NOVA IMS

DATA MINING PROJECT

ANASTASIA NICA |M20210516
INÊS RODRIGUES |M20210557
JOÃO ASCENSÃO | M20210178
GROUP AI

TABLE OF CONTENTS

TABLE OF TABLES

TABLE OF FIGURES

1. INTRODUCTION

This project aims to develop a customer segmentation of the ABT (Analytic Based Table) for an insurance company.

With the information collected by our data scientists, the Portuguese insurance company can segment their customers based on their characteristics and choices and consequently tailor the marketing approaches, which will allow the company to benefit from cost reduction and to have more satisfied customers as they will receive less offers but more targeted ones matching their interests and consumption patterns.

2. DATA AND VARIABLES DESCRIPTION

The company's ABT from 2016 has **10 296 observations** and **14 variables** that are described in the following table (table 1).

VARIABLE	TYPE	DESCRIPTION
Cust_ID	int64	Customer ID
First_Year	float64	First year as a customer
Birthday	float64	Customer's Birthday Year
Education	object	Customer's Education Level (1-Basic, 2-High school, 3-BSc/Msc, 4-PhD)
Montly_Salary	float64	Gross Monthly Salary (€)
Area	float64	Categorical variable from 1 to 4 that identifies the living area (there is no information about the meaning of the area codes)
Children	float64	Binary variable that tells if the customer has children (1) or not (0)
CMV	float64	Customer Monetary Value/Lifetime value = (annual profit from the customer) x (number of years that they are a customer) - (acquisition cost)
Claims_Rate	float64	Amount paid by the insurance company (€)/ Premiums (€) (Note: in the last 2 years)
Motor	float64	Annual premiums in Motor (€)
Household	float64	Annual premiums in Household (€)
Health	float64	Annual premiums in Health (€)
Life	float64	Annual premiums in Life (€)
Work_Compensation	float64	Annual premiums in Work Compensations (€)

Table 1 - Variable Description

We should note that in the premium variables we can have negative values that manifest reversals occurred in the current year (2016), paid in the previous one(s). This means that the clients with negative values cancelled the respective insurance.

3. DATA PREPARATION

In order to have reliable data to apply the clustering methods and to obtain trustworthy results we have to clean our data. We started by checking and removing **duplicate observations**. We found out that we had **3 duplicate observations**. This means we now have **10293 observations**.

Previously, when analyzing the data types for each column, we noticed that *Education* was stored as an object. When imported, values for this variable were like “1 - Basic”. Since each level of *Education* has a correspondent number, we decided it would be more practical to keep only that number. Therefore, we converted this variable into an ordinal one.

3.1. DESCRIPTIVE STATISTICS

In order to get insights from our data, we decided to create two tables with **descriptive measures of our variables**. One with the categorical variables (table 3) and other with numerical (table 2), as the measures presented are different depending on the attributes type.

VARIABLE	MIN	P25	P75	MAX	MEAN	STD	MISSING VALUES
First_Year	1974	1980	1992	53784	1991.06	511.34	30
Birthday	1028	1953	1983	2001	1968.01	19.71	17
Monthly_Salary	333	1706	3290	55215	2506.62	1157.52	36
CMV	-165680	-9.44	399.86	11875.9	177.63	1946.09	0
Claims_Rate	0	0.39	0.98	256.2	0.74	2.92	0
Motor	-4.11	190.59	408.3	11604.4	300.50	211.94	34
Household	-75	49.45	290.05	25048.8	210.42	352.64	0
Health	-2.11	111.8	219.04	28272	171.55	296.44	43
Life	-7	9.89	57.79	398.3	41.85	47.48	104
Work_Compensation	-12	10.67	56.79	1988.7	41.28	51.52	86

Table 2 - Descriptive Statistics from Numerical Variables

Looking at the statistics presented above, we noticed that there will be a lot to work on and so, we highlighted the values that seem to be more problematic. Most variables look like they have **outliers**, due to the difference between the minimum value and the percentile 25 and/or the maximum and the value in percentile 75. For example, for *Claims_Rate*, we can see that the percentile 75 is 0.98 while the maximum value is 256.2 - this means that we probably have upper outliers. Another example is *CMV*. Here, the percentile 25 is -9.44 while the minimum value is -165680, probably indicating the presence of at least one dramatic lower outlier.

The big standard deviations in comparison to the mean in variables like *CMV* (deviation equal to 1946.09 and mean equal to 177.63) are also a sign of outliers. Another red flag would be, for example, the maximum value for *First_Year* (53784), which does not make sense. We will deal with these cases later when doing the **coherence checks**.

Something that also deserves our attention are the **missing values**. We noticed that some variables like *Life* have many missing values (104). Later, we will have to decide what to do with the individuals that have null values in some attributes.

VARIABLE	MISSING VALUES	UNIQUE CATEGORIES	MODE	MODE FREQ
Area	1	4	4	4142
Children	21	2	1	7260
Education	17	4	3 - BSc/MSc	4799

Table 3 - Descriptive Statistics for Categorical Variables

The number of categories stated is in line with what we expected, and we consider that there is no red flags when looking at the remaining data on this table, besides the missing values.

3.2. COHERENCE CHECKING

We needed to confirm if the data we had was reasonable in the given context. For that, we created a set of rules to evaluate it, that resulted in a new variable for each rule - in these variables the value 1 means that the observation is not obeying to the rule.

The first rule we created was about *Birthday*. We decided that the values for this variable could not be bigger than 2016 or smaller than 1896. The data was stored in 2016 and, for this reason, it is impossible that someone born after this year is included in the data frame. Also, we chose 1896 because the oldest person alive certified by the Guinness World Records[1] is 116 years old, so we believe that it would not make sense to have someone older than 120 in our data. After applying this rule, we found that there is **only one observation** that does not comply with it - the customer 7196 was born in 1028.

The second rule is about the *First_Year*. It does not make sense for the *First_Year* to be smaller than 1896 and we cannot have a record of something that did not happen yet, so it cannot be bigger than 2016. After applying this rule, we found out that there is also **only one observation** that does not obey it. Customer 9295 will have his first policy made in the year of 53784. This is clearly not right so we decided to delete this row.

The third rule is also about *First_Year*. Since this variable represents the first contact of the client with the insurance firm, it does not make sense that this contact happened before the person was born - *First_Year* cannot be smaller than *Birthday*. After applying this rule, we were shocked to find that **1997 observations did not comply with it**. Since this is a huge number, we could not delete all the incoherent rows. So, as *First_Year* was calculated by the company and *Birthday* was probably submitted by the customers, we decided that it is more likely that the customers gave wrong information. Assuming this, it would mean that 1997 customers filled the forms wrongly. If this many customers gave wrong data, nothing can guarantee us that others did not. With this in mind, we decided to **delete the *Birthday* column**.

Looking back at the first coherence rule, the problem does not apply anymore once we deleted the *Birthday* column. The wrong data about the customer born in 1028 was also deleted.

Finally, the last rule is about *Monthly_Salary* and *Premiums*. A person cannot spend more money than what they earn. We established that the sum of all the premium variables cannot be higher than the annual salary (*Monthly_Salary* times twelve).

Only one observation was not coherent with this last rule - customer 9150 spent 29331.32€ in insurance premiums in 2016, but his annual salary is 11844€. The amount spent exceeds the amount earned by 40%. We found this quite odd - even though some people may have savings, it is not normal that someone would spend an amount so much bigger than the annual salary on insurance premiums. After exploring the values for each category of premiums for this customer, we noticed that 96.4% of the money spent was in health premiums (28272€). When compared to the remaining values for the Health column, this value is oddly high so we concluded that even if it had not raised a red flag in the coherence check, it would in outlier detection. Therefore, we decided to delete the observation. After deleting this row, we were left with **10291 observations**.

In order to facilitate the understanding of the coherence check rules, we created the following summary table (table 4).

[1] https://www.guinnessworldrecords.com/news/2019/3/worlds-oldest-person-confirmed-as-116-year-old-kane-tanaka-from-japan/

INCOHERENCE NUMBER	INCOHERENCE PSEUDOCODE	WHY THIS RESTRICTION
1	1 if (Birthday > 2016 or Birthday < 1896) else 0	We consider it is not possible to have people older than 120 years old or people that were not born in 2016.
2	1 if (First_Year > 2016 or First_Year < 1896) else 0	As we will not have people older than 120 and we don't have data from the future, we have to make sure the year of the first policy is between 1986 and 2016.
3	1 if First_Year < Birthday else 0	People can not have insurance policies in their name before they are born.
4	1 if (2016-Birthday < 16 and Monthly_Salary > 0) else 0	As the minimum legal age to work in Portugal is 16 years old, people younger than this shouldn't have a salary.
5	1 if (2016-Birthday <= 16 and (Education == '3 - BSc/MSc' or Education == '4 - PhD') else 0	We find it odd that a person with only 16 years old already has a Bachelor or a PhD.
6	1 if (sum(Motor, Household, Life, Health, Work_Compensation) > (Monthly_Salary*12)) else 0	People, normally, do not spend more money than they earn. It is not impossible to happen, but we want to take a closer look at these cases.

Table 4 - Summary of Coherence Rules

Since we removed some rows due to the rules applied, we figured we should re-do the descriptive statistics table so that we were able to analyze it and perceive the outliers. According to the table below (table 5), some of the outliers seem to have been removed, while others are still there. We can notice that *Claims_Rate*, for example, still has a maximum value of 256.2 while the percentile 75 is 0.98, indicating that there might be some upper outliers. However, values that did not make sense, like the maximum value for *First_Year* on the last descriptive statistics table were now corrected. The real maximum for *First_Year* is now 1998.

VARIABLE	MIN	P25	P75	MAX	MEAN	STD	MISSING VALUES
First_Year	1974	1980	1992	1998	1986.02	6.61	30
Monthly_Salary	333	1706	3290.5	55215	2506.62	1157.52	36
CMV	-165680	-9.44	399.86	11875.9	177.63	1946.09	0
Claims_Rate	0	0.39	0.98	256.2	0.74	2.92	0
Motor	-4.11	190.59	408.3	11604.4	300.50	211.94	34
Household	-75	49.45	290.05	25048.8	210.42	352.64	0
Health	-2.11	111.8	219.04	7322.48	171.55	296.44	43
Life	-7	9.89	57.79	398.3	41.85	47.48	104
Work_Compensation	-12	10.67	56.79	1988.7	41.28	51.52	86

Table 5 - Descriptive Statistics After Coherence

3.3. OUTLIERS

Taking into consideration the descriptive statistics on table 5 and as we already said, most of the variables seem to have outliers. Therefore, we will go step by step throughout the variables, analyzing the **histograms** and **boxplots** for each one that raises a red flag.

As we can see from the figure 1, regarding *Monthly_Salary*, there are 2 extreme positive observations that are influencing the distribution of the variable. Although we do not display the histogram as it is virtually impossible to see due to the presence of outliers, we have taken it into account. Thus, we **removed the 2 observations above 30 000€** and the distribution obtained is similar to a normal.

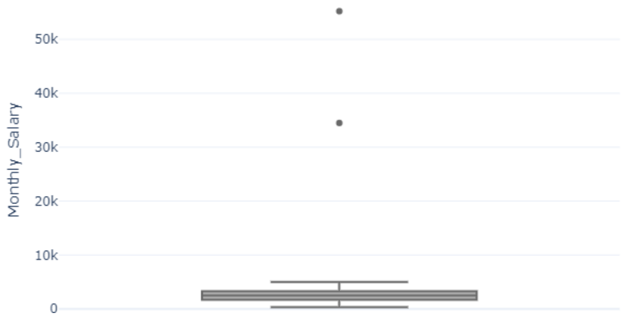


Figure 1 - Boxplot for *Monthly_Salary*

Regarding *CMV*, the boxplot is totally flat (figure 2) highlighting the presence of multiple outliers. The outliers skew the average *CMV* and the standard deviation of the distribution, which should be recalculated after the outlier's removal. We started by **deleting the biggest negative outlier** and, as we still got a flat boxplot, we **deleted 15 more negative outliers**. After this process, we focused on the positive outliers and, although these ones mean they are good customers because their annual contribution to the company is bigger than the acquisition cost, it is imperative to delete outliers because they will influence our analysis. We **removed 23 observations** from the main data frame.

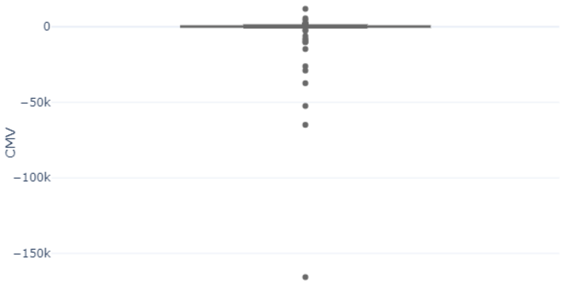
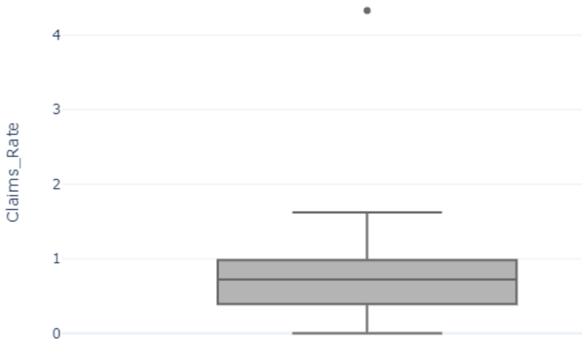


Figure 2 - Boxplot for *CMV*

Observing *Claims_Rate's* boxplot (figure 3) and histogram (figure 4), there was **one upper outlier** that we eliminated.



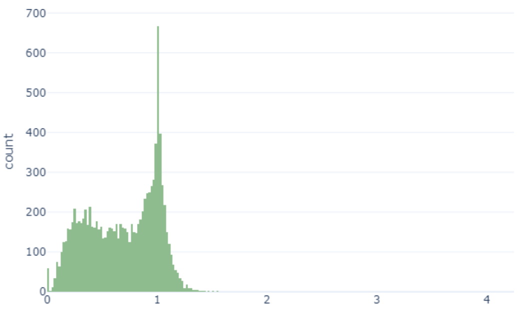


Figure 3 - Boxplot for *Claims_Rate*

Figure 4 - Histogram for *Claims_Rate*

Regarding Motor, there was a huge difference between the percentile 75 and the maximum value pointing out the presence of outliers as we can notice from the figure 5. For this reason, we **removed the 3 observations**, getting a distribution that resembles a normal with a slight negative skew.

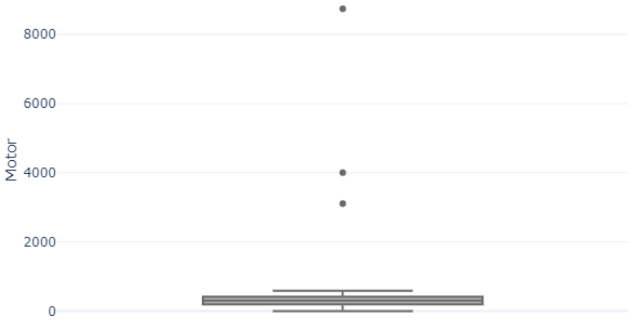


Figure 5 - Boxplot for *Motor*

According to the boxplot in figure 6, about Health, we can see that there are only some moderate outliers and thus, we **removed 5 outliers** considering the first big break from left to right.

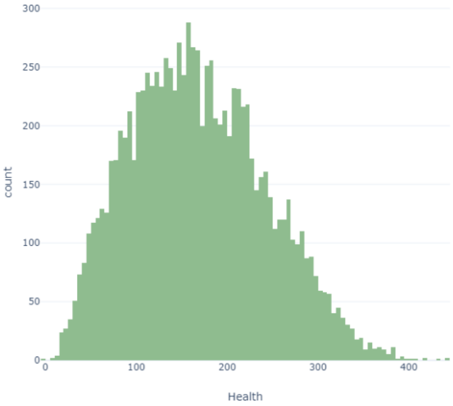


Figure 6 - Histogram for *Health*

Referring to *Life* and looking at the respective histogram (figure 7), there are some breaks on the distribution which led to the **removal of 17 outliers**.

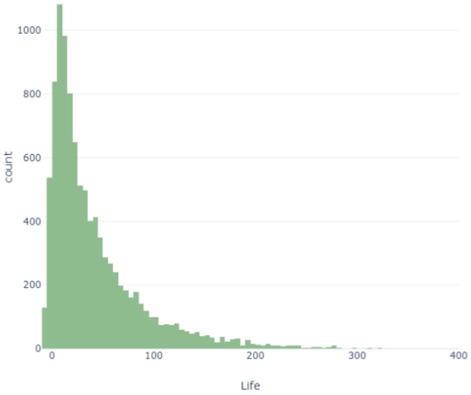


Figure 7 - Histogram for *Life*

Considering *Household*, its distribution is influenced by the extreme upper outliers creating a flat boxplot (figure 8). Hence, we first **removed the three big outliers** and as the boxplot continued flat, we **deleted the consecutive two outliers**, obtaining the distribution in figure 9. There were still some breaks in the distribution so we decided to remove the observations until the first break (from right to left), **deleting 20 more observations**.

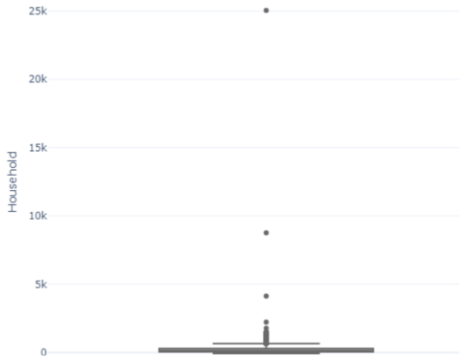


Figure 8 - Boxplot for *Household*

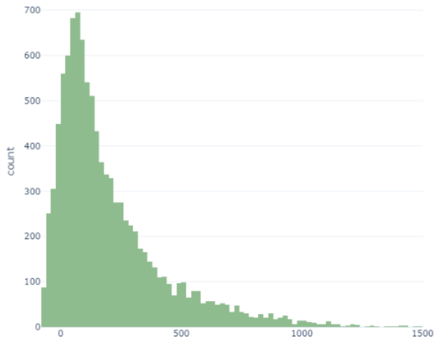


Figure 9 - Histogram for *Household*

Finally, regarding *Work_Compensation*, we can note from the figure 10 that we have 3 evident upper outliers that were removed, obtaining the distribution in figure 11, with a long tail to the right. As we can observe, there are still some breaks in the distribution that led to the **removal of 15 more outliers**.

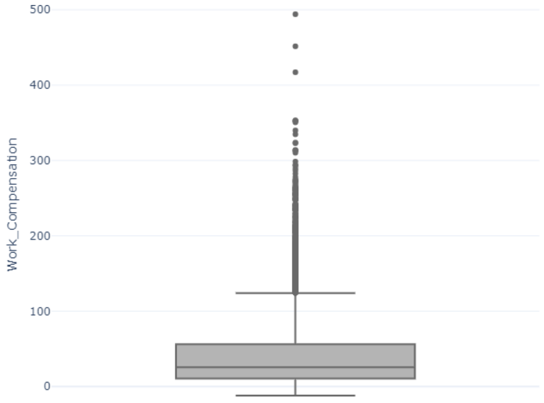


Figure 10 - Boxplot for *Work_Compensation*

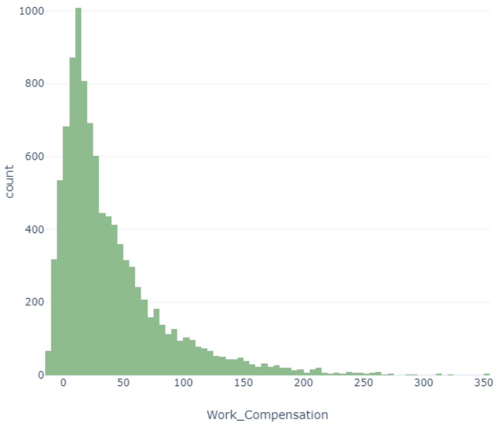


Figure 11 - Histogram for *Work_Compensation*

After the outlier's removal, we ended with **10 185 observations.** Deleting, in total, **106 observations** (**1%** of the total observations) in this phase.

For further analysis, we decided to create figure 12, to show the relations between every pair of variables so that we could look for **bidimensional outliers**. It also includes the final distribution for each variable in the main diagonal.

As we can see, variables like *Household*, *Life* and *Work_Compensation* (fifth, seventh and eighth rows of the plot) have positive skew, which we will deal with later. The order by which the variables appear on the plot is *Monthly_Salary*, *CMV*, *Claims_Rate*, *Motor*, *Household*, *Health*, *Life* and *Work_Compensation*. By looking at the scatter plot between *CMV* and *Claims_Rate*, we can already notice that they will be highly correlated.

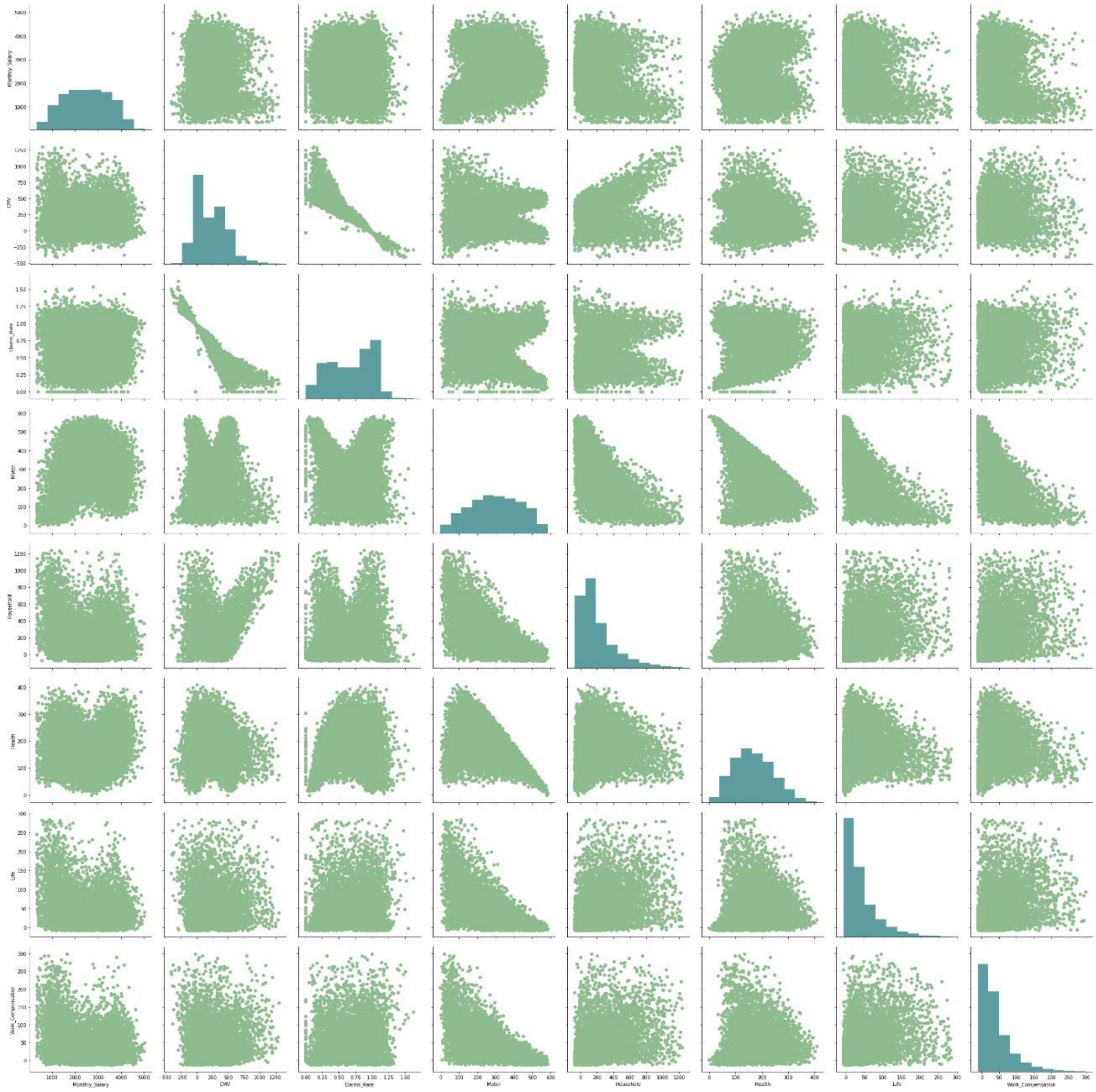


Figure 12 – Pairplot

3.4. CORRELATIONS

In order to understand the relations between variables, we computed the correlations between them using the **Spearman method**. At first, we thought about using the Pearson method, but after some discussion and after taking into account its assumption that the variables follow a normal distribution and the fact that it only evaluates linear relationships between variables we went for Spearman. The final argument for this reasoning was the existence of ordinal variables like *Education*. At this point we will not see the correlations with *Area*, as this is a nominal variable. We will deal with it later.

We ended up noticing that there are no really strong correlations between any variable in the data frame as most of the squares in the graphic matrix (figure 13) are pale, except for *Claims_Rate* and *CMV*, that have a correlation of -0.97, meaning that they vary in the opposite direction in almost the same measure. The biggest correlations after this one are between *Motor* and all the other premiums (always between -0.65 and -0.7).

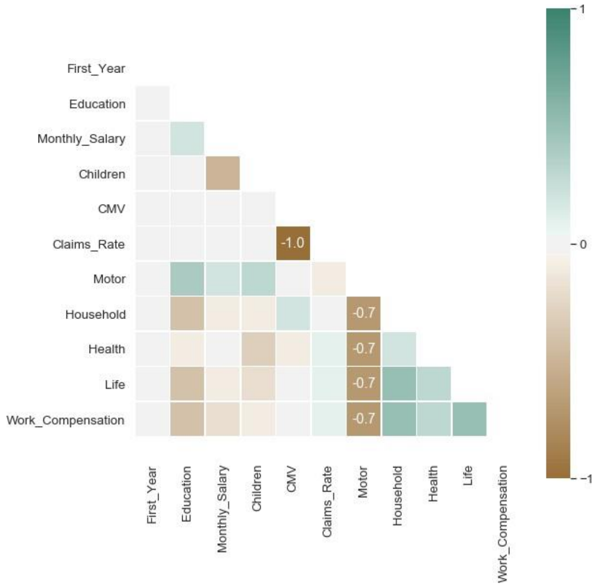


Figure 13 - Correlations Heatmap

3.5. MISSING VALUES TREATMENT

In table 7, we found out that the *Area* only had one missing value. Since we have 10 249 rows and only one has a missing value for this variable, we decided to delete this row.

We can also notice that the *First_Year* had **30 missing values** and we thought about imputing these values, but after a critical discussion about what variables should we use that made sense to impute these values, we ended up concluding that none of the variables could help us to do that. Thus, we deleted these 29 rows (one corresponded to the row deleted before) ending with **10 219 observations**.

VARIABLE	MISSING VALUES
First_Year	30
Education	17
Area	1
Children	21
Monthly_Salary	36
CMV	0
Claims_Rate	0
Motor	33
Household	0
Health	42
Life	103
Work_Compensation	86

Table 6 - Missing values by variable

For the variables referring to premiums, we decided to **replace missing values with 0**. We believe that, in this case, if no data is recorded, the customer did not spend any money on that specific premium service. It is not that strange that we have 265 null values in these attributes because if someone has one specific insurance, nothing guarantees that this person also has the others. They might not even have a car or a house of their own. We ended up replacing 33 missing values in the *Motor* premiums, 42 in *Health*, 104 in *Life* and 86 in *Work_Compensation*.

Education had 2 missing values and, since this is a categorical variable, we decided to **use a K Neighbors Classifier to impute the nulls** in it. We used 7 neighbors and based their weights on the euclidean metric. After looking at the correlations for *Education* and using our knowledge of the theme, we decided to use *Motor*, *Health*, *Household*, *Life* and *Work_Compensation* to create the classifier.

We moved on to filling in the 36 missing values regarding the *Monthly_Salary*. This is a continuous variable, so we decided to **use a regression**. For a start, we analyzed the correlations to evaluate which variables would be better to predict the salary. The most correlated variable was *Children* (-0.46), followed by *Life* and *Work_Compensation* (0.24 on both), *Household* (-0.23), *Motor* (0.22) and *Education* (0.17). We wanted to use a K Neighbors Regressor, but first wanted to be sure that the variables were relevant for the regression. We calculated their p-value for a linear regression and the final R square, obtaining a p-value of 0 for all the chosen variables and an R square of 0.36, meaning that approximately 36% of the total variance in *Monthly_Salary* is tackled by this regression. Even though 36% is not as high as we would wish, since we will be using the neighbors, we will use more local information. With this in mind, we considered the results obtained from the regression as a good sign to move on with the K Neighbors Regressor. For it, we used 10 neighbors, weighted according to the euclidean distance. After applying this imputation, we saw the variable distribution and nothing had changed.

Finally, we had 13 missing values on the *Children* variable. As this attribute is categorical, we applied the same method as in *Education*. After looking at the correlations for *Children* and discussing critically, we decided to use *Monthly_Salary*, *Motor*, *Health*, *Household*, *Life* and *Work_Compensation* to create the classifier.

We used *Monthly_Salary* since the correlation between this one and *Children* is relatively high (-0.46). This correlation expresses an opposite effect between both variables that can be explained in two ways, first if someone has a high salary it is probably focused on trips, enjoying life and on responsibilities, there is no time to have many children to take care off and second, if someone has a lower salary, probably it also has a low education and therefore, less information about contraceptives that may lead to have more children. Thus, we thought that this variable would totally make sense.

The inclusion of the premiums in the classifier is explained by the fact that if someone has children of their own, they are probably more concerned about ensuring financial stability and health insurance in case of any casualty, that is why we considered important to include them in the KNN classifier.

3.6. DESCRIPTIVE STATISTICS AFTER DATA TREATMENT

After the data treatment where we removed noisy values, outliers and imputed missing values, discarding in total 141 observations (1.4% of our dataset) and 1 variable, we ended up with **10 155 observations** and 13 variables in total. Consequently, it is clear that the descriptive statistics changed because almost every variable suffered transformations. Thus, the following table (table 7) presents the final descriptive statistics.

VARIABLE	MIN	P25	P75	MAX	MEAN	STD	MISSING VALUES
Numerical							
First_Year	1974	1980	1992	1998	1986	6.6	0
Monthly_Salary	333.0	1701.0	3289.0	5021.0	2503.9	984.3	0
CMV	-416.7	-9.1	398.8	1455.9	216.3	255.0	0
Claims_Rate	0.0	0.4	1.0	1.6	0.7	0.3	0
Motor	-4.1	190.4	407.4	585.2	296.3	138.6	0
Household	-75.0	49.5	289.8	1513.1	205.4	230.7	0
Health	-2.1	111.0	218.9	442.9	167.4	74.7	0
Life	-7.0	9.9	57.0	398.3	41.4	47.4	0
Work_Compensation	-12.0	9.9	55.9	353.2	40.5	46.3	0
Categorical							
Area	-	-	-	-	-	-	0
Children	-	-	-	-	-	-	0
Education	-	-	-	-	-	-	0

Table 7 - Final descriptive statistics

4. DATA PREPROCESSING

4.1. TRANSFORM VARIABLES

In order to try to gain explainability and partitioning power for a better and more precise customer segmentation, we decided to create new variables that are basically transformations of variables we already had in the data frame.

The first variable created was ***Client_Years***, that measures the number of years since the customer's first policy. We believe it is easier to analyze this kind of data instead of having only the first policy's year .

We also decided to create ***Yearly_Salary*** since the amount spent in each premium is expressed as the value for one year, we figured it would be useful to have the annual salary.

Then, we wanted to have the total spent on premiums, so we created ***Total_Premiums*** that sums up the value of the money spent in each premium category, except the cancelled ones. This variable tells us how much money each customer spent in our company in 2016.

After creating the *Total_Premiums* we realized that some customers did not have any purchases made for the year of 2016. Since we only have data for this year, we figured it did not make sense to include them in our analysis. With this in mind, we decided to remove these 12 customers from our data frame, ending with **10 143 observations**.

Using these last two new variables, we created the ***Effort_Rate***, which measures the proportion of the salary spent in our company. This may be a good measure of a client's commitment to the company as it measures the effort of each customer to be a client.

Similarly to the effort rate, we created a ratio for each premium category. ***Motor_Ratio***, ***Household_Ratio***, ***Health_Ratio***, ***Life_Ratio*** and ***Work_Ratio*** measuring the proportion of the total money spent on premiums that was spent in each specific category. Using these variables, we can see in which kind of premium is the customer investing the most. When the customer cancelled the premium, the value for the respective ratio is 0, as he/she did not spend money on that specific premium.

Regarding the customers who cancelled premiums, we wanted to compute the sum of the cancelled value as we believe it is different to cancel a 300€ premium versus a 1€ one. Having this in mind, we created a variable that represents the sum of values for the cancelled premiums and called it ***Negative***.

Lastly, we created a variable called ***Cancelled***. This is a binary variable that assumes the value 1 if the customer cancelled any premium and 0 otherwise.

A summary for all the new variables created can be found in table 8 below.

ID	NAME	VARIABLE PSEUDO-CODE	DESCRIPTION
1	Client_Years	2016 - First_Year	Number of years since the client's first policy
2	Yearly_Salary	12 * Monthly_Salary	Gross yearly salary (€)
3	Total_Premiums	(Motor>0) + (Household>0) + (Health>0) + (Life>0) + (Work_Compensation>0)	Total value spent in premiums in the year of 2016 or paid in advance (€)
4	Effort_Rate	Total_Premiums / Yearly_Salary	Proportion of the yearly salary spent in premiums
5	Motor_Ratio	Motor / Total_Premiums (if Motor<0, Motor_Ratio=0)	Proportion of the total value spent in premiums that was spent in motor premiums
6	Household_Ratio	Household / Total_Premiums (if Household<0, Household_Ratio=0)	Proportion of the total value spent in premiums that was spent in household premiums
7	Health_Ratio	Health / Total_Premiums (if Health<0, Health_Ratio=0)	Proportion of the total value spent in premiums that was spent in health premiums
8	Life_Ratio	Life / Total_Premiums (if Life<0, Life_Ratio=0)	Proportion of the total value spent in premiums that was spent in life premiums
9	Work_Ratio	Work_Compensation / Total_Premiums (if Work_Compensation<0, Work_Compensation_Ratio=0)	Proportion of the total value spent in premiums that was spent in work compensation premiums
10	Negative	(Motor<0) + (Household<0) + (Health<0) + (Life<0) + (Work_Compensation<0)	Sum of the values for cancelled premiums
11	Cancelled	1 if any premium <0, else 0	Takes the value 1 if the customer cancelled any premium, 0 otherwise

Table 8 - New variables

We decided to apply the **square root transformation** to the variables *Life*, *Household* and *Work_Compensation* as a measure to reduce their skewness, because data with heavily skewed variables may lead to very elongated clusters that are not well captured by methods like K-Means and Ward's, which favours roughly spherical-shaped clusters. As we had negative numbers and it is not possible to compute the square root of a negative number, we had to sum to all observations the absolute value of the minimum value for each variable. Firstly, we tried to apply a log transformation to the same variables, but the result wasn't as good.

We followed the same line of thought for the variables *Effort_Ratio*, *Life_Ratio*, *Work_Ratio* and *Household_Ratio* after taking a look to their distributions.

Besides this transformation, we also removed outliers found in the recently created variables *Total_Premiums*, *Effort_Rate*, *Household_Ratio*, *Life_Ratio* and *Work_Ratio*, which gave a total of 95 rows dropped.

Since the *Monthly_Salary* gives us the same information as *Yearly_Salary*, we decided to **drop *Monthly_Salary*** The same happens between the variables *First_Year* and *Client_Years*. We kept only the *Client_Years* variable because the number itself is more countable than the year of the first policy.

At this point, we had removed from our dataset 2.4% of the observations and we now have **10 048 records** that will be used in the clustering analysis. We are now ready to proceed to the next phase.

4.2. CORRELATIONS

At this phase, we find it important to **go back to the correlations** and check how the new variables are related to each other and to the original ones, as we should not use high correlated variables when performing clustering analysis because it can inflate the importance of some variables, leading to wrong segmentation definitions.

It is normal that we have several variables with a correlation equal or higher than 0.65 (0.7 in the graph below), like the Premium variables and their respective ratios, since the new variables were created from the existing ones. The same happens with the square root transformations. The following pairs of variables represent the ones that can never be in the same clustering: (a) *CMV* and *Claims_Rate*; (b) *Total_Premiums*, *Household* and *Motor* and the respective ratios and sqrt; (c) *Effort_Rate* and *Yearly_Salary*. Correlations are represented in figure 14.

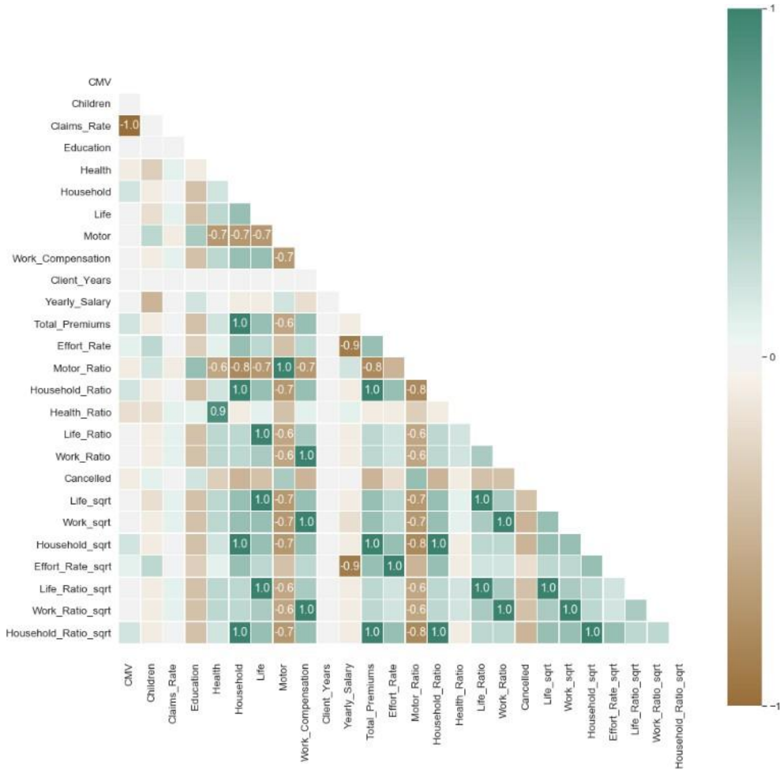


Figure 14 - Correlations with new and transformed variables

We will have all these correlations in mind when choosing the variables to use in the clustering phase.

4.3. VARIABLE SELECTION

In order to increase the performance of our models in the next phases, it is important to make an exploratory analysis between our variables to see how they behave with each other and to have a baseline of what should be the best set of variables to consider when applying clustering algorithms.

A variable that since the beginning was a little ambiguous is *Area*. We do not have any kind of information about what this variable truly means and so, we decided to explore it. We plotted boxplots along 6 variables that we found important for the next phases - *CMV, Claims_Rate, Total_Premiums, Client_Years, Yearly_Salary and Effort_Rate*. Observing the figure 15, we can conclude that there are no significant differences between categories in each of the 6 variables, therefore, our suspicions were correct. This variable has no discriminant power and will not be considered in the clustering phase - **we removed it**.

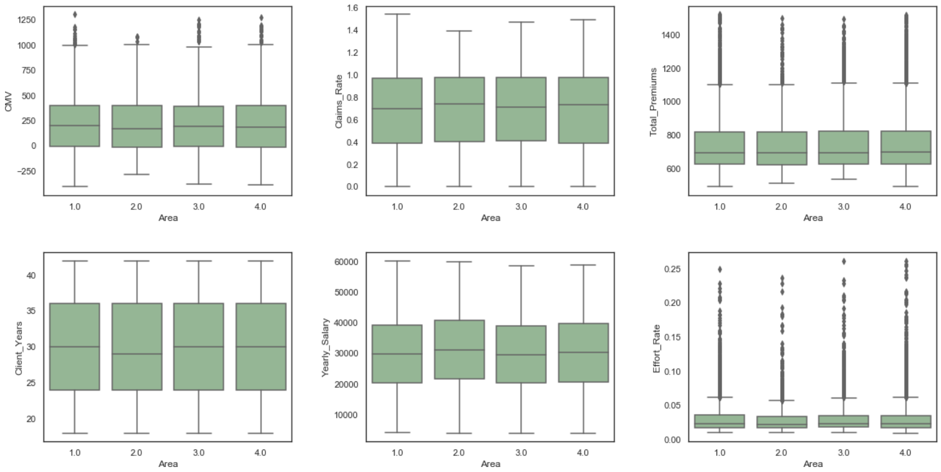


Figure 15 - Boxplots for *Area*

Considering the variables *Education* and *Children* and knowing that we critically think that these variables could have discriminant power as they are almost the only variables on the sociodemographic view, we decided to do the same analysis for these variables as we did for *Area* but in this case we also considered the ratio variables.

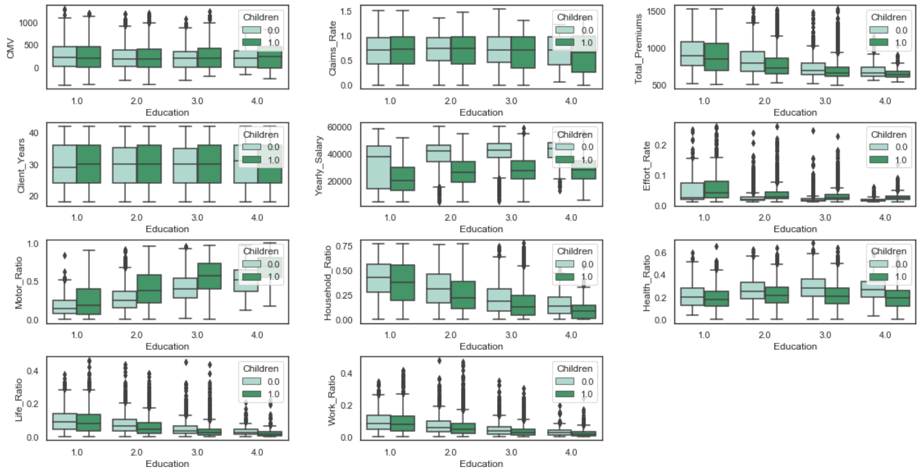


Figure 16 - Boxplots for categorical variables

As we can see from the figure 16, there are significant differences between the categories of *Education* in some variables mainly in *Total_Premiums*, *Motor_Ratio*, *Household_Ratio*. Regarding *Children*, there are also some differences mainly in *Yearly_Salary*. Hence, we should consider these variables as discriminators for the clustering analysis.

Now it is time to reason about the need to divide our variables in different views. When we look to our data, we believe there are three different areas under analysis: the client's Sociodemographic information, the Value of the client for the company and the Products bought. Knowing this, we will not perform clustering with all variables at once, but we will perform a clustering analysis for each view and then join the results.

The variables that are part of each view are the following:

a. **Value**: *Children*; *Education*; *Yearly_Salary*

b. **Product**: *Health*; *Household*; *Life*; *Motor*; *Work_Compensation* (original, ratios and square roots)

c. **Sociodemographic**: *CMV*; *Cancelled*; *Claims_Rate*; *Client_Years*; *Effort_Rate* (original and square root); *Total_Premiums*

At this phase, we decided to standardize the numeric variables from the Value and Socio Demographic views as they all have different scales and we do not want any of them to have a higher weight than others when creating the clusters. We will not standardize the attributes in the Product view as they all have the same unit of measure.

5. CLUSTERING ALGORITHMS

5.1. K-PROTOTYPES/K-MEANS & HIERARCHICAL

K-Prototypes is a clustering algorithm that is based on a mix between K-Means and K-Modes. In the K-Modes algorithm, distance is measured by the number of common categorical attributes shared by the two observations. On the other hand, the K-Means algorithm identifies k centroids and allocates each observation to the centroid closer to it. Summing up, K-Modes deals with categorical variables and K-Means with numerical variables, what makes K-Prototypes able to deal with both.

Throughout this project, we used K-Means and K-Prototypes with k equal to 20 (number of centroids). Then, we applied hierarchical clustering to those centroids using the Ward method alongside with the Euclidean distance.

SOCIODEMOGRAPHIC SEGMENTATION

Since the sociodemographic view included both types of variables, we applied K-Prototypes for the clustering. Since K-Means has some trouble dealing with skewed distribution, we confirmed that *Yearly_Salary* does not have a tail neither to the left nor to the right.

For defining the number of clusters, we looked at the dendrogram (figure 17) resulting from the hierarchical clustering. By looking at it, we decided to go with three clusters.

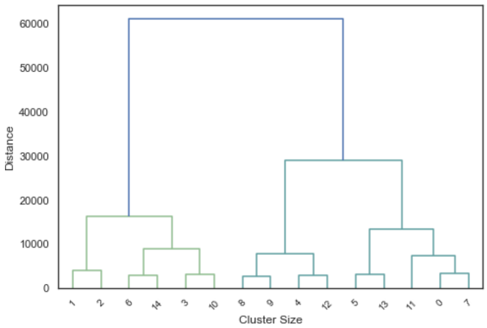


Figure 17 - Dendrogram for sociodemographic view

The profiling of the resulting segments was as follows (figure 18):

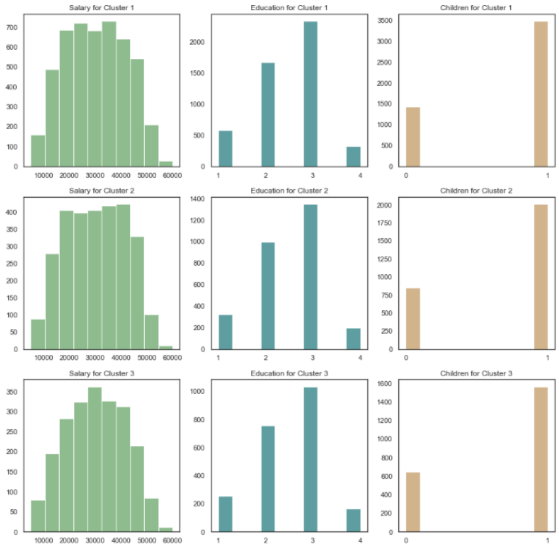


Figure 18 - Profiling for sociodemographic with K-Prototypes

Unfortunately, there are no major differences between the three clusters. All of them have most of the people with salaries close to the average, and both *Education* and *Children* look almost the same in between segments. Having this in mind, we decided not to use this algorithm.

VALUE SEGMENTATION

For the value segmentation we used *CMV*, *Effort_Rate* and *Total_Premiums*. At first, we thought of including *Client_Years* and *Cancelled*. However, after testing various combinations of all the variables and looking at the various resulting profiling plots, we concluded that these last two variables were not very relevant for differentiating the clusters and were overshadowing the others. Also, it is not very recommended to use K-Means with binary variables like *Cancelled*. Taking all of this into consideration, we chose the three variables mentioned before but kept Cancelled in the final profiling plot.

To define the number of clusters, we looked at the dendrogram (figure 19) resulting from the hierarchical clustering. By looking at it, we decided to go with three clusters.

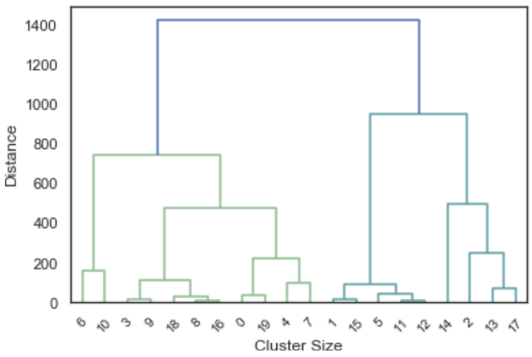


Figure 19 - Dendrogram for Value with K-Means

The profiling of the resulting segments was as follows (figure 20):

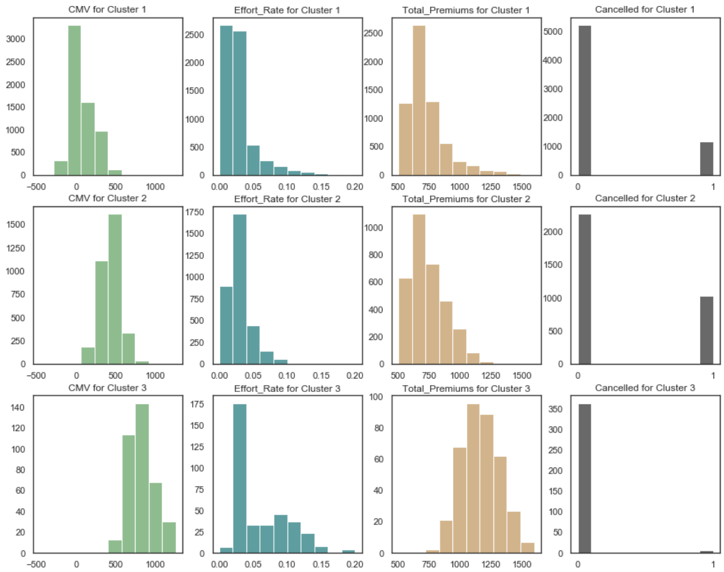


Figure 20 - Profiling for value with K-Means

Taking a look at the plot above we can clearly notice a differentiation based on *CMV*. The first clusters has clients with lower value, the second with intermediate value and the third with higher value. Also, the last cluster includes the clients with higher values for *Total_Premiums* and with a majority of zeros for *Cancelled*. Summing up, the last cluster includes clients that spend more and do not cancel premiums, making it the cluster of valuable clients.

One huge problem with this segmentation was the size of the clusters. Both the first and second clusters have around 5000 observations, while the last cluster only has 666 observations, making it much smaller than the others.

PRODUCT SEGMENTATION

For the product segmentation we wanted to study how customers were grouped according to what they bought. For this view we tried using K-Means with the premium ratios created before and then with the original premiums. We used square roots to remove the tails of distributions and standardized the variables before inputting them into the algorithm.

The results for the clustering with non-ratio variables proved to be better than the ones with the ratio ones. Even though *Household* differentiated more the clusters as a ratio, *Health* was better used as originally received.

For defining the number of clusters, we looked at the dendrogram (figure 21) resulting from the hierarchical clustering. By looking at it, we decided to go with three clusters.

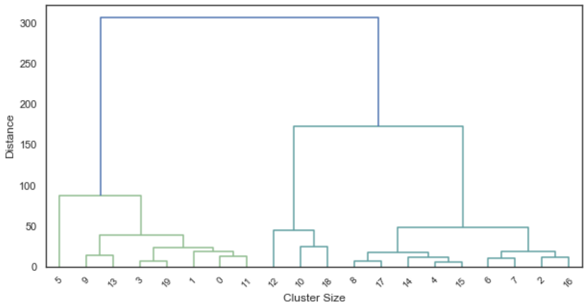


Figure 21 - Dendrogram for product view

The profiling of the resulting segments was as follows:

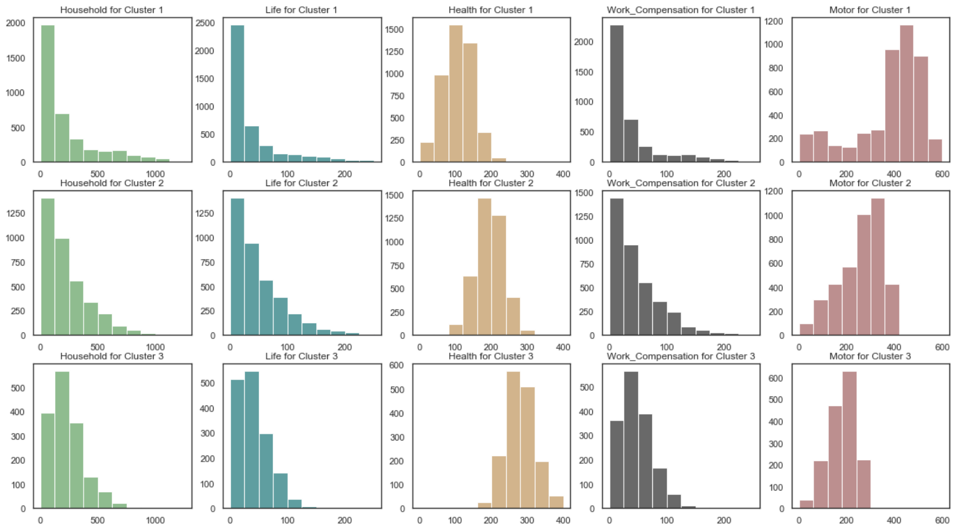


Figure 22 - Profiling for product with K-Means

As mentioned above there is a clear difference between clusters based on *Health*. Similarly to what happened with *CMV* on the value segmentation, the first cluster has the lower values, the second intermediate ones and the last higher ones. This means that the people on the first cluster tend to buy less *Health* premiums and the people on the last cluster more. However, the other variables did not differentiate the clusters a lot, so we decided to try doing this segmentation with other algorithms.

5.2. SELF ORGANIZING MAPS & HIERARCHICAL

One of the most popular clustering algorithms is the Self Organizing Maps (SOM) as it provides dimensionality reduction transforming high-dimensional input space into typically two-dimensional. Adding to this, one of the best characteristics of SOM relies on not making assumptions regarding the variable's distributions, having strong capabilities when dealing with nonlinear relationships and skewed distributions (Petra Perner (Ed.), 2010). So, when applying clustering in each view we set aside the square root transformed variables. Apart from that, we did not perform SOM in the Sociodemographic view as this view is mostly composed by categorical variables and SOM does not handle categorical variables well.

In this algorithm, we have to decide the number of units and sometimes we still get an abstract painting, to overcome this we applied Hierarchical Clustering on top of SOM using the units we got as observations.

VALUE SEGMENTATION

In this view, we had some choices to make regardless of whether to put *CMV* or *Claims_Rate* in the model and of course, the number of clusters to perform. We performed the algorithms with two different configurations for 4, 3 and 2 clusters: *Claims_Rate*, *Client_Years*, *Effort_Rate*, *Total_Premiums* and *Cancelled* for 4, 3 and 2 clusters - **first configuration**; *CMV*, *Client_Years*, *Effort_Rate*, *Total_Premiums* and *Cancelled* - **second configuration**. We concluded that in both configurations, the *Client_Years* variable was not a good discriminator and we did not include it in our analysis.

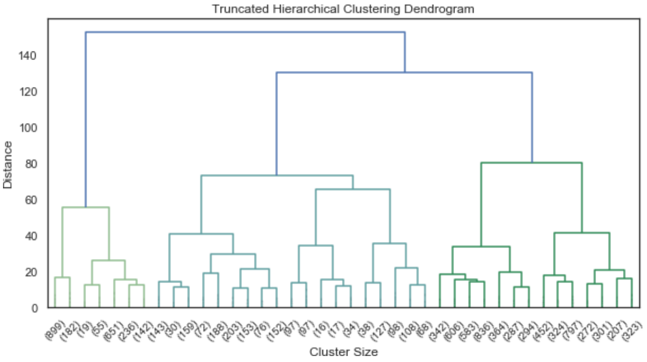


Figure 23 - Dendrogram for Value with SOM

Focusing now on which is the optimal number of clusters for the Value Segmentation, looking at the dendrogram above (figure 23), the best option would be between 2, 3 and 4 clusters. We found the profiling impossible with 4 and to choose between 2 and 3 clusters we compared the histograms in both configurations. Keeping in mind that the value of the customer is an important feature when targeting marketing campaigns, we got a better and more meaningful differentiation of customers with 3 clusters, that we will use from now on.

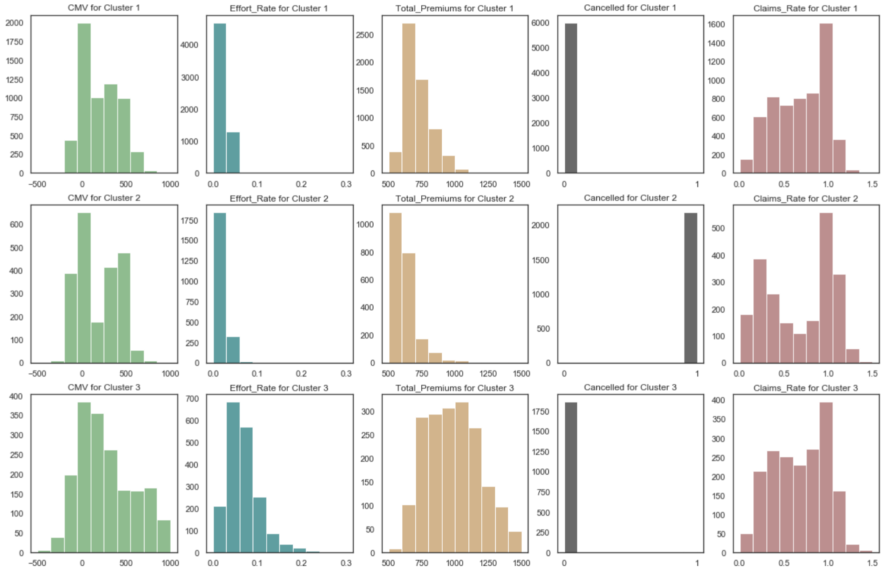


Figure 24 - Profiling for value with SOM

Regarding the configuration 1 and 2, both without *Client_Years*, we can see that *Claims_Rate's* distribution over the 3 clusters is similar and there is no significant difference between them. On the other hand, for *CMV*, cluster 3 has the customers with higher value. We decided to keep the second configuration as our final model for the value segmentation.

Analyzing figure 24, cluster 2 (**2184 customers**) is characterized by customers that spend less than the majority of the customers in the other clusters, with low effort rates and that cancelled at least one insurance for 2016; cluster 1 is composed by **5988 customers** that spend a little more or the same as clients in cluster 2, that did not cancel any insurance(s) and with low effort rates; cluster 3 (**1876 customers**) is the most valuable segment having the customers that spend more with higher effort rates and no cancellations. Having in mind what we just described, we can define the clusters as follows: **Cluster 1: Silver; Cluster 2: Bronze; Cluster 3: Gold**.

PRODUCT SEGMENTATION

On the Product view the work was easier, we had to choose between the original variables - *Household*, *Life*, *Health*, *Work_Compensation* - and the ratios. The number of clusters in this case was clear - 3 clusters, as we can see from the dendrogram below (figure 25).

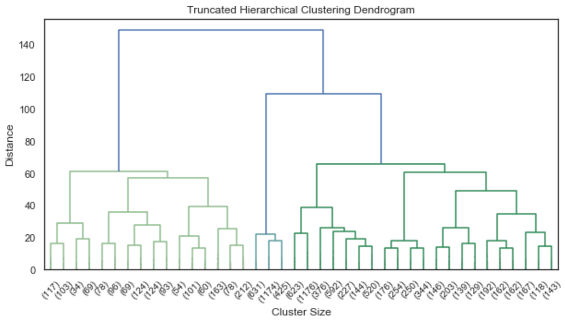


Figure 25 - Dendrogram for product with SOM

When comparing both configurations, the results with ratio and non-ratio variables were almost the same. Analyzing deeply each histogram and comparing them, we can conclude that only *Household* was better with the ratios while the difference between clusters on the other variables is more evident when using non ratio variables, so, we kept the configuration with the original variables as our final one. Although *Motor* was not used to perform the analysis, we took it into account when profiling the clusters, as it is shown on figure 26 below.

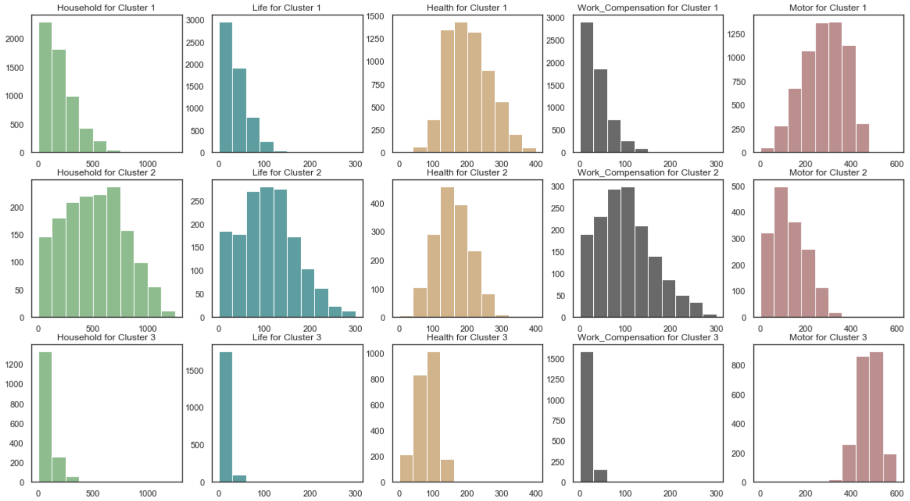


Figure 26 - Profiling for product with SOM

<u>Cluster 3</u> has customers that spend a lower amount in almost every premium except for *Motor*, having the customers that spend more in *Motor*; <u>cluster 1</u> is composed by customers that spend a little more or the same as the clients in <u>cluster 3</u> for every premium except for *Health* and *Motor*, having customers who spend more in health insurance and less in *Motor*; <u>cluster 2</u> includes customers that spend the most in *Household*, *Life* and *Work* and less in *Motor*. We can label the clusters has follows: **Cluster 1: *Health*** (6243 clients); **Cluster 2: *Household* & *Life* & *Work*** (2230 clients); **Cluster 3: *Motor*** (1575 clients).

5.3. DBSCAN

DBSCAN is a density-based clustering method, that uses the concept of distance and takes into account a specific minimum number of points to create groups of observations. Besides this, we could have used this method to detect outliers in an earlier phase, because it also marks as outliers the points found in low density regions.

We will give a try to this method as it has some advantages over the usual K-Means, making possible the discovery of clusters with arbitrary shapes and not only spherical. Unfortunately, it does not only have advantages but also drawbacks that we found difficult to overcome in our analysis, mainly in one specific view. DBSCAN cannot deal well with clusters of varying density since the clustering depends on two parameters that will be used to discover all clusters and cannot be specified for each one of them. After some discussion, we considered that as we were not too strict when excluding the outliers from the analysis, this might be a problem because we would possibly have clusters of different densities and some outliers might be found.

The other difficulty we faced was setting the right values for the two parameters needed, we tried several combinations and ended up choosing the one that gave us the best silhouette value - a measure of how similar an object is to its own cluster compared to other clusters.

When performing the clustering for the Product view we could only get either two clusters when using the original variables - one with 9279 observations and the other with 22 - or one cluster, when using the ratios. And these were the best results we got in terms of silhouette value, when trying different combinations of the parameters, so we discarded these hypotheses right away.

In the Value view, after testing distinct values for the parameters we got very similar results in the two combinations of variables, with both resulting in two clusters very identical to two of the clusters achieved with SOM and 545 observations marked as noise. Having this, we decided to discard the results in this view too, considering that the ones given by SOM were better.

SOCIODEMOGRAPHIC SEGMENTATION

At the end, we were left with the Sociodemographic view. In order to apply DBSCAN to the Sociodemographic variables we decided to normalize the *Yearly_Salary* to have all variables in a similar scale and we created 4 dummy variables to account for the 4 levels of *Education*.

This view gave the best results achieved through DBSCAN with a silhouette value of 0.811 and 8 clusters. To see a graphical representation of the clustering results we did a pca with the same variables and plotted the observations according to the labels given. Figure 27 below shows what we were left with.

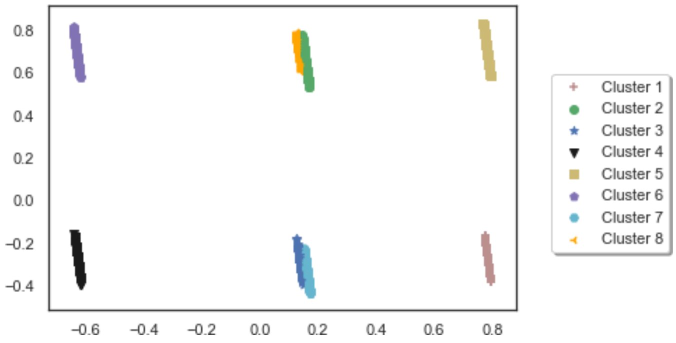


Figure 27 - DBSCAN findings for sociodemographic view

At first, by looking at the figure above, we thought we could join 2 clusters to other 2 in order to have only 6, due to the proximity of their observations in the 2 most important principal components. But when we took a look at the histograms we saw that there were 2 combinations of clusters, 3 by 3, that were very similar - only the *Education* varied -, so at the end we decided to join some clusters and keep only 4 clusters to move one, that have the following composition (figure 28).

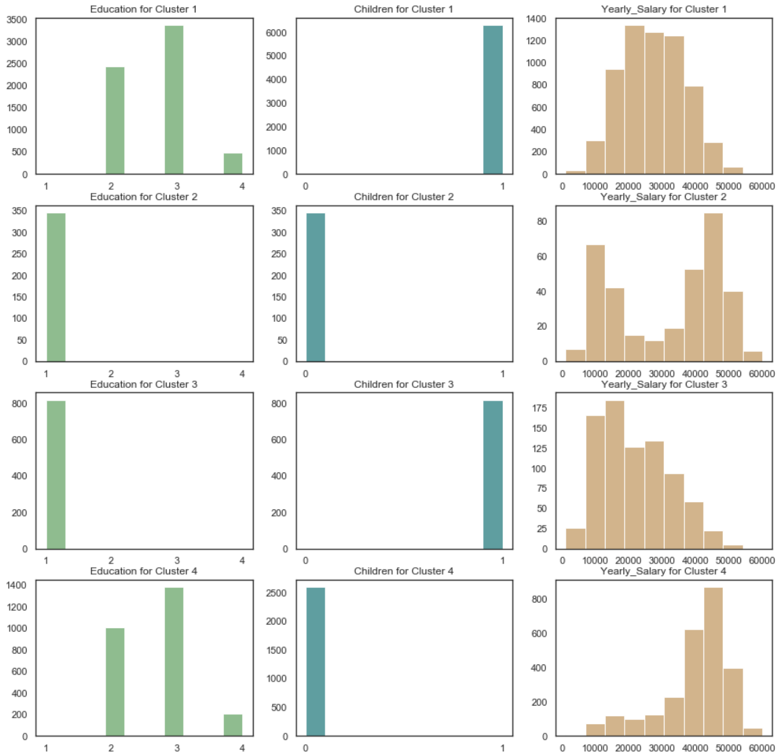


Figure 28 - Profiling for sociodemographic with DBSCAN

Cluster 1 has the customers with high school education and higher, who have children and no extreme values in Salary - has 6292 observations. Cluster 2 has 2591 observations and people with basic education, no children and all types of salary, on the other hand cluster 3 has 819 observations and the customers with basic education, children and less salary but also some with higher ones. To finalize, cluster 4 has 346 observations and the customers with High School education or higher, no children and some of the people with the highest salaries. Summing up: **Cluster 1 - High Education with children; Cluster 2 - Basic Education without children; Cluster 3 - Basic Education with children; Cluster 4 - High Education without children**.

5.4. CLUSTERS OBTAINED

Considering the results obtained for each clustering algorithm in the various views, we chose to stick with the clusters created by DBSCAN for the sociodemographic view since in the results for K-Prototypes there were no major differences between clusters.

For the product view, we decided to use the results created by SOM.

The value view had good results in SOM and K-Means. However, we considered that the differences shown between the clusters created by K-Means were only based on *CMV*. For the first cluster, we clearly had the customers with lower *CMV* and for the third cluster we had the customers with higher *CMV*. The differences noted between clusters for *Total_Premiums* reflected the ones for *CMV* - it is normal that clients with higher value buy more and, therefore, have higher values for *Total_Premiums*. The clusters created by SOM have differences both for *Total_Premiums* and Cancelled. We can also note some minor differences in *Effort_Rate*. Keeping this in mind, we believe it is more beneficial to keep these results instead of the ones derived from K-Means.

6. REASSIGNMENT OF INDIVIDUALS TO CLUSTERS

After joining all clusters from the three views we got 33 clusters where some of them had a small number of observations that does not justify the existence of the cluster itself. We decided to create a cutoff of 200, meaning that we droped the clusters that had less than 200 clients. At the end, we were left with **11 clusters** and **1377 individuals not assigned to any cluster**.

Having this, we had to reassign these individuals to the most similar existing cluster. Even though we did this reassignment, we believe this technique is not perfect as we are basing it in a clustering that may not perfect itself and it might lead to some misclassifications

To perform this reassignment, we used the **Decision Tree algorithm**, one of the most used for classification, having as features only the variables that were used in the clustering phase. We split the data frame with labels into train and test in order to fit the model with the training samples and then checked the algorithm's performance. We got a model that makes good predictions in 92.5% of the times. This metric is not always the best one, because we do not know if there are classes being better predicted than others as this is an overall measure of the model's performance. For further investigation, we also checked the f1 score - a combined measure of precision and recall - for each class and we found that only cluster 16 is being predicted with an f1 score below 0.85, what is a good value for this measure. The most important features for the classification made by the decision tree were *Cancelled* (0.23) and *Children* (0.20) followed by *Health* (0.13), *Motor* (0.12) and *Effort_Rate* (0.12).

It is now important to analyze where the individuals in each dropped cluster went to, in order to see if the results given by our decision tree were acceptable.

When doing the comparison, we found that there are some individuals that previously belonged to the same cluster, that got split into more than one cluster. But there are also cases where all of them went to the same one. The ones that got separated can be explained if we think that although they were in the same clusters, they are not exactly the same. Additionally, we found that 91.58% of the individuals were assigned to a cluster that only differs by one view from the one where they previously were and that was what we were expecting.

After the reassignment of observations, we looked at the profiling again in order to check whether the composition of the clusters remained the same. As expected, the results obtained were not so good, probably because of the misclassifications that might have occurred. However, comparing the profiling for each cluster, we ended up joining two clusters as they only differed in *Monthly_Salary*; two others because they only differed in *Education*; two others due to their similarity that is only broken by the variable *Cancelled*. The final clusters and the respective labels can be found in the following table (table 9). The biggest cluster has 2635 individuals and the smallest 331. To note that the labels give the characterization for each cluster.

Clusters	Value	Product	Sociodemographic	N
1	Silver	Health_Motor	HighEduc_Child	3249
2	Gold	Household_Life_Work	Child	802
4	Bronze	Motor	HighEduc_Child	1970
5	Silver	Health_Motor	HighEduc_NoChild	2033
8	Bronze	Health_Motor	HighEduc_Child	759
10	Bronze	Health_Motor	HighEduc_NoChild	421
18	Silver	Household_Motor_Health	BasicEduc_Child	331
21	Gold	Household_Life_Work	NoChild	483

Table 9 - Final Clusters

For the classification we also tried Neural Networks and K Nearest Neighbors in order to see if it could get a better performance but both algorithms gave poorer results than the Decision Tree.

At this point, the outliers excluded from the previous analysis were assigned to an existing cluster using the same decision tree as above. At first, we did a profiling of the outliers by cluster, but they were so few that we found the analysis irrelevant. Unfortunately, when analyzing the outliers classification, the results are not conclusive and thinking about costs, this step could be skipped if our marketing campaign cost per person is too high once this classification gives us low confidence.

7. MARKETING APPROACHES

Considering the characteristics of the customers in the final clusters, we can suggest some approaches to consider when doing the marketing campaigns for each cluster.

Gold_Household_Life_Work_Child: Create a pack of Premiums of High Quality, that includes Life Premiums for all members of the family, Household and Work Compensation Premiums;
Gold_Household_Life_Work_NoChild: Create a pack of Premiums of High Quality, that includes Life, Household and Work Compensation Premiums, following an up-selling approach;

Silver_Health_Motor_HighEduc_Child: Campaigns focused on Family Health and Motor Premiums with the offer of a discount when purchasing Household Premiums at the same time, promoting cross-selling;
Silver_Household_Motor_Health_BasicEduc_Child: Simple and straightforward campaigns focused in Motor, Household and Family Health Premiums;
Silver_Health_Motor_HighEduc_NoChild: Campaigns focused in Health and Motor Premiums with the offer of a voucher at the time of purchase, to use in Life Premiums;
Bronze_Motor_HighEduc_Child: Campaigns focused in Motor Premiums, with a cross-selling approach for Family Health Premiums, given that they have children;
Bronze_Health_Motor_HighEduc_Child: Campaigns focused in Health and Motor Premiums, with the offer of a discount when purchasing another Premium at the same time;
Bronze_Health_Motor_HighEduc_NoChild: Campaigns focused in Health and Motor Premiums with the offer of a discount if the Premiums are bought for 2 years.

8. CONCLUSION

Regardless of the fact that our initial variables lacked on data about the clients – which made difficult the clustering regarding the sociodemographic view – and essential data to do a RFM analysis, that would have been interesting, we succeeded to meet the initial goal. We identified customers' groups for the insurance company and briefly specified marketing campaigns for each group.

When applying the marketing campaigns, we have to be careful with the budget and probably define priorities. Having this in mind, looking at the clusters' labels and at the number of individuals in each cluster, we can define cluster 1 and 5 as the first priority since they simultaneously represent Silver customers and the largest groups of customers. Cluster 2, 18 and 21 should follow the priorities by this order as they represent Silver and Gold customers. The reason why we are prioritizing Silver and Gold customers instead of Bronze is pretty obvious, we should not spend our money in clients that present an unstable behavior. Between Gold and Silver customers, Gold customers already give profit to the company and, in our opinion, it would be more profitable to invest in Silver customers as they have a good prospect of buying more and need more influence for it than the Gold ones.

9. REFERENCES

Plots, Transformations, and Regression: An Introduction to Graphical Methods of Diagnostic Regression Analysis, by Anthony C. Atkinson

Bhattacharyya, S. (2019, December 11). DBSCAN Algorithm: Complete Guide and Application with Python Scikit-Learn. Medium. https://towardsdatascience.com/dbscan-algorithm-complete-guide-and-application-with-python-scikit-learn-d690cbae4c5d

Chung-Chian Hsu. (2006). Generalizing self-organizing map for categorical data. IEEE Transactions on Neural Networks, 17, 294–304.

Ester, M., Kriegel, H.-P., & Xu, X. (n.d.). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. 6.

How DBSCAN works and why should we use it? - Towards Data Science. (n.d.). Retrieved January 3, 2020, from https://towardsdatascience.com/how-dbscan-works-and-why-should-i-use-it-443b4a191c80

Petra Perner (Ed.). (2010). Advances in Data Mining: Applications and Theoretical Aspects. Berlin ; New York : Springer, ©2010.

Silhouette (clustering). (2019). In Wikipedia. https://en.wikipedia.org/w/index.php?title=Silhouette_(clustering)&oldid=931344504

Standardization in Cluster Analysis. (2018, September 24). Alteryx Community. https://community.alteryx.com/t5/Alteryx-Designer-Knowledge-Base/Standardization-in-Cluster-Analysis/ta-p/302296

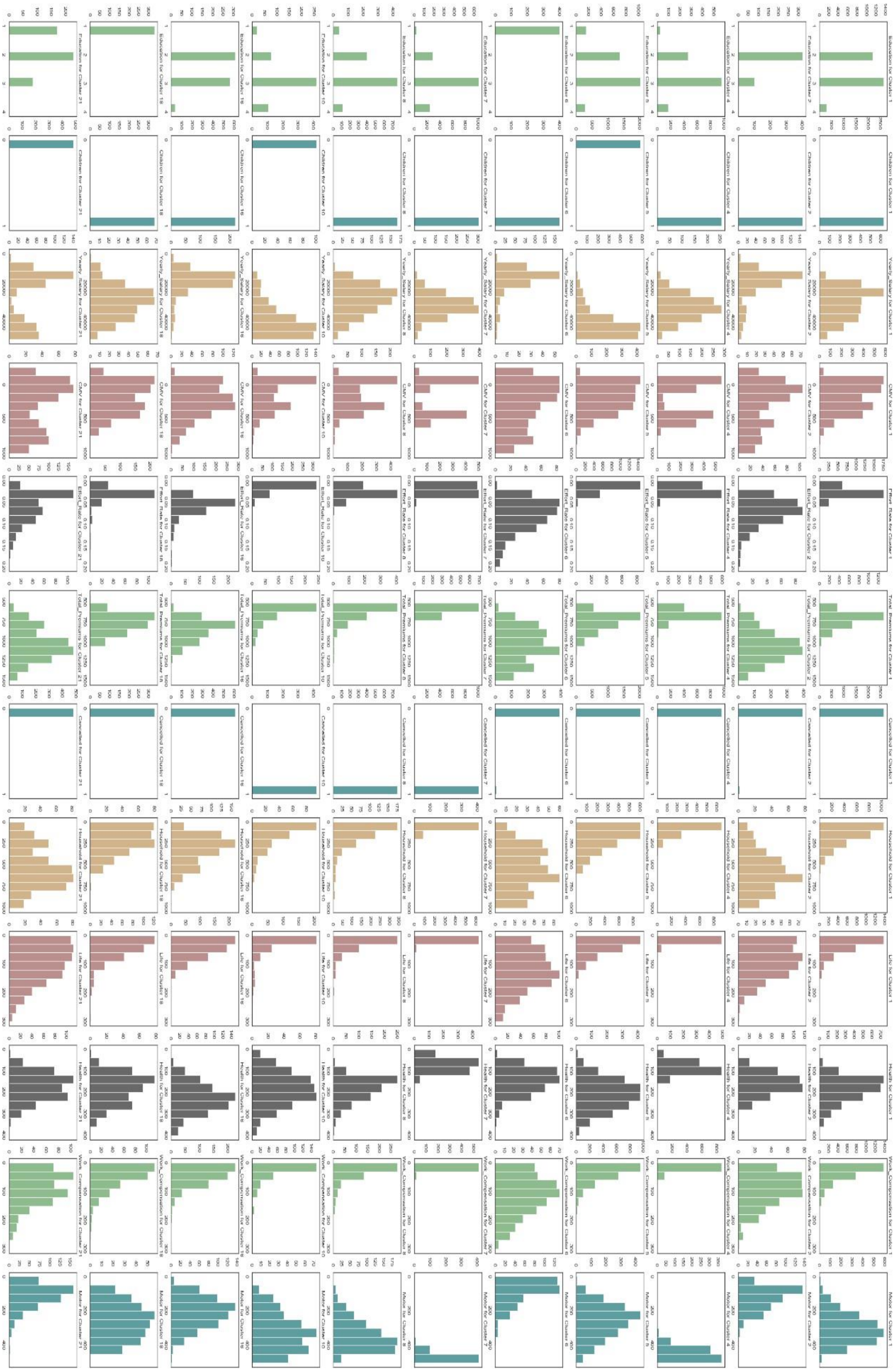
Wade, C. (2018, August 21). Transforming Skewed Data. Medium. https://towardsdatascience.com/transforming-skewed-data-73da4c2d0d16

Yuan, Y. (2019, September 17). RFMT Segmentation Using K-Means Clustering. Medium. https://towardsdatascience.com/rfmt-segmentation-using-k-means-clustering-76bc5040ead5

Asan, Umut & Ercan, Secil. (2012). An Introduction to Self-Organizing Maps. 10.2991/978-94-91216-77-0_14.

ANNEXES

Final Clusters Profiling


Education for Cluster 1
Children for Cluster 1
Yearly_Salary for Cluster 1
CMV for Cluster 1
Effort_Rate for Cluster 1
Total_Premiums for Cluster 1
Cancelled for Cluster 1
Household for Cluster 1
Life for Cluster 1
Health for Cluster 1
Work_Compensation for Cluster 1
Motor for Cluster 1
Education for Cluster 2
Children for Cluster 2
Yearly_Salary for Cluster 2
CMV for Cluster 2
Effort_Rate for Cluster 2
Total_Premiums for Cluster 2
Cancelled for Cluster 2
Household for Cluster 2
Life for Cluster 2
Health for Cluster 2
Work_Compensation for Cluster 2
Motor for Cluster 2
Education for Cluster 4
Children for Cluster 4
Yearly_Salary for Cluster 4
CMV for Cluster 4
Effort_Rate for Cluster 4
Total_Premiums for Cluster 4
Cancelled for Cluster 4
Household for Cluster 4
Life for Cluster 4
Health for Cluster 4
Work_Compensation for Cluster 4
Motor for Cluster 4
Education for Cluster 5
Children for Cluster 5
Yearly_Salary for Cluster 5
CMV for Cluster 5
Effort_Rate for Cluster 5
Total_Premiums for Cluster 5
Cancelled for Cluster 5
Household for Cluster 5
Life for Cluster 5
Health for Cluster 5
Work_Compensation for Cluster 5
Motor for Cluster 5
Education for Cluster 6
Children for Cluster 6
Yearly_Salary for Cluster 6
CMV for Cluster 6
Effort_Rate for Cluster 6
Total_Premiums for Cluster 6
Cancelled for Cluster 6
Household for Cluster 6
Life for Cluster 6
Health for Cluster 6
Work_Compensation for Cluster 6
Motor for Cluster 6
Education for Cluster 7
Children for Cluster 7
Yearly_Salary for Cluster 7
CMV for Cluster 7
Effort_Rate for Cluster 7
Total_Premiums for Cluster 7
Cancelled for Cluster 7
Household for Cluster 7
Life for Cluster 7
Health for Cluster 7
Work_Compensation for Cluster 7
Motor for Cluster 7
Education for Cluster 8
Children for Cluster 8
Yearly_Salary for Cluster 8
CMV for Cluster 8
Effort_Rate for Cluster 8
Total_Premiums for Cluster 8
Cancelled for Cluster 8
Household for Cluster 8
Life for Cluster 8
Health for Cluster 8
Work_Compensation for Cluster 8
Motor for Cluster 8
Education for Cluster 10
Children for Cluster 10
Yearly_Salary for Cluster 10
CMV for Cluster 10
Effort_Rate for Cluster 10
Total_Premiums for Cluster 10
Cancelled for Cluster 10
Household for Cluster 10
Life for Cluster 10
Health for Cluster 10
Work_Compensation for Cluster 10
Motor for Cluster 10
Education for Cluster 16
Children for Cluster 16
Yearly_Salary for Cluster 16
CMV for Cluster 16
Effort_Rate for Cluster 16
Total_Premiums for Cluster 16
Cancelled for Cluster 16
Household for Cluster 16
Life for Cluster 16
Health for Cluster 16
Work_Compensation for Cluster 16
Motor for Cluster 16
Education for Cluster 18
Children for Cluster 18
Yearly_Salary for Cluster 18
CMV for Cluster 18
Effort_Rate for Cluster 18
Total_Premiums for Cluster 18
Cancelled for Cluster 18
Household for Cluster 18
Life for Cluster 18
Health for Cluster 18
Work_Compensation for Cluster 18
Motor for Cluster 18
Education for Cluster 21
Children for Cluster 21
Yearly_Salary for Cluster 21
CMV for Cluster 21
Effort_Rate for Cluster 21
Total_Premiums for Cluster 21
Cancelled for Cluster 21
Household for Cluster 21
Life for Cluster 21
Health for Cluster 21
Work_Compensation for Cluster 21
Motor for Cluster 21