Data Mining Project

MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS

Customer Segmentation for Sitima - Companhia de Seguros, S.A.

Group U:

Mafalda Garcia, number: 20210763

Simão Pereira, number: 20210250

Tiago Santos, number: 20210548

December, 2021

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

INDEX

1. Introduction

Sitima - Companhia de Seguros, S.A. is a fictional insurance company in Portugal that offers and provides its customers insurance and services capable of meeting their needs, it works with a vast network of mediators and partners.

A dataset with information regarding 10,290 customers was made available and through it we will develop a Customer Segmentation in order to facilitate the work of Sitima's Marketing Department.

During this project, we intend to explore and retrieve the data with the objective of a better understanding of all the different Customers' Profiles. We intend to define the best variables for segmentation and reach the best way of clustering the customers.

Lastly, we will elaborate a marketing approach for each cluster.

1.1. Exploring Data

We start the project by exploring the data, observing its composition, variables and other information's that will be useful for a better knowledge of the data and, in turn, a better Customer Segmentation.

The first step was loading the customer's dataset, noting that it has a total of 10296 rows and 14 columns that represent the different variables of the dataset. Then in order to have more deep vision of the dataset we decided to use the ". describe(include='all')" method that provides us with information about the count, frequency, mean, standard deviation, minimum, maximum and the respective quartiles.

Using ".info()" we checked that the data type mostly have float64 and only one object (the education degree of the customer) and still, using the same method we verify that there are missing values in the following variables ("FirstPolYear"; "BirthYear", "EducDeg"; "MonthSal"; "GeoLivArea"; "Children"; "PremMotor"; "PremHealth"; "PremLife" ; "PremWork") so we decided to replace these missing values with "nan" in order to null the blank spaces.

To finalize the exploring part of the project we checked if there were any repeated records, which do not exist, and we used the "pandas_profiling" to get a better overview and check if there was any important information we forgot to explore.

2. Data Preparation

In this phase we manipulate and clean the data in order to improve performance and avoid misleading results.

2.1. Filling Missing Values

We started the manipulation by dividing the variables by metric and non-metric features and then filled in the missing value of the non-metric feature (Education Degree) with its respective mode (b'3 - BSc/MSc') and for the rest missing values we chose to use the median instead of the mean because we assumed a normal distribution and because is less “sensitive” to the outliers, so it would be safer option given the fact the we had not yet observed the outliers.

2.2. Incoherence

During the analysis we found two incoherence’s that don't make sense.

The first one found was in the variable 'FirstPolYear' which indicates the year of the customer's first policy and since the dataset refers to the year 2016, it makes no sense to have customers in which the year of the first policy is > 2016.

The second was in the 'BirthYear' variable where, once again, given that the reference year is 2016, it makes no sense to have customers whose birth year is less than 1900.

Therefore, to deal with these incoherence’s, we decided to create metric filters where the 'FirstPolYear' variable has only <= 2016 years and the 'BirthYear' variable has only customers born >= 1900.

However, after that, we noticed that there were many customers with insurance that were not even born, so we decided to delete the variable "BirthYear" and to keep the variable "FirstPolYear".

2.3. Outliers

To find and remove outliers *(Fig.2.3)* we used three different techniques:

2.3.1. Inter-quartile Range(IQR)

Initially we used the normal range for IQR, the 25th and 75th percentile but we would eliminate a lot of data, keeping in mind that as rule of thumb we should not delete much more than 3% so in order to decrease the percentage of deleted data we defined the percentiles as 3rd and 97th.

So, we ended up eliminating 0.44% from the dataset.

2.3.2. Local outlier factor(LOF)

The Local Outlier Factor (LOF) algorithm is an unsupervised anomaly detection method created with the goal of computing the local density deviation of a given data point about its neighbours. The algorithm is designed to recognize as outliers the samples that have a significantly lower density than their neighbors.

We have considered the 50 nearest neighbors and left the rest of the parameters as default and we detect around 1.90% of outliers, which is acceptable.

We ended up eliminating 0.99%.

2.3.3. Z-score

In addition, we thought that we should also test another method of outlier detection and as a result of the Z-score, approximately 0.16% of records were considered as outliers.

Conclusion: After performing all these techniques, we decided to use LOF because it was the method that removed more outliers and yet is inside the rule of thumb of 3%.

3. Data Pre-processing

3.1. Feature Engineering

In order to improve our analysis, we decided to create a new variable called **"PremTotal"** which consists of the annual total premiums per customer.

Also, for an easier understanding of the variable **"FirstPolYear"** we replaced it with how many years they have been loyal to the company (**"LoyaltyYears"**).

3.2. Feature selection

In order to evaluate the importance of all features and decide whether or not we should remove some input variables we used the correlation matrix *(Fig. 3.3),*with the spearman method to see the correlation between variables.

After analyzing the matrix, we ended up having some variables with very high correlation meaning that they give us pretty much the same information and so we removed the feature "CustMonVal" (highly correlated with "ClaimsRate") plus the variable that we created before ("PremTotal") because we concluded that it does not give meaningful information as well.

3.3. Data standardization

To put all metric features in the same scale we had to standardize the data and for that we used the MinMaxScaler, the StandardScaler and RobustScaler.

We ended up using the RobustScaler method because it works better with the presence of outliers.

3.4. Encoding

We also transformed the categorical data, in this case just the education degree variable into numeric outputs and for that we used the one-hot encoding method to put the values of the variable being 0 and 1.

3.5. Principal Component Analysis (PCA)

We used PCA as another technique of dimensionality reduction and based on the principal components that we got , it's fair to say that we probably had too many variables, since that just 5 principal components could explain about 80% of the variance of the data meaning that 5 original variables could explain 80% of the whole variance and at that point, we still had 10 features in the input space.

So, according to the cumulative variance (*Fig. 3.5)* of the principal components and their correlation *(Fig 3.5.1)* with the original variables we concluded that the features "LoyaltyYears" and "GeoLivArea" did not have a big importance in the whole dataset, so we decided to remove them.

4. Clustering

4.1. K-Means

To initialize this method of clustering we defined how many clusters we wanted to divide the instances. To do this, we used a plot that showed the evolution of inertia with the increase of the number of clusters *(Fig 4.1).*

Based on the elbow method we would choose 3 or 4 clusters. To double-check our choice, we checked the silhouette score of each number of clusters and decided that 4 would be the best number of clusters.

Now that we had the number of clusters, all we needed to do was run the K-Means algorithm and concatenate the labels to each individual.

4.2. Hierarchical Clustering

In order to find the optimal linkage for the algorithm we decided to plot the r2 (*Fig.4.2*) and realized that, for the 4 clusters that we had, the Complete linkage *(Fig. 4.2.1)* was better. This method is based on the distance between the most distant elements from each cluster.

Each point is considered a cluster and the distance between each cluster and the others is calculated. The two clusters with smaller distance are grouped in a new cluster. After this, the distances between clusters are recalculated and the process repeats itself.

4.3. DBScan

The main premise of this model, and Mean-shift, is to find the density points.

From start we knew that most algorithms that are based on density don't usually produce good results on big datasets. Nonetheless we tried this method.

The disadvantage of this method is that we need to define two very important parameters: the maximum distance of 2 points to be considered as neighbours and the minimum samples in a neighbourhood to make the point a core point. The "MinPts" parameter ("min_samples" in the algorithm) is usually chosen as twice the number of variables. Therefore, we defined it as 16. We ended up with a final R squared of 0.0008.

4.4. Mean-Shift

Mean-shift is a sliding-window based algorithm that foccuses on finding dense areas of data points to cluster.

When it finds the finds the optimal solution, based on closeness, it removes duplicate seeds - if two seeds are close to each other, they're assumed as just one and the remaining seeds are considered cluster centroids. In this point, the method calculates the distance between each point and the centroids assigns them to the closest cluster. The most important parameter to define is the "bandwith", which controls the size of the Sliding Window. We used the Scikit-Learn method "estimate_bandwidth". We ended up with a final R squared of 0.4454.

4.5. Self-Organizing Maps (SOM)

We tried another method for the clustering called Self-Organizing Maps (SOM) *(Fig.4.5).* The SOM objective is to adjust the units to the data in the input space, so that the network is (as best as possible) representative of the training dataset.

Also, this method is used for many ends and one of them is for clustering detection and is an unsupervised learning method that produce a high-dimensionality representation of the data.

For this approach we need to define the grip shape a priori the unfolding phase and the fine-tuning phase which we defined as 50x50 and 100 iterations respectively .

We merged this method with both K-means and Hierarchical Clustering and ended up with a final R squared of 0.379 which means that this approach is not very suitable for our clustering model.

4.6. K-Modes

K-modes is, in a way, a K-means for the categorical data. While k-modes uses distances to find centroids and cluster, KModes choses and updates the centroids based on Means, or the number of mismatches of each centroid to similar observation.

In KModes, the most important parameter to define is the number of clusters. For this, we plotted cost vs no. of clusters and concluded 3 clusters was the optimal result.

After this we just had to run the KModes code for 3 clusters.

4.7. Clustering Method – Decision

After the data was clean and easy to compare, we used different clustering methods to find patterns so we could group the clients and get the best results for our goal. In order to do this, we use the methods represented in the attached figure *(Fig.4.7).*

Concerning the decision of the best clustering method, by using the R2 scores *(Fig.4.7.1 and Fig.4.7.2)* we plotted the percentage of the response variable variation that is explained by each method and confirmed the best number of clusters. We concluded that K-means was the best method for our data and that the number of clusters we should get was 4.

The previous analysis (silhouette) showed us that the best number of clusters was between 3 and 4, but this new analysis leaves no questions that the percentage increase as we increase the clusters from 3 to 4 is still very significant. After 4, the increase of percentage is very slim.

In this analysis, besides SOM, we didn't include Mean-shift and DBScan because we confirmed that, as previously noted, algorithms that are based on density don't usually produce good results on big datasets.

5. Cluster Analysis and Marketing Strategies

Now that we have our final clusters, we need to determine what makes them different so we can create a different marketing approach to each one of them. We plotted the average values for each numerical variable and checked the main differences *(Fig. 5.1 and Fig. 5.1.1)*

5.1. First Cluster (C0) – Older people with high income

Regarding this cluster, knowing the only two variables that distinguish it from the average it their Higher Health Premium and the fact that they are the cluster with less probability of having children, we infer that it represents older people, probably retired.

Given that this cluster has older and richer population, we suggest promoting a partnership with "Viagens Abreu" and "Atlântico", with discounts for the clients.

5.2. Second Cluster (C1) – Younger people starting the adult life

We are inferring that this cluster represents young adults with kids. We consider this because of their comparatively higher Motor Premium, and lower rest of premiums. We made this assumption because most times young adult invest more in car insurance that house and health, even though these are very likely to have children.

Regarding this cluster that has a younger population, we recommend a LinkedIn-ads and Instagram-ads strategy. Considering the conclusion we got from their characteristics, the company should offer discounts on the main house appliances stores with a ceiling of 500€ and the offer of a microwave.

5.3. Third Cluster (C2) – Middle-aged people with low income

Considering their salaries are way below the average, they are not certain to have kids and they have high Household, health, life, and work premiums, we suppose they are middle-aged people with a not so stable-financial situation.

For this cluster, we suggest a mail-marketing and Facebook-ads strategy given that these are their main formal-communication form and most middle-aged people spend a lot of time on Facebook. Applying these types of communication, we want to offer a Lamborghini test-drive in *Autódromo do Estoril*.

5.4. Fourth Cluster (C3) – Middle-aged people with low income and financial-stability

What differentiate C3 from C2 is that, on average, C3 has a slightly lower salary, Motor, Household, Health and Life premiums. The only premium in where C3 invests more than C2 is Work probably because their work is of higher risk. So, given this and that, in general, C3 invests less in their insurances we can assume that they have a less stable financial life.

For this cluster, we suggest a mail-marketing and Facebook-ads strategy given that these are their main formal-communication form and most middle-aged people spend a lot of time on Facebook. Using these types of communication, we are going to promote the offer of the first and second month premiums for new clients and the next month for old clients who apply for it.

6. Conclusion

We concluded that k-means is the best approach to perform the customer segmentation, given this data, in an efficient and trustworthy way.

We would also like to leave some recommendations to Sitima for future reference. Firstly, we would advice the company to make a better data retrieving from their customers in order to avoid missing, miswritten or incoherent data. Secondly, for future cluster segmentation needs, it would helpful if Sitima gathered more information about the client's working conditions, for example if they have a stable job or not. Also, to make the process more efficient there's no need to keep record of some specific information's about the customers as the living area.

It is important to emphasize that despite the method of clustering they clearly have 4 types of clients to deal with, so the company can optimize the marketing strategies according to each.

7. References

(s.d.). Obtido de Geeks for geeks: https://www.geeksforgeeks.org/python-programming-language/

(s.d.). Obtido de Scikit-Learn: https://scikit-learn.org/stable/modules/classes.html#

(s.d.). Obtido de Stack Overflow: https://stackoverflow.com/questions

Brownlee, J. (2020). Data Preparation - How to Scale Data With Outliers for Machine Learning. *Machine Learning Mastery*.

Han, J., Micheline, K., & Jian, P. (2012). *Data Mining Concepts and Techniques.* Morgan Kaupmann.

8. Appendix

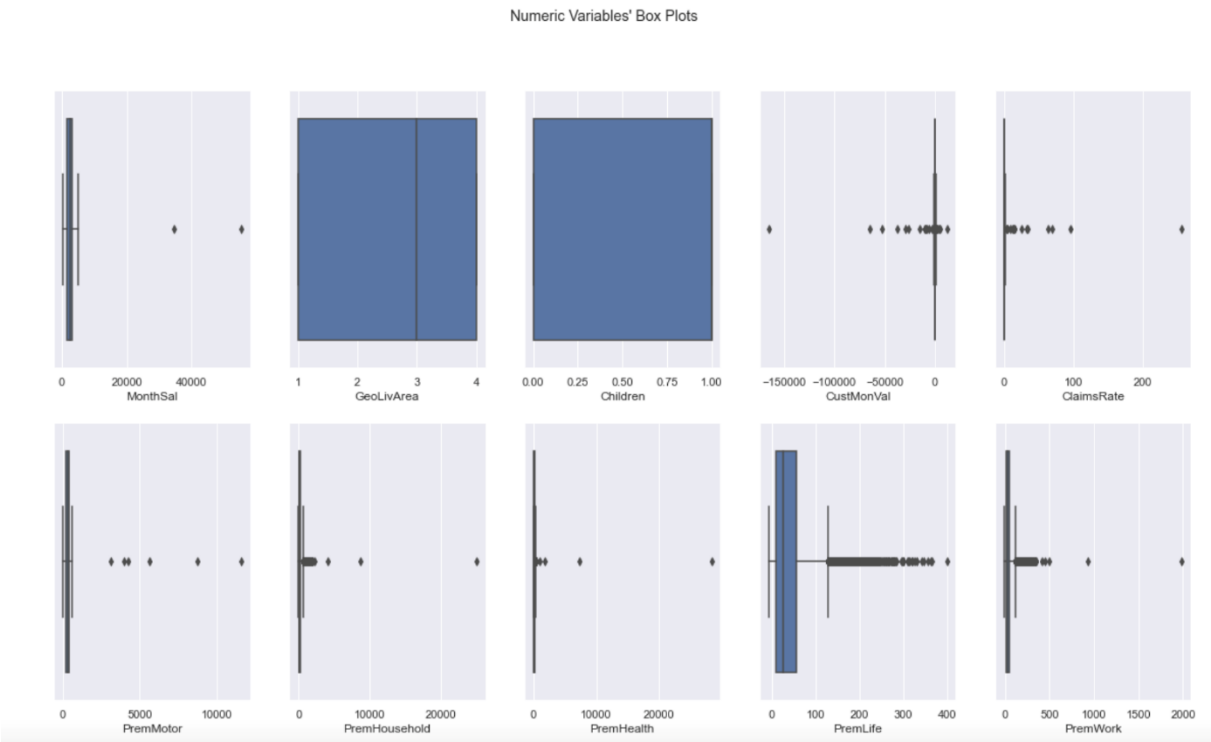

Figure 2.3 – Outliers before the removal

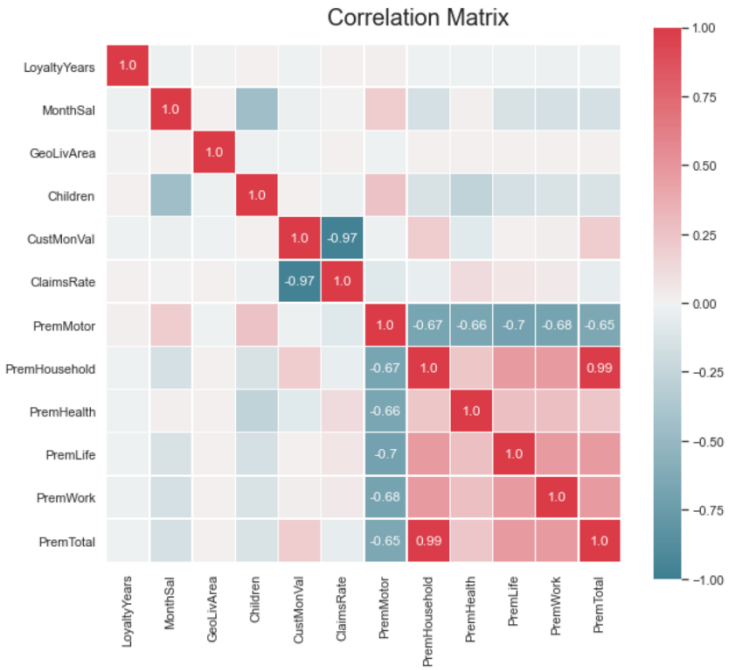

Figure 3.3 – Correlation Matrix

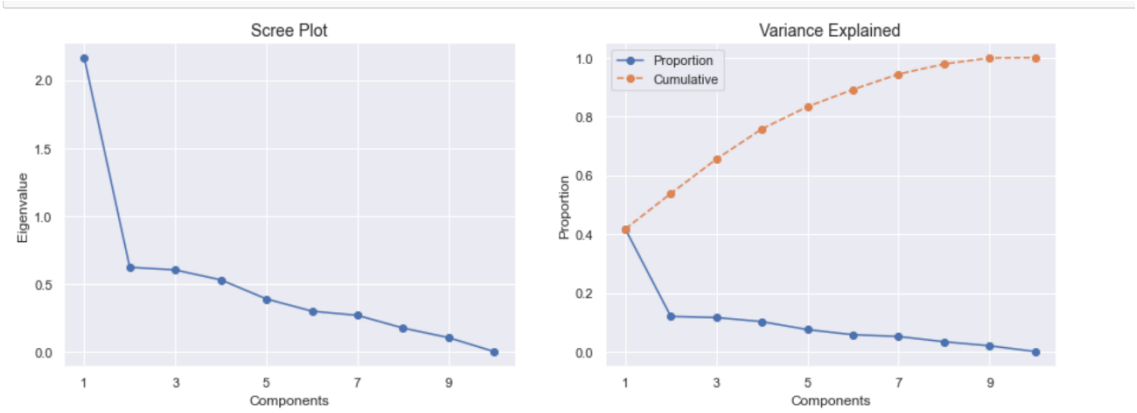

Figure 3.5 – Elbow method: cumulative variance of the PCA

	PC0	PC1	PC2	PC3	PC4
LoyaltyYears	-0.014284	0.000607	-0.026276	0.010230	0.094660
MonthSal	-0.346773	-0.119582	0.415679	0.033738	-0.785775
GeoLivArea	0.008870	-0.014174	0.009152	-0.014339	-0.030725
Children	-0.151319	0.167526	-0.483487	-0.039352	0.573027
ClaimsRate	0.079022	-0.064677	0.235557	-0.238605	0.139502
PremMotor	-0.909813	0.109604	-0.365525	-0.030896	-0.109069
PremHousehold	0.760257	-0.027946	-0.123107	0.627595	-0.054935
PremHealth	0.278510	-0.211421	0.859604	0.031751	0.337840
PremLife	0.768657	-0.498984	-0.223865	-0.319458	-0.080406
PremWork	0.779054	0.570064	0.049526	-0.236761	-0.094349

Figure 3.5.1 – Correlation PCA

Figure 4.1 - The inertia plot - K-means

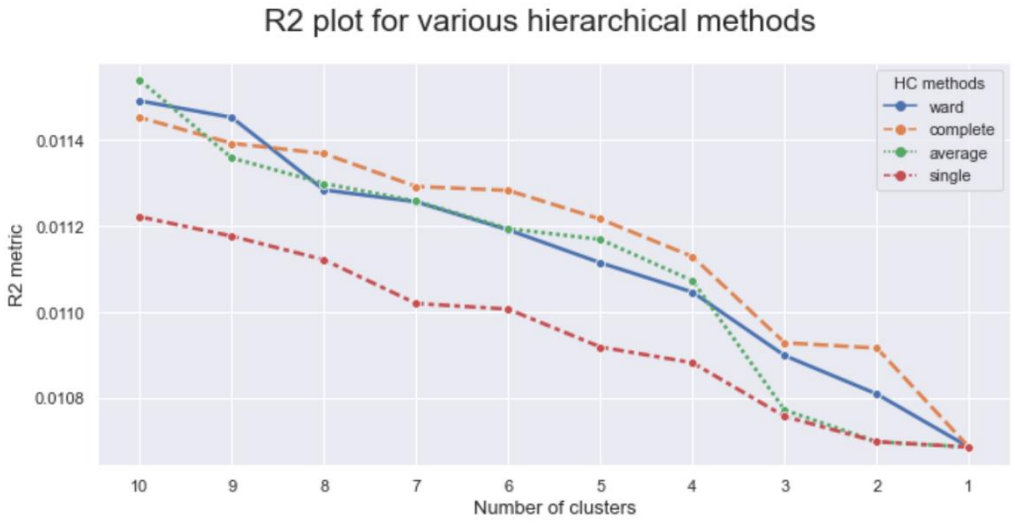

Figure 4.2 – R2 plot for Hierarchical Methods

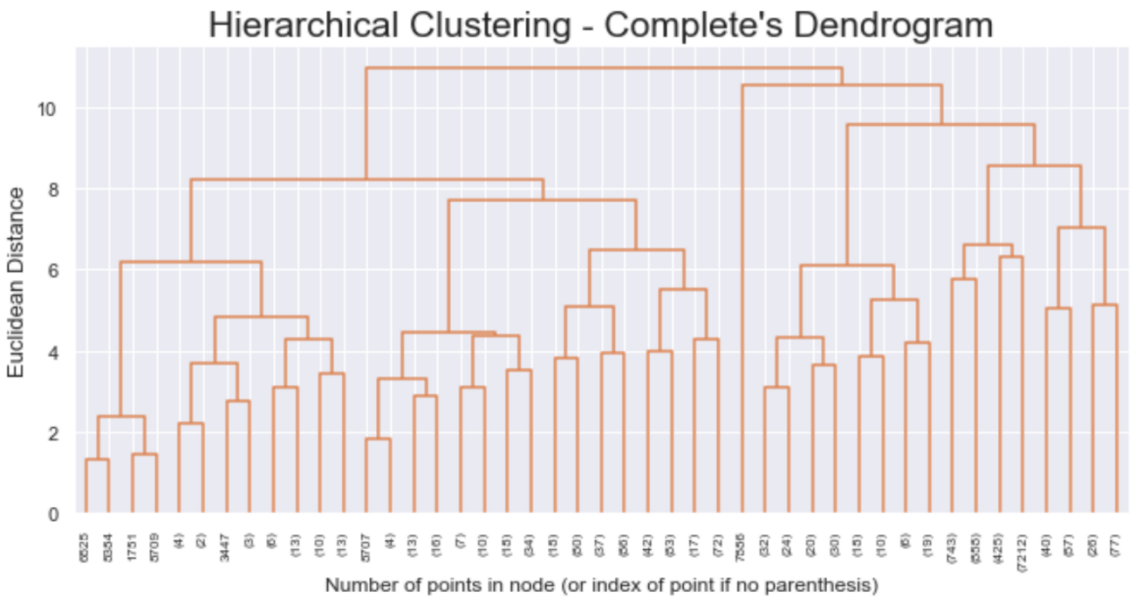

Figure 4.2.1 – Complete Linkage Dendrogram

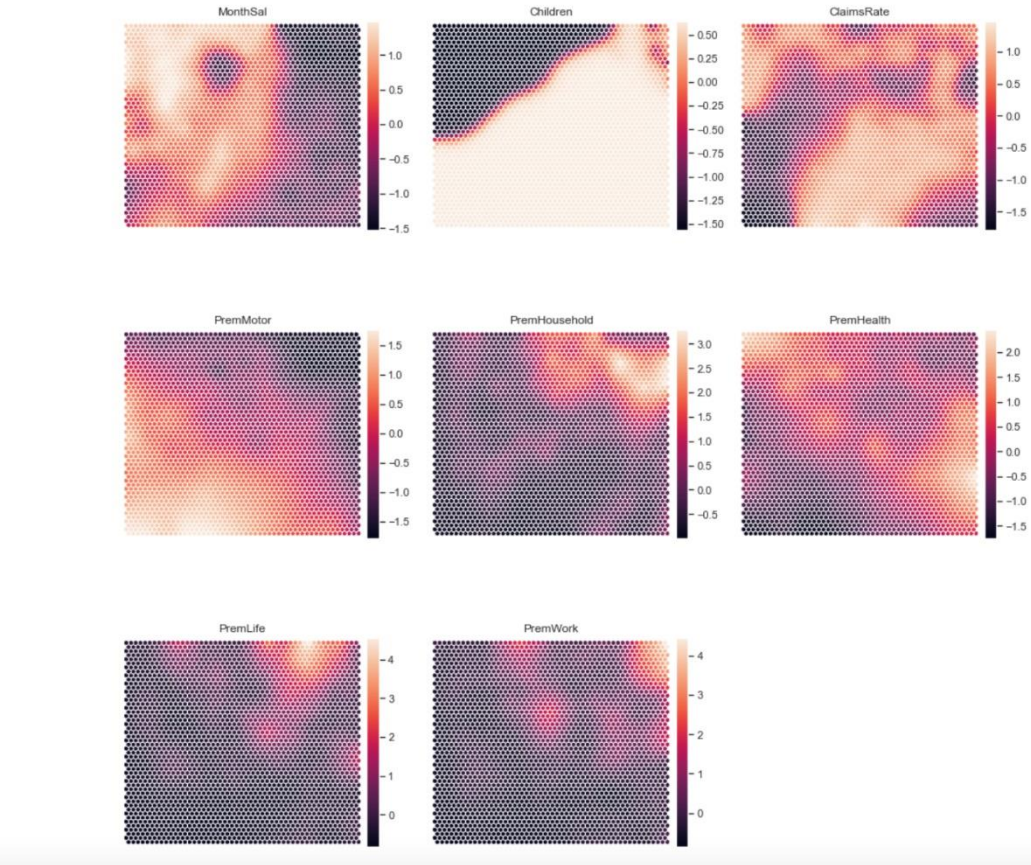

Figure 4.5 – Component Planes (SOM)

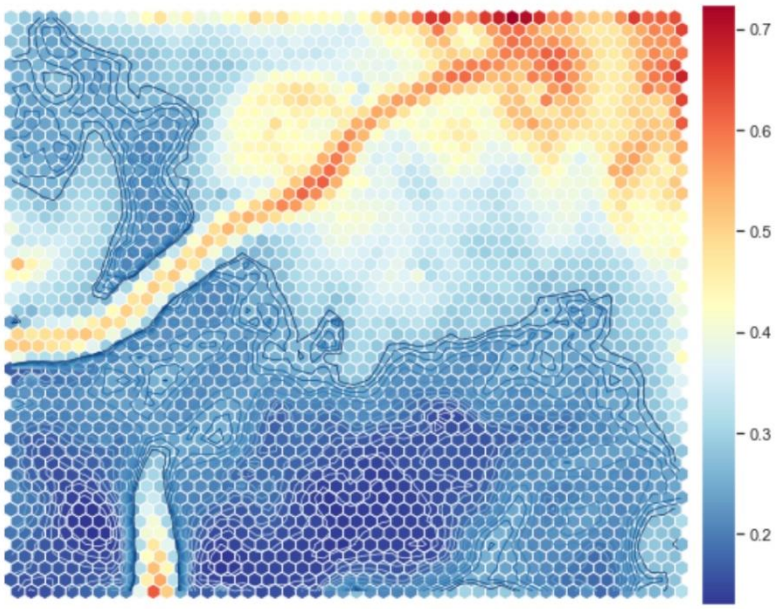

Figure 4.5.1 – U-matrix SOM

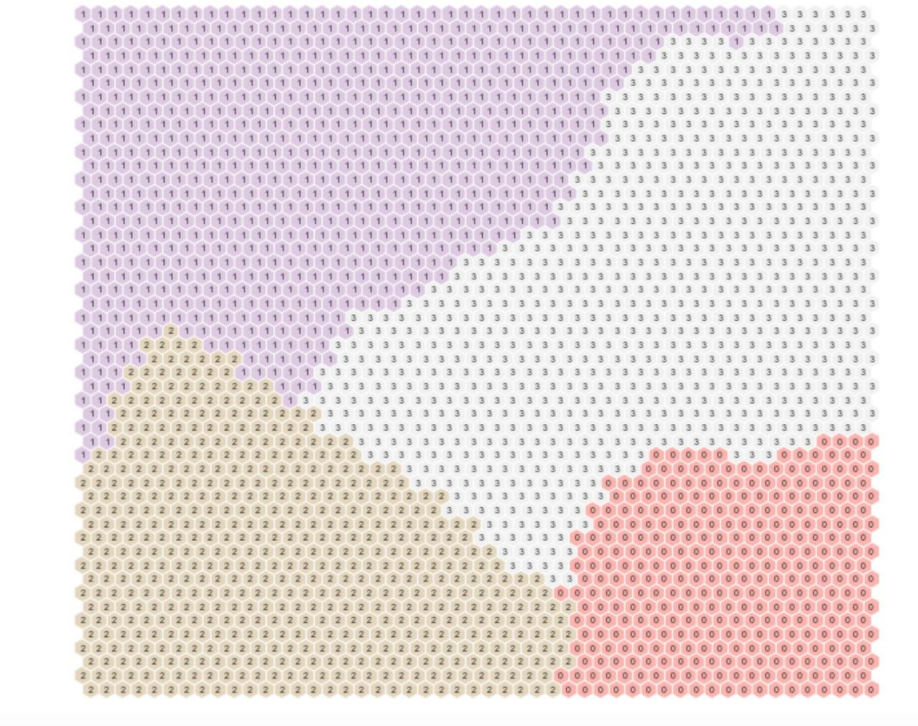

Figure 4.5.2 – Clustering SOM

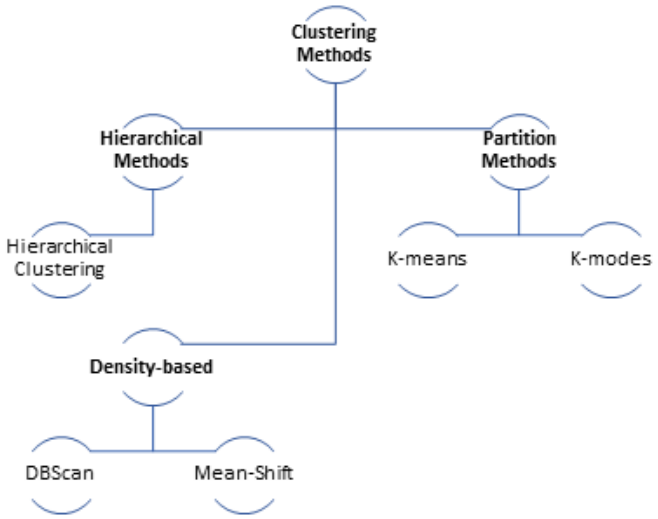

Figure 4.7 - Clustering Methods Illustration

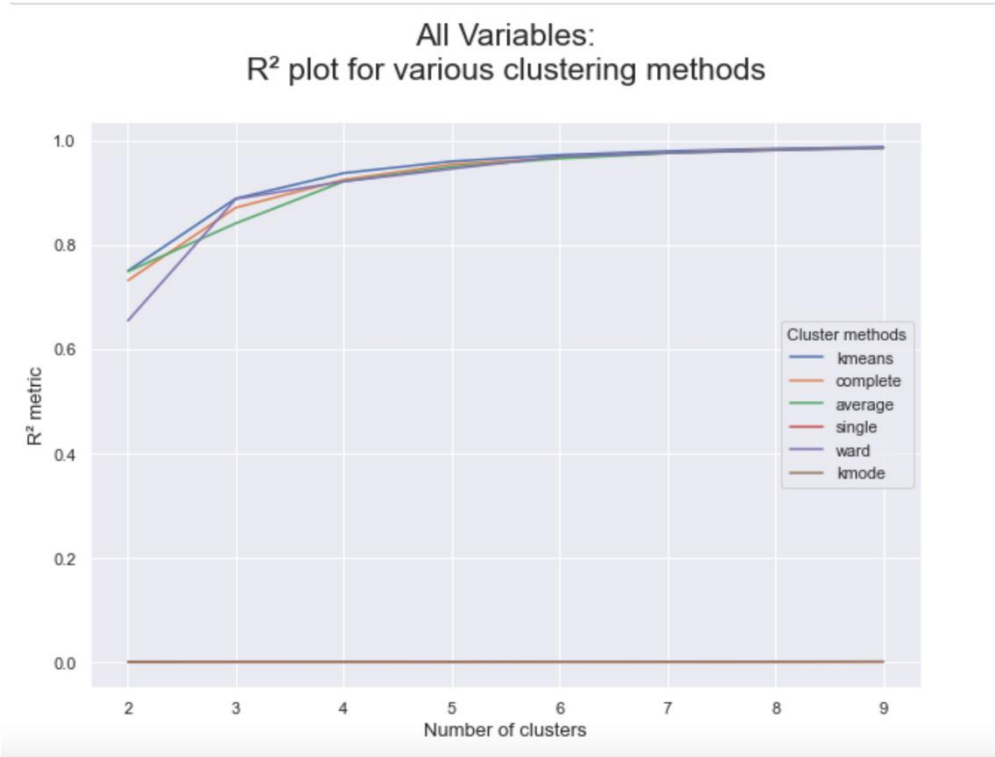

Figure 4.7.1 – R2 plot scores for each clustering method

	kmeans	complete	average	single	ward	kmode
2	0.749949	0.700312	0.730042	0.000165	0.726606	0.000016
3	0.888610	0.879038	0.877337	0.000196	0.877960	0.000100
4	0.937411	0.918879	0.931327	0.000233	0.928654	0.000192
5	0.959938	0.953024	0.949157	0.000277	0.952241	0.000121
6	0.972201	0.964649	0.965433	0.000363	0.966926	0.000223
7	0.979555	0.975854	0.975240	0.000399	0.977795	0.000201
8	0.984325	0.982494	0.980201	0.000554	0.982157	0.000465
9	0.987613	0.985996	0.984853	0.000682	0.985604	0.000561

Figure 4.7.2 – Optimal Cluster

Simple Data

Cluster	0	1	2	3	Average
MonthSal	2 744	2 581	1 788	1 681	**2 498**
Children	0,56	0,86	0,66	0,69	**0,71**
ClaimsRate	0,73	0,63	0,68	0,70	**0,68**
PremMotor	239	430	123	111	**298**
PremHousehold	214	68	548	465	**204**
PremHealth	225	117	165	160	**168**
PremLife	41	14	138	75	**41**
PremWork	42	14	54	152	**40**

Figure 5.1 - Average variable values for each cluster

Cluster	Salary	Children	Motor	Household	Health	Life	Work
0		Have	High	Low	Low	Low	Low
1	Poor		Low	High		High	High
2	Poor		Low	High		High	
3		Might not			High		

Figure 5.1.1 – Main cluster characteristics

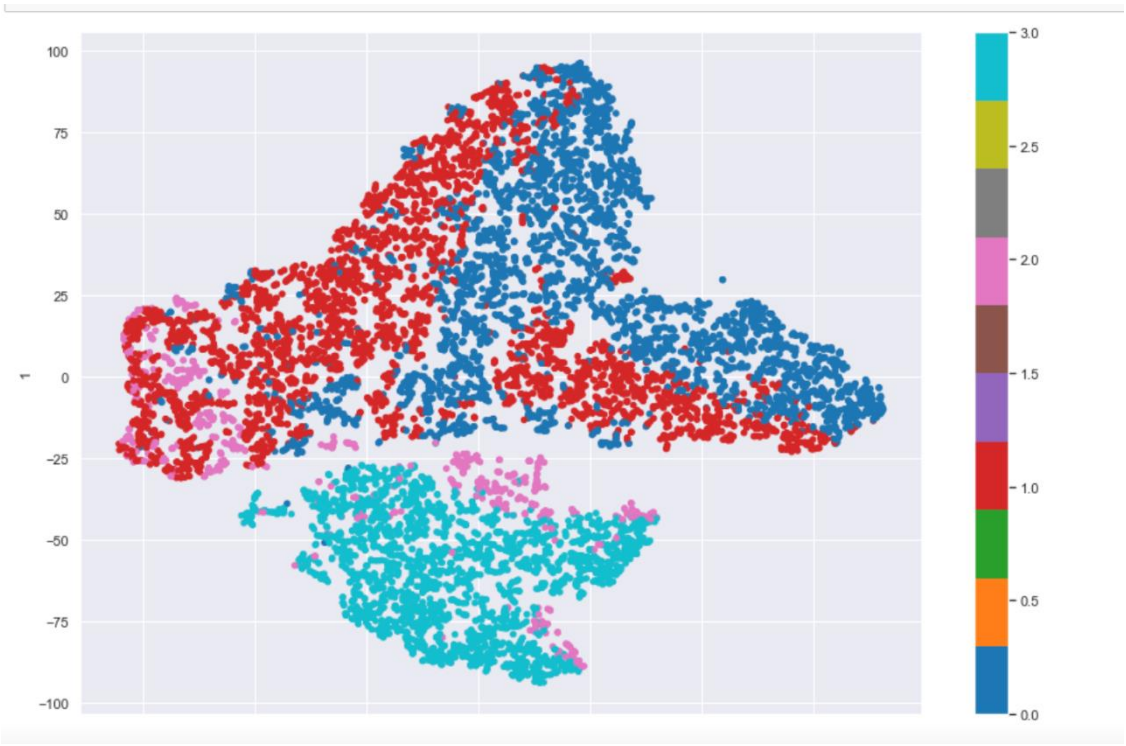

Figure 5.1.4 - Cluster visualization using t-SNE