, computed for the different linkage methods used to calculate the distance between clusters:

* Ward – the criterion for choosing the pair of clusters to merge at each step is based on the optimal value of an objective function.
* Complete – It returns the maximum distance between each data point.
* Average – It returns the average of distances between all pairs of data points.
* Single –Returns minimum distance between two points, where each point belongs to two different clusters. [7]

After some analysis (R2 comparison and dendrogram visualization), we decided to proceed with complete linkage. In figure 4.5, it is presented the R2 for each linkage.

Chart, line chart

Description automatically generated

Figure 4.5 - R2 plot for the various hierarchical methods.

To create and visualize the hierarchical clustering dendrogram, we have used two functions: *full\_tree* and *full\_tree\_visual*. Looking at Figure 4.6 we can conclude the optimal number of clusters from this method is 4.

Chart

Description automatically generated

Figure 4.6 - Dendrogram.

The R2 obtained from this method was 0.21 and for this reason, as mentioned above, we decided to discard this approach and focus on the K-Means clustering, which shows a better R2 metric value.

After splitting the data into training and test, we also applied a decision tree classifier to test our  
solution. It was able to predict 93.4% of the customers correctly. In terms of feature importance, the most relevant feature on this prediction is *PremMotor* (41%) followed by Customer’s *Age* (27%) and *Seniority* (18%).

Chart, bar chart

Description automatically generated

Figure 5.1 - Features importance.

# Results

Diagram

Description automatically generated

Looking at the clusters individually we can characterize them as follows:

* Defend (Cluster 0): This is the segment with the 2nd oldest customers and the ones that have contracts for longer. They have the 2nd lowest premiums for household, work, life and health; however, this group presents the 2nd highest premium for motor.
* Attack (Cluster 1): This group is composed by the 2nd most recent clients, average salary and with the highest premium for motor. However, they present, the lowest premiums for household, work, life and health.
* Attract (Cluster 2): These clients are the youngest ones and with the lowest monthly gross salary. However, they present the highest monthly value and highest premiums for household, life and work.
* Retain (Cluster 3): Unlike the previous segment, this one is the oldest one, with the highest monthly gross salary and with the 2nd longest contracts.
* Growth (Cluster 4): This is the 2nd youngest group and the 2nd lowest monthly salaries. It presents the highest premium for health but lowest monthly value.

# Conclusions

As stated in section 2.2 – Business objectives, two of the main objectives of this project were to identify the key characteristics that best distinguish the customers and understand which and how many customer segments there are in the provided database. Our final solution was able to detect 5  
segments of customers. We were able to describe the key characteristics of customers in each segment in the section 4 e 5.

# References

[1] Chapman, P, Clinton, J, Kerber, R., Khabaza, T., Reinartz, T, Shearer, C. & Wirth, R. (2000). CRISPDM 1.0, CRISP-DM consortium

[2] Kwak, S, Kim, J 2017, Statistical data preparation: management of missing values and outliers, US National Library of Medicina, viewed 2 January 2022, <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5548942/>

[3] Mazzanti, S 2021, “Isolation Forest”: The Anomaly Detection Algorithm Any Data Scientist Should Know, Towards Data Science, viewed 15 December 2021, <https://towardsdatascience.com/isolation-forest-the-anomaly-detection-algorithm-any-data-scientist-should-know-1a99622eec2d>

[4] Satyam, K 2021, 5 Anomaly Detection Algorithms every Data Scientist should know, Towards Data Science, viewed 15 December 2021, <https://towardsdatascience.com/5-anomaly-detection-algorithms-every-data-scientist-should-know-b36c3605ea16>

[5] Htoon, K 2020, A Guide To KNN Imputation, Medium, viewed 16 December 2021, <https://medium.com/@kyawsawhtoon/a-guide-to-knn-imputation-95e2dc496e>

[6] Brownlee, J 2020, How to Use StandardScaler and MinMaxScaler Transforms in Python, Machine Learning Mastery, viewed 2 January 2022, <https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/>

[7] Prajapati, K 2020, Hierarchical clustering and linkage explained in simplest way, Medium, viewed 2 January 2022, <https://medium.com/@codingpilot25/hierarchical-clustering-and-linkage-explained-in-simplest-way-eef1216f30c5>

# Appendix

## Appendix I – Features description

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Seniority | Number of years since the first policy |
| Age | Age of the customer |
| MonthSal | Gross monthly salary (€) |
| CustMonVal | Customer Monetary Value |
| PremMotor | Premiums (€) in LOB: Motor |
| PremHousehold | Premiums (€) in LOB: Household |
| PremHealth | Premiums (€) in LOB: Health |
| PremLife | Premiums (€) in LOB: Life |
| PremWork | Premiums (€) in LOB: Work Compensations |