Data Mining Project

MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS

Cluster approach in an insurance company

Group CD

Danilo Arfeli, number: 20211296

Gabriel Avezum, number: 20210663

Tomás Peixoto, number: 20210993

Janeiro, 2022

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

INDEX

1. Introduction

Nowadays the availability of data is so large, fast or complex that it's difficult or impossible to process it using traditional methods. It is becoming increasingly important to be able to handle the availability of big data. In that sense, if a company wants to increase their competitiveness, modern techniques shall be used to extract important insights from the data. For this project, the data is regarding a fictional insurance company in Portugal, a customer clustering technique is suggested and considered to be essential to have a better understanding of the customers.

Clustering models are used to build segmentation profiles, to help areas like the Marketing Department understand the different customers. It can also be used for different applications like improving strategies, detecting outliers, and improving classification or regression models. The project's goal is to produce a clustering segmentation, with its interpretation output delivered to the Marketing Department. Therefore, they will have a better understanding about customer's behaviour and adjust their campaigns accordingly.

This project contains a brief description of the theoretical model, methodology strategy, descriptive analysis, heuristic interpretations of the dimensions, suggested cluster segmentation and strategies that could be used in them.

To produce results in a cluster model several paths exist that will lead you to the result, in this project it was decided to use the Cross Industry Standard Process for Data Mining (CRISP-DM) with few adjustments. CRISP-DM is a methodology that shows flexibility and it is useful when solving analytical business issues. It is composed with six phases that naturally describe the data science life cycle and is presented in figure 7.1.

2. Business and Data Understanding

2.1. Data Understanding

For this project the 10.296 customers of an insurance company from 2016 were provided with 14 columns, all the customers have their own ID that means that we have just one row per customer. The variables provided and created can be found in the table 7.1, with the feature description and missing values percentage.

2.2. Coherence check

The coherence check summary is presented on table 7.2. The following paragraphs provide further explanation.

Following some data exploration, the two variables that have clearly shown a lack of coherence were the "FirstPolYear" against the "BirthYear".

The customers with the first policy year smaller than the birth year account for a total of 1994 instances, which represents approximately 19.3% of the database. A plausible explanation could be that these observations refer to family insurances, whereby the customer was now the child with the respective birth year, but the policy year would refer to the year the family has started the insurance, for instance, through his parents. However, out of the 1994 observations, 337 had no children (condition nº1).

With regards to the ones who had children, considering the child was less than 18 years old, a total of 159 observations have shown up. These instances present values to variables such as “MonthSal” and “PremMotor”, which do not make sense for a person of that age. Therefore, one possible explanation could be that some variables of each observation refer to the parents and others to the child (condition nº2).

With a total of 496 observations (337+159) that cannot be properly explained, we have decided to swap the 1994 values from the two referred variables, assuming it was an imputation or informatic error in the database.

The condition nº3 refers to customers with less than 16 years old, but with a child, which despite possible is very unlikely.

We have verified there are no customers with less than 16 years old, but with a Bachelor or PHD degree (condition nº4).

Finally, we have found a Birthyear equals 1028 and a first policy year equal to 53784. Each value results from the condition number 5 and 6, respectively. These values were replaced by missing values in order to be treated in the following steps.

2.3. Missing values

The percentage of missing values is very low, as can be seen in Table 7.1. The variable “Life” is the one with the highest value around 1%. The missing values were initially replaced using the median and mode, for the numeric and categorical variables, respectively. That’s necessary to properly use the LOF method, a multidimensional outlier detection technique that will be presented in the following subchapter.

After the outlier treatment, the conditions were met to properly handle the numerical variables missing values. Firstly, the original missing values have been reintroduced in the database. Then, the KNN imputation method has been applied, using the parameters (neighbours=5, weights='distance').

Categorical variables distribution after the fast-missing values treatment can be seen in the barplot below.

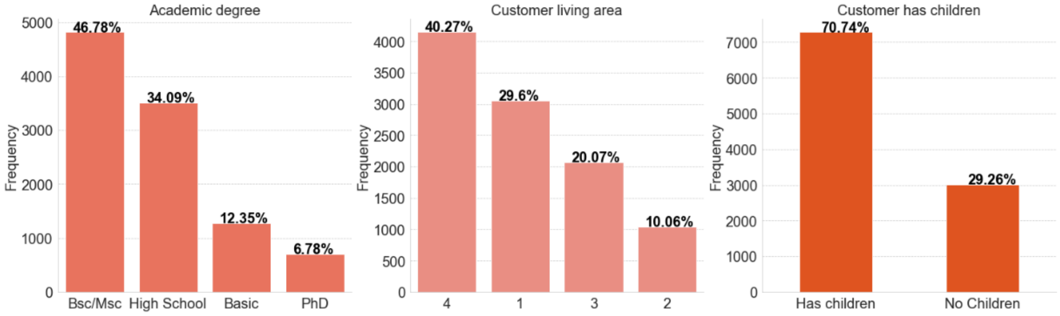


Figure 2.3.1 – Frequency for the categorical variables

From the graph above we can conclude that most of our customers has an academic degree equal Bsc/Msc or High School. The customer living area is concentrated in the areas 4 and 1, but none description has been given. The last barplot presents that the most of our costumers has children.

2.4. Outlier treatment

2.4.1. Boxplot Visualization and IQR

We can consider that all numerical variables have outliers, through the visualization of the boxplots shown in figure 7.2.

The variable “MonthlySal” has two points that do not fit to the distribution given by the boxplot causing its distortion. These two points represent extreme values, which corresponds to a very small percentage of records compared to the full data. We have decided to apply the IQR method getting a maximum upper threshold value of 5659€, that corresponded to the removal of the two outliers points mentioned above.

The input “CustMontVal” has several outliers in negative and positive directions. We could test several thresholds to find the values that would fix the distribution, but this could be tedious and infeasible. In this case, we have used the upper and lower limit given by the IQR. The lower and upper limit values are -623 and 1013, respectively, corresponding to dropping 1.07% of the total records.

With regards to the “PremMotor” input, we could have removed manually the 6 outliers’ points, but instead we have used the IQR method to get on optimum threshold for the feature that will have an upper limit value of 731. It led to the removal of the referred 6 points, that represent 0.06% of records.

The variable “Claimsrate” have few outliers in the positive direction. Using the IQR method, we also got an optimum threshold with an upper limit value of 1.865, which corresponded to drop 0.15% of records. An important observation has to be considered for this variable because the extreme values can mean that the company is probably losing money for some customer and this can indicate fraudulent behaviors of the customers.

The “Premheath” variable has multiple outliers. The IQR method returned an upper limit value of 380, dropping only 0.29% of data.

The boxplots with the outliers removed is presented in figure 7.3.

For the variables not mentioned above, the IQR method was not considered because it would lead to a high number of records dropped. We have decided to use another strategy presented below.

2.4.2. Local Outlier Factor (LOF)

We would like to use a multidimension outlier method to detect more accurately possible outliers not covered on the previous subchapter. In addition, the variables “PremHousehold”, “PremLife” and “PremWork” had still to be treated. Therefore, we have decided to use Local Outlier Factor algorithm to accomplish both goals.

According to the reference [1], the algorithm outputs the score for each observation and with it, we can define a threshold. There's only one parameter, the nearest neighbours (MinPts), that needs to be set to define the local neighbourhood of the observations in the input space. Following the same reference [1], the best way to define the parameter is to test the algorithm in a range of MinPts and retrieve the maximum LOF score for each observation, with the score being tested using all the values between the range 20 and 70 MinPts. For the sake of clarity, different observation scores can use different MinPts, and it is the algorithm that searches for the MInPts that maximize the score for each observation.

We have used the algorithm from the package sklearn.neighbors. In this package, the LOF score is negative, hence, we have used the minimum LOF score as the score of each observation. After all observation scores have been calculated, the figure 2.4.2.1 was built. Using the figure and testing different thresholds, we have decided that all observations with a score below -1.40 was an outlier.

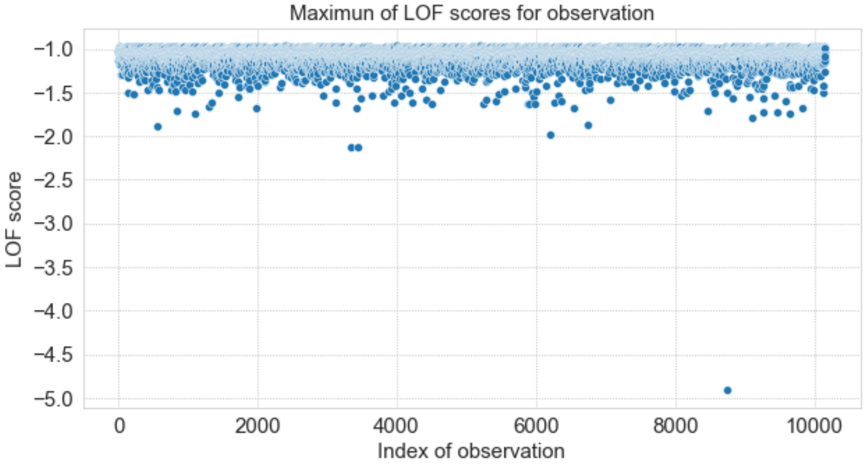


Figure 2.4.2.1 – LOF scores for each observation

After this process, the continuous variables boxplots were updated (figure 7.3). It's possible to see an improvement of the variable's distributions. The total number of outliers removed was 2.42% percentage of the initial database.

2.5. Feature engineering

To extract and have a better insight from the database, we have decided to build new features. This can be accomplished by transforming one variable or by combining the initial variables from the database.

In table 7.1, we have presented the new variables. Now, we will explain how the calculation was done and provide a brief description about them.

1. "ActualAge": is the customer age. We have considered the year of 2016 as the base, because it says so in the project description. The calculation is done by subtracting the variable "BirthYear" from 2016.
2. "TimeCustm": Time passed since the first policy. Following the same logic as described in "ActualAge" this variable was calculated by subtracting the variable FirstPolYear from 2016.
3. "PremTotal": Sum of all premiums. Simply the sum of all premiums

4. “income_commit”: Sum of all premiums “PremTotal” divided by the annual salary (12 times “MonthSal”). We wanted to understand the proportion of the salary that the customers spent in the policy.
5. “AgeFirstPol”: Age of the customer in their first policy. Simply subtracting “BirthYear” from “FirstPolYear”.

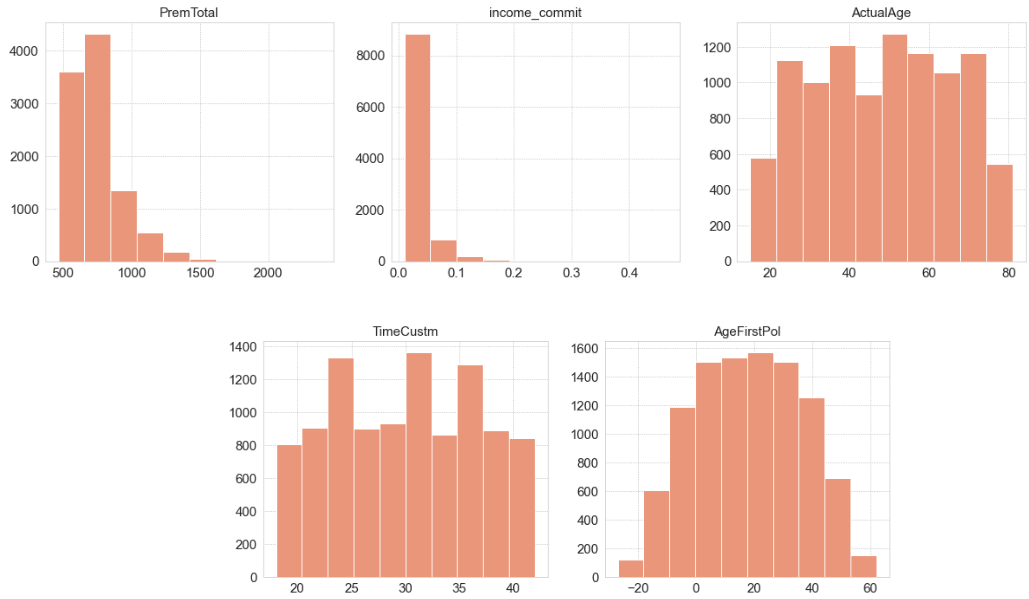


Figure 2.5.1- Histograms for the new variables.

In the figure above, we can see that the variable “TimeCustm” is evenly distributed. “ActualAge” has the both ends with a lower frequency, but the values between is evenly distributed. “AgeFirstPol” has a normal like distribution with the values between -20 and 60. The last two variables, “PremTotal” and “income_commit”, are right-skewed. Since we have treated the outliers before, the new variables will not present outliers.

3. Modelling

3.1. Feature selection

Looking to our variables, it is possible to understand they have two different groups of origin. The first one is the **Sociodemographic**, that contains the variables age, salary, time of customer, education degree and children. The second is the group of **products_values**, that encompass the variables related to all premiums, claims rate and the income commit. So, we have decided to split the cluster into two different sets of variables and use a cluster method for each one. Now, before doing the feature selection we need to keep in the mind these splits.

To select our variables, we will use the two main points of feature selection. One is relevancy, where the variables without impact in our dataset will be set aside and will not be used in the cluster modelling. The other one is redundancy between important variables, because to build the model is important to not have the same information being repeated.

3.1.1. Correlation

The spearman correlation has been applied to all numerical variables, including the new ones created, to avoid using redundant variables when creating the clusters. The results are presented on Figure 7.4 in the appendix.

The pairs of variables highly correlated (threshold above 0.90) are listed below. The ones kept are highlighted in bold. The decision has been based on the variables' meaning easiness of interpretation, meaning to say, it is easier to read the actual age of customer rather than the Birthyear. In addition, it has been accounted for each variable, the total number of variables with which they are highly correlated to. The ones with the highest value have been removed.

- **ClaimsRate** & CustMonVal
- **PremHousehold** & PremTotal
- **TimeCustm** & FirstPolyYear
- **Actual Age** & Birthyear
- **Actual Age** & AgefirstPol

With both the numerical and categorical variables to use decided to create the final clusters, we have chosen to segment them based on their meaning and type of the problem studied, with the goal to create better clusters and to facilitate its interpretation.

3.1.2. Categorical variables

Firstly, the figure 7.5 has been created to understand the meaning of each continuous variable with respect to the variables "geoarea" and "children". We see that most of the graphs, present the lines split between "No children" and "children" almost parallel, so we can conclude this variable is important to our clustering. However, in both lines the values inside the geo area levels doesn't change, so this variable is not that relevant.

Doing the same analysis with the education degree the figure 7.6 was built. We can see that in all graphs there is a relation between the continuous variables and the education degree. For example, in the "PremMotor" graph, it is possible to see that in each different level of education degree, the mean of the continuous variable is different and is increasing.

3.2. Clusters

3.2.1. Hierarchical/K-means

The hierarchical and k-means methods are the ones we have considered to create the clusters with the **product_value** variables. The reason is all are numerical continuous variables, with no binary ones which are known to have a certain impact on the creation of the clusters.

Firstly, we have standardized the data using the StandardScaler method.

Afterwards, both methods have been evaluated using the R^2 metric as a measure of the homogeneity of the cluster's solution. For each method, it has been considered a total number of clusters varying from 2 to 9. In addition, for the hierarchical method, 4 different aggregation rules have been assessed.

The results summary is presented on the figure below.

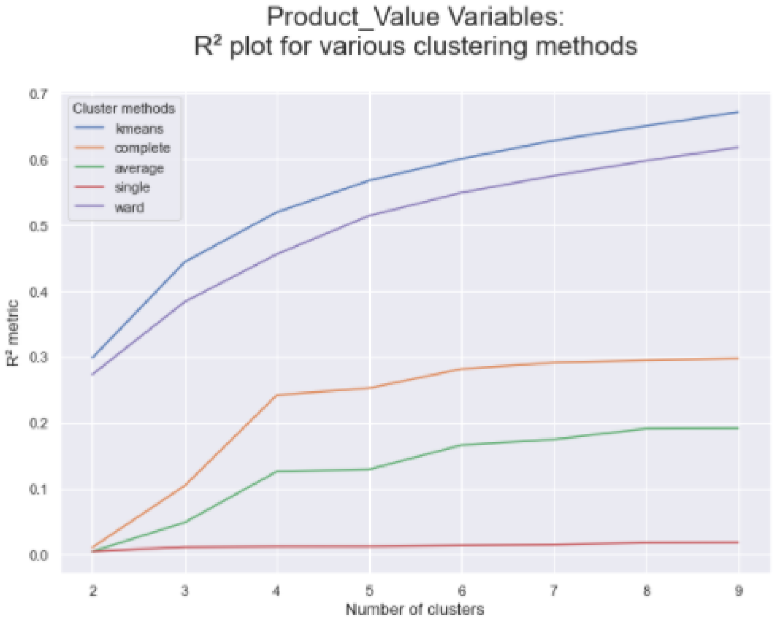


Figure 3.2.1.1: R^2 metric for different cluster methods

Kmeans technique reveals to provide better results than the hierarchical.

Then, we have combined both techniques to check if the R^2 value would improve and to have a graphical view with the Ward's dendrogram to better understand the distance between the different clusters. It has started by clustering with kmeans using 50 clusters. The hierarchical clustering is applied on top of it, with the results presented on Figure 7.7 in appendix. The combination of both Figure 3.2.3.1 and Figure 7.7 suggests 4 as the optimum number of clusters.

The solution with only Kmeans and the one with hierarchical on top of the Kmeans as presented above, outputs a R^2 value of 0.520 and 0.394, respectively for 4 clusters. Hence, only Kmeans has been the technique chosen to create the final product_value clusters. The centroids are show on figure 7.8, which clearly show a good segmentation in respect to the different variables.

3.2.2. K-prototypes

One alternative to use categorical data in a cluster algorithm that uses Euclidean distance is to use a method called K-Prototype (proposed by Huang), which is a method that can use mixed data types (combines the k-means and k-modes algorithms). It measures distance between numerical features using Euclidean distance (same as K-means), but also measures the distance between categorical features using the number of matching categories (same as K-modes).

The k prototype algorithm was applied in the features that we have called **Sociodemographic**. The method will use the different distance measures for the numerical variables (MonthSal, ActualAge, TimeCustm) and for the categorical ones (EducDeg and Children).

As an initial step of the algorithm, it is necessary to choose a number of K. For this purpose, it was done an elbow curve fitting k in a range from 2 to 10. Unlike k-Means, k-Prototypes do not provide us with inertia scores, but instead the cost score (sum distance of all points from their respective cluster centroids) variable to assess the goodness of fit. The elbow curve with the cost score is presented below.

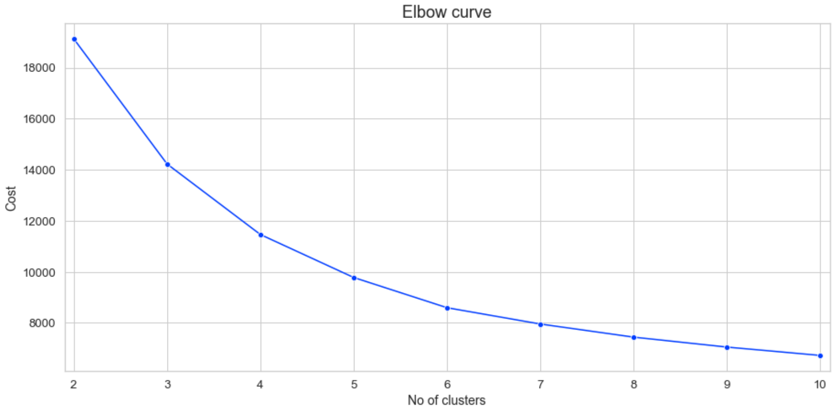


Figure 3.2.2.1: Elbow curve for the K-prototype method

The elbow curve plot suggests that an optimum number of clusters is around 4 or 5, as we can see that the cost tends to have lower decrease further the k. It was tested the summary statistics for these 2 numbers of k, and the best segmentation was presented with k equals to 4.

It is observed from the figure 7.9, the algorithm has provided distinguished clusters in terms of categorical features, especially the Children input. The cluster labelled as 1 has the highest no children frequency. In terms of education degree, the cluster 3 outputs the lower levels of education. In the numeric features, we have an expected segmentation on “MonthSal” and “ActualAge” (correlated features). Their averages values are as following per each cluster: cluster 1 > cluster 2 > cluster 0 > cluster 3. In addition, cluster 0 and 1, each with an average of “TimeCustm” around 33-35 years differentiates well from the cluster 3 and 2, each with an average of “TimeCustm” around 23 years.

3.2.3. Final Cluster

The deployment of two different clustering techniques on the different variables group of origin has resulted on four clusters for each one. Their combination results in a total of sixteen clusters, which is too many to be able to construct a proper marketing strategy. To reduce these number a hierarchical cluster solution was used. Before using it, we had to calculate the centroid of each sixteen clusters. First, we had to scale our variables using the StandardScaler. Then, we have grouped the dataset by the sixteen clusters and took the mean of the numerical variables, which were used in the K-means and in K-prototypes.

Now we can pass the centroids in the hierarchical cluster and see the Ward’s dendrogram to understand how many clusters we will keep. Based on the figure 7.10, we have decided to keep six clusters. Before moving on, we had to check the number of observations in each one and see the UMAP to determine if this is a possible solution.

In figure 7.11, we can see that the cluster 0 has less than 500 observations, which represents approximately 2% of the total observations. We have decided to regroup this cluster. To do it, we can look at the UMAP and see which cluster is closer. Looking in the figure 7.12, it is possible to see that cluster 0 has points across three different clusters, so is not possible to group all cluster 0 into a single cluster.

Now to overcome this problem, we have decided to build a KNN classifier to predict the new class of observations in cluster 0. To train the KNN we have removed the cluster 0 from our train set and test set and used this model to predict their class. The KNN model returned a 95% accuracy in our test set.

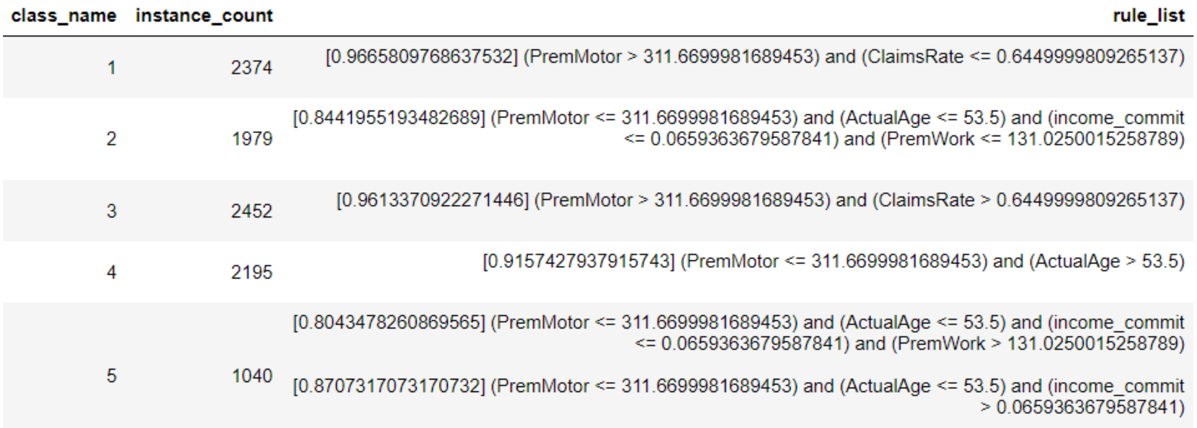
After this procedure, the final merged cluster solution distribution is presented in figure 7.13. The five clusters have a good number of observations each. In addition, the UMAP visualization shown in figure 7.14 allows to conclude the final clusters are well defined and split, which is a good indicator we had a good final cluster solution.

4. Business Strategies

4.1. Cluster understanding and Marketing strategies

We have used different techniques to interpret the final clusters.

The decision tree classifier algorithm has allowed to understand which are the most important attributes and their values to distinguish the different clusters. The figure below, a screenshot from the notebook, presents a summary. The first value on each "rule_list" column, indicates the percentage of the respective cluster that complies with the rule presented. For instance, for the cluster number 1 is 96.7% approximately.



class_name	instance_count	rule_list
1	2374	[0.9665809768637532] (PremMotor > 311.6699981689453) and (ClaimsRate <= 0.6449999809265137)
2	1979	[0.8441955193482689] (PremMotor <= 311.6699981689453) and (ActualAge <= 53.5) and (income_commit <= 0.0659363679587841) and (PremWork <= 131.0250015258789)
3	2452	[0.9613370922271446] (PremMotor > 311.6699981689453) and (ClaimsRate > 0.6449999809265137)
4	2195	[0.9157427937915743] (PremMotor <= 311.6699981689453) and (ActualAge > 53.5)
5	1040	[0.8043478260869565] (PremMotor <= 311.6699981689453) and (ActualAge <= 53.5) and (income_commit <= 0.0659363679587841) and (PremWork > 131.0250015258789) [0.8707317073170732] (PremMotor <= 311.6699981689453) and (ActualAge <= 53.5) and (income_commit > 0.0659363679587841)

Figure 4.1.1 – Rules for each cluster

Based on above, the variables with the most discriminative power are "PremMotor", "ClaimsRate", "income_commit", "ClaimRate" and "ActualAge".

Moving to graphic visualization to further understand the different clusters along the numerical variables, a pointplot graph with the variable's average per cluster is shown on the figure 7.15.

In addition, the figures 7.16 and 7.17 clearly illustrate the distribution of the categorical variable's "education" and "children", respectively, per cluster.

Based on the information above, the Table 7.3 presents a self-explanatory descriptive summary with the main characteristics of each cluster and with a label attached to it.

The cluster interpretation allows us to tailor the following marketing recommendations.

Motor profit are the ones that value their cars and are safe drivers. Long-time customers that have returned highest amount of money in total to the insurance company. The marketing team should benefit these customers, by providing higher discounts in relation to the other premiums, since they invest little on these and are a safe customer.

Health concerned are the customers that value their health, still being young. Since they have children, the marketing campaign should focus on the premium Health, by offering packages to all family.

Motor loss are the customers that value their cars but are unsafe drivers. Must have a lot accidents, hence, the CMV is null despite being a client for almost 30 years. The marketing campaign should send videos to raise the awareness about the risk of unsafe driving. In addition, based on their behaviours and since they have children, a focus on the premiums health and life shall be given, to support them and their family, respectively, in case of a serious accident.

Retired are the customers with an average of 68 years and no children. They have a high salary but not high premiums apart from the health premium. Campaigns mostly focused on premium life, but also motor and household should be the target.

Most valuable are intriguing customers since they have low salary, are the youngest ones with an average of 30 years old, have children, but still are the ones with the highest total premiums, and that's the reason we consider our most valuable customer. Instead of focusing in selling more premiums to them, the campaign should only focus in selling cheap packages to their family, not only to get new customers, but mostly as a gesture of appreciation of their effort and commit towards the company.

4.2. Outliers and new customers

Following the marketing strategy solution, we had to build a method to associate possible future new customers to the clusters. So, a decision tree classifier with an accuracy of 91% was built. Using this tree, we classified the outlier observations and delivery a final database with all the customers assigned to at least one cluster. The referred decision tree is depicted in figure 7.18.

5. Conclusion

It is worthwhile to mention, we have tried DBSCAN and Manshift clustering algorithm but the initial results were not promising, with some observations being defined as outliers despite we had done the outliers treatment before. On the other hand, these algorithms had trouble to create segmented clusters despite our efforts finetuning their algorithm parameters.

With regards to SOM algorithm, the visualization advantage is surpassed by the UMAP and is considerably more computationally expensive than the other clustering algorithms.

Since, we were getting good cluster segmentation by using both K-means and Kprototypes, we have decided to not further investigate the SOM.

The proposed cluster segmentation allows the marketing area to explore very well different kinds of customers, since its observed segments that distinguish customers profiles in motor premiums with profitable and non-profitable clients, one cluster focused exclusive for health, segments with children and non-children, old clients and new clients and one cluster of top customers. This gives a huge opportunity to make better and data driven strategies.

6. References

[1] - Breunig, M. M., Kriegel, H. P., Ng, R. T., & Sander J. (2000, May). LOF: identifying density- based local outliers. *ACM sigmod record, vol 29, No. 2*, pp 93-104

7. Appendix

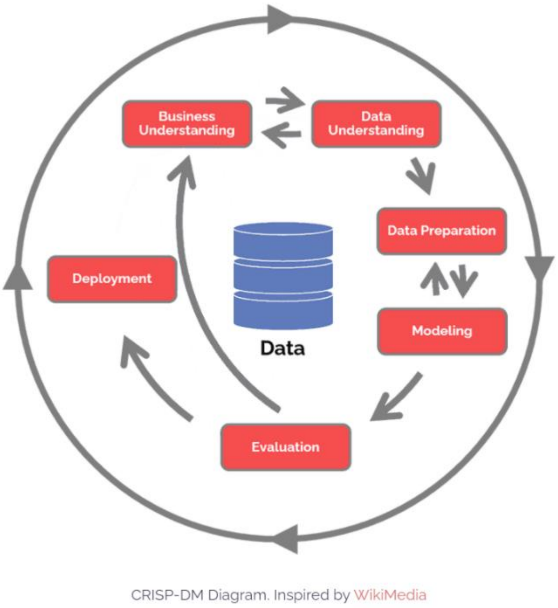


Figure 7.1: Crisp-DM diagram

Variable	Type	% Miss	Description
ID	Num	0.00%	Customer unique ID
First Policy	Num	0.29%	The first year that the customer did a policy
Birthday	Cat	0.16%	Customer birthday year
Education	Cat	0.16%	Academic degree
Salary	Num	0.34%	Customer gross monthly salary (€)
Área	Cat	0.01%	Customer living area
Children	Bin	0.20%	Customer has children (1) or no (0)
CMV	Num	0.00%	Customer Monetary Value
Claims	Num	0.00%	Claims rate: company paid / Premiums (€) (Last 2 years)
Motor	Num	0.33%	Premiums (€) in LOB (2016): Motor
Household	Num	0.00%	Premiums (€) in LOB (2016): Household
Health	Num	0.41%	Premiums (€) in LOB (2016): Health
Life	Num	1.10%	Premiums (€) in LOB (2016): Life
Work Compensation	Num	0.83%	Premiums (€) in LOB (2016): Work Compensation
*ActualAge	Num	**0%	Customer age based in 2016
*TimeCustm	Num	**0%	Time passed since the first policy based in 2016
*PremTotal	Num	**0%	Sum of all premiums
*income_commit	Num	**0%	Sum of all premiums divided by the annual salary
*AgeFirstPol	Num	**0%	Age of the customer in their first policy

Table 7.1: Variables provided and new features created

* : means that the variable was created in the feature engineer.

** : the created variables will have 0% missing values because they were created after the missing values treatment.

Number	Condition (" Code")	Frequency
1	(FirstPolYear < BirthYear and Children==0)	337
2	(FirstPolYear < BirthYear and Children ==1 and BirthYear >=1998)	159
3	(FirstPolYear > BirthYear and Children ==1 and BirthYear >2000)	0
4	(BirthYear >=2000 and (Education == ("BSc/MSc" or "PhD"))	0
5	(BirthYear <1890 or BirthYear >2016)	1
6	(FirstPolYear <1890 or FirstPolYear >2016)	1

Table 7.2: Summary of coherence check

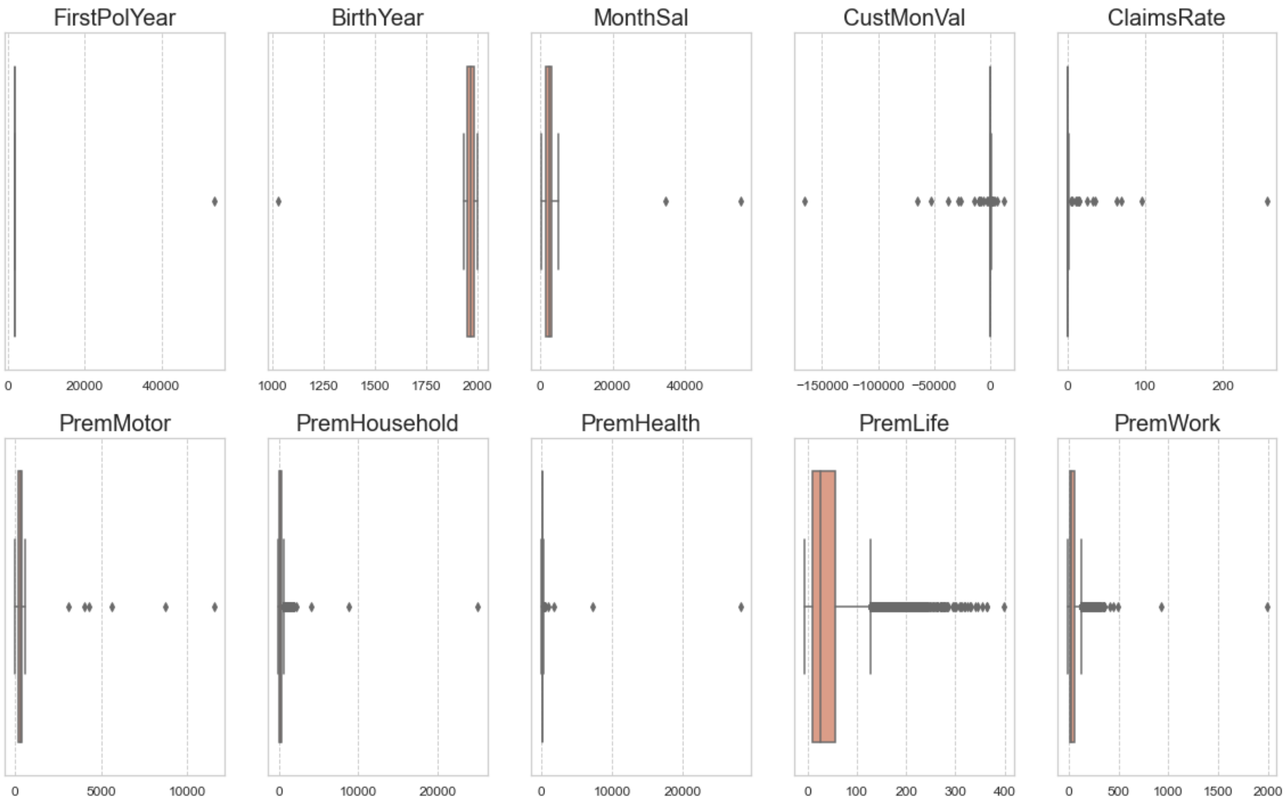


Figure 7.2 – Boxplots of continuous variables before outlier detection

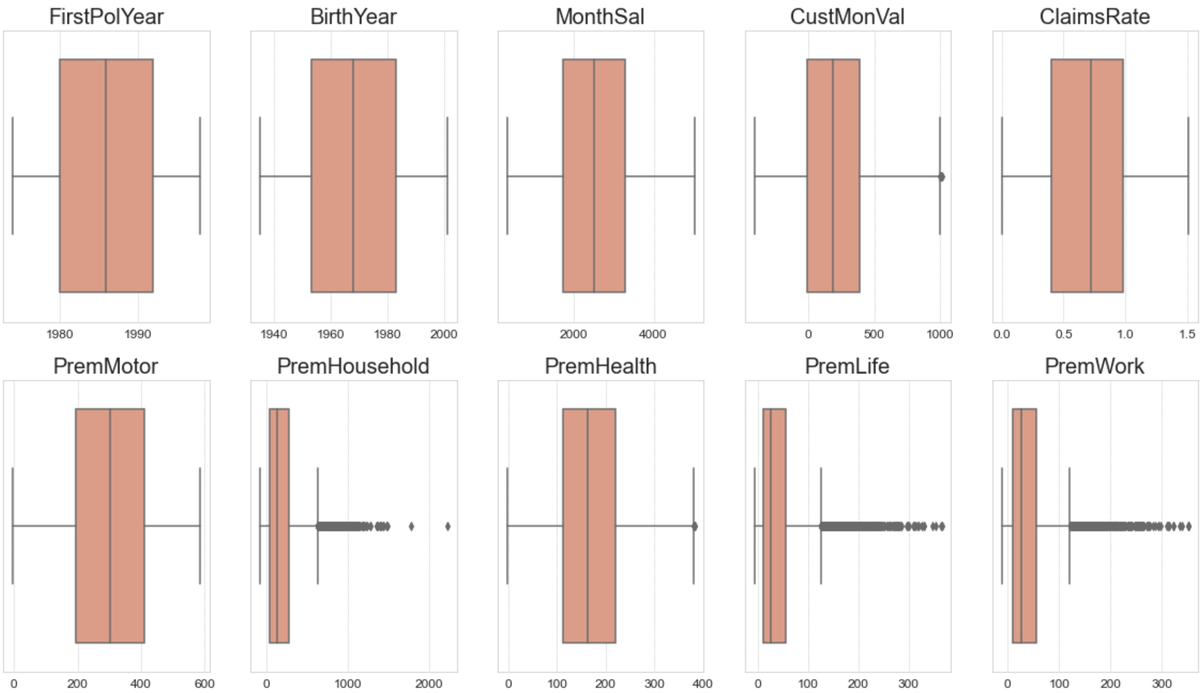


Figure 7.3 – Boxplots of continuous variables after outlier detection

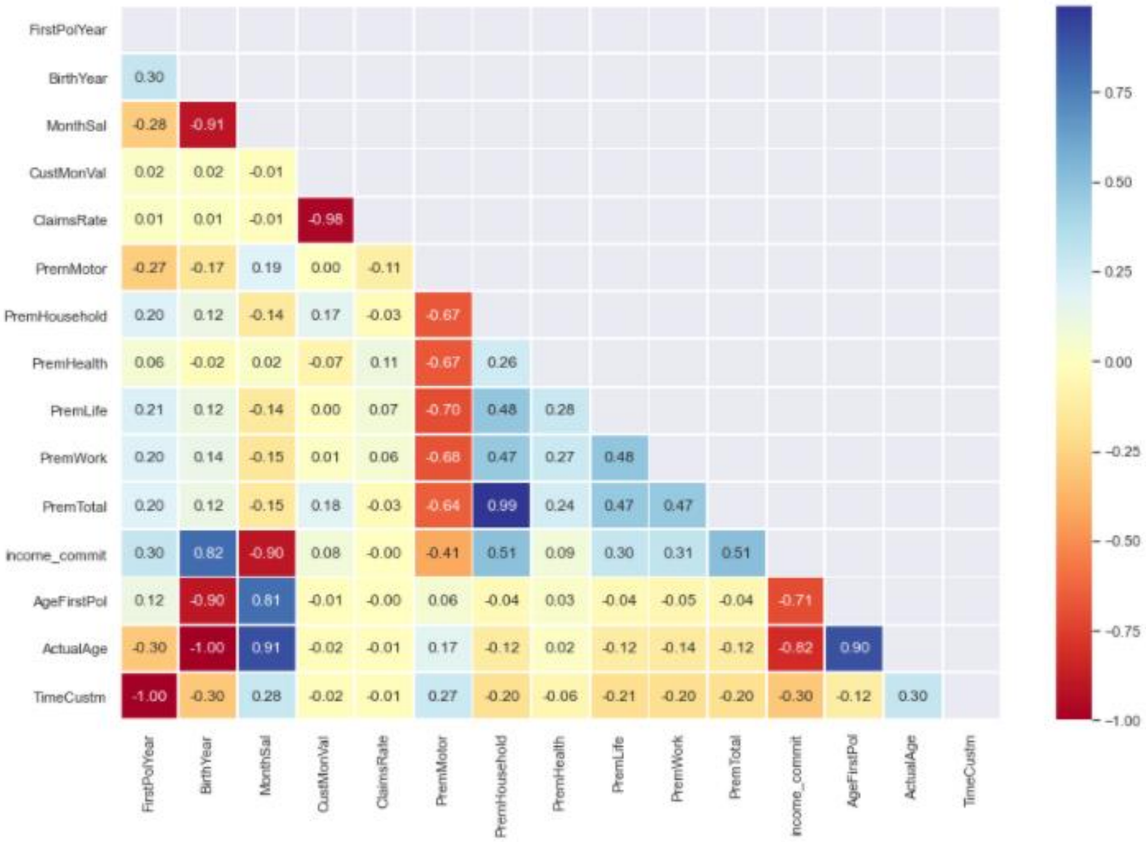


Figure 7.4: Spearman correlation numerical variables

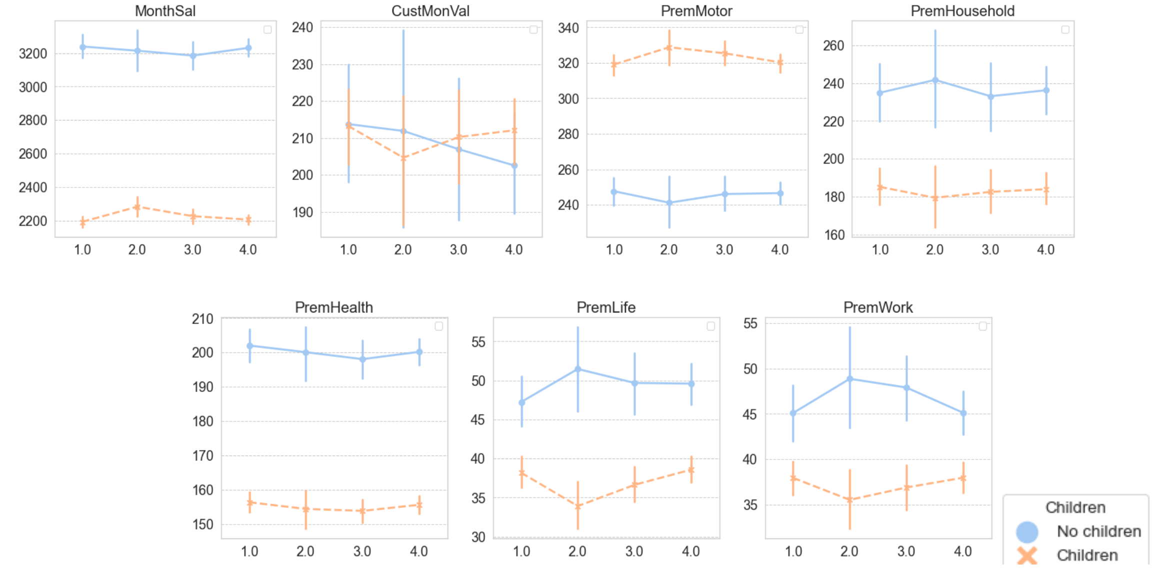


Figure 7.5 - Pairplot of the geoarea degree with a break in Children in all numeric variables.

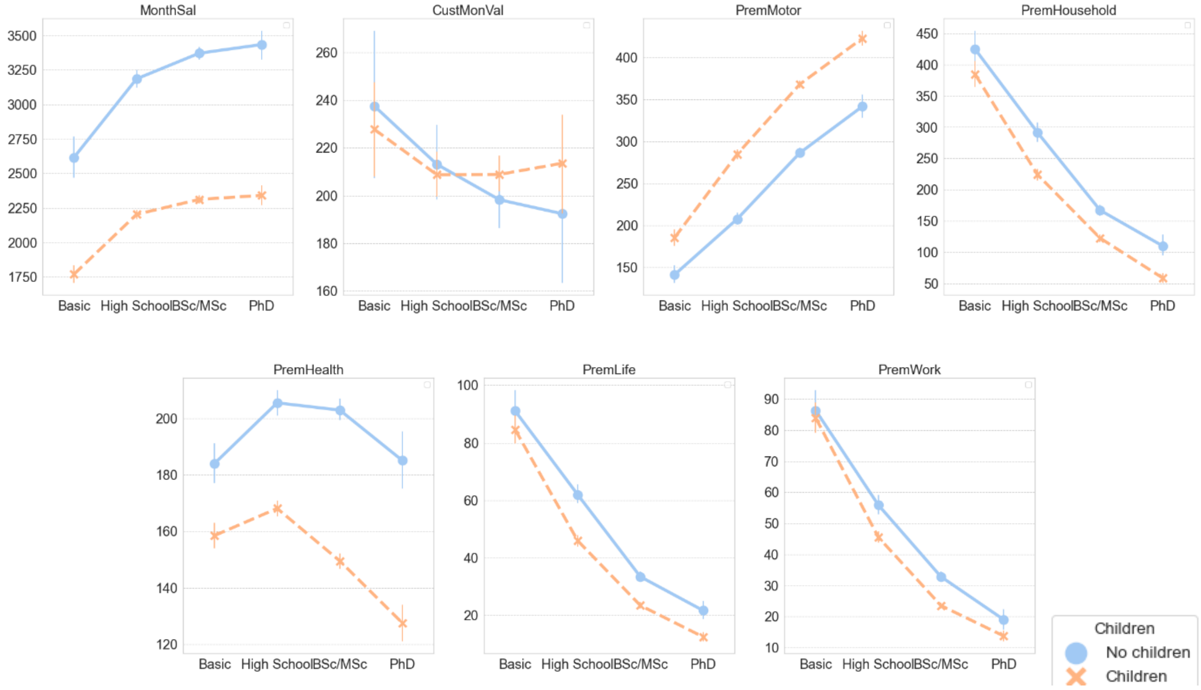


Figure 7.6 – Pairplot of the education degree with a break in Children in all numeric variables.

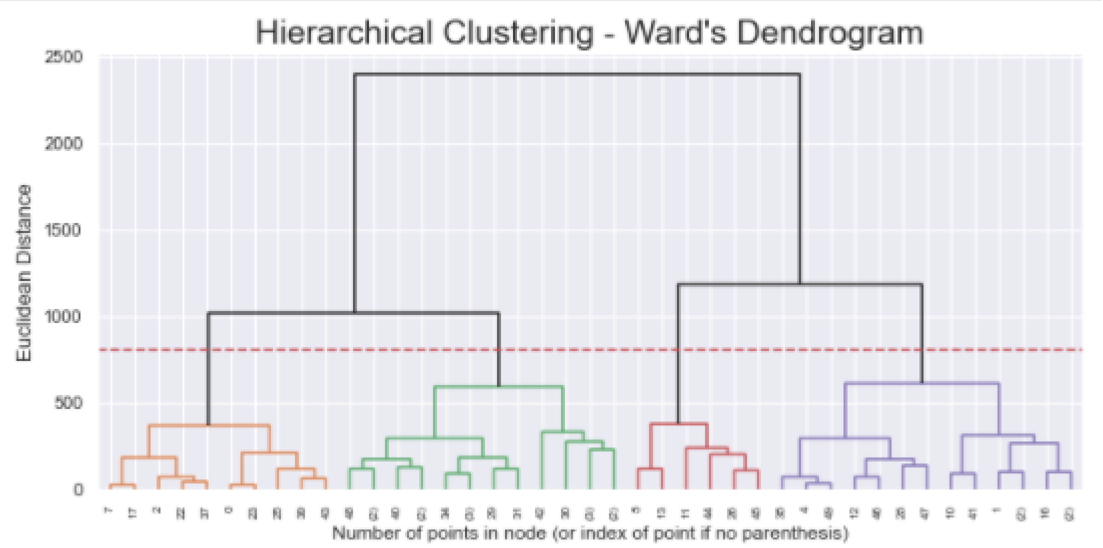
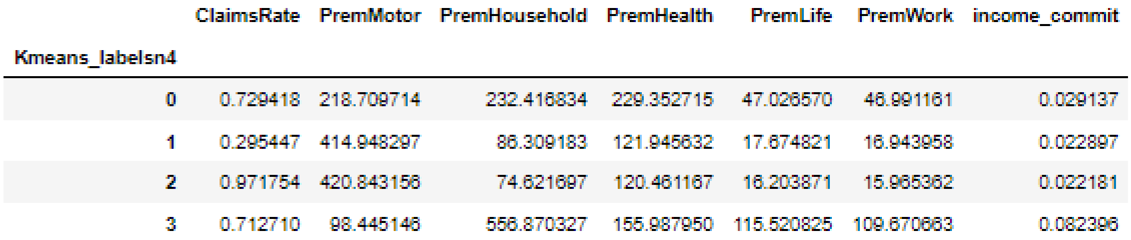
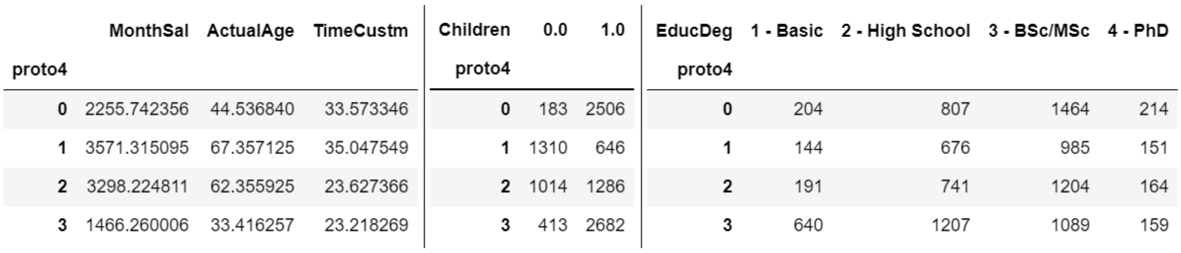


Figure 7.7: Hierarchical clustering – Ward's Dendrogram



	ClaimsRate	PremMotor	PremHousehold	PremHealth	PremLife	PremWork	income_commit
Kmeans_labelsn4							
0	0.729418	218.709714	232.416834	229.352715	47.026570	46.991161	0.029137
1	0.295447	414.948297	86.309183	121.945632	17.674821	16.943958	0.022897
2	0.971754	420.843156	74.621697	120.461167	16.203871	15.965362	0.022181
3	0.712710	98.445146	556.870327	155.987950	115.520825	109.670663	0.082396

Figure 7.8: Kmeans centroids for product_value variables



	MonthSal	ActualAge	TimeCustm
proto4			
0	2255.742356	44.536840	33.573346
1	3571.315095	67.357125	35.047549
2	3298.224811	62.355925	23.627366
3	1466.260006	33.416257	23.218269

Children	0.0	1.0
proto4		
0	183	2506
1	1310	646
2	1014	1286
3	413	2682

EducDeg	1 - Basic	2 - High School	3 - BSc/MSc	4 - PhD
proto4				
0	204	807	1464	214
1	144	676	985	151
2	191	741	1204	164
3	640	1207	1089	159

Figure 7.9: table containing the results for the k-protype clusters.

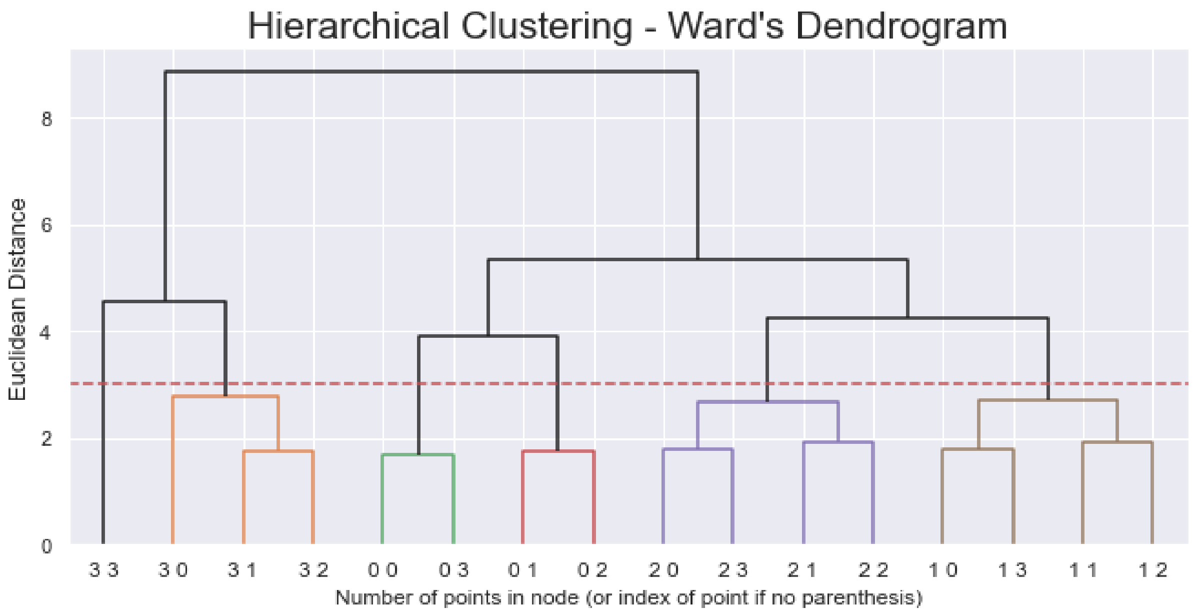


Figure 7.10– Ward’s Dendrogram for the combined cluster’s methods

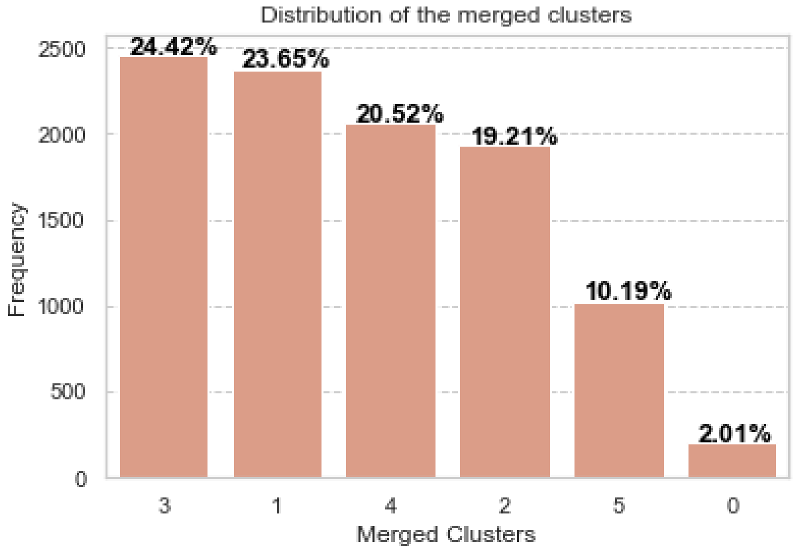


Figure 7.11 – Histogram of the cluster’s frequency with their percentage.

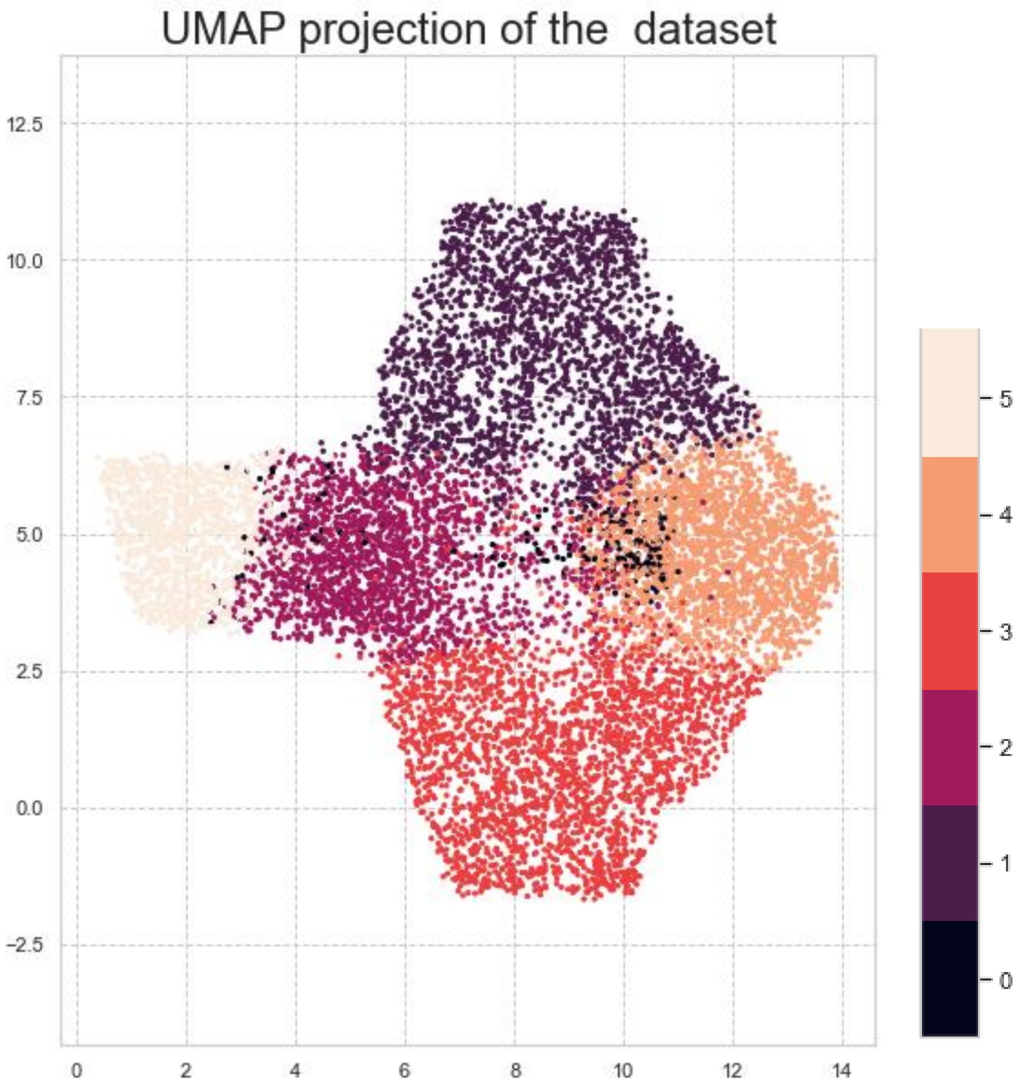


Figure 7.12– UMAP for the merged clusters

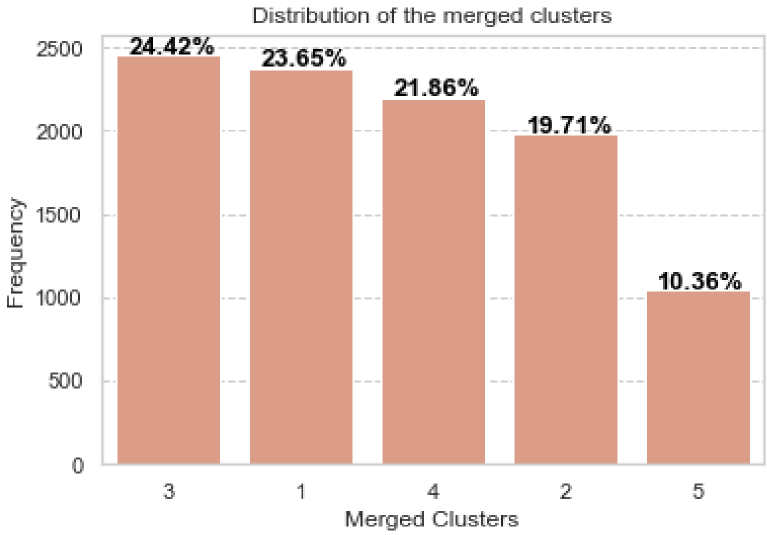


Figure 7.13 – Histogram of the cluster’s frequency with their percentage after replacement of cluster 0.

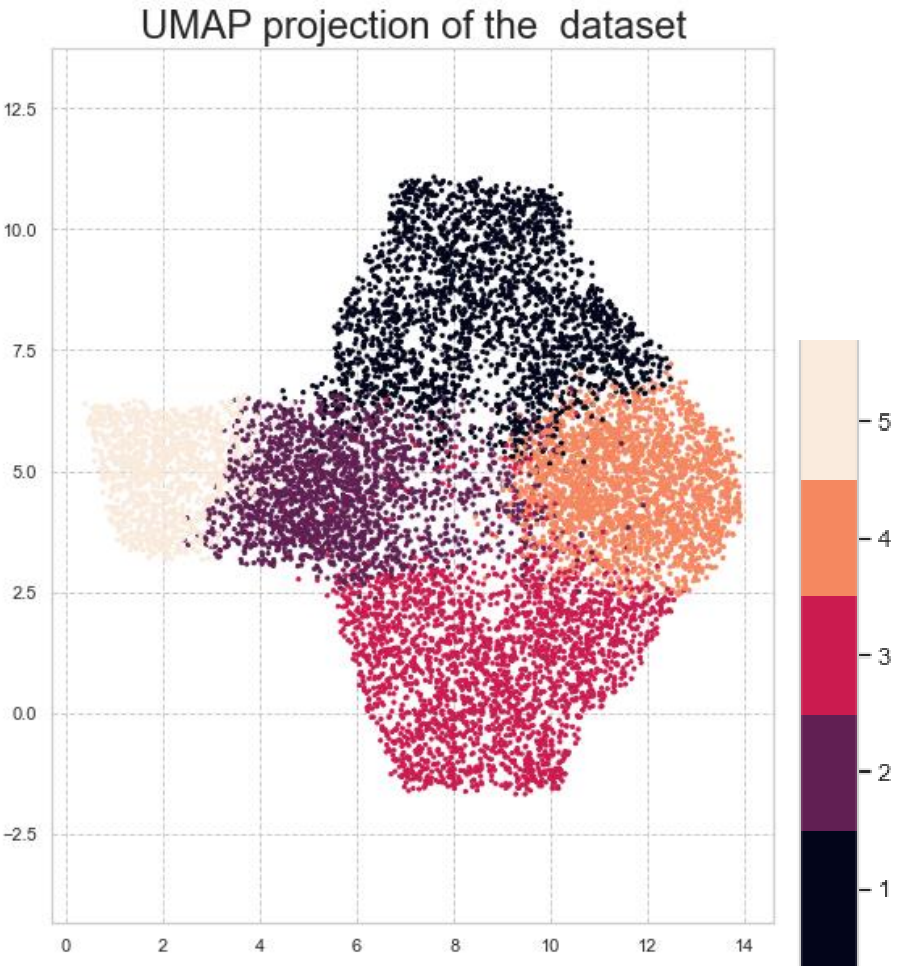


Figure 7.14 – UMAP for the merged clusters after replacement of cluster 0.

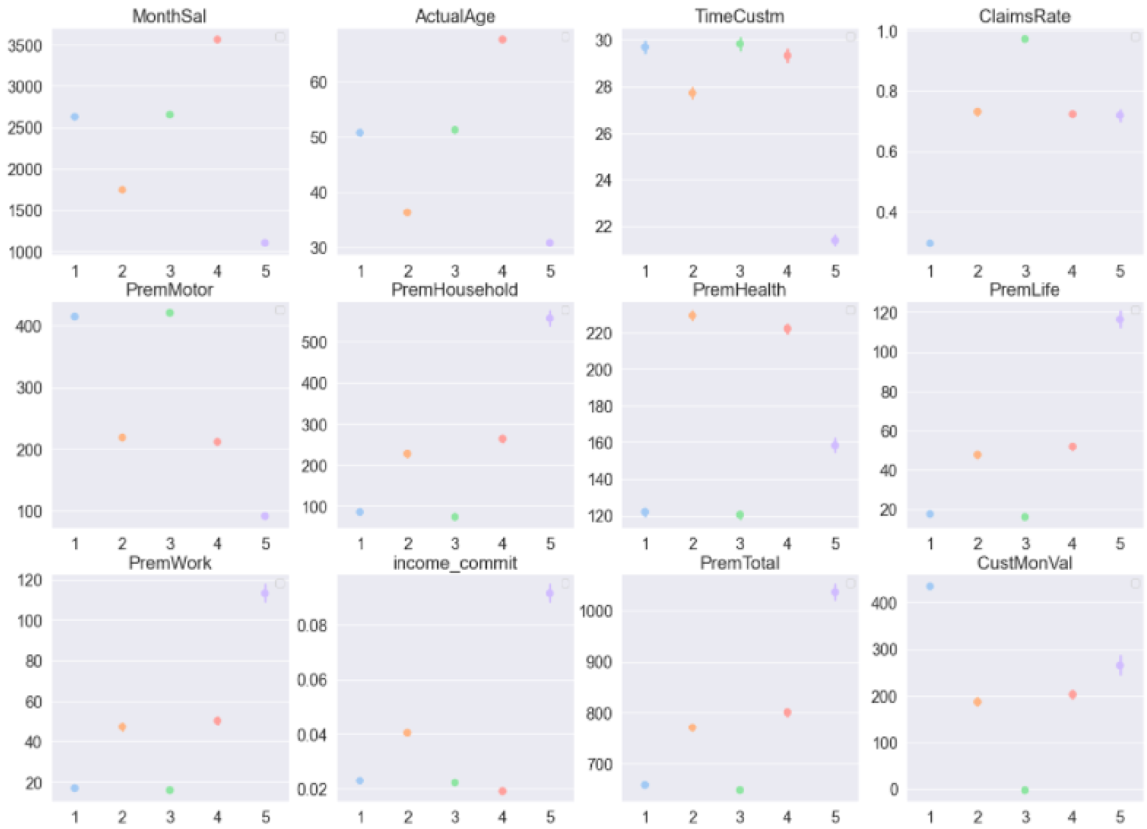


Figure 7.15 – Numerical variables' average per cluster

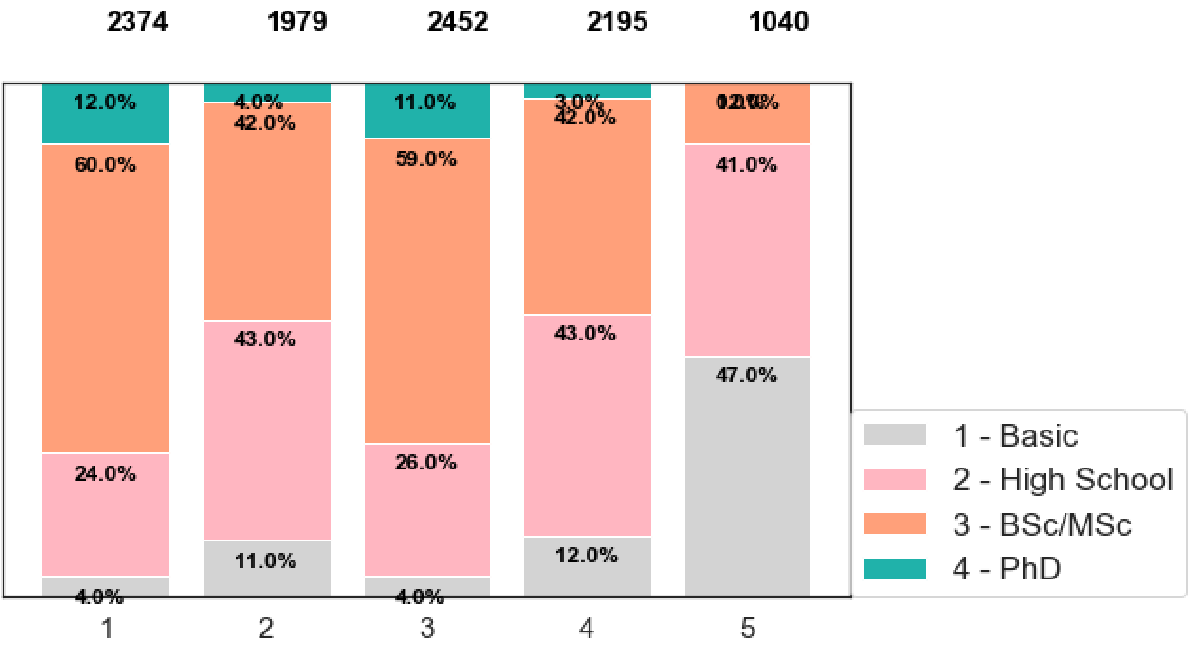


Figure 7.16 – Distribution of the education in each cluster.

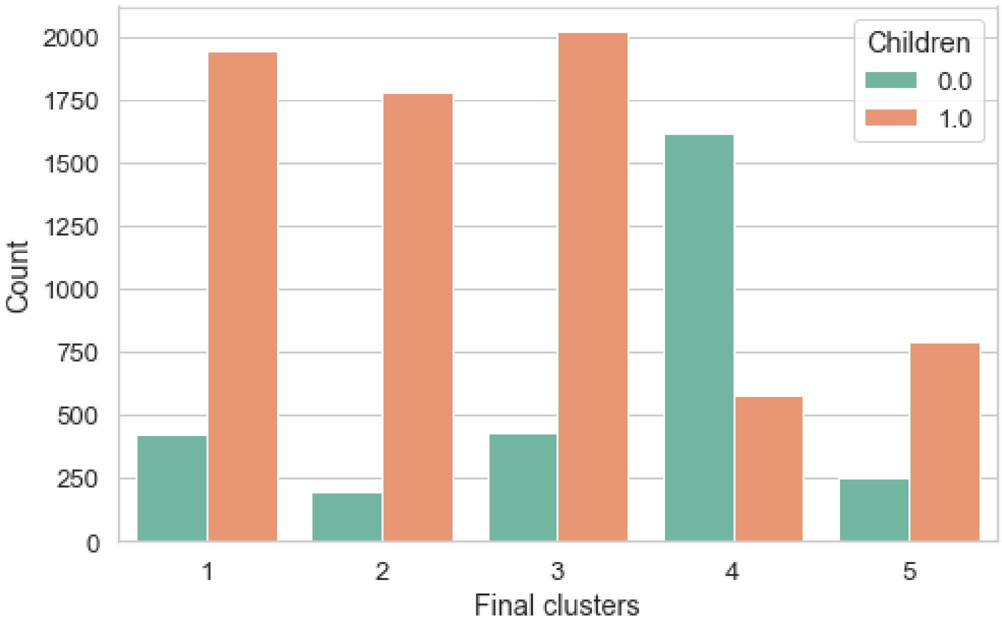


Figure 7.17 – Distribution of the children in each cluster.

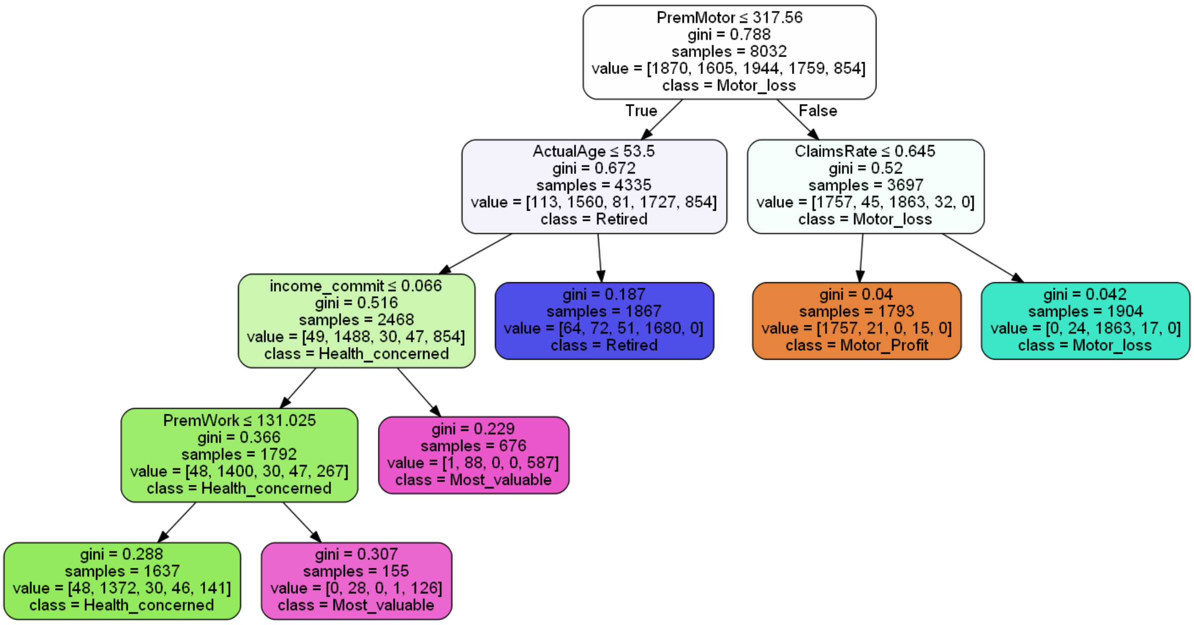


Figure 7.18 – DecisionTree for classifying new observations

Cluster Number	Cluster Label	Product_Value Description			Sociodemographic Description			
		High Values	Low Values	Other	Age Average	Education	Children	Salary
1	Motor Profit	. PremMotor . CMV	. Claims Rate . Total Premiums		50	High	Yes	Average High
2	Health Concerned	PremHealth		Balanced in all other Premium Insurances	38	Mix	Yes	Average Low
3	Motor Loss	.PremMotor . Claims Rate	. Total Premiums . CMV		50	High	Yes	Average High
4	Retired	PremHealth	Incomme commit	Balanced in all other Premium Insurances	68	Mix	No	High
5	Most Valuable	. PremHousehold, PremLife and PremWork . Total premiums	PremMotor		30	Low	Yes	Low

Table 7.3 - Clusters Interpretation