NOVA IMS Information Management School

Data Mining Project

Report

Group S

-

20190688 Alexandra Ordina

20190932 Ana Cláudia Alferes

20190417 Pedro Carvalho

Teachers

-

Fernando Lucas Bação

Jorge Antunes

Index

1. Introduction

The following report aims to do a Customer Segmentation in such a way that it will be possible for the Marketing Department to better understand all the different Customer's Profiles.

The first step taken was analysing the insurance business itself to have a better understanding of the concepts we are working with, followed by the Data Understanding where we explore the data we were given and check for some kind of anomalies. The next step we face was Data Preparation consisting on data cleaning and formatting in order to have a precise data to work with. Then we proceed to the Modeling phase where the data was split based on data type and analysed using several unsupervised cluster techniques - K-means, Hierarchical (dendogram), DBSCAN, SOM and K-Modes. After evaluating the performance of these techniques we selected K-Means for our final segmentation and described the result, giving possible suggestions regarding marketing strategy for the client segments.

2. Contextualization

The provided dataset is an Analytical Based Table counting 10.290 customers. Descriptive metadata was provided explaining the meaning of variables collected by the insurance company. Administrative metadata on how the data was collected, who has responsibility to collect and maintain it, was not made available.

We know that the dataset represents a single year 2016.

We were provided with database file and an Excel file which upon examination appeared to be identical in contents so all further steps described in the report are based on the single source of Excel file.

Also no business expectations were provided as to the desired number of client groups or levels of importance of variables in dataset so we will be applying several unsupervised data mining techniques to define the appropriate number of groups we think would be a good fit based on provided data and will treat all variables as having the same weight.

3. Business Understanding

In the beginning of the project this important to understand the business concept of an insurance company to get an idea of which variables are the most important in this type of business and where we will focus to have a detailed analysis of our client dataset. Insurance companies base their business model around assuming and diversifying risk from customer to customer. Suppose the insurance company is offering a policy with a $100,000 conditional payout. It needs to assess how likely a prospective buyer is to trigger the conditional payment and extend that risk based on the length of the policy.

This is where insurance underwriting is critical. Without good underwriting, the insurance company would charge some customers too much and others too little for assuming risk. This could price out the least risky customers, eventually causing rates to increase even further. If a company prices its risk effectively, it should bring in more revenue in premiums than it spends on conditional payouts.

To summarize, there are some particular issues that we focus on throughout the report:

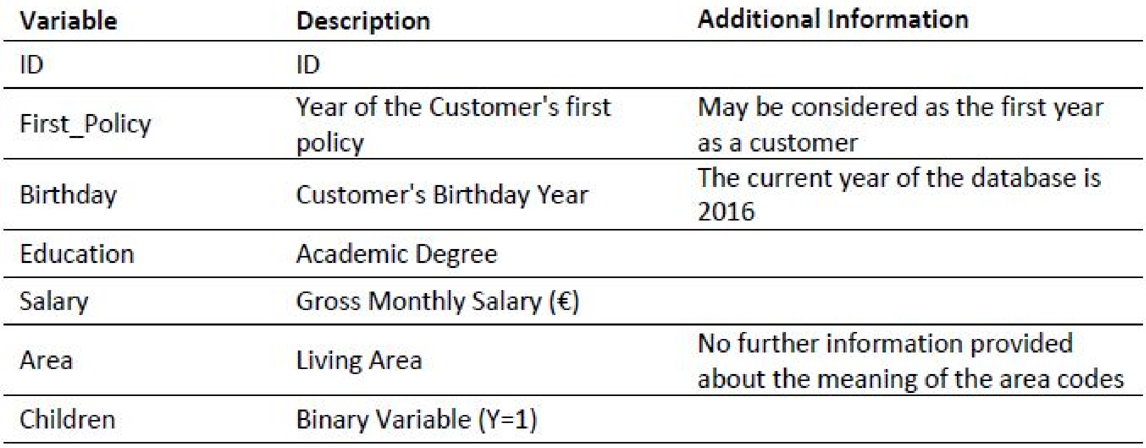
a. What type of customers are interested in what type of policy(*ies)?
b. What are the profile of our clients?
c. What are the profile of clients that bring more value to the company?

In conclusion, we hope to understand who the best clients are as well as the best type of policy in order to help marketing managers to make the best possible decisions.

4. Data Exploration

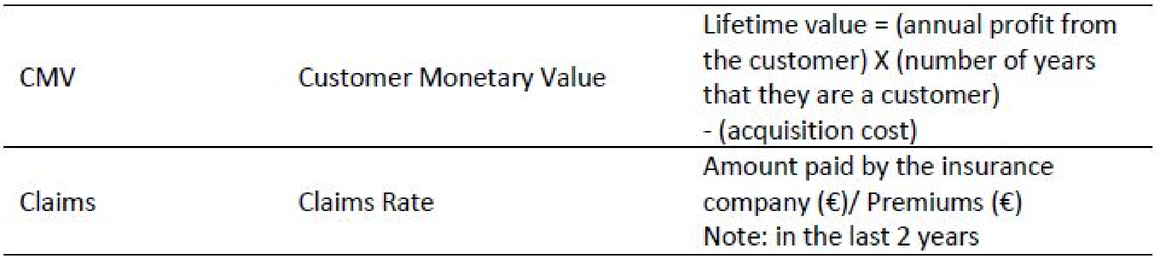
In this phase to familiarise with the client dataset and their features we divided the variables into Customer Demographics -concerning the customer information, Customer Value - concerning the value of the customer for the company, and Policies - concerning the information about the premiums of the policies:

Customer Demographics:



Variable	Description	Additional Information
ID	ID	
First_Policy	Year of the Customer's first policy	May be considered as the first year as a customer
Birthday	Customer's Birthday Year	The current year of the database is 2016
Education	Academic Degree	
Salary	Gross Monthly Salary (€)	
Area	Living Area	No further information provided about the meaning of the area codes
Children	Binary Variable (Y=1)	

Customer Value:



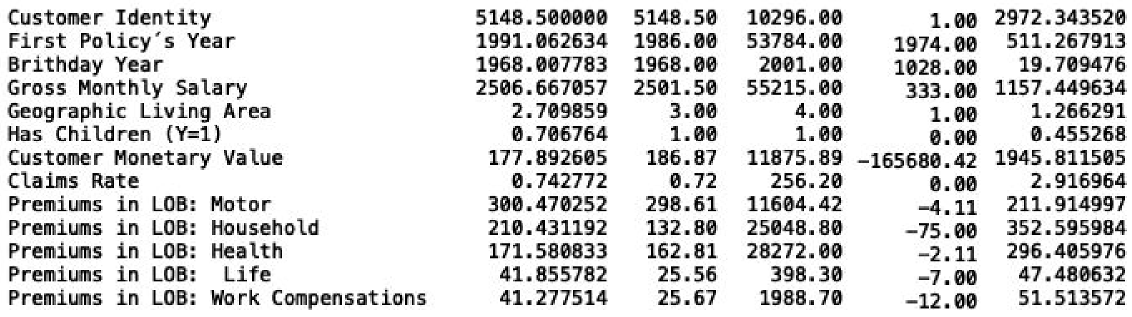
CMV	Customer Monetary Value	Lifetime value = (annual profit from the customer) X (number of years that they are a customer) - (acquisition cost)
Claims	Claims Rate	Amount paid by the insurance company (€)/ Premiums (€) Note: in the last 2 years

Policies:

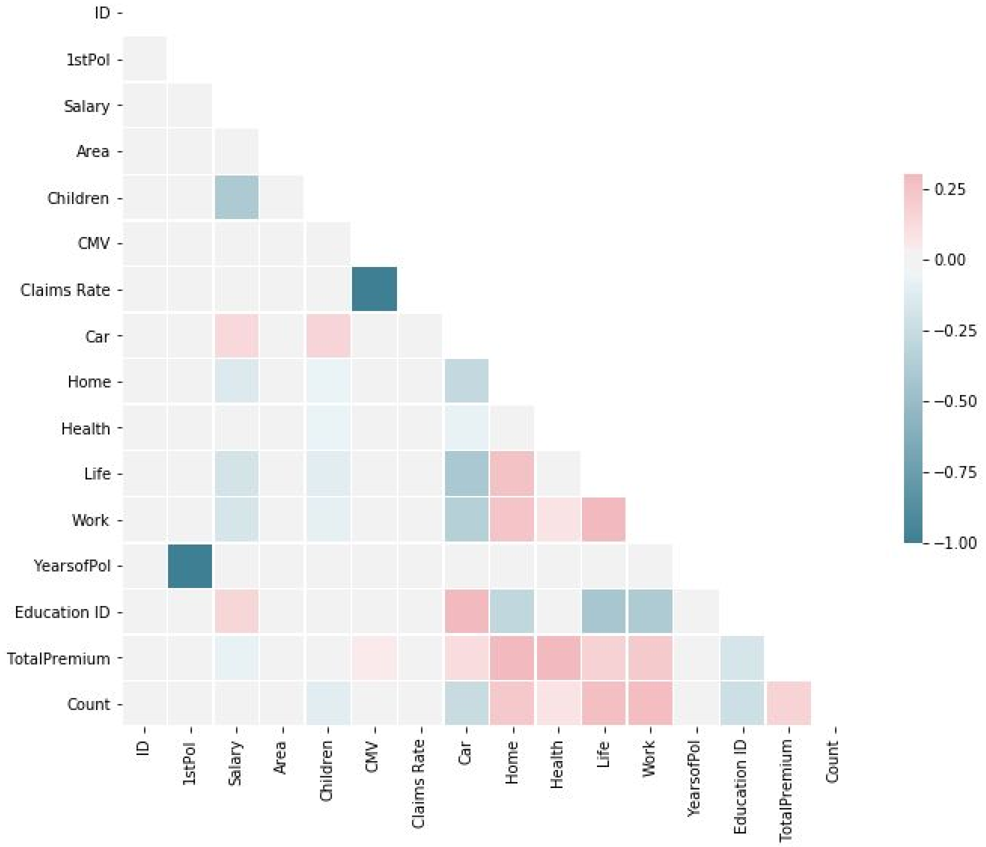
Motor	Premiums (€) in LOB: Motor	Annual Premiums (2016) Negative Premiums may manifest reversals occurred in the current year, paid in previous one(s)
Household	Premiums (€) in LOB: Household	
Health	Premiums (€) in LOB: Health	
Life	Premiums (€) in LOB: Life	
Work_Compensation	Premiums (€) in LOB: Work Compensations	

Then we check on some of the basic statistics of the DataFrame in order to have an overall look for the variables:

```
        Variables                                   Mean        Median    Max        Min          StD
```



```
Customer Identity                      5148.500000  5148.50  10296.00       1.00  2972.343520
First Policy´s Year                    1991.062634  1986.00  53784.00    1974.00   511.267913
Brithday Year                          1968.007783  1968.00   2001.00    1028.00    19.709476
Gross Monthly Salary                   2506.667057  2501.50  55215.00     333.00  1157.449634
Geographic Living Area                    2.709859     3.00      4.00       1.00     1.266291
Has Children (Y=1)                        0.706764     1.00      1.00       0.00     0.455268
Customer Monetary Value                 177.892605   186.87  11875.89 -165680.42  1945.811505
Claims Rate                               0.742772     0.72    256.20       0.00     2.916964
Premiums in LOB: Motor                  300.470252   298.61  11604.42      -4.11   211.914997
Premiums in LOB: Household              210.431192   132.80  25048.80     -75.00   352.595984
Premiums in LOB: Health                 171.580833   162.81  28272.00      -2.11   296.405976
Premiums in LOB:  Life                   41.855782    25.56    398.30      -7.00    47.480632
Premiums in LOB: Work Compensations      41.277514    25.67   1988.70     -12.00    51.513572
```



Customer Demographics Analysis

After all the previous analyzing steps we start creating a few variables more to add up information needed to our dataframe.

Age (2016 - Birthday Year) was one of the new variables added, in order to check the age of our clients. From the beginning we can check that something went wrong with the data corresponding to the Birthday Year variable (Figure 1).

To have a better look over this type of variable we remove the clients with age over 100 and we end up with this result.

The result showed within the graph still seen unusual, since we have clients under 18, amount other problems found (Figure 2).

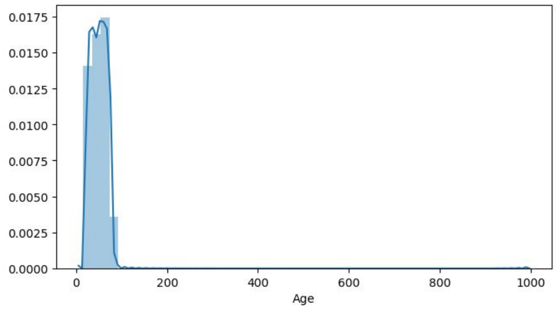


Figure1

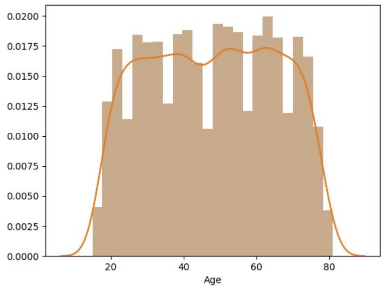


Figure 2

YearsP (2016 - First Policy's Year) regarding the number of years that is our customer.
As for age we can see that something went wrong (Figure 3), so we remove the clients that have this variable lower than 0, and this was our result. (Figure 4)
The new results show that most of our clients have been with us between 30 to 35 years (2218,22%).

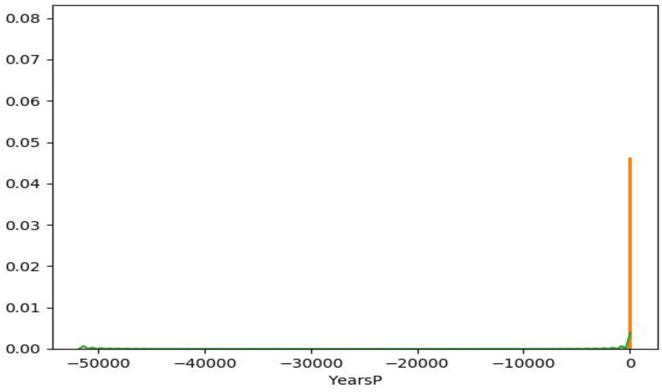


Figure 3

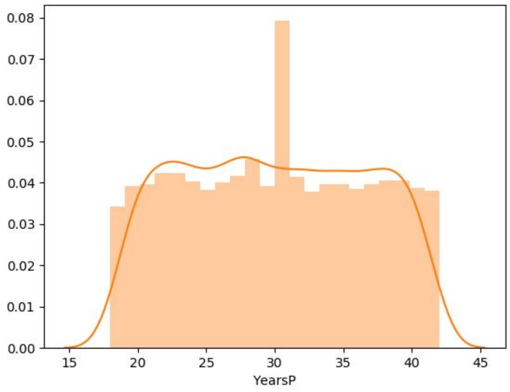
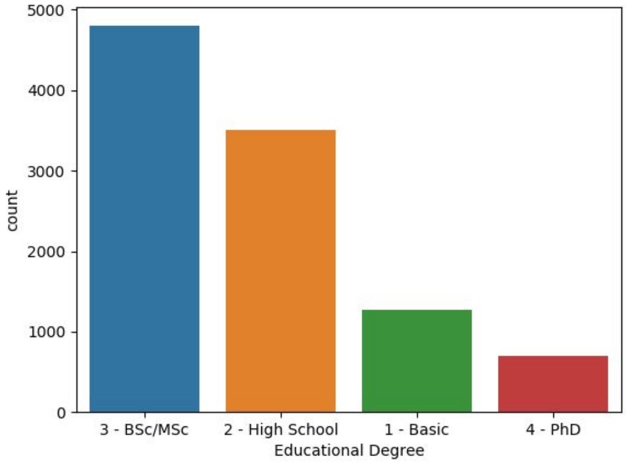


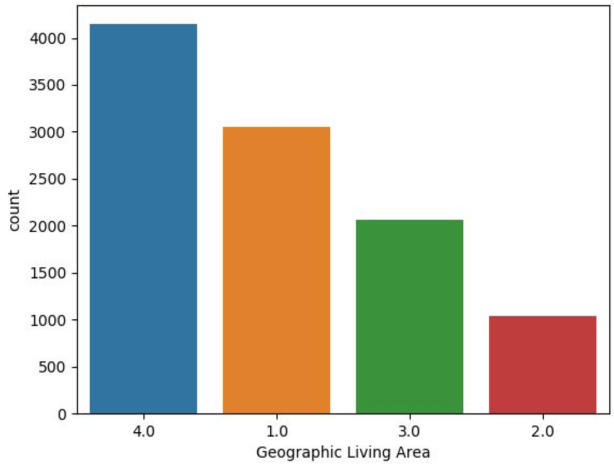
Figure 4

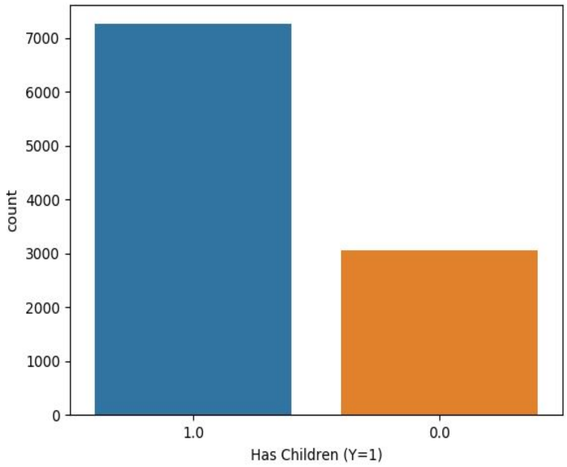
To have a closer look to this new created variables we created the following line of code - `mydata[(mydata['YearsP'] < mydata['Age'])&(mydata['Age']>= 18)& (mydata['First Policy´s Year']-mydata['Brithday Year'] >= 18)].count()` - this code counts the clients that we consider to have the **right** data and the result was 5196 (50%) clients. Now we face a problem that we need to solve, since we cannot only work with 50% of the data we assumed that the column regarding age was an incorrect input in database and as gives too many faulty records, we decided to drop age variable.

Regarding the Education level, we can see that 46% of our clients have a Bachelor/Master Degree, 34% corresponding to High School Degree, 13% to Basic and 7% to PhD.

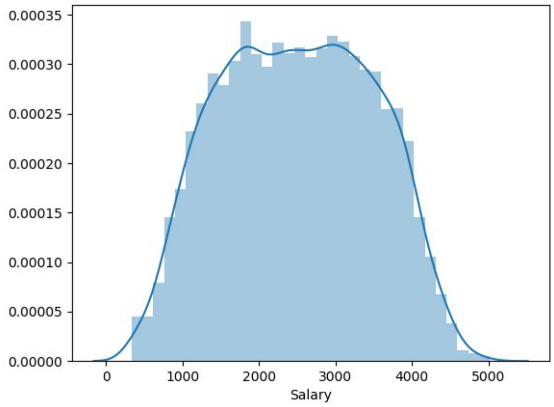


The next two graphs show distribution of the clients per Geographic Living Area, with 40% of clients living in area 4, 30% living in area 1, 20% living in area 3 and 10% living in area 2. Followed with the Has Children (Y=1) graph, with 71% of the clients having children and the remaining 29% not having children.



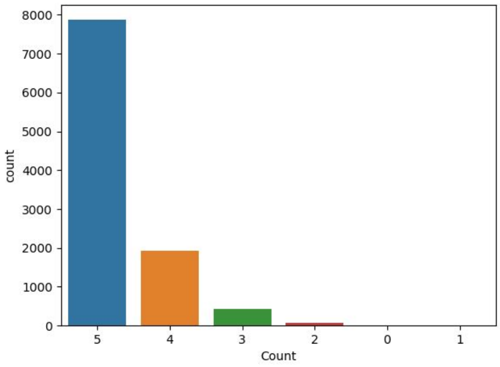


Then we check for the variable Salary, to have a better understanding of our major profile clients we exclude the one that have a salary higher than 10000. We can see that the Salary of our clients vary between 500 and 50000, with a rough accent between 1600 and 3300 (25%).

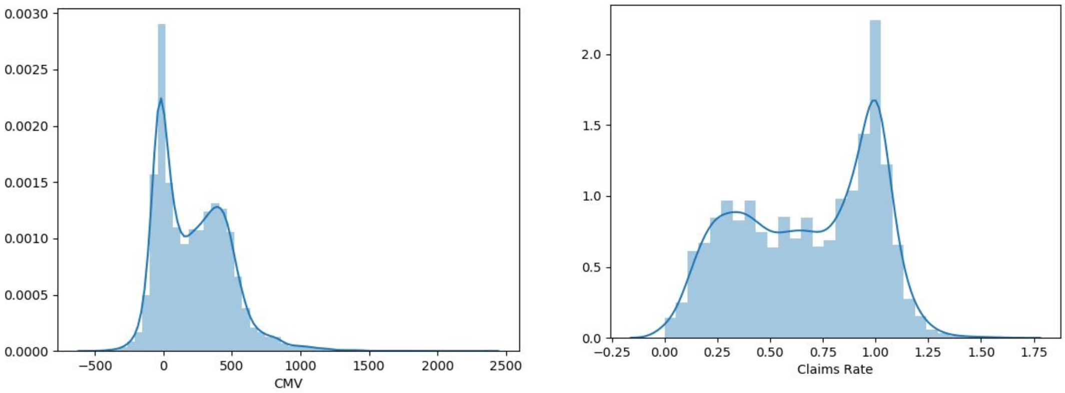


Customer Value Analysis

Count concerning the number of policies that the client has purchased. After checking the results we can see that 76% of the clients have all the 5 policies and 19% have 4 policies in the year 2016.

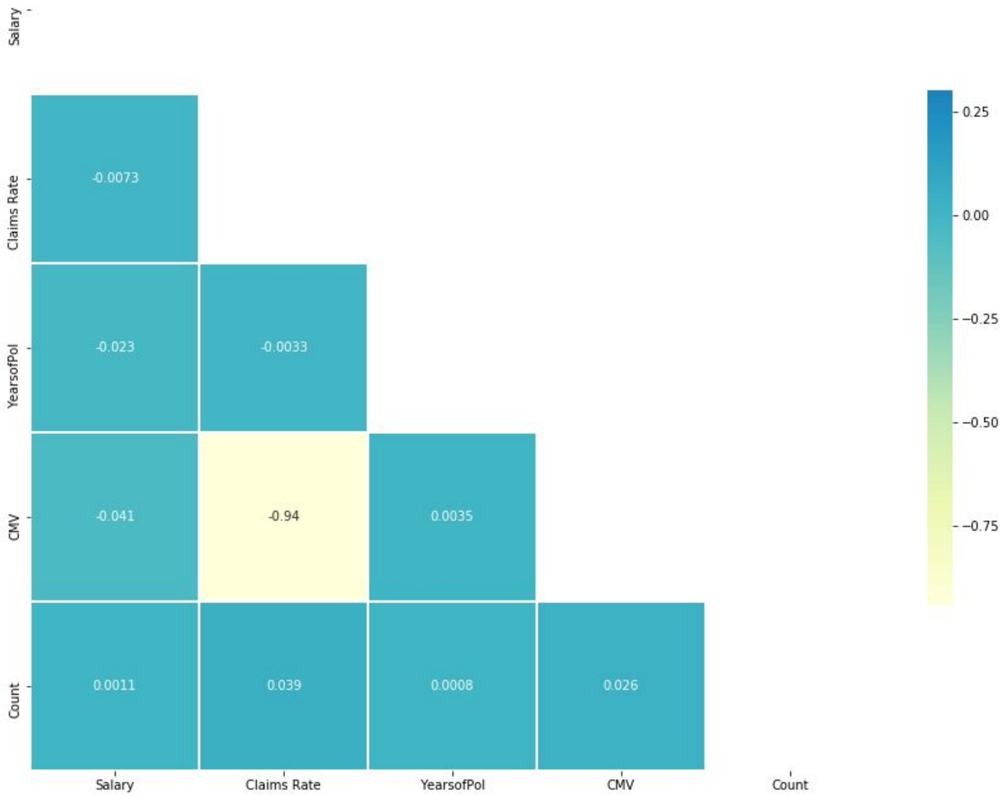


Regarding the CMV (Customer Monetary Value) we see that they vary mostly between -30 and 30 corresponding to 16% of our clients. For the Claims Rate the highest values are between 0.90 and 1.00 corresponding to 13% of our clients.



Finally, we also look at how customer numerical variables are correlated: CMV and Claims rate are highly correlated which is expected due to the nature of CMV indicator calculation.

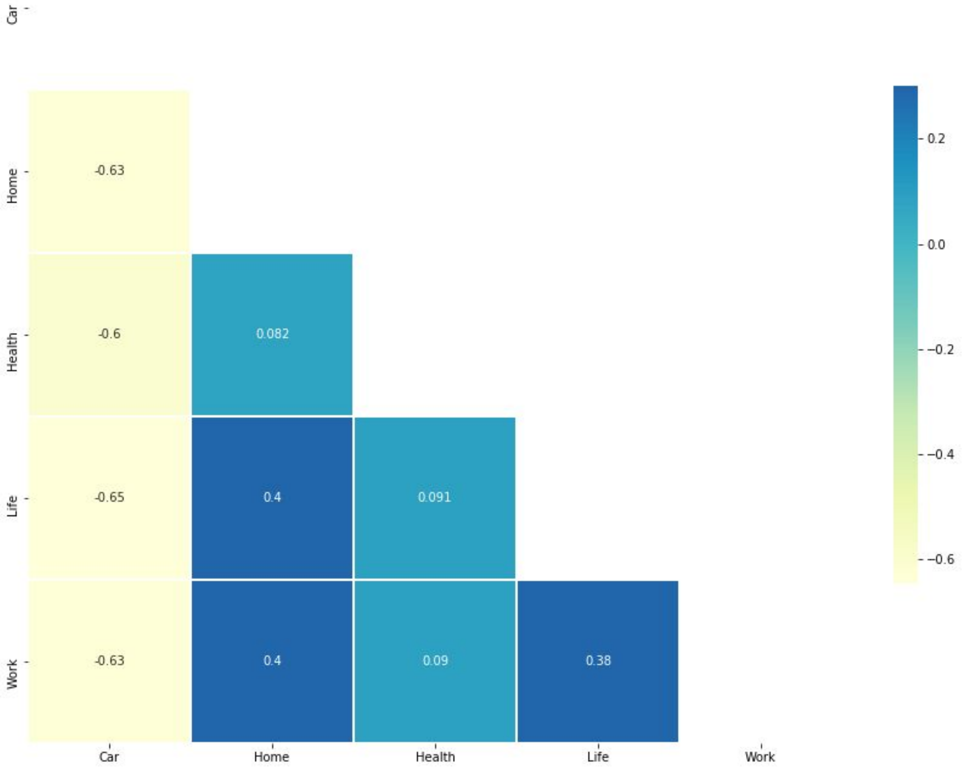
We will remove Claims Rate from input area for clustering.



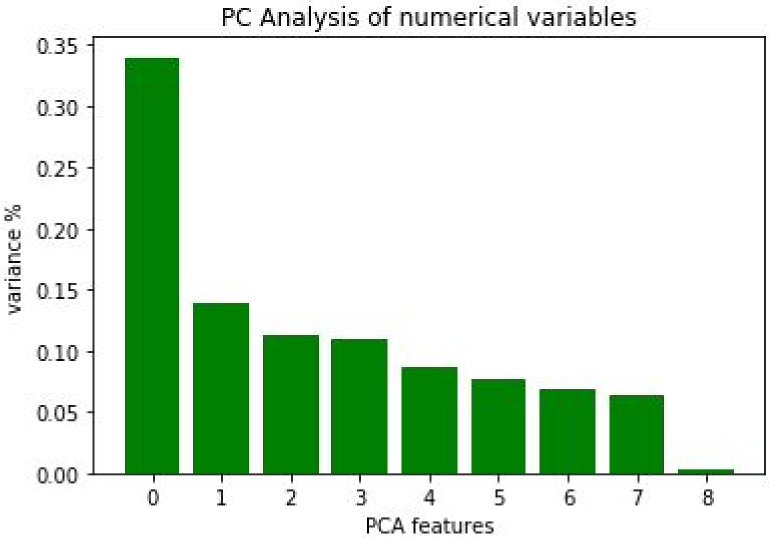
Policies Analysis

From below chart we can see that Life insurance is correlated to both Work compensations and House insurances. House insurance is correlated to Work compensations. We are likely to see clusters with a mix of Life, House and Work compensation.

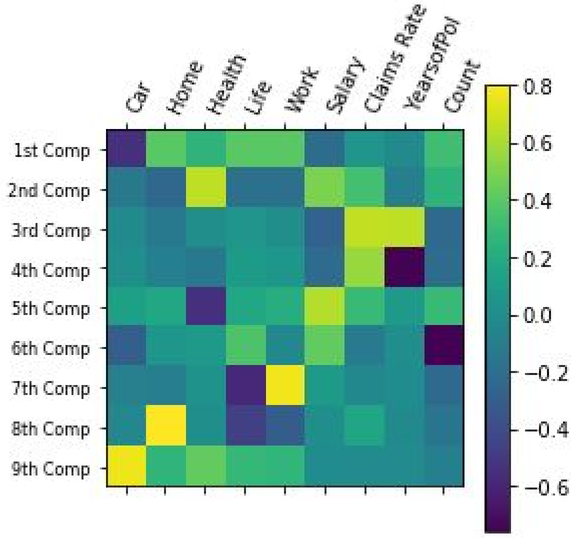
Car insurance is negatively correlated to all other policies which is an indicator that if we have a segment of clients buying a car insurance, they are not purchasing other types of policies.



We also performed PCA analysis on numerical values of customers together with policies to see which variables will have the highest explanatory power. Around 60% are explained by first 3 PCA's, then each additional PCA explains roughly another 10%.



We created heatplot to see how the features mixed up to create the components:



First PCA is majorly influenced by Car insurance, second PCA - by Health, and the third - by Years of policies and Claims rate. This gives us an indication of the features which will contribute mostly to our classifier in segmenting customers by policies they purchase.

We also performed PCA analysis on Policies and Customer numerical separately and the charts can be consulted in Appendix I.

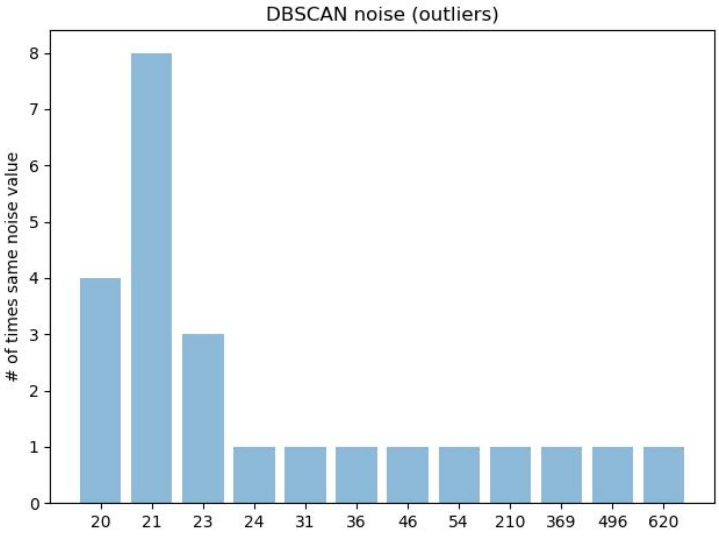
5. Data Pre-processing and cleaning

- **Data types** - numerical and categorical, we identified suitable approaches to deal with both types
- **Data completeness and null values**
  Data transformations performed:
  - deleted variable Birthday (replacing was not deemed suitable due to large number of false records)
  - replaced all null values in policies payments (absence of payment = zero payment
  - removed all other records with null values (it is the quickest and simplest and results in 78 records which is a small percent of total dataset. We considered replacing null values not to be resource efficient and might create noise)
- **Input space reduction**
  - Removed correlated variables: Claims Rate (left CMV)
- **Outlier detection** - our decisions to do visual univariate identification of thresholds for outliers based on most prominent variables and not follow interquartile methods as they detect too many outliers. So we first identified outliers for individual variables and then reviewed and re-confirmed them using DBSCAN for our full multivariable dataset stripped of null values.

21 Outliers removed by thresholding:

1. Salary over 30,000
2. Customer Monetary Value under -25,000 and over 10,000
3. Claims rate over 20
4. Total Premium over 20,000
5. Years of Policy under zero
6. Car insurance premiums over 2,000
7. House insurance premiums over 8,000
8. Health insurance premiums over 7,000
9. Work compensation policy premiums over 1,000

- Outliers were reconfirmed using DBSCAN which produced the same number of outliers (21) with 20 full matches. There was just 1 client identified as outlier by DBCAN in addition to existing outliers. This process has added confidence that the outliers considered previously are indeed data that have no significant value. Different sample quantities and radii were iteratively tested and the most stable and data inclusive instances were used as the final results. The same variables were used for both thresholding and DBSCAN.



- Data normalization due to different scales (using StandardScaler)

6. Data modelling

To each of the identified dimensions of the customers we applied the same clustering techniques in order to achieve the best segmentation result and stable clusters.

All variables enter the clustering algorithms unweighted.

Clustering customers based on Policies

Methods applied to numeric values of policies:

- K-means
- Hierarchical
- SOM
- DBSCAN

Methods applied to evaluate quality of clusters and clustering technique:

- Intra-cluster distances
- Silhouette scores and graph

Running elbow graph we considered the optimal cluster number would be between 2 and 4 clusters. Testing all three options we decided upon 2 clusters as input into k-means clustering which produced two well-pronounced groups of Car insurance buyers and Home, Life and Work insurance buyers. 2 clusters also got the best silhouette score and smallest intra-cluster distances.

We then applied Hierarchical clustering and SOM which re-confirmed our decisions on cluster number and composition from the k-means clusters.
DBSCAN proved unsuitable to identify clusters as the dataset has no distinct density based groupings so it yielded no significant results, however we decided to use it to review selected outliers as previously stated.

We also evaluated cluster results obtained by above techniques and k-means showed best silhouette score and shortest distances. K-means is also one of the most widely used methods and has less computational cost compared to other algorithms, so our final segmentation into 2 clusters is based on k-means, and the results of other clustering methods are saved in Appendix II for reference only.

Clustering customers based on Customer value and demographics

- all above methods were also performed for numeric values of clients value, and the final segmentation was done based on k-means algorithm
- k-modes was applied for categorical

Running elbow graph we considered the optimal cluster number would be between 2 and 4 clusters as for policies. Testing these options we decided upon 3 clusters as input into k-means clustering. Based on silhouette scores the smallest distances were achieved by 2 clusters, however with 3 clusters we get better interpretable groups with distinct salary ranges, years of policies and count of policies. Our decision was to go ahead with 3 clusters to make full use of these features. Full clustering results are referenced in Appendix II.

As for categorical values we have tested k-modes however we decided not to use the resulting clusters in our final segmentation. The set contains a relatively small number of features (3: Children, Education and Area) and we decided to use them for profiling the client segments we obtained for each dimension in previous steps instead of introducing another sub-segmentation.

Finally, in order to classify the clients from our outliers dataset, we applied both decision tree and k-nearest neighbours methods (70% training, 30% test set). Test result for decision tree showed better accuracy (30% mislabelled clients compared to 49% of mislabelled clients in decision tree) so we labelled outliers based on decision tree. We took the decision not to classify 78 clients with null values as they represent a very small share of the client dataset and can be removed from customized mailing lists and other marketing communications with no effect on overall performance. Since almost all client records show salary and year of first policy and analysis showed these are important segmentation features, we would recommend that these fields are set mandatory in the insurance database to prevent human errors and to help standardize the process in future.

Further details are in Appendix III.

Details of the final segmentation for each dimension will be presented in the following section[1].

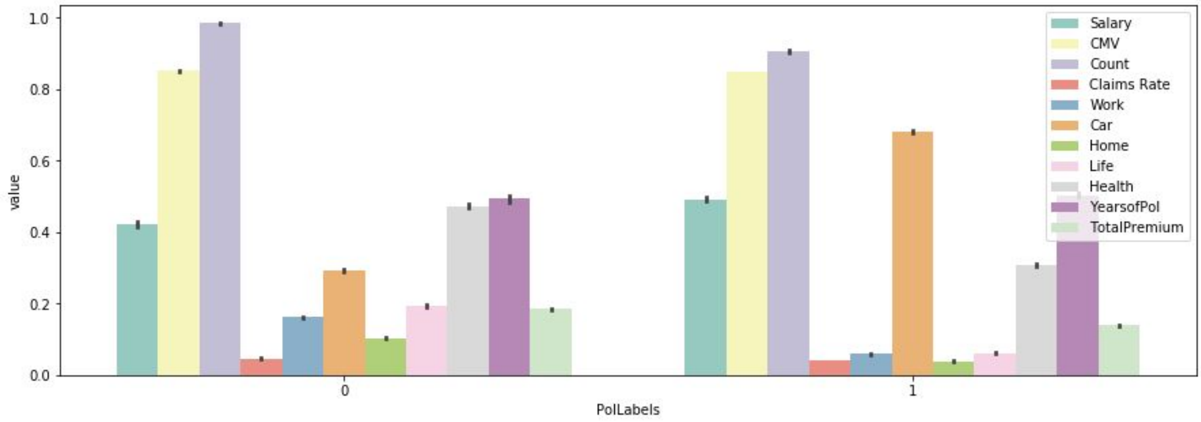
7. Final result and clusters description

Our final segmentation is based on two dimensions: policies portfolio and client portfolio.
Policies portfolio shows which insurance type has generated most income in policy payments. Client portfolio shows clients' financial profile focusing on their salary range and monetary value.

2 segments based on Policies portfolio:

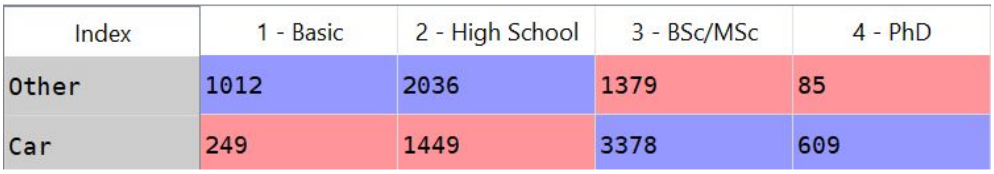
- Car insurance buyers
- House, Health and Work compensation buyers

[1] As initial centroids are set randomly, the segmentation output will slightly differ each time the code is run, however the overall result and cluster features remain the same and the attached excel file with labels can be used as final



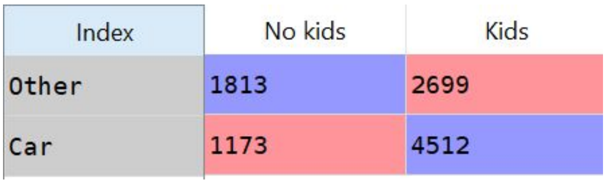
We used 3 categorical values of clients to profile these segments and show how the segments differ in terms of Education level, family situation and living area. Below graphs show clients count in each section:

Education profiles:



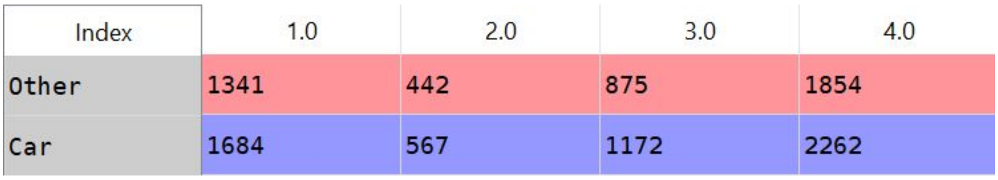
Index	1 - Basic	2 - High School	3 - BSc/MSc	4 - PhD
Other	1012	2036	1379	85
Car	249	1449	3378	609

Family profile:



Index	No kids	Kids
Other	1813	2699
Car	1173	4512

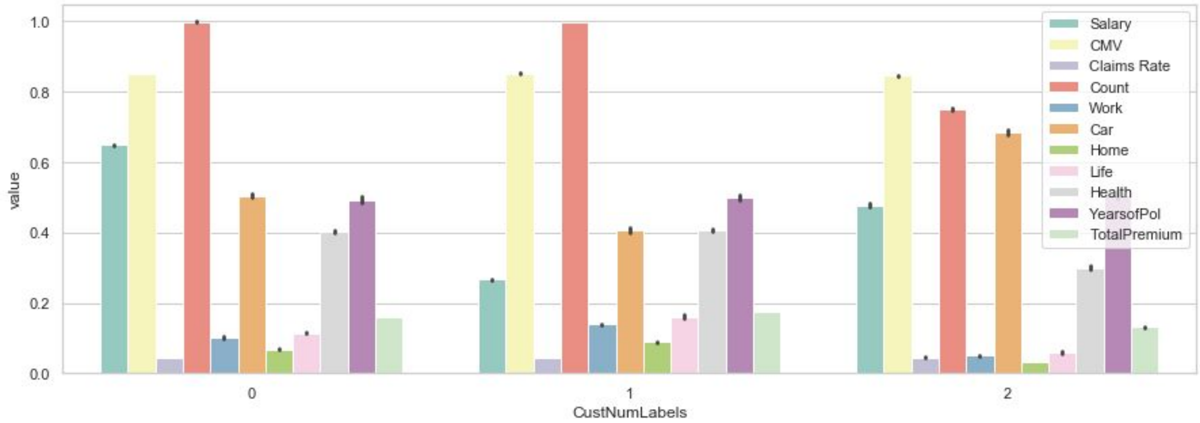
Area profile:



Index	1.0	2.0	3.0	4.0
Other	1341	442	875	1854
Car	1684	567	1172	2262

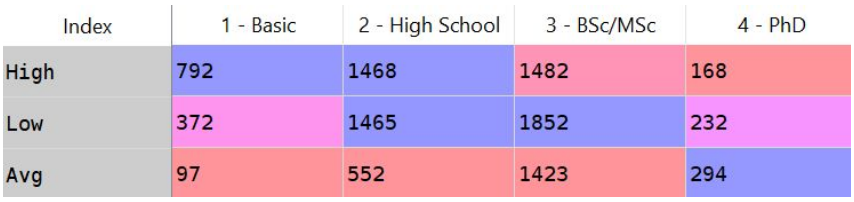
3 segments based on Client portfolio ('Salary','Claims Rate','YearsofPol','Count'):

- High earners, average monetary value, 5 policies
- Low earners, high monetary value, 5 policies
- Average earners, lower monetary value, lower policies count



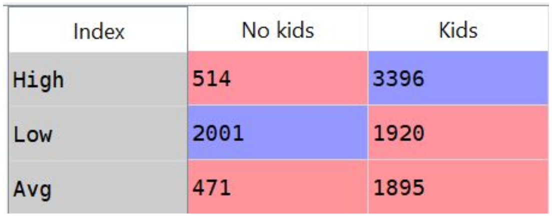
We used 3 categorical values of clients to profile these segments and show how the segments differ in terms of Education level, family situation and living area.

Education profile:



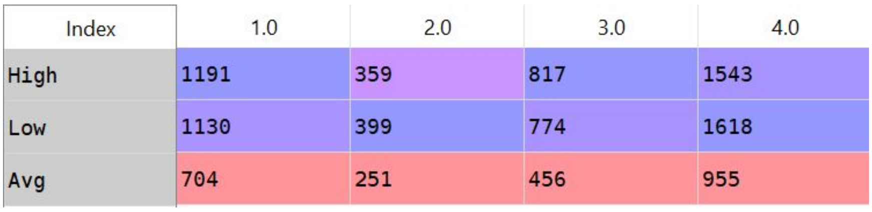
Index	1 - Basic	2 - High School	3 - BSc/MSc	4 - PhD
High	792	1468	1482	168
Low	372	1465	1852	232
Avg	97	552	1423	294

Family profile:



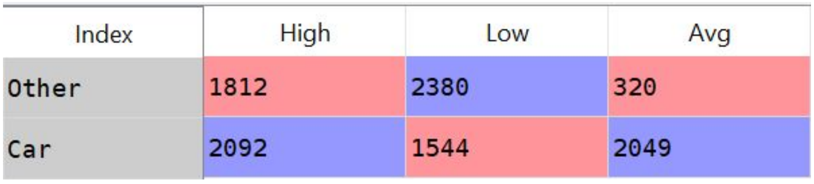
Index	No kids	Kids
High	514	3396
Low	2001	1920
Avg	471	1895

Area profile:

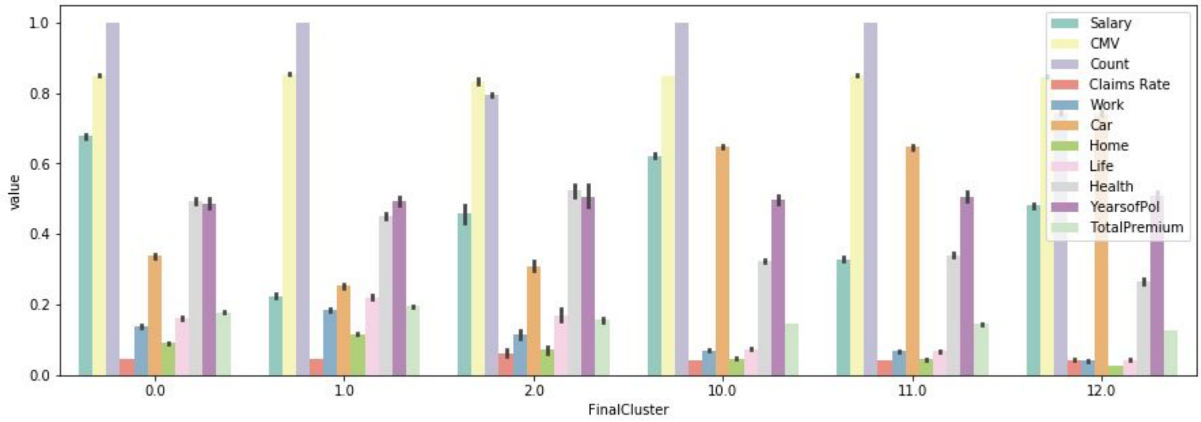


Index	1.0	2.0	3.0	4.0
High	1191	359	817	1543
Low	1130	399	774	1618
Avg	704	251	456	955

When crossing the segments we obtained for each of the dimensions we ended up with a total of **6 micro-segments.** Below is the clients count for each segment:



Index	High	Low	Avg
Other	1812	2380	320
Car	2092	1544	2049



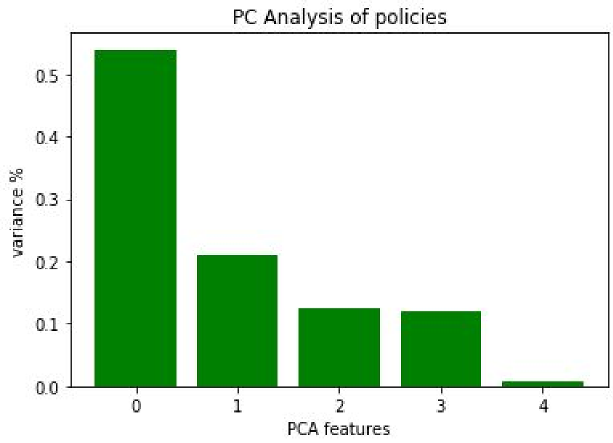
- For high earners groups (High-Other and High-Cars) we can boost loyalty and retain those clients by offering customized discounts after N'th year as a client provided the claims rate is not reaching a certain ceiling. Although they are not the target, this strategy can be used to encourage reduction in claims rate (and not as a churn measure) on the Average and Low customers.

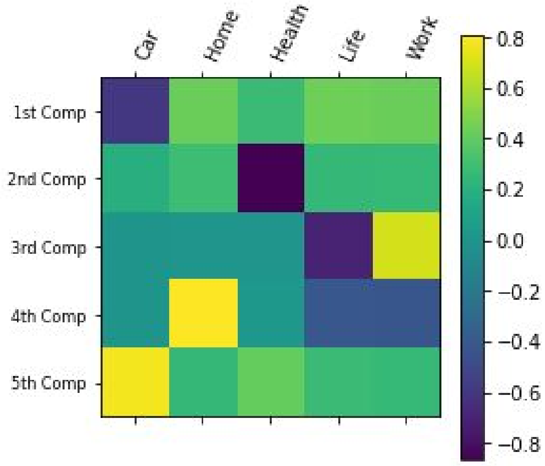
- For average earners (Avg-Other, Avg-Car) who purchased less than 5 policies we advise to cross-sell the insurances (if client has two or more insurances with the company - offer other available products at a discounted rate)

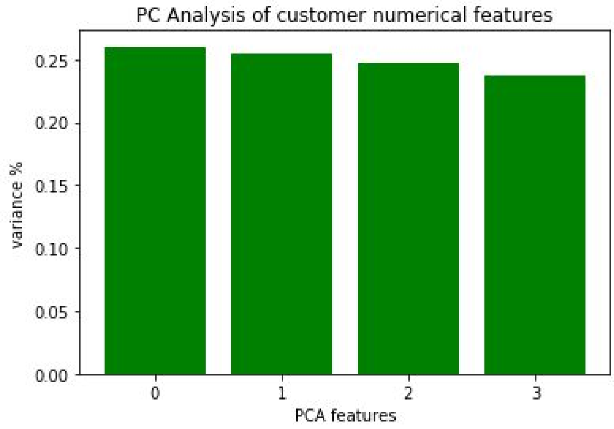
- In addition, we generated **red-flag** for over 1.5 thousand clients who dropped one or more policies. We decided to only consider Health, Life, Work compensation as we have no additional information based on clients home or car ownership. This will allow the marketing team to analyse different customer profiles and evaluate on how to bring them back and prevent dropping in future. We also recommend adding to customers database a field for reason of clients stopping the policy which requires mandatory input from sales person. This will help in further analysis.

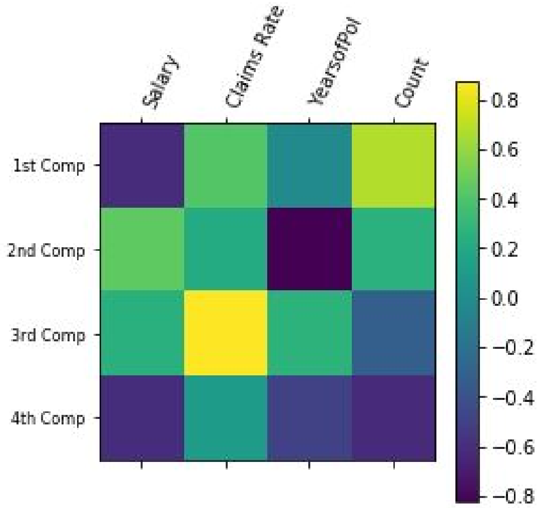
Appendices

Appendix I. Principal Components Analysis







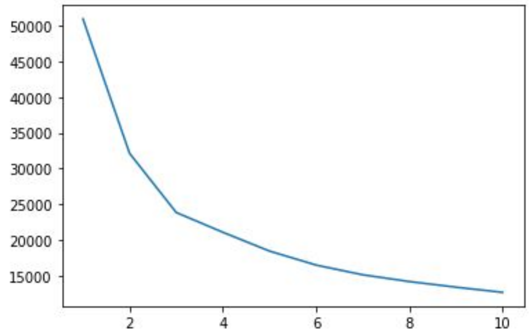


Appendix II. Clustering graphs

1. Clustering customers based on policies

K-MEANS

Elbow graph for policies shows turning point at either 2, 3 or 4:



Result with 4 clusters:

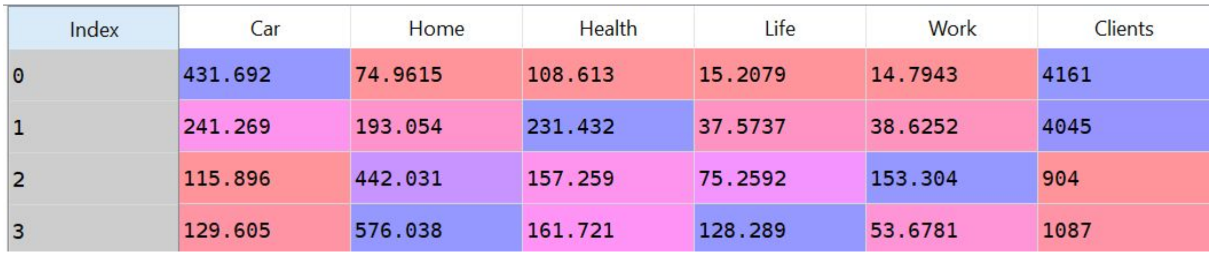
<u>Clusters</u>:

0 = Car insurance

1 = Health insurance

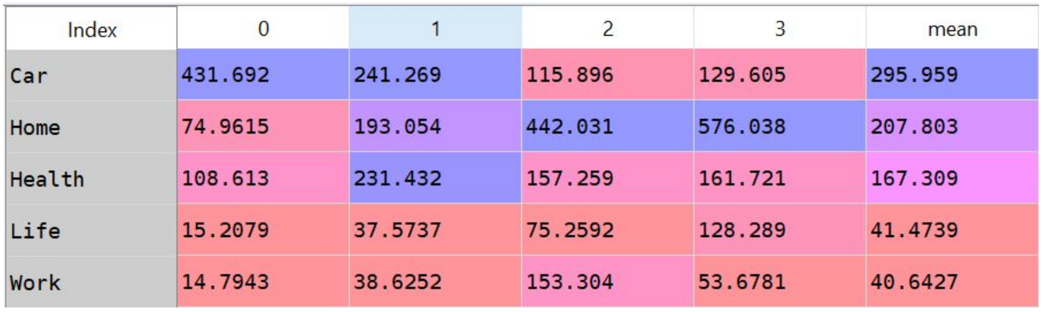
2 = Home, life & work compensation

3 = Home, life & work compensation (very similar to 2)

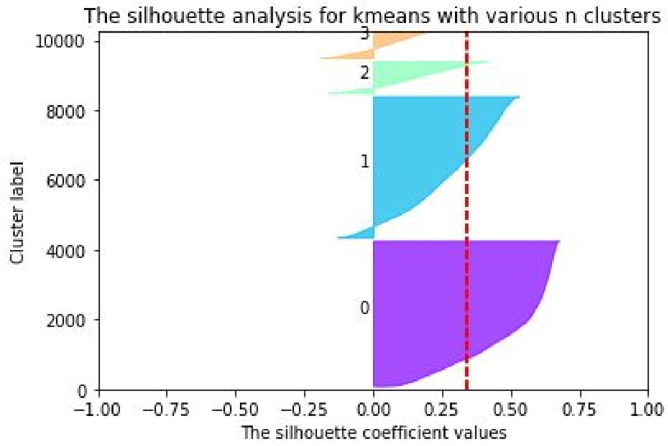


Index	Car	Home	Health	Life	Work	Clients
0	431.692	74.9615	108.613	15.2079	14.7943	4161
1	241.269	193.054	231.432	37.5737	38.6252	4045
2	115.896	442.031	157.259	75.2592	153.304	904
3	129.605	576.038	161.721	128.289	53.6781	1087

Comparing with the average:



Index	0	1	2	3	mean
Car	431.692	241.269	115.896	129.605	295.959
Home	74.9615	193.054	442.031	576.038	207.803
Health	108.613	231.432	157.259	161.721	167.309
Life	15.2079	37.5737	75.2592	128.289	41.4739
Work	14.7943	38.6252	153.304	53.6781	40.6427



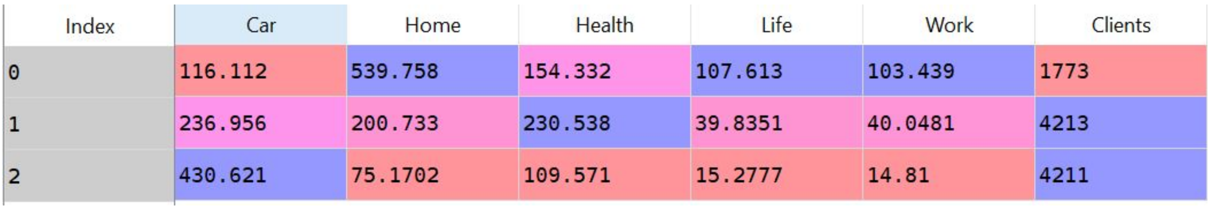
Result with 3 clusters:

Clusters:

0 = Home, Life and Work above average

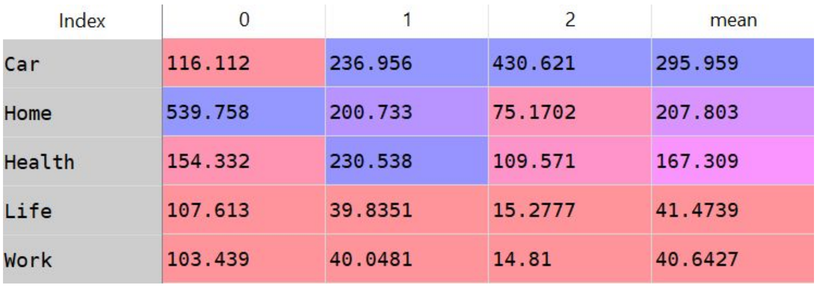
1 = Focusing on Health (above average) and some average spending on other insurances

2 = Clients focusing on car insurance (other insurance are lower than average)

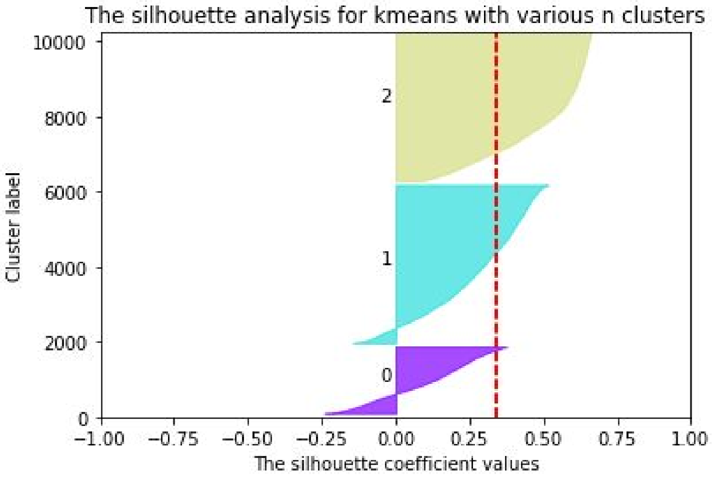


Index	Car	Home	Health	Life	Work	Clients
0	116.112	539.758	154.332	107.613	103.439	1773
1	236.956	200.733	230.538	39.8351	40.0481	4213
2	430.621	75.1702	109.571	15.2777	14.81	4211

Comparing with the average:



Index	0	1	2	mean
Car	116.112	236.956	430.621	295.959
Home	539.758	200.733	75.1702	207.803
Health	154.332	230.538	109.571	167.309
Life	107.613	39.8351	15.2777	41.4739
Work	103.439	40.0481	14.81	40.6427

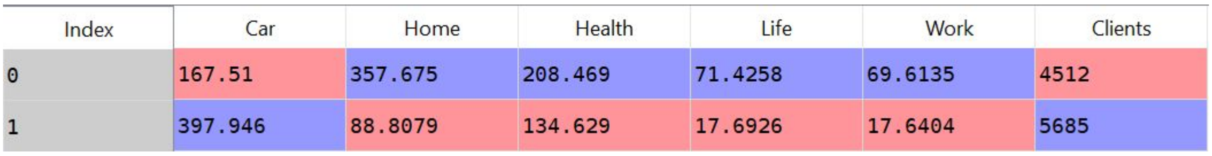


Result with 2 clusters:

Clusters:

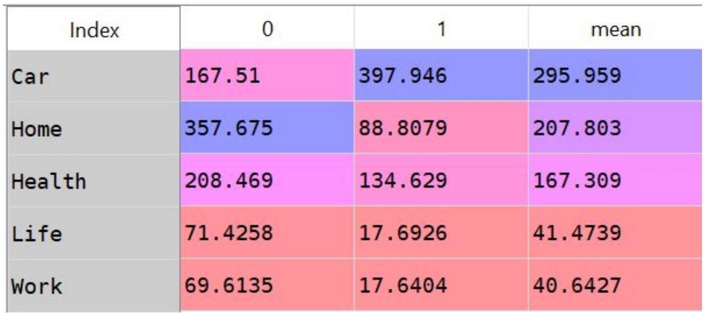
0 = Home,Health, Life and Work are above average, low on Cars

1 = Car insurance purchasers

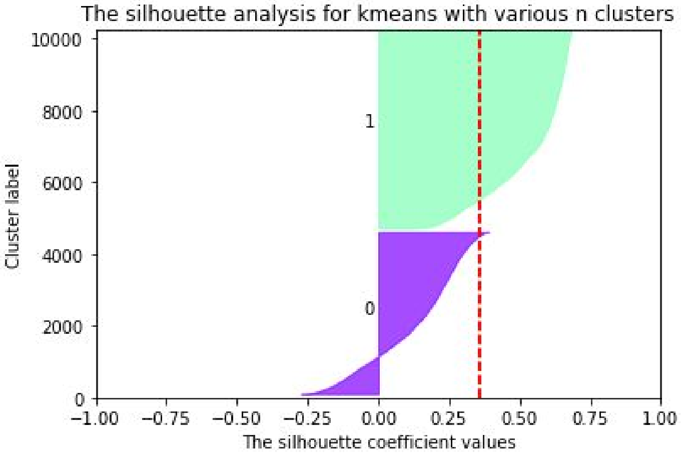


Index	Car	Home	Health	Life	Work	Clients
0	167.51	357.675	208.469	71.4258	69.6135	4512
1	397.946	88.8079	134.629	17.6926	17.6404	5685

Comparing with the average:



Index	0	1	mean
Car	167.51	397.946	295.959
Home	357.675	88.8079	207.803
Health	208.469	134.629	167.309
Life	71.4258	17.6926	41.4739
Work	69.6135	17.6404	40.6427



Evaluation:

For n_clusters = 4 The average silhouette_score is : 0.34200983821515124

For n_clusters= 4 The Distance Mean: 353.85919846274174

For n_clusters = 3 The average silhouette_score is : 0.3399405880993841

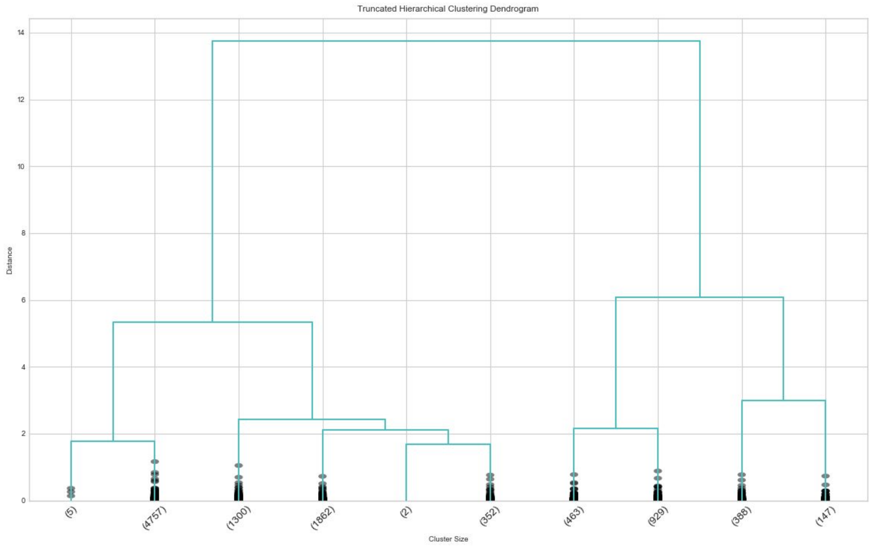
For n_clusters = 3 The Distance Mean: 330.397898414558

For n_clusters = 2 The average silhouette_score is : 0.36065399507323265

For n_clusters = 2 The Distance Mean: 292.411055926153

HIERARCHICAL CLUSTERING

Dendogram graph for policies shows a cut line around 4 clusters:



<u>Clusters</u>:

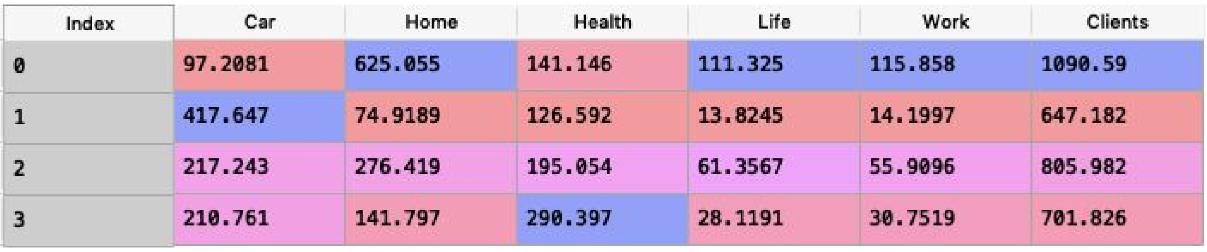
The result of hierarchical applied over policies are 4 clusters divided by:

0 = Home highly above average; Life and Work above the average; low on Cars and Health

1 = Car above the average and every other variable below the average

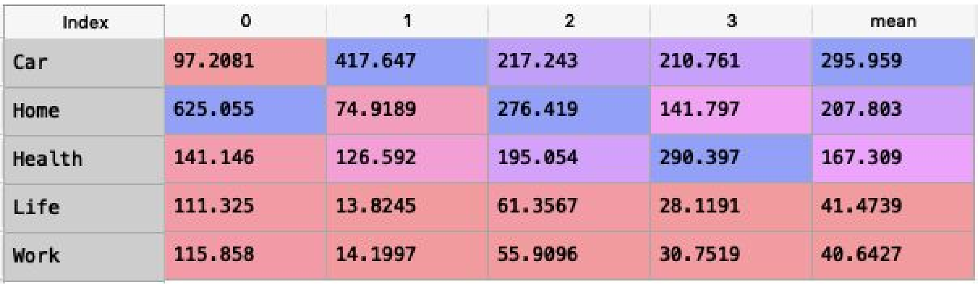
2 = Home and Health above the average; Car, Life and Work below

3 = Health above average and every other variable below the average

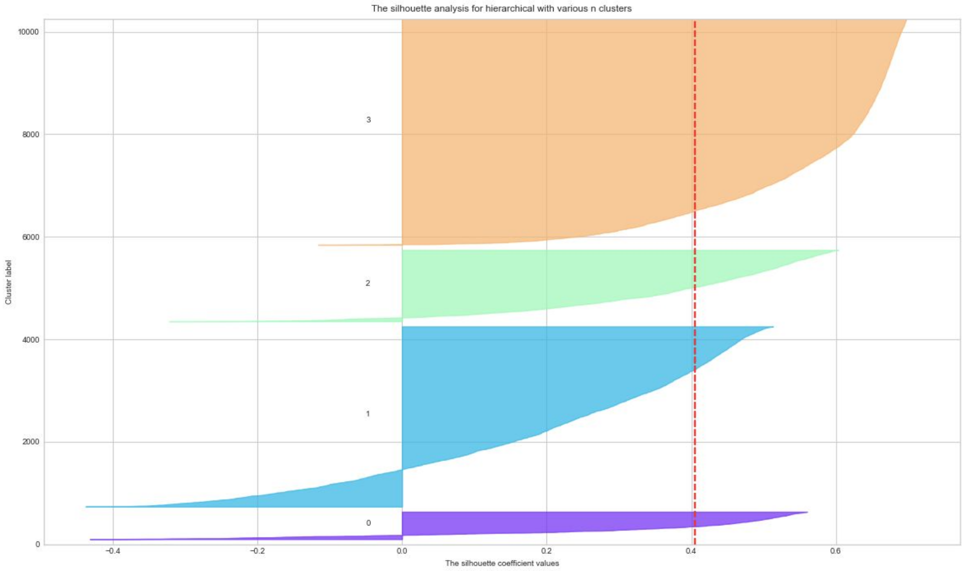


Index	Car	Home	Health	Life	Work	Clients
0	97.2081	625.055	141.146	111.325	115.858	1090.59
1	417.647	74.9189	126.592	13.8245	14.1997	647.182
2	217.243	276.419	195.054	61.3567	55.9096	805.982
3	210.761	141.797	290.397	28.1191	30.7519	701.826

Comparing with the average:



Index	0	1	2	3	mean
Car	97.2081	417.647	217.243	210.761	295.959
Home	625.055	74.9189	276.419	141.797	207.803
Health	141.146	126.592	195.054	290.397	167.309
Life	111.325	13.8245	61.3567	28.1191	41.4739
Work	115.858	14.1997	55.9096	30.7519	40.6427



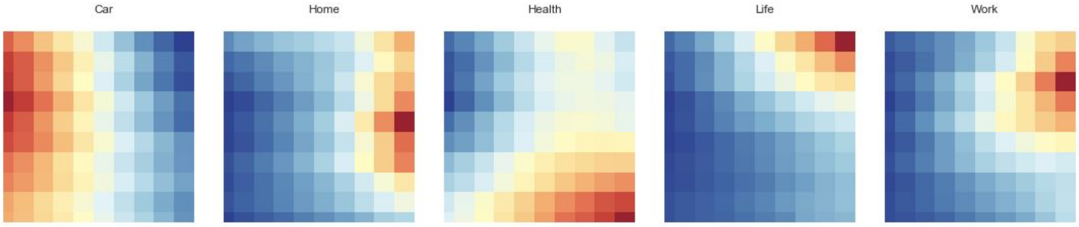
<u>Evaluation</u>:

- For n_clusters = 4 The average silhouette_score is : 0.25806696116016864
- For n_clusters = 4 The Distance Mean: 339.79456310432766

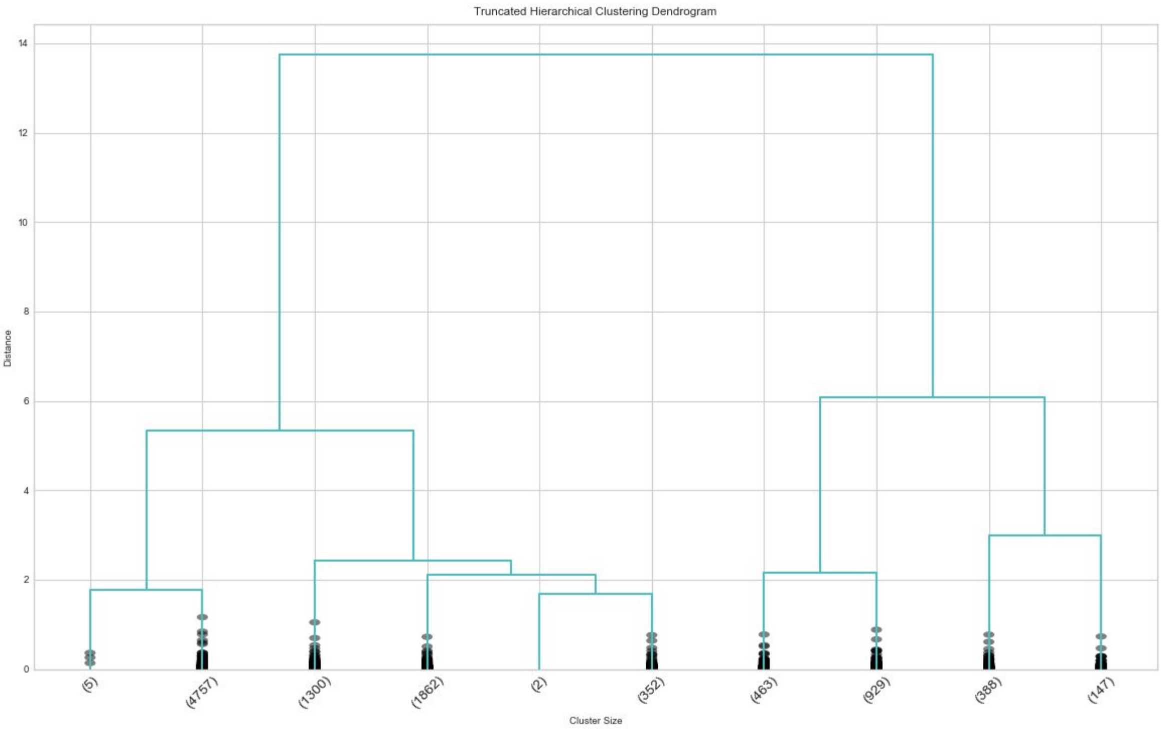
SELF-ORGANIZING MAPS CLUSTERING

Visually the result of self-organizing maps applied over policies is showing the optimal distribution to 3 client groups based on the type of insurance they hold:

0 = Car insurance buyers
1 = More of household + mix of others
2 = More of health, no other



BUILDING DENDOGRAM OVER SOM



Result with 4 clusters:

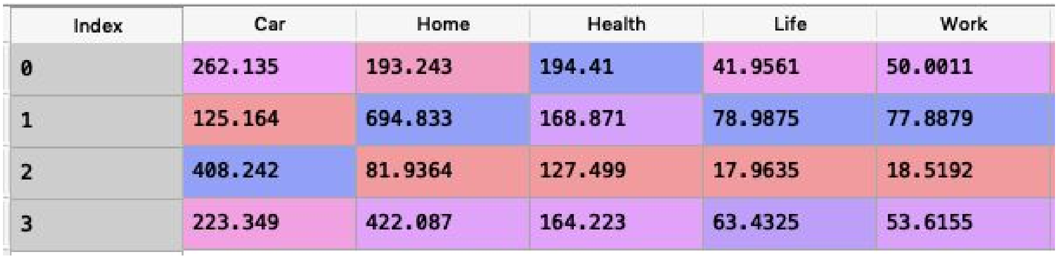
<u>Analysis</u>

0 = Everything average

1 = Low car, high home, high life, high work

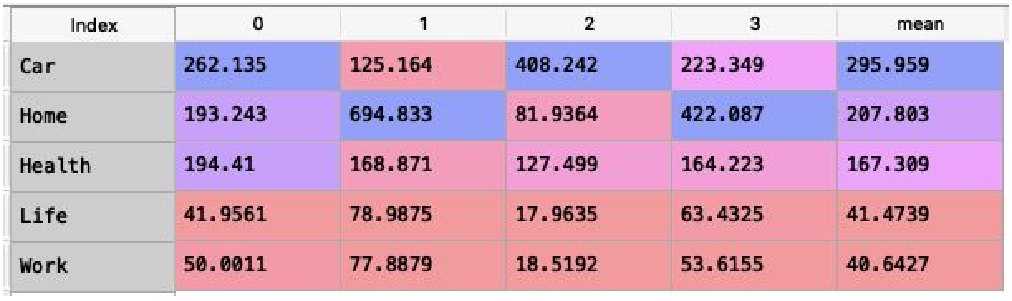
2 = High car, low home, low life, low work

3 = High home, high life



Index	Car	Home	Health	Life	Work
0	262.135	193.243	194.41	41.9561	50.0011
1	125.164	694.833	168.871	78.9875	77.8879
2	408.242	81.9364	127.499	17.9635	18.5192
3	223.349	422.087	164.223	63.4325	53.6155

Comparing with the average:

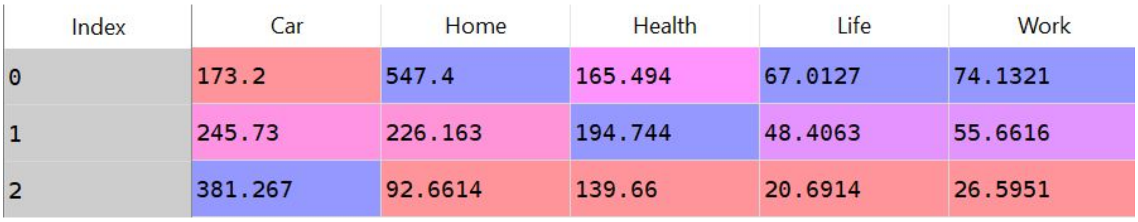


Index	0	1	2	3	mean
Car	262.135	125.164	408.242	223.349	295.959
Home	193.243	694.833	81.9364	422.087	207.803
Health	194.41	168.871	127.499	164.223	167.309
Life	41.9561	78.9875	17.9635	63.4325	41.4739
Work	50.0011	77.8879	18.5192	53.6155	40.6427

Result with 3 clusters:

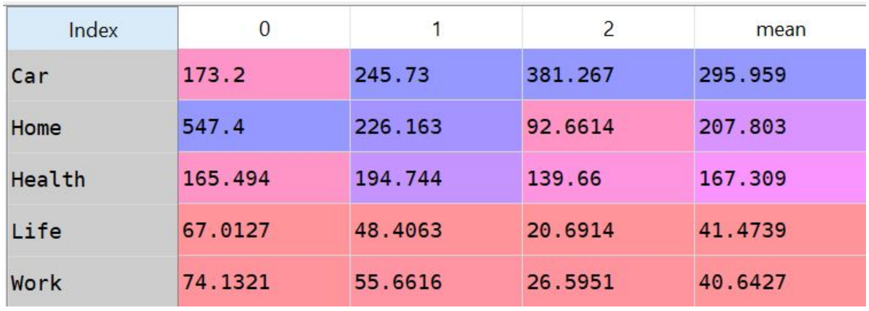
<u>Analysis:</u>

0= Low Car, High Home, Life and Work
1= Everything average
2= High Car, low on the other variables



Index	Car	Home	Health	Life	Work
0	173.2	547.4	165.494	67.0127	74.1321
1	245.73	226.163	194.744	48.4063	55.6616
2	381.267	92.6614	139.66	20.6914	26.5951

Comparing with average of full clients dataset:

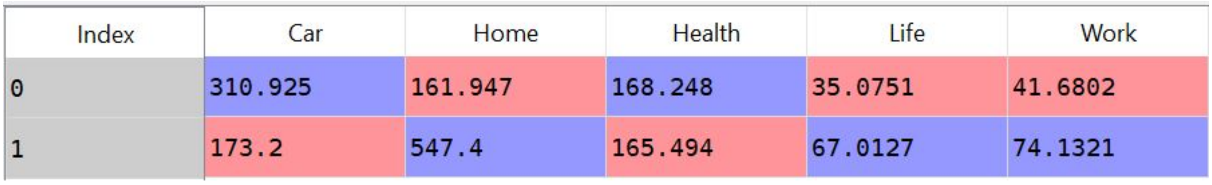


Index	0	1	2	mean
Car	173.2	245.73	381.267	295.959
Home	547.4	226.163	92.6614	207.803
Health	165.494	194.744	139.66	167.309
Life	67.0127	48.4063	20.6914	41.4739
Work	74.1321	55.6616	26.5951	40.6427

Result with 2 clusters:

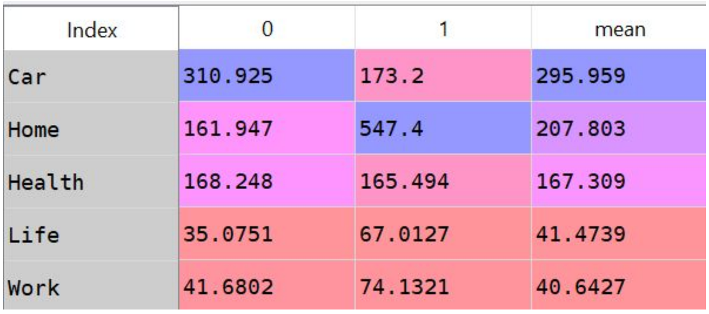
Clusters:

0 = Car insurance buyers
1 = House insurance buyers + mix of others



Index	Car	Home	Health	Life	Work
0	310.925	161.947	168.248	35.0751	41.6802
1	173.2	547.4	165.494	67.0127	74.1321

Comparing with average of full clients dataset:



Index	0	1	mean
Car	310.925	173.2	295.959
Home	161.947	547.4	207.803
Health	168.248	165.494	167.309
Life	35.0751	67.0127	41.4739
Work	41.6802	74.1321	40.6427

Evaluation:
For n_clusters = 2 The Distance Mean: 332.2916141509375
For n_clusters = 3 The Distance Mean: 312.09399762389404
For n_clusters = 4 The Distance Mean: 354.588523850971

DBSCAN

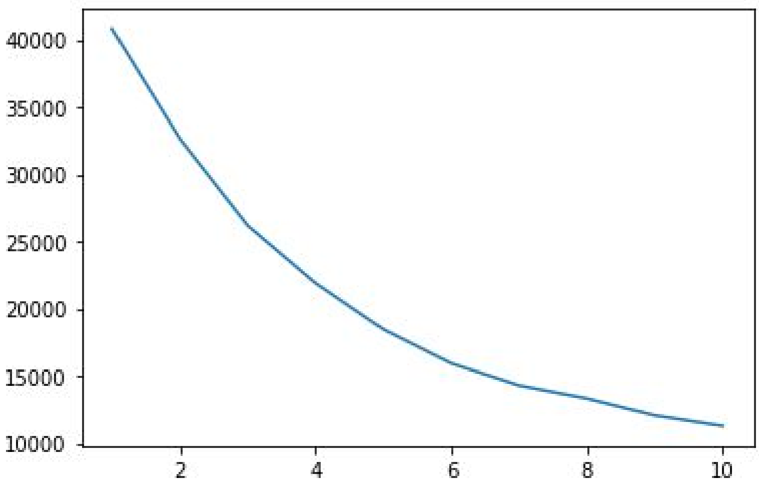
For the purpose of describing and retrieving inferences from the data, DBSCAN was used as a clustering method. Unfortunately, it provided very poor clusters, always having a huge one and many small ones and/or noise. Only the variables pertaining to the policies value from each customer were used and tested with different epsilon and minimum data contained within yielding persistently the same results. It is impossible to infer anything from it since it has no interpretative value and so this method was just used as an outlier detector to confirmed already determined possible outliers as stated earlier.

2. Clustering customers based on demographics and value

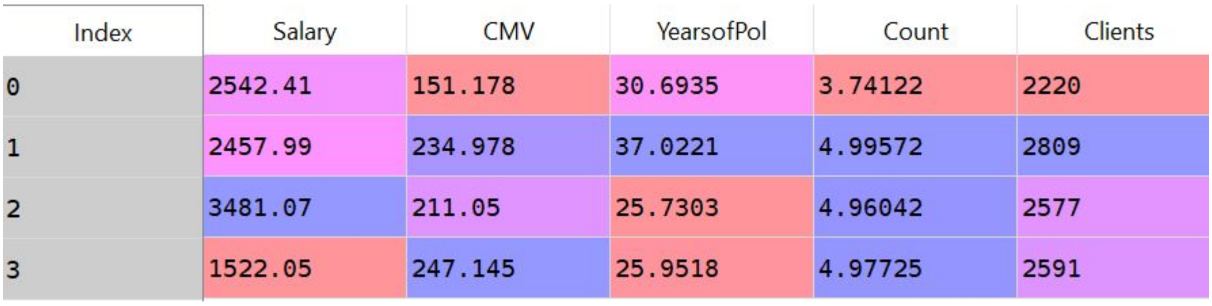
Customer numerical values

K-MEANS

Elbow graph for customers shows turning point at either 2, 3 or 4:

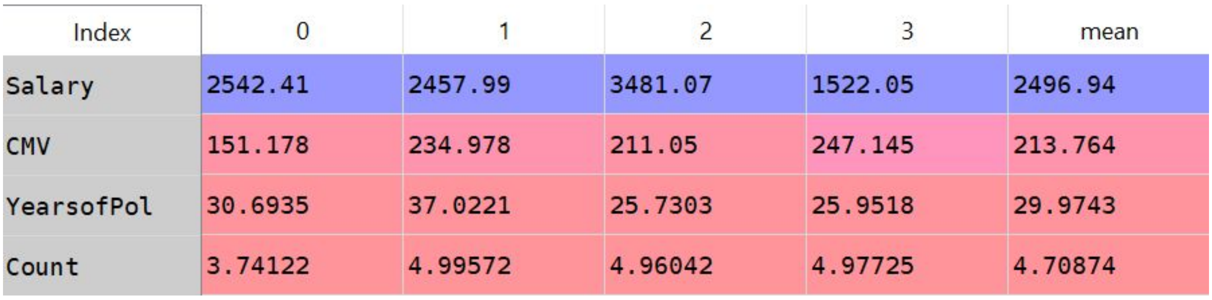


Result with 4 clusters:
<u>Clusters:</u>
0 = Average earners, few policies, low CMV
1 = Average earners, all policies, average CMV, long years with the company
2 = High earners, average CMV
3 = Low earners, average CMV

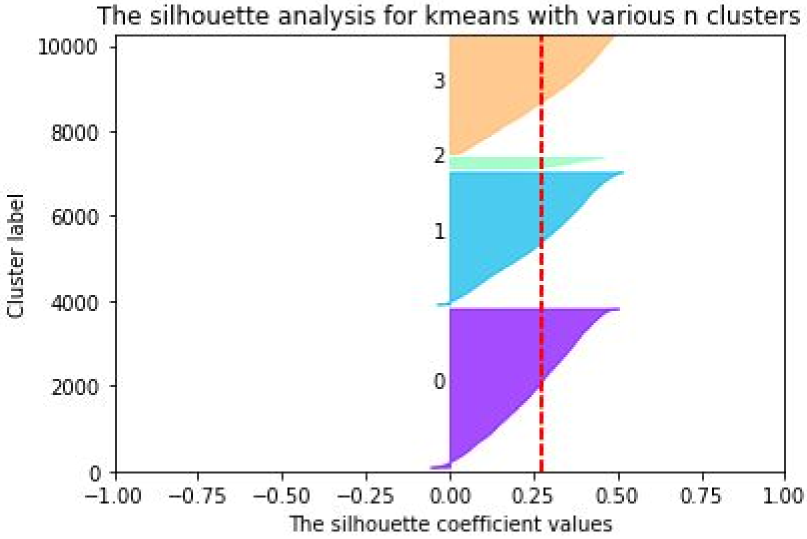


Index	Salary	CMV	YearsofPol	Count	Clients
0	2542.41	151.178	30.6935	3.74122	2220
1	2457.99	234.978	37.0221	4.99572	2809
2	3481.07	211.05	25.7303	4.96042	2577
3	1522.05	247.145	25.9518	4.97725	2591

Comparing with average of full clients dataset:



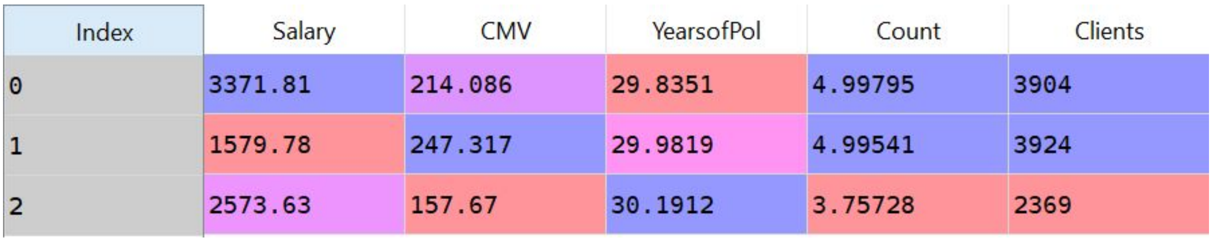
Index	0	1	2	3	mean
Salary	2542.41	2457.99	3481.07	1522.05	2496.94
CMV	151.178	234.978	211.05	247.145	213.764
YearsofPol	30.6935	37.0221	25.7303	25.9518	29.9743
Count	3.74122	4.99572	4.96042	4.97725	4.70874



Result with 3 clusters:

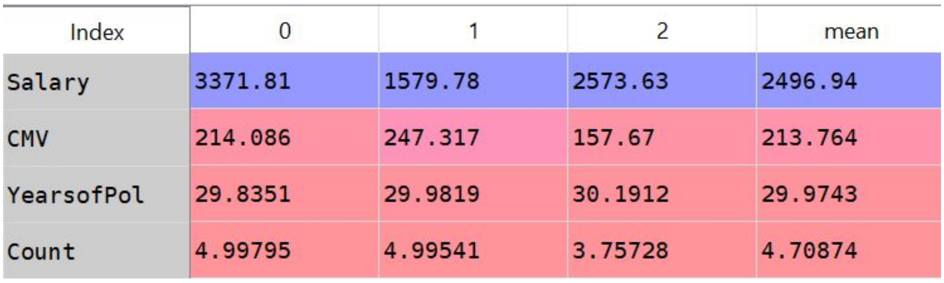
<u>Clusters:</u>

0 = High earners, average monetary value, 5 policies
1 = Low earners, high monetary value, 5 policies
2 = Average earners, lower monetary value, lower policies count

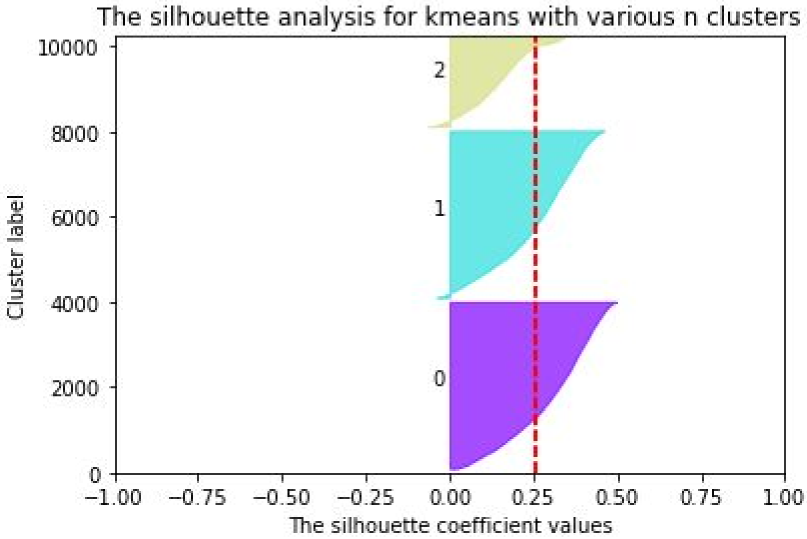


Index	Salary	CMV	YearsofPol	Count	Clients
0	3371.81	214.086	29.8351	4.99795	3904
1	1579.78	247.317	29.9819	4.99541	3924
2	2573.63	157.67	30.1912	3.75728	2369

Comparing with average of full clients dataset:



Index	0	1	2	mean
Salary	3371.81	1579.78	2573.63	2496.94
CMV	214.086	247.317	157.67	213.764
YearsofPol	29.8351	29.9819	30.1912	29.9743
Count	4.99795	4.99541	3.75728	4.70874

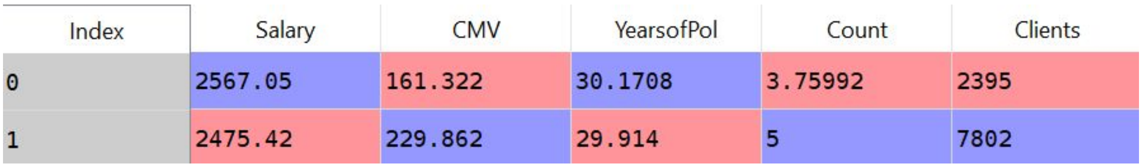


Result with 2 clusters:

Clusters:

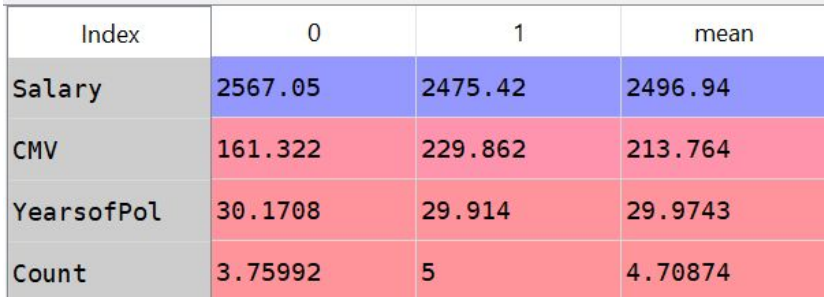
0 = Average earners, low CMV, less policies

1 = Average earners, high CMV, 5 policies

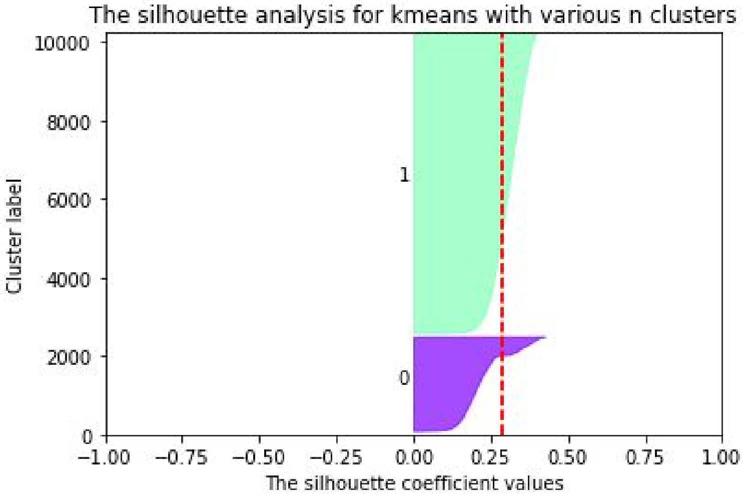


Index	Salary	CMV	YearsofPol	Count	Clients
0	2567.05	161.322	30.1708	3.75992	2395
1	2475.42	229.862	29.914	5	7802

Comparing with average of full clients dataset:



Index	0	1	mean
Salary	2567.05	2475.42	2496.94
CMV	161.322	229.862	213.764
YearsofPol	30.1708	29.914	29.9743
Count	3.75992	5	4.70874



Evaluation:

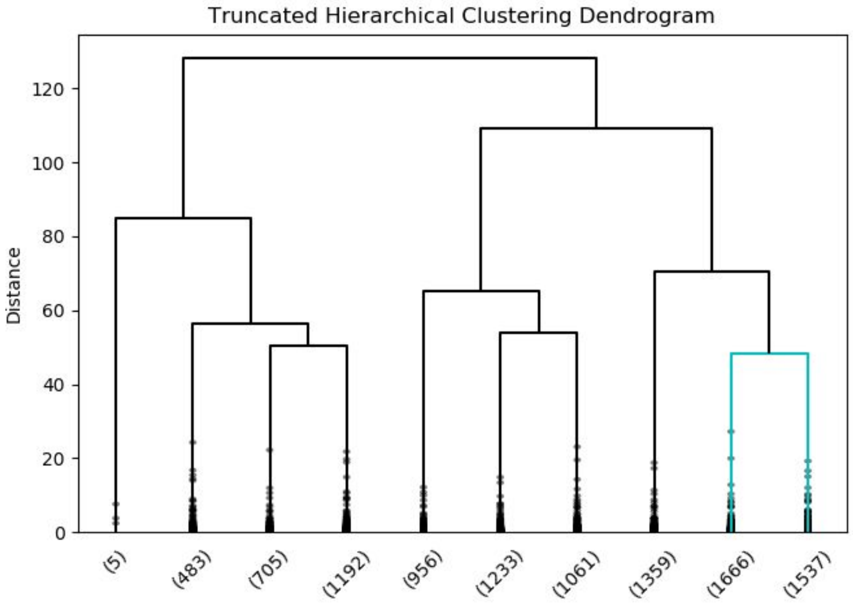
For n_clusters = 4 The average silhouette_score is : 0.24182319719009954
For n_clusters = 4 The Distance Mean: 1042.620297901392

For n_clusters = 3 The average silhouette_score is : 0.25671402661382137
For n_clusters = 3 The Distance Mean: 1060.1984963018197

For n_clusters = 2 The average silhouette_score is : 0.2894027819483478
For n_clusters = 2 The Distance Mean: 895.2135835795928

HIERARCHICAL CLUSTERING

Dendogram graph for policies shows a cut line around 3/4 clusters:



Result with 4 clusters:

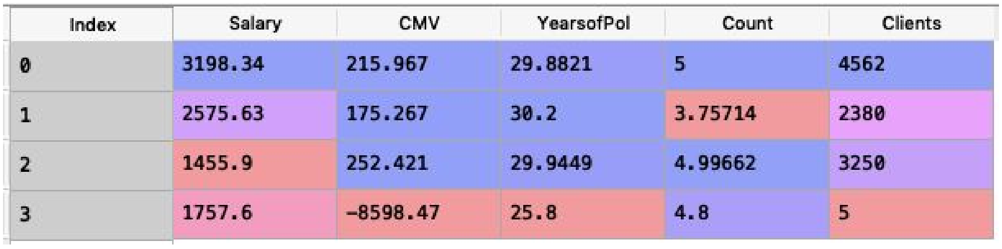
Analysis:

0 = High earners, average CMV, 5 policies

1 = Average earners, low CMV, less policies

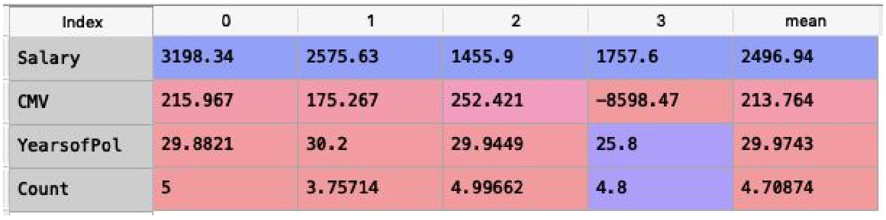
2 = Low earners, high CMV, +/- 5 policies

3 = Low earners, VERY low CMV, +/- 5 policies ---> Small Cluster



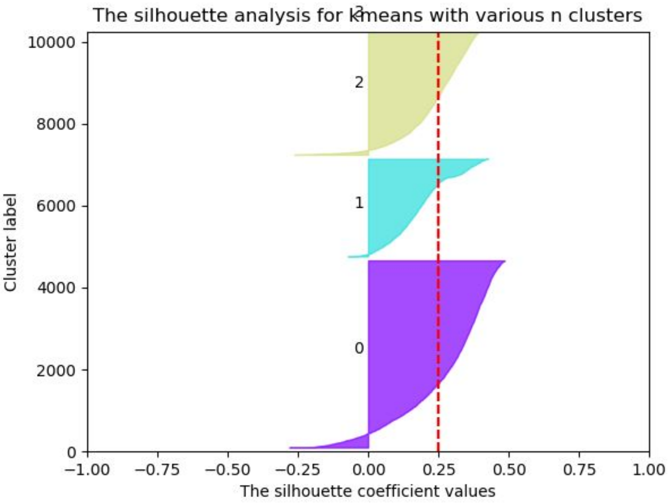
Index	Salary	CMV	YearsofPol	Count	Clients
0	3198.34	215.967	29.8821	5	4562
1	2575.63	175.267	30.2	3.75714	2380
2	1455.9	252.421	29.9449	4.99662	3250
3	1757.6	-8598.47	25.8	4.8	5

Comparing with the average:



Index	0	1	2	3	mean
Salary	3198.34	2575.63	1455.9	1757.6	2496.94
CMV	215.967	175.267	252.421	-8598.47	213.764
YearsofPol	29.8821	30.2	29.9449	25.8	29.9743
Count	5	3.75714	4.99662	4.8	4.70874

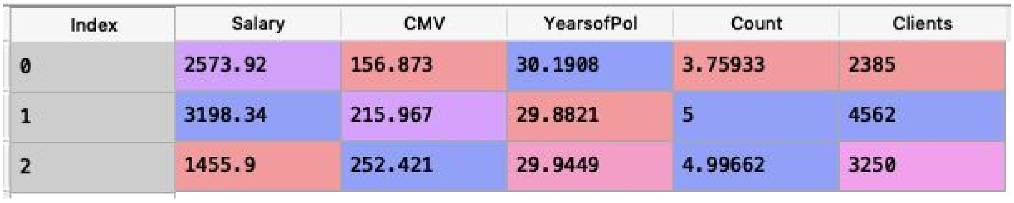
The particular size of cluster 3 does not allow it to appear on silhouette graph



Result with 3 clusters:

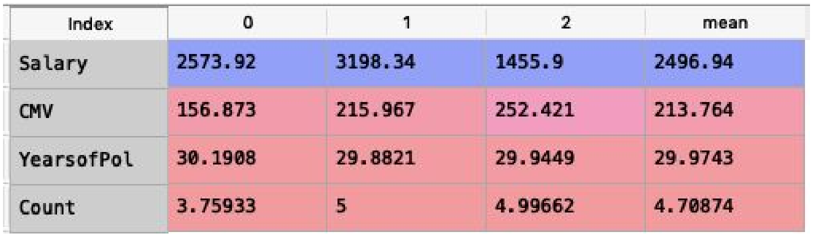
Clusters:

0 = Average earners, low CMV, less policies
1 = High earners, average CMV, 5 policies
2 = Low earners, high CMV, +/- 5 policies

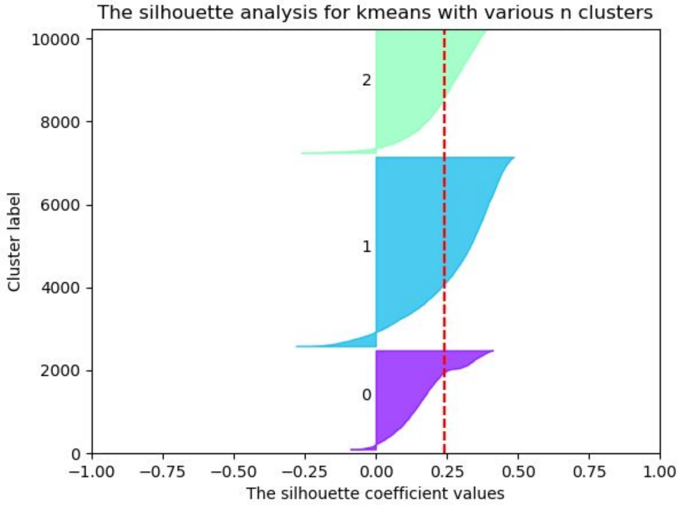


Index	Salary	CMV	YearsofPol	Count	Clients
0	2573.92	156.873	30.1908	3.75933	2385
1	3198.34	215.967	29.8821	5	4562
2	1455.9	252.421	29.9449	4.99662	3250

Comparing with the average:



Index	0	1	2	mean
Salary	2573.92	3198.34	1455.9	2496.94
CMV	156.873	215.967	252.421	213.764
YearsofPol	30.1908	29.8821	29.9449	29.9743
Count	3.75933	5	4.99662	4.70874



Evaluation:

For n_clusters = 3 The average silhouette_score is : 0.24344552882561546
For n_clusters = 3 The Distance Mean: 1057.3448729732822

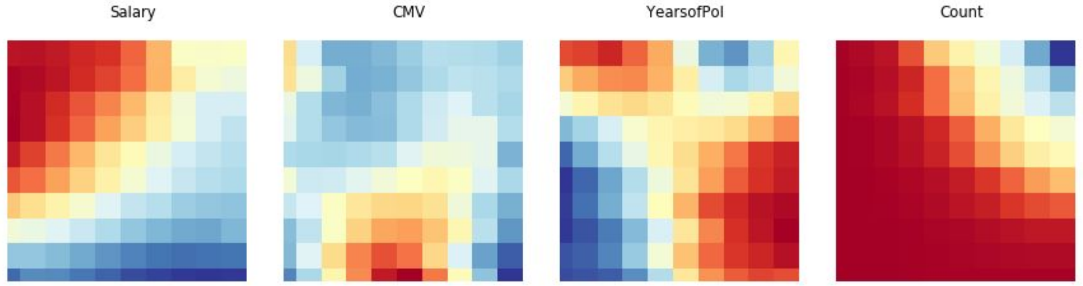
For n_clusters = 4 The average silhouette_score is : 0.24799751671308426
For n_clusters = 4 The Distance Mean: 3017.3499395252825

SELF-ORGANIZING MAPS CLUSTERING

Visually the result of self-organizing maps applied over customer numeric values are showing the optimal distribution to 3 client groups based on their features:

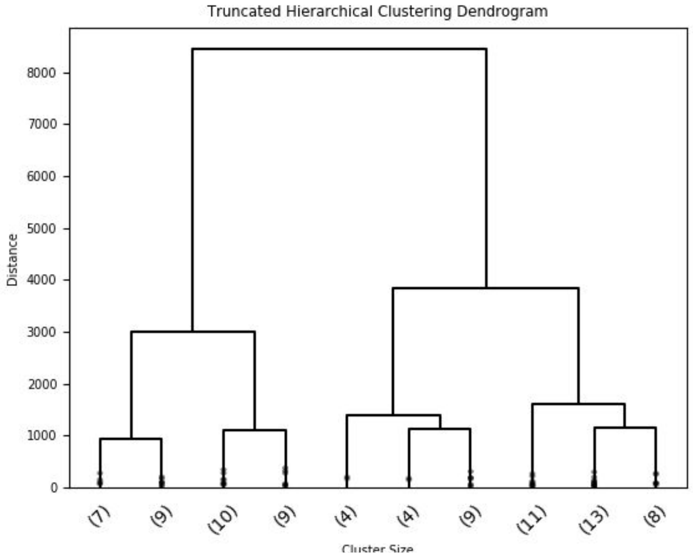
0 = Salary
1 = CMV

2 = Loyalty, years of policy



BUILDING DENDOGRAM OVER SOM

Dendogram graph for policies shows a cut line around 2, 3 or 4 clusters:



Result with 4 clusters:

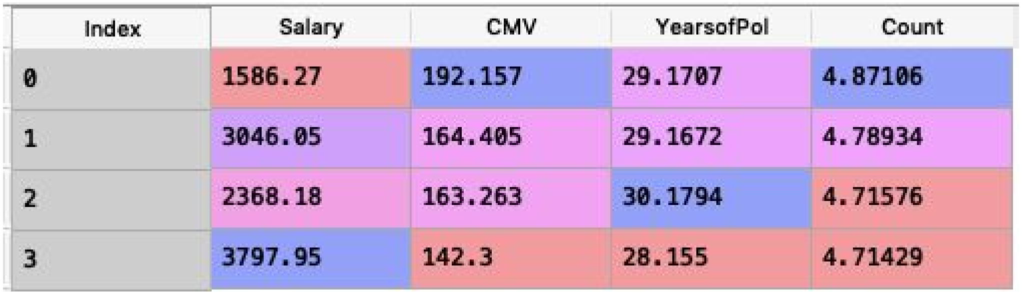
Analysis

0= Low Salary

1= Average Salary

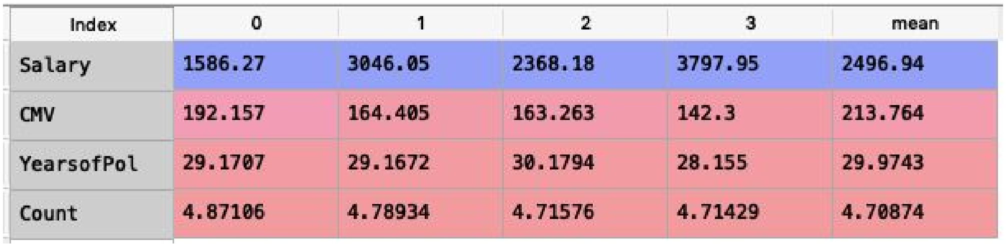
2= Average Salary

3= High Salary , low CMV



Index	Salary	CMV	YearsofPol	Count
0	1586.27	192.157	29.1707	4.87106
1	3046.05	164.405	29.1672	4.78934
2	2368.18	163.263	30.1794	4.71576
3	3797.95	142.3	28.155	4.71429

Comparing with the average:

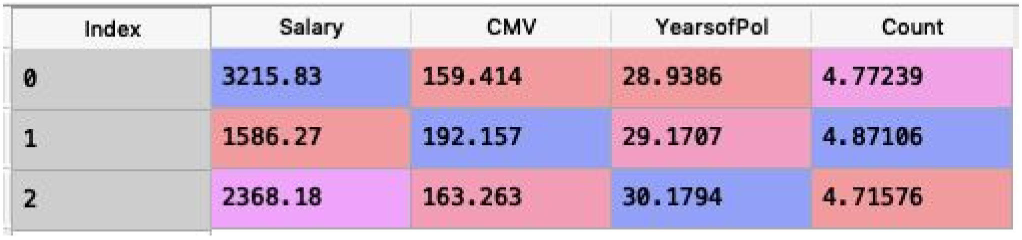


Index	0	1	2	3	mean
Salary	1586.27	3046.05	2368.18	3797.95	2496.94
CMV	192.157	164.405	163.263	142.3	213.764
YearsofPol	29.1707	29.1672	30.1794	28.155	29.9743
Count	4.87106	4.78934	4.71576	4.71429	4.70874

Result with 3 clusters:

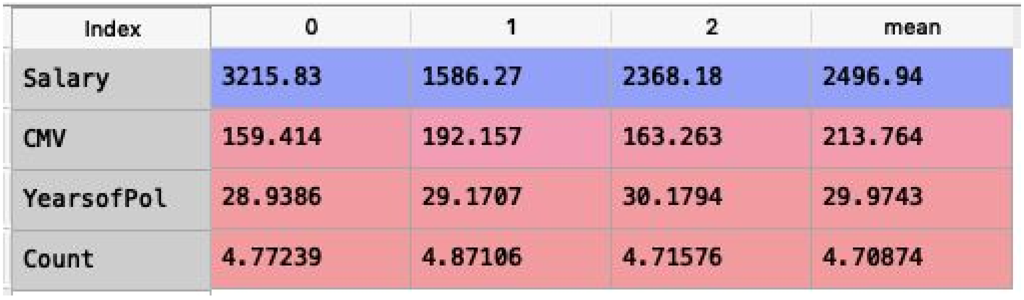
Clusters:

0= High Salary
1= Low Salary
2= Average Salary



Index	Salary	CMV	YearsofPol	Count
0	3215.83	159.414	28.9386	4.77239
1	1586.27	192.157	29.1707	4.87106
2	2368.18	163.263	30.1794	4.71576

Comparing with the average:

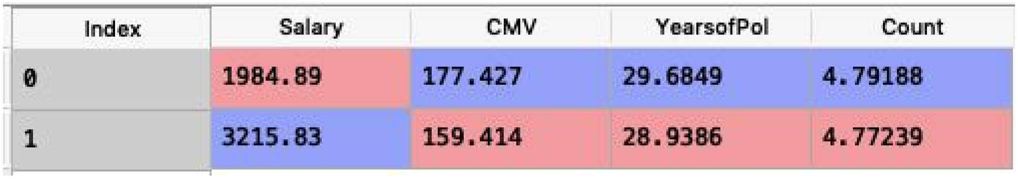


Index	0	1	2	mean
Salary	3215.83	1586.27	2368.18	2496.94
CMV	159.414	192.157	163.263	213.764
YearsofPol	28.9386	29.1707	30.1794	29.9743
Count	4.77239	4.87106	4.71576	4.70874

Result with 2 clusters:

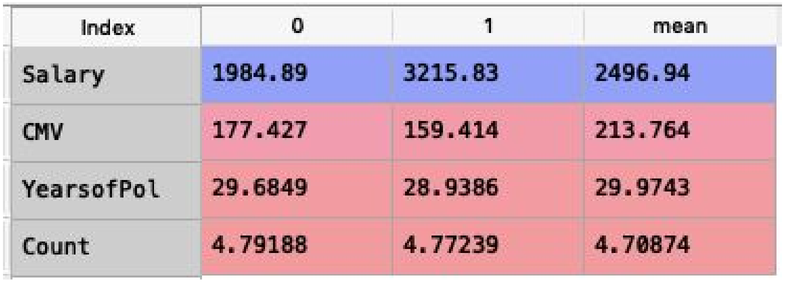
Clusters:

0= Low Salary, low CMV
1= High Salary, low CMV



Index	Salary	CMV	YearsofPol	Count
0	1984.89	177.427	29.6849	4.79188
1	3215.83	159.414	28.9386	4.77239

Comparing with the average:



Index	0	1	mean
Salary	1984.89	3215.83	2496.94
CMV	177.427	159.414	213.764
YearsofPol	29.6849	28.9386	29.9743
Count	4.79188	4.77239	4.70874

Evaluation:

For n_clusters = 2 The Distance Mean: 1013.9669763641602
For n_clusters = 3 The Distance Mean: 1034.7237022367667
For n_clusters = 4 The Distance Mean: 1111.3002148993135

DBSCAN

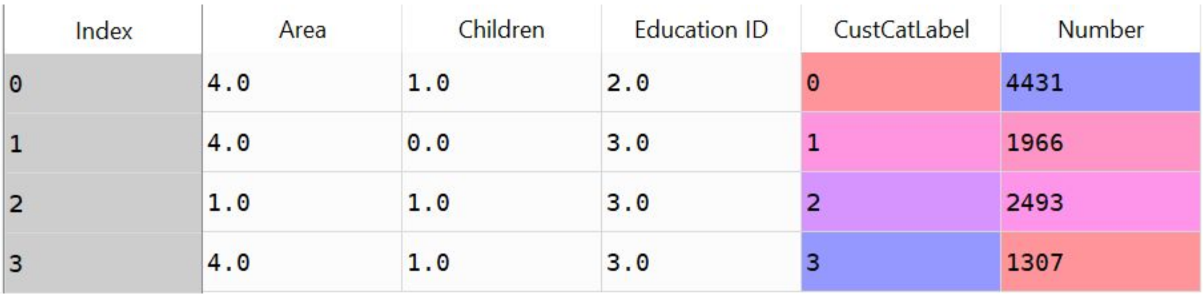
Customer categorical values

K-MODES

Clusters:

0 = Kids, High school, Area 4
1 = No kids, Master, Area 1
2 = Kids, Master, Area 1
3 = Kids, Master, Area 4

Result with 4 clusters:

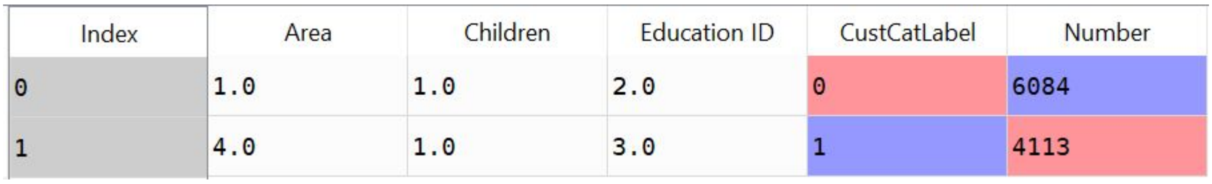


Index	Area	Children	Education ID	CustCatLabel	Number
0	4.0	1.0	2.0	0	4431
1	4.0	0.0	3.0	1	1966
2	1.0	1.0	3.0	2	2493
3	4.0	1.0	3.0	3	1307

Result with 2 clusters:

Clusters:

0 = High school, Area 1
1 = Master, Area 4

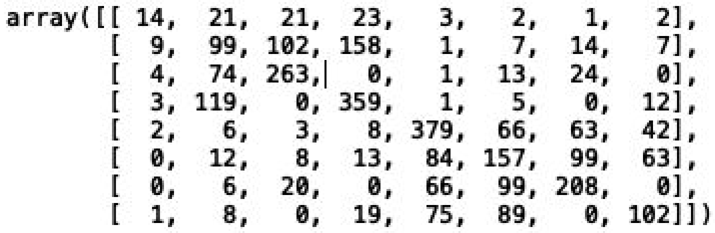


Index	Area	Children	Education ID	CustCatLabel	Number
0	1.0	1.0	2.0	0	6084
1	4.0	1.0	3.0	1	4113

Appendix III. Reintroducing the outliers

Confusion Matrix

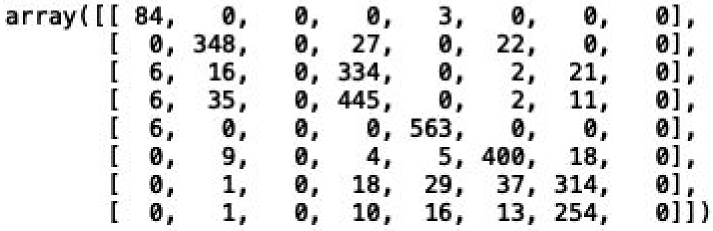
KNeighbors:



```
array([[ 14,  21,  21,  23,   3,   2,   1,   2],
       [  9,  99, 102, 158,   1,   7,  14,   7],
       [  4,  74, 263,   0,   1,  13,  24,   0],
       [  3, 119,   0, 359,   1,   5,   0,  12],
       [  2,   6,   3,   8, 379,  66,  63,  42],
       [  0,  12,   8,  13,  84, 157,  99,  63],
       [  0,   6,  20,   0,  66,  99, 208,   0],
       [  1,   8,   0,  19,  75,  89,   0, 102]])
```

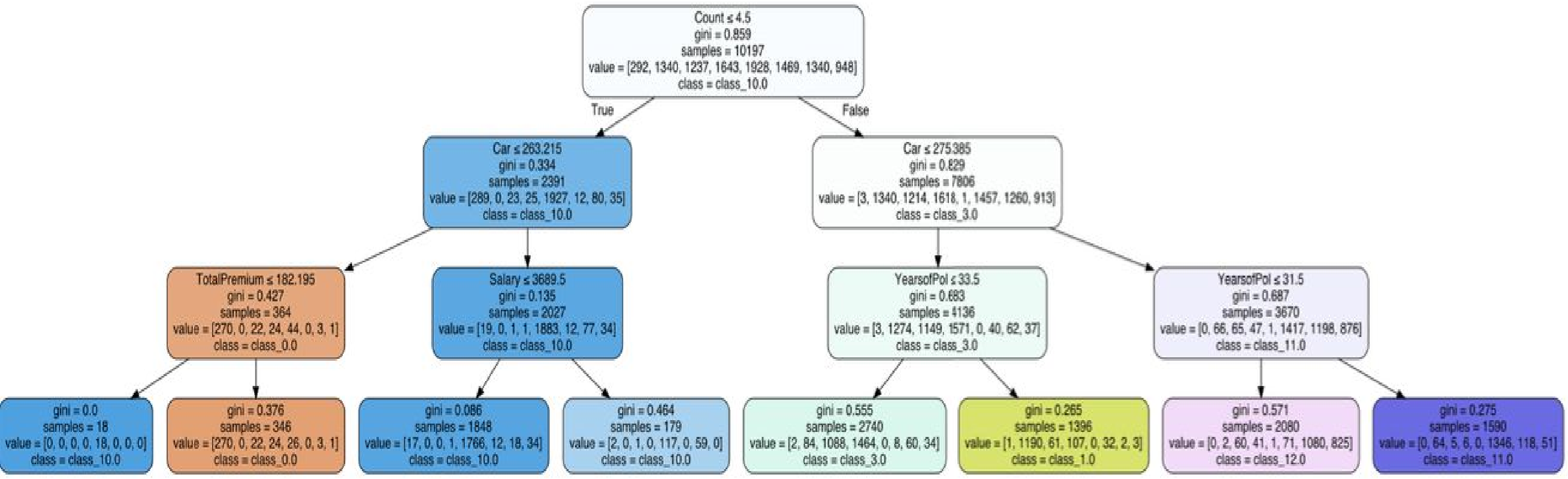
Accuracy:
(tp+tn)/(tp+tn+fp+fn) ⇔
⇔ 1581/3060= 51%

Decision Tree:



```
array([[ 84,   0,   0,   0,   3,   0,   0,   0],
       [  0, 348,   0,  27,   0,  22,   0,   0],
       [  6,  16,   0, 334,   0,   2,  21,   0],
       [  6,  35,   0, 445,   0,   2,  11,   0],
       [  6,   0,   0,   0, 563,   0,   0,   0],
       [  0,   9,   0,   4,   5, 400,  18,   0],
       [  0,   1,   0,  18,  29,  37, 314,   0],
       [  0,   1,   0,  10,  16,  13, 254,   0]])
```

Accuracy:
(tp+tn)/(tp+tn+fp+fn) ⇔
⇔ 2154/3060= 70%


Count ≤ 4.5
gini = 0.859
samples = 10197
value = [292, 1340, 1237, 1643, 1928, 1469, 1340, 948]
class = class_10.0
True
False
Car ≤ 263.215
gini = 0.334
samples = 2391
value = [289, 0, 23, 25, 1927, 12, 80, 35]
class = class_10.0
Car ≤ 275.385
gini = 0.829
samples = 7806
value = [3, 1340, 1214, 1618, 1, 1457, 1260, 913]
class = class_3.0
TotalPremium ≤ 182.195
gini = 0.427
samples = 364
value = [270, 0, 22, 24, 44, 0, 3, 1]
class = class_0.0
Salary ≤ 3689.5
gini = 0.135
samples = 2027
value = [19, 0, 1, 1, 1883, 12, 77, 34]
class = class_10.0
YearsofPol ≤ 33.5
gini = 0.683
samples = 4136
value = [3, 1274, 1149, 1571, 0, 40, 62, 37]
class = class_3.0
YearsofPol ≤ 31.5
gini = 0.687
samples = 3670
value = [0, 66, 65, 47, 1, 1417, 1198, 876]
class = class_11.0
gini = 0.0
samples = 18
value = [0, 0, 0, 0, 18, 0, 0, 0]
class = class_10.0
gini = 0.376
samples = 346
value = [270, 0, 22, 24, 26, 0, 3, 1]
class = class_0.0
gini = 0.086
samples = 1848
value = [17, 0, 0, 1, 1766, 12, 18, 34]
class = class_10.0
gini = 0.464
samples = 179
value = [2, 0, 1, 0, 117, 0, 59, 0]
class = class_10.0
gini = 0.555
samples = 2740
value = [2, 84, 1088, 1464, 0, 8, 60, 34]
class = class_3.0
gini = 0.265
samples = 1396
value = [1, 1190, 61, 107, 0, 32, 2, 3]
class = class_1.0
gini = 0.571
samples = 2080
value = [0, 2, 60, 41, 1, 71, 1080, 825]
class = class_12.0
gini = 0.275
samples = 1590
value = [0, 64, 5, 6, 0, 1346, 118, 51]
class = class_11.0