Data Mining Project

MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS

CUSTOMER SEGMENTATION: INSURANCE COMPANY

Group AS

Ana Miguel Monteiro, 20221645

Ana Rita Viseu, 20220703

Sara Galguinho, 20220682

January 2023

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

Abstract

This project aims to develop a strategy of customer segmentation for a Portuguese insurance company. By finding the relevant customer segments related to this business, we were able to gain insights on the value, demographics and spending habits of each customer group, which in turn helped us create targeted marketing approaches as well as informed business applications. In order to do this, we were granted access to a sample of customers from the company's active database.

We started by performing data exploration, using data visualization tools and descriptive statistics, and then preprocessed the data by handling incoherences, inconsistencies, missing values and outliers. Afterwards, we performed feature engineering by creating new variables and altering existing ones, and posterior dimensionality reduction, by considering the relevancy and the redundancy of all variables. Then we advanced to the clustering phase, where we divided our remaining features into a demographic and a product perspective and proceeded to apply the clustering methods we found to be most appropriate for the problem at hand. After analysing and comparing the different clustering techniques, we manually merged our perspectives, one that was built using the K-Prototypes algorithm and another one using K-Means on top of a Self Organizing Map. As a result, we reached a final solution of five distinct customer segments. Each one of these segments was then analysed and marketing approaches were developed considering their charactheristics. As a final step, we assessed the feature importance of our clustering solution and reclassified the previously removed outliers.

Keywords: Data Mining, Data Exploration, Data Preprocessing, Data Preparation, Feature Engineering, Dimensionality Reduction, Clustering, Insurance

Index

1. Introduction 1
2. Data Exploration 2
3. Data Preprocessing 3
   - 3.1 Coherence Checking 3
   - 3.2 Missing Values 3
   - 3.3 Outliers 4
   - 3.4 Feature Engineering 5
     - 3.4.1. Creating New Variables 5
     - 3.4.2. One-Hot Encoding 5
   - 3.5 Dimensionality Reduction 5
     - 3.5.1. Redundancy 5
     - 3.5.2. Relevancy 6
4. Redoing Data Exploration 6
5. Feature Scaling 6
6. Clustering 7
   - 6.1 Demographic Perspective (K-Prototypes) 7
   - 6.2 Product Perspective 7
     - 6.2.1 Cluster Comparison 8
7. Customer Segmentation and Marketing Approaches 8
8. Feature Importance and Reclassification of Outliers 10
9. Conclusion 10
10. References 10
11. Appendix 12

1. Introduction

A2Z Insurance is a long-standing company that became one of the largest insurers in Portugal in 2016. Due to a lack of data driven culture that led to poorly maintained databases, A2Z has simply mass-marketed everything over the years and is now making efforts to start differentiating customers and develop more focused programs.

Clustering is an unsupervised machine learning technique used in many fields, enabling the gain of valuable insights from data by discovering its patterns. These algorithms aim to group datapoints, so that points in the same group have features as similar as possible and points belonging to different clusters are as dissimilar as possible.

In the business problem at hand, our clustering analysis aims to divide a customer base into groups of individuals that have similarities relevant to marketing, such as demographic, psychographic, geographic and behavioral characteristics. After all, in today's era, companies need to ensure they do not fall into the one size fits all trap. With a better understanding of their current customers and the recognition of different groups, the company will be able to not only increase customer loyalty, but also attract new customers, by being able to make strategic choices about product definition, positioning, pricing, promotions and target marketing.

We have been provided with data regarding different customers from A2Z Insurance's database, with the current year being 2016. This data contains:

- Sociodemographic attributes — year of birth, academic degree, gross monthly salary, living area and whether the customer has children.
- Company related attributes — year of the first policy (considered as the first year as a customer), lifetime monetary value, claims rate and annual premiums (Motor, Household, Health, Life and Work Compensation).

The aim of this project is to segment the provided database and give A2Z insights on each customer segment, as well as business and marketing suggestions considering this segmentation.

By applying the appropriate clustering methods, using different perspectives, as well as combining and analyzing the results, we were able to find the relevant clusters of customers and develop our own strategies for each cluster.

2. Data Exploration

Our first step was to perform an initial exploration of the available information and try to identify data patterns, structural problems and take early conclusions. We used descriptive statistics and data visualization tools, such as pandas profiling, to generate an overall report of the data and discover inconsistencies and potential insights.

After importing the needed libraries and the dataset, we started by setting the index to the customer's ID (*CustID*), after making sure that all the values were unique (meaning no two customers had the same ID).

By analyzing our dataset using the tools mentioned above, we gained insights such as:

- A total of 10296 observations and 13 variables;
- 3 duplicated observations;
- Datatypes that need to be corrected (more specifically in *FirstPolYear*, *BirthYear*, *GeoLivArea* and *Children*);
- A total of 389 missing values (not accounting for possible unidentified missing values). All variables contain missing data except for *CustMonVal*, *ClaimsRate* and *PremHousehold*, so we expect to test various imputation methods;
- One instance where *FirstPolYear* has a value higher than 2016 (53784);
- Instances where *BirthYear* is higher than *FirstPolYear*, meaning there are customers who have their first policy before even being born;
- Some very low values and very high values in the variable *MonthSal* that could potentially be outliers;
- One instance with an extremely low value in the variable *CustMonVal* (-165680.42);
- Some extremely high values in *ClaimsRate*, considering the minimum of this variable is 0 and the mean is approximately 0.74, but the maximum is 256.2;
- Potential outliers in *PremMotor*, *PremHousehold*, *PremHealth* and *PremWork* given away by the inspection of the minimum, maximum, mean and quartiles of these variables;
- High Spearman Correlation between *BirthYear* and *MonthSal*;
- High Spearman Correlation between *CustMonVal* and *ClaimsRate*;
- *PremMotor* is negatively correlated to *PremHousehold*, *PremHealth*, *PremLife* and *PremWork*;
- The variable *GeoLivArea* can take on the values 1.0, 2.0, 3.0 or 4.0, however there is no further information that allows us to understand the meaning of these values.
- A somewhat low percentage of customers live in area 2.0 (10.1%), while the most common area of residency is area 4.0 (40.3%);
- Most of the customers have children (70.6%);
- The most frequent education level is Bachelor/ Masters (46.6%) and the least frequent is PhD (6.8%).

All of these potential problems will be solved in the preprocessing stage. The variable information of the numerical variables can be well systemized in **Table 1** (Appendix), where we can confirm almost all variables seem to contain extreme outliers by analysing their respective central tendency measures. As for the categorical variables, we visualized their absolute frequencies (Appendix, **Figure 1**) in order to get the general idea of how many observations there are in each category.

3. Data Preprocessing

Regarding the preprocessing of our data, we corrected some of the incoherences detected above, imputed missing values and removed outliers. Then we performed feature engineering and dimensionality reduction. Throughout this stage, we made sure to create numerous copies of the different versions of the dataset in case we decided to change something in our preprocessing strategy later. We also made sure to keep track of how much data we were removing in each step, so we could account for it in the end.

Before checking and correcting incoherences, we changed the datatypes of the variables *FirstPolYear*, *BirthYear*, *GeoLivArea* and *Children* to more appropriate ones and dropped the 3 duplicates identified previously, leaving us with 10293 observations (meaning we removed 0.0003% of the original data).

3.1 Coherence Checking

As previously noticed, there are customers that have a *FirstPolYear* value lower than their *BirthYear* value, meaning they became clients before being born. Upon further inspection there are 1997 observations with this issue. Considering the customer's first policy year is data from the company and the year of birth is much more likely to be wrongly given by the customer or simply wrongly inserted in the database, we considered three possible solutions: either we replace these incorrect values in the column *BirthYear* with NaNs and then input artificial data, we assume that *BirthYear* is equal to *FirstPolYear*, or we delete the column *BirthYear*, discarding this variable completely. We decided to apply the latter because 1997 is too many instances to artificially change.

Since the current year of the database is 2016, it wouldn't make sense to have values that are higher than that in the variable *FirstPolYear*. There was only one record where this happened, so we dropped it, removing 0.0001% of data.

Once these problems were dealt with, we defined the following features as non-metric: *GeoLivArea*, *Children* and *EducDeg*. All other features were defined as metric (Appendix, **Table 2**).

3.2 Missing Values

As there could be missing values not identified as a NumPy NaN, we started by correcting this possible issue. After checking how many missing values each feature had and realizing that all of them contained a very small percentage of NaNs (with the highest percentage being only 1.01% in *PremLife*), it wouldn't make sense to delete any of them.

The missing values in *PremMotor*, *PremHealth*, *PremLife*, and *PremWork* (34, 43, 104 and 86 respectively) are most likely customers that don't have that type of insurance, given that this is information controlled by the company and is probably MNAR (missing not at random). Therefore, we decided to fill those NaNs with 0.

In the remaining features, the variables that had missing values were *FirstPolYear*, *EducDeg*, *MonthSal*, *GeoLivArea* and *Children* (30, 17, 36, 1 and 21 respectively). We decided to try different imputation methods such as Median and Mode, K-Nearest Neighbor and Iterative Imputer.

I. Median and Mode: We filled the NaNs in the metric features with their respective medians, whereas for the non-metric features we filled them with their modes. Considering *GeoLivArea* only

had one missing value, this method would've been enough for this variable. However, for all other features, we considered a more complex method would be more fitting.

II. K-Nearest Neighbor (KNN) with weights='distance': Since it wouldn't make sense to apply KNN to the non-metric features, we filled those NaNs with their modes. For the metric features, we used the MinMax Scaler to normalize the data, since KNN is distance-based, and then applied this method using 5 neighbors and the Euclidean distance as parameters. We also defined 'weights=distance' so that the closer datapoints have more weight in the prediction of the missing values, which helps reduce the influence of outliers and focuses the prediction on the more "representative" datapoints. After imputation, we reverted the scale to get the unscaled original values back.

III. Iterative Imputer: To apply this method, we first encoded the feature *EducDeg*. By using ordinal encoder, we couldn't define the specific order of each category, so we decided to do it manually. After that, we applied the Iterative Imputer using the Decision Tree Classifier for the non-metric features and the Decision Tree Regressor for the metric ones.

We decided to use the Iterative Imputer as our final choice because it is a multivariate imputing strategy that estimates the missing values based on other features, making it a more sophisticated approach.

3.3 Outliers

Detecting and removing outliers prior to performing clustering is a very important step, given that a lot of clustering algorithms are sensitive to them.

By analyzing the absolute frequencies of the non-metric features after data imputation (Appendix, **Figure 2**), we concluded that there were no outliers to treat in these variables, since all categories have a good number of observations.

As for the metric features, we had already suspected the existence of extreme outliers, which is confirmed by analyzing their respective histograms and box plots (Appendix, **Figure 3** and **Figure 4**). To ensure the data is treated in the most beneficial manner, we tried six different ways to detect outliers and then chose the most effective one and with as little loss of relevant information as possible.

I. Manual Filtering: We manually defined filters based on data visualizations and kept 98.17% of data.

II. IQR (Interquartile Range) Method: In this method, we defined an upper and lower bound using the interquartile range (IQR) multiplied by 3 (we chose this value based on visualizations and on the percentage of data kept). We then excluded all observations outside of that boundary, keeping 95.68% of the data.

III. Manual & IQR: Here we combined the two previous methods and kept 98.25% of the data.

IV. Z-Score: Using this method, we standardized the data and removed all observations that are 4 standard deviations apart from the mean (0). The value 4 was chosen for a similar reason to the value in the IQR method. We kept 98.33% of the data.

V. DBSCAN: With DBSCAN, we excluded all points that aren't considered core or border. Firstly, we determined the optimum epsilon to use as a parameter (0.08) using the Elbow Method and considering visualizations of the data and the percentage of data removed. We kept 98.85% of the data.

VI. Isolation Forest: For this method, we defined the contamination as 0.03, as we intended to remove 3% of the data. This value was chosen, again, considering data visualizations and the fact that we wanted to transform our dataset as little as possible. Naturally, we kept 97% of the data.

After considering how each outlier removal method impacted the metric variables' distributions and how much data was removed, we decided to proceed with the Z-Score method. Since we still appeared to have a few remaining outliers, we decided to manually remove those and then visualized the variables' distributions to confirm our removal was successful (Appendix, **Figure 5** and **Figure 6**). As a result, we ended up removing 98.28% of data. The removed outliers were stored in a new dataset, so we could reallocate them once clustering was performed.

Lastly, we checked the pairwise relationship of the metric variables (Appendix, **Figure 7**) to detect possible bivariate outliers, however the data did not contain any visible multi-variate outliers that needed treatment.

3.4 Feature Engineering

3.4.1. Creating New Variables

To conduct a more accurate cluster analysis, new variables were created, based on transformations of the features already present in the dataset (Appendix,

Table 3). With these new variables we now have possibly more relevant information, which is fundamental to improve the results of our clustering methods. These new variables were introduced both in the dataset with and without outliers and the metric features were updated.

Then we checked the descriptive statistics to analyse the central tendency measures of our new features (Appendix, **Table 4**).

3.4.2. One-Hot Encoding

We created a second dataset where we applied *OneHotEncoding* to the categorical features *GeoLivArea* and *EducDeg* to convert each value into a binary variable. It may be more useful to deal with these types of variables when interpreting clusters. We ended up with these new variables: *GeoLivArea_2*, *GeoLivArea_3*, *GeoLivArea_4*, *HighSchool*, *Bsc/MSc*, *PhD*.

3.5 Dimensionality Reduction

After creating new features, we are now faced with too many variables, so it's essential to reduce the dimensionality of the dataset, as it could negatively impact the performance of the clustering algorithms. In order to do this, we considered both the redundancy and the relevance of the new and the original variables.

3.5.1. Redundancy

In order to remove redundant features, we computed the Spearman Correlation between the metric features (Appendix, **Figure 8**). As expected, some of the new variables have very high correlations with

the original ones that were used to create them. After analyzing the correlation matrix, we decided to drop the following features: *FirstPolYear*, *MonthSal*, *CustMonVal*, *ClaimsRate* and *PremTotal*.

Although Principal Component Analysis (PCA) is a great way to reduce the input space by creating linearly uncorrelated variables, we decided not to use it, since its main disadvantage is low interpretability, which is crucial to identifying the clusters formed.

3.5.2. Relevancy

To determine which variables were relevant and which ones weren't, we performed an exploratory analysis. To get an overview of the relationship between our features and see the effect they have on each other, we analysed the pairwise relationship (Appendix, **Figure 9**) of all variables. Both *GeoLivArea* and *Children* seemed to display some lack of relevancy, so we created box plots and pairwise relationships for each one of these variables to confirm our suspicions.

Through the analysis of the graphs, we concluded that *GeoLivArea* has no relevance to other features (Appendix. **Figure 10** and **Figure 11**). On the contrary, *Children* is definitely relevant (Appendix, **Figure 12** and **Figure 13**). Thus, we decided to drop *GeoLivArea* for the clustering phase.

4. Redoing Data Exploration

Before proceeding to the clustering stage, we redid some data exploration, since new problems could have arisen with the creation of the new variables. In fact, we confirmed the presence of some remaining outliers in *CustMonVal_Year* and *PremHousehold* (Appendix, **Figure 14** and **Figure 15**). We manually removed these outliers, deleting 0.81% of the data. The newly detected outliers were then added to the dataset with the first outliers and data visualizations were checked to make sure the removal was successful (Appendix, **Figure 16** and **Figure 17**). Overall, 2.57% of the original data was removed.

The feature *PremMotor* does have a correlation of -0.7 with the rest of the premium related variables (Appendix, **Figure 18**), but we considered that to be moderately high and not particularly problematic. The variable *CustomerYears* was removed since it seems to have little impact in the remaining variables, and we already have the feature *CustMonVal_Year*.

5. Feature Scaling

Once our final variables were selected, the final step was to scale our data. It's crucial to normalize our variables and make them comparable, since all the clustering methods that are going to be used deal with distances. This way we will avoid variables with larger scales having more weight and importance on the cluster solutions. We tried three different methods of data scaling, namely the MinMax Scaler, the Standard Scaler and the Robust Scaler. The latter was deemed unnecessary for our situation, since all extreme outliers had already been removed. As for the remaining two, we tested our clustering algorithms with both and ended up choosing the Standard Scaler. The descriptive statistics can be seen in **Table 5** (Appendix).

6. Clustering

In order to perform a more thorough clustering analysis, two perspectives were developed. A demographic one, that evaluates aspects related to family, education and salary, and a product perspective, that assesses aspects related to the company.

6.1 Demographic Perspective (K-Prototypes)

For this perspective, we used five features: *HighSchool*, *BSc/MSc*, *PhD*, *Children* and *AnnualSal*. Since we were dealing with both numerical and categorical data, we used the K-Prototypes algorithm, which handles clustering with both data types by combining K-Means and K-Modes.

In order to select the optimal number of clusters to use, we computed each clustering cost from 1 to 10 clusters. This cost is defined as the sum distance of all points to their respective cluster centroid and combines the calculation for numerical and categorical variables. By examining the Elbow plot (Appendix, **Figure 19**), 3 clusters seems to be the most appropriate number of groups to form.

After running the K-Prototypes algorithm and analyzing the results (Appendix, **Table 6**), it is possible to conclude that the first cluster (Cluster 0) is generally composed by customers that have children, a medium-low level of education and a low salary. The second cluster (Cluster 1) is composed by customers that also have children, a high level of education and a medium salary. Lastly, Cluster 2 is generally composed by customers that don't have children, have a medium level of education and a high salary. All clusters have few customers with a PhD, which is only natural, considering it was the least frequent level of education.

In this clustering solution, we have 2947 customers belonging to Cluster 0, 4061 in Cluster 1, and 3025 in Cluster 2. A simple profiling of these clusters can be seen in **Figure 20** (Appendix).

6.2 Product Perspective

In the product perspective, we included the following features: *CustMonVal_Year*, *PremMotor*, *PremHousehold*, *PremHealth*, *PremLife* and *PremWork*. Fortunately, all variables in this perspective are metric, which gives us the possibility of testing various clustering methods.

The first two methods we used were Hierarchical Clustering on K-Means and Hierarchical Clustering on K-Medoids (K-Medoids is a variant of K-Means where the centroids of the clusters are actual datapoints close to the center of the cluster instead of being the average of the positions of the datapoints). For the first approach, we started by applying the K-Means algorithm with a large number of clusters (35) and determining their respective centroids. Then, we computed the R^2 (using the Euclidean distance) for Hierarchical Clustering with various linkage methods and numbers of clusters (Appendix, **Figure 21**) and concluded that the Ward's linkage method is the best, since it was the one with the highest R^2 in almost every situation. By observing the dendrogram (Appendix, **Figure 22**), we decided that the optimal number of clusters is 4 and ran Hierarchical Clustering on the initial 35 clusters. Using K-Medoids, the approach was the same, except the final number of clusters was 2.

Besides that, we applied two more clustering algorithms: K-Means on top of SOM (Self Organizing Maps) and Hierarchical Clustering on top of SOM. For the first one, we trained our model with a map size of [20, 20] and a Gaussian neighborhood. Using the Elbow method (Appendix, **Figure 23**) we concluded the best number of clusters was 3 and then performed K-Means on top of the 400 units of the SOM. The process was very similar when performing Hierarchical Clustering on top of SOM and also resulted in 3 final clusters, as indicated by the dendrogram (Appendix, **Figure 24**).

6.2.1 Cluster Comparison

In order to choose our final clustering solution for the product perspective, we considered how many clusters each method contained, their respective sizes, and how distinguishable they seemed to be. When comparing the cluster profiling of each method (Appendix, **Figure 25**), we decided the most appropriate number of clusters for this perspective was 3, which left us with one of the Self Organizing Maps as an option. Given that the profiling results showed similar outputs, our final decision relied on having the better-balanced data between clusters. Therefore, we considered K-Means on top of Self Organizing Maps to be the best solution.

By analyzing the centroids of each cluster (Appendix, **Table 7**), we can deduce that Cluster 0 is generally composed by customers that spend the most on premiums in LOB Health and have the lowest monetary value for the company, meaning they could represent older or sick people. The customers in Cluster 1 generally spend the most on premiums in LOB Household, LOB Life and LOB Work Compensation, while having the best monetary value for the company. This cluster could possibly represent families on the wealthier side. Finally, Cluster 2 is generally composed by customers that spend the most on premiums in LOB Motor and the least in all other types of premiums. They have a medium monetary value for the company.

In this clustering solution, we have 3694 customers belonging to Cluster 0, 2458 in Cluster 1, and 3881 in Cluster 2.

7. Customer Segmentation and Marketing Approaches

After choosing our final clustering method, we proceeded with the merging of the two perspectives, which resulted in a semi-final clustering solution of 9 clusters (Appendix,). Through the analysis of the contingency table (Appendix, **Table 8**), it was clear that some clusters had few individuals, which indicates the existence of clusters that share some similar characteristics. Having that said, we decided to merge the clusters with fewer observations with the bigger ones, using two different methods: manual merging and hierarchical merging. Both presented similar results, having clusters with approximately the same size and similar values for the centroids. We chose to continue with the manual merging, applying it twice. To know which of the biggest clusters to merge with the smallest (chosen previously), this method uses the proximity of each cluster to the others and merge the ones that are closer. Since we're using distances, we performed the merging using only the metric features, because categorical features might cause a biased result. As a final solution, we have 5 different clusters. Their respective centroids and profiling can be seen on **Table 9** and **Figure 27** (Appendix). As a form of visualization, we used the T-SNE method, using only the metric features (Appendix, **Figure 28**).

Finally, we conducted a careful analysis of each customer segment and developed possible marketing approaches considering their respective characteristics.

Cluster 2: This cluster has 1328 individuals and is characterized by customers that have children, a medium-low education, and a low salary. They spend mainly on Household, Health, Life and Work premiums. It is interesting that despite having the lowest Annual Salary, these customers, along with the ones on Cluster 8, are part of the best clients of the insurance company. This might be related to the fact that this group is largely composed by customers with dependents. Having children to take care of seems to make these customers want to invest more on insurances. A possible marketing strategy for this segment would be selling combinations of two or three of these four premiums, instead of offering them separately, and name them, for example, as "Family Packages", especially considering the clients' low wages. This strategy could also attract more customers sharing the same resemblances.

Cluster 4: This cluster has 2199 observations and is mainly composed by customers that do not represent a significant weight in the insurer's profit. These clients are the ones with the second lowest average annual salary, despite having a high education. They are also characterized by having dependents, which might explain the fact that their highest expense is on Health premiums. With this type of customers, it would be a good idea to implement advertising campaigns, for example by offering a 3-6 month discount on the acquisition of a new premium. This would work as an incentive for these customers to spend more on other premiums.

Cluster 6: The customers in this cluster (3881 individuals) stand out because despite having a high education (the highest when comparing with the ones in the other clusters), they do not earn a very high salary. Although they have children, these clients do not invest a lot in Health and Life premiums, as the ones in Cluster 2. Their mainly acquisition is actually the Motor premium and they spend the least on every other type of premium. Investing on a marketing campaign promoting premiums such as Health, Life and Household could be interesting since most of the clients in this cluster have children and it would probably be of their interest to acquire premiums such as these.

Cluster 7: This is an interesting group (with 1495 individuals) because it is composed by the customers with the highest annual salary that less contribute to the company's profit. Most of the clients don't have children and have a high education. Their major investment is on Health premiums, being indeed the best customers of this particular premium. Having a high salary and no children, offering these customers a discount in other premiums might not be the best approach since they probably don't feel the need to acquire them. A good idea would be increasing the insurance fee so that the company's profit with these clients could also improve.

Cluster 8: This cluster contains 1130 clients that share some resemblances with the ones in Cluster 2. Their mainly expense on the company premiums is also with the Household, Health, Life and Work premiums and they're the ones that most contribute to the company's profit. Despite these similarities, they differ because these customers generally don't have children, have a low education, but a high salary. An idea for this type of customer, in contrast to the one suggested in Cluster 2, would be applying a similar approach to the one in Cluster 7, since these clients have a high salary and already invest in a good amount of premiums.

8. Feature Importance and Reclassification of Outliers

With the aim of assessing which features are more important in our clustering solution, we decomposed the general R^2 into the R^2 for each variable. We also computed a Decision Tree Classifier. In both cases, Motor Premiums and the annual salary were the two features with the highest importance, followed by Premiums in Health and in Life. The level of education and the monetary value are the least important features in our clustering analysis.

Now that our clients are labeled and our customer segments are constructed and studied, it's important to classify the observations previously considered as outliers, since they are still customers, and it wouldn't make sense to leave them unclassified. In order to do this, we divided our dataset into a train set (to fit the model) and a test set (to evaluate the algorithm's performance) and constructed two predictive models: Decision Tree Classifier and K-Neighbors Classifier. For the Decision Tree Classifier, we set a hyperparameter *max_depth* of 3, to prevent overfitting. This model estimated that, on average, we could predict 81.33% of our customers correctly. For the K-Neighbors Classifier, we used 10 neighbors and were able to predict, on average, 90.8% of our customers correctly.

Since this model gave the highest percentage of correct predictions, we used it to reclassify the outliers. As a result, 254 observations were assigned to Cluster 7, 3 observations to Cluster 8 and 2 observations to Cluster 6.

9. Conclusion

The aim of this project was to segment the customers into groups so that they could clearly be told apart from one another. The segmentation was performed having as purpose applying the best marketing strategy possible.

Considering we started with a very noisy dataset, we needed to do a careful exploration and preprocessing of our data. We started by treating some incoherences, proceeded with the imputation of the missing values through the iterative imputer and then performed a first outlier removal. As a way of having features as relevant as possible, we created new variables and reduced the input space, discarding highly correlated variables and irrelevant ones. After this step we did a second outlier removal, having in mind that new outliers could arise. Finally, we decided to transform our categorical features into dummys and scaled the data using the Standard method.

In the clustering phase we divided our data into two perspectives: "Demographic", were we included some personal aspects of our customers, and "Product", where we have all the meaningful features for the company business. For the "Demographic" perspective we used K-Prototypes and for the "Product" perspective we decided to go with the SOM + K-Means algorithm. In the end we were confronted with 9 distinct clusters that we manually merged, ending up with the final 5 clusters that we thoroughly analysed.

With this final result it was possible to not only identified our best clients, but also ways of attracting new ones.

10. References

BrownLee, J. (2020). Iterative Imputation for Missing Values in Machine Learning. Retrieved December 10, 2022, from https://machinelearningmastery.com/iterative-imputation-for-missing-values-in-machine-learning/

Mindrila, D., & Balentyne, P. ScatterPlots and Correlation. Retrieved December 17, 2022, from https://www.westga.edu/academics/research/vrc/assets/docs/scatterplots_and_correlation_notes.pdf

Aprilliant, A. (2021). The k-prototype as Clustering Algorithm for Mixed Data Type (Categorical and Numerical). Retrieved December 18, 2022, from https://towardsdatascience.com/the-k-prototype-as-clustering-algorithm-for-mixed-data-type-categorical-and-numerical-fe7c50538ebb

Soliya, S. (2021). Customer Segmentation using k-prototypes algorithm in Python. Retrieved December 18, 2022, from https://medium.com/analytics-vidhya/customer-segmentation-using-k-prototypes-algorithm-in-python-aad4acbaaede

Gogia, N. (2019). Why Scaling is Important in Machine Learning. Retrieved December 20, 2022, from https://medium.com/analytics-vidhya/why-scaling-is-important-in-machine-learning-aee5781d161a

Kumar, S. (2020). Understanding K-Means, K-Means++ and, K-Medoids Clustering Algorithms. Retrieved December 20, 2022, from https://towardsdatascience.com/understanding-k-means-k-means-and-k-medoids-clustering-algorithms-ad9c9fbf47ca

Shiledarbaxi, N. (2021). Comprehensive Guide To K-Medoids Clustering Algorithm. Retrieved December 20, 2022, from https://analyticsindiamag.com/comprehensive-guide-to-k-medoids-clustering-algorithm/

11. Appendix

	count	mean	std	min	25%	50%	75%	max
FirstPolYear	10266.0	1991.062634	511.267913	1974.00	1980.00	1986.00	1992.0000	53784.00
BirthYear	10279.0	1968.007783	19.709476	1028.00	1953.00	1968.00	1983.0000	2001.00
MonthSal	10260.0	2506.667057	1157.449634	333.00	1706.00	2501.50	3290.2500	55215.00
GeoLivArea	10295.0	2.709859	1.266291	1.00	1.00	3.00	4.0000	4.00
Children	10275.0	0.706764	0.455268	0.00	0.00	1.00	1.0000	1.00
CustMonVal	10296.0	177.892605	1945.811505	-165680.42	-9.44	186.87	399.7775	11875.89
ClaimsRate	10296.0	0.742772	2.916964	0.00	0.39	0.72	0.9800	256.20
PremMotor	10262.0	300.470252	211.914997	-4.11	190.59	298.61	408.3000	11604.42
PremHousehold	10296.0	210.431192	352.595984	-75.00	49.45	132.80	290.0500	25048.80
PremHealth	10253.0	171.580833	296.405976	-2.11	111.80	162.81	219.8200	28272.00
PremLife	10192.0	41.855782	47.480632	-7.00	9.89	25.56	57.7900	398.30
PremWork	10210.0	41.277514	51.513572	-12.00	10.67	25.67	56.7900	1988.70

Table 1 – Numerical Variables' Descriptive Statistics

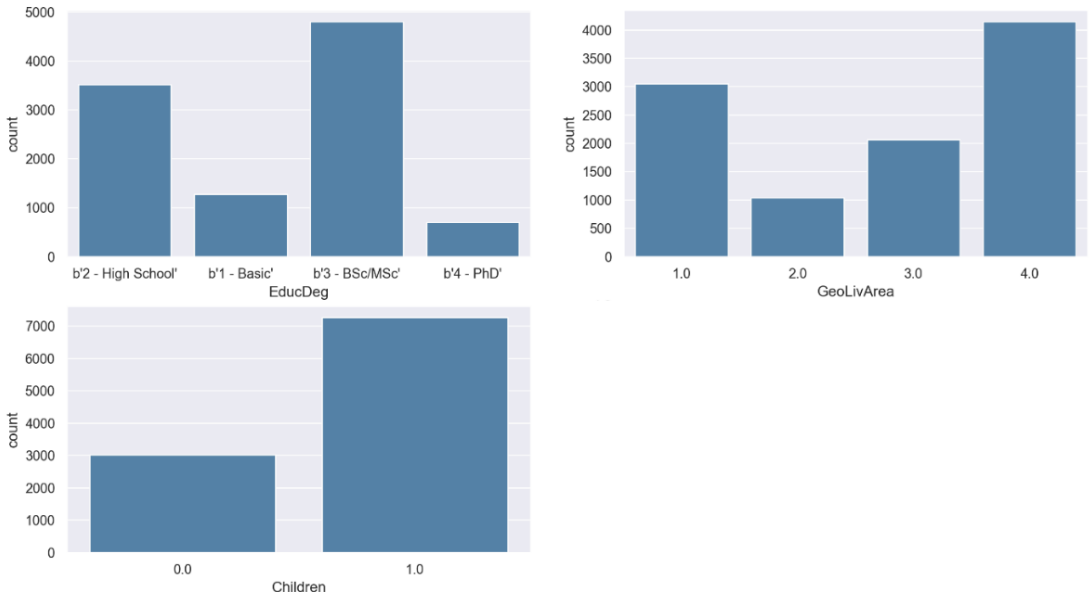

Figure 1 – Categorical Variables' Absolute Frequencies

Metric Features	Non-Metric Features
GeoLivArea	FirstPolYear ; MonthSal
Children	CustMonVal; ClaimsRate
EducDeg	PremMotor; PremHousehold
	PremHealth; PremLife; PremWork

Table 2 – Metric and Non-Metric Features

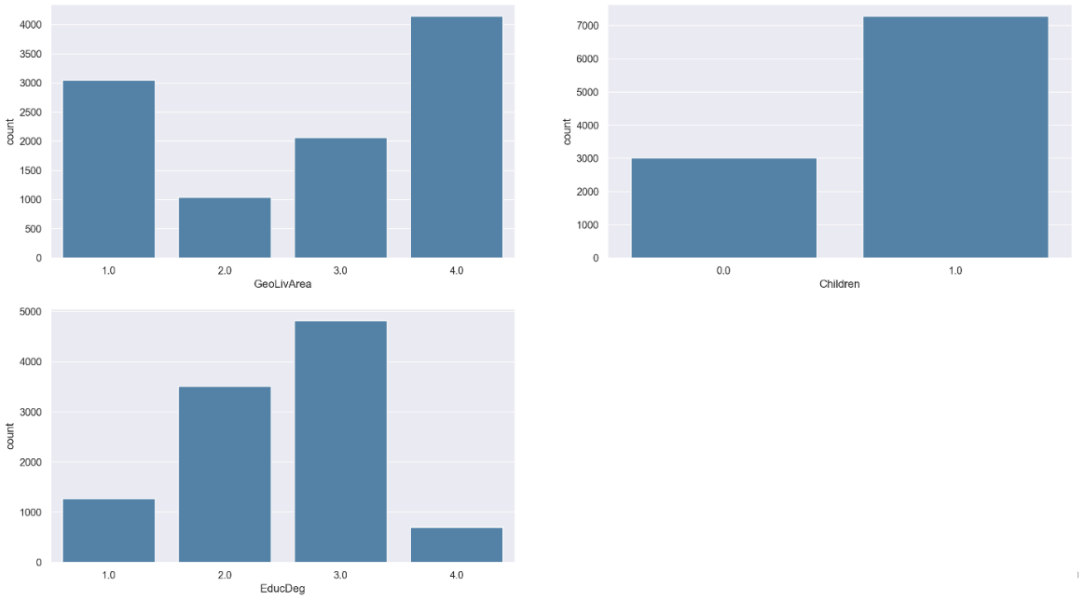

Figure 2 – Categorical Variables' Absolute Frequencies After Missing Values Imputation

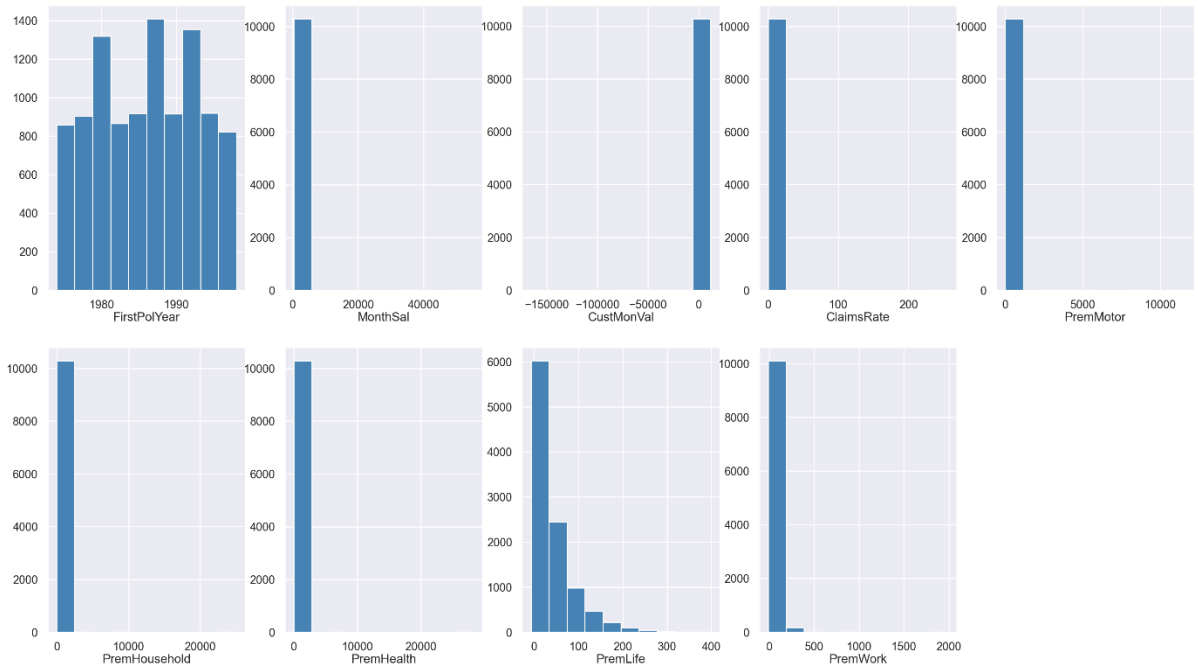

Figure 3 – Metric Variables' Histograms Before Outlier Removal

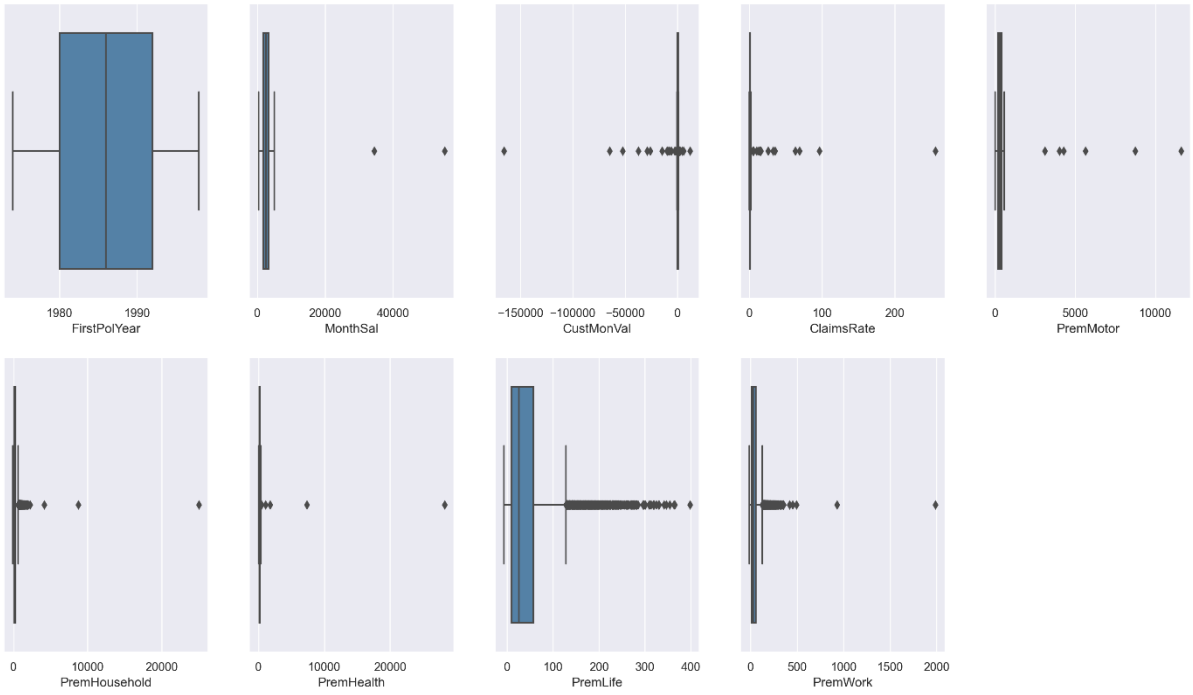

Figure 4 – Metric Variables' Box Plots Before Outlier Removal

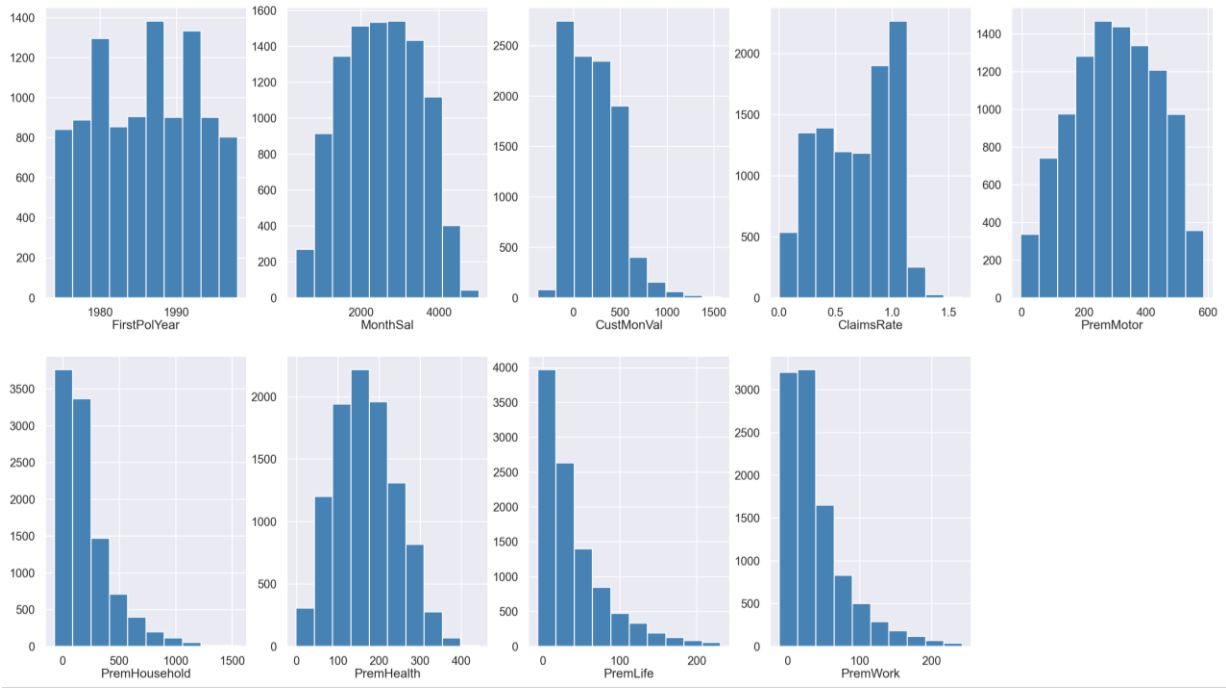

Figure 5 – Metric Variables' Histograms After Outlier Removal (Z-score & Manual)

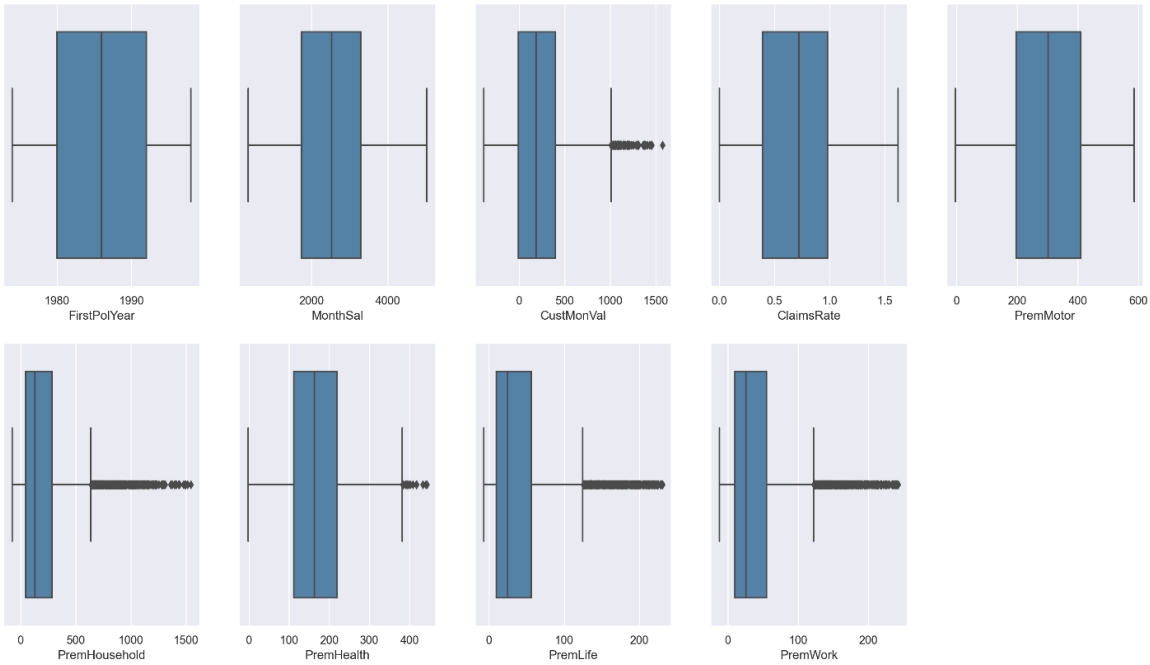

Figure 6 – Metric Variables' Box Plots After Outlier Removal (Z-score & Manual)

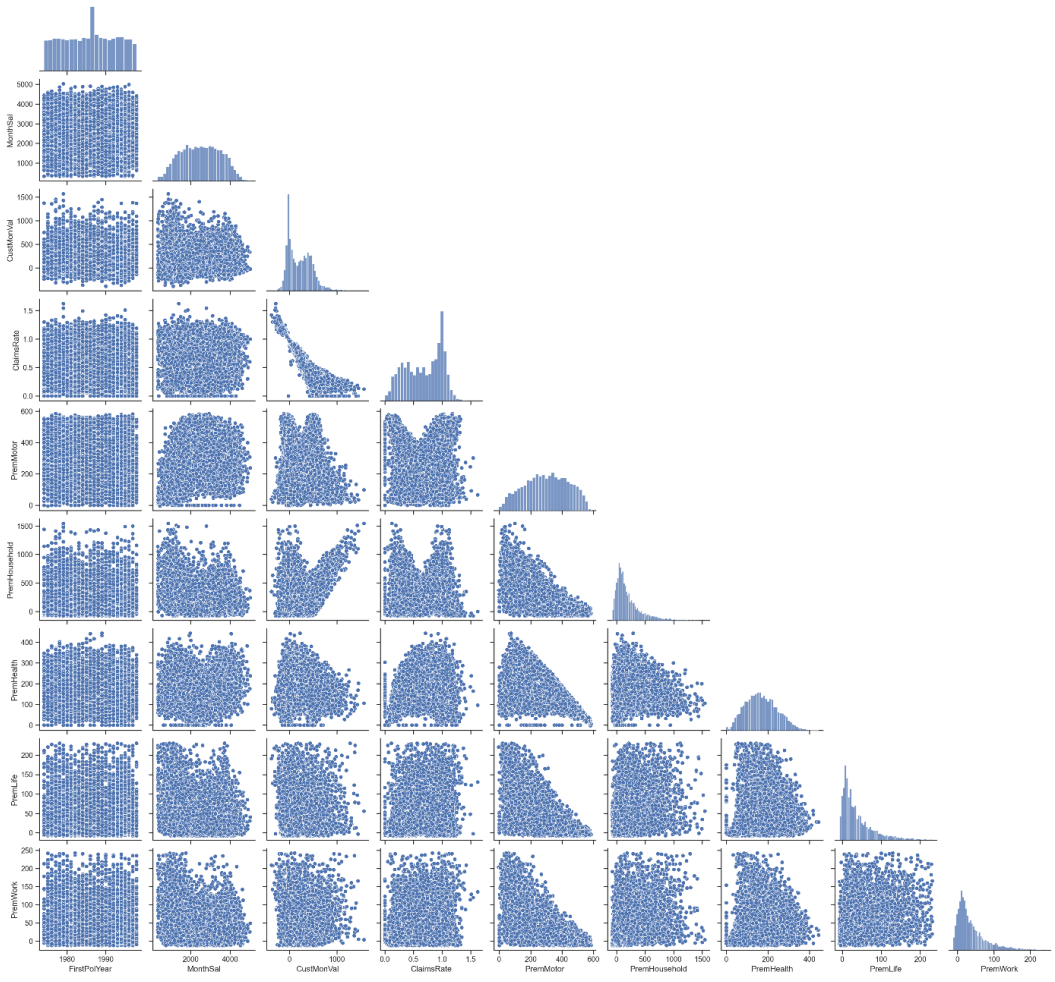

Figure 7 - Pairwise Relationship of Metric Variables

Variable Name	Definition	Description
CustomerYears	2016 - *FirstPolYear*	Number of years as a customer
AnnualSal	*MonthSal* * 14 (multiplied by 14 due to vacations and Christmas grants)	Annual Salary
PremTotal	*PremMotor + PremHousehold + Prem-Health + PremLife + PremWork*	Sum of all Premiums in 2016
RatioPremTotal	*PremTotal / AnnualSal*	Proportion of the annual salary spent on Premiums
CustMonVal_Year	*CustMonVal / CustomerYears*	Approximate annual value

Table 3 – New Variables Created

	count	mean	std	min	25%	50%	75%	max
FirstPolYear	10115.0	1986.014983	6.596894	1974.000000	1980.00000	1986.000000	1992.000000	1998.000000
EducDeg	10115.0	2.495996	0.786767	1.000000	2.00000	3.000000	3.000000	4.000000
MonthSal	10115.0	2515.115986	974.285994	333.000000	1739.50000	2525.000000	3293.000000	5021.000000
GeoLivArea	10115.0	2.707266	1.266857	1.000000	1.00000	3.000000	4.000000	4.000000
Children	10115.0	0.707958	0.454724	0.000000	0.00000	1.000000	1.000000	1.000000
CustMonVal	10115.0	217.177399	253.613803	-386.180000	-9.10000	186.260000	398.240000	1571.760000
ClaimsRate	10115.0	0.679484	0.317722	0.000000	0.39000	0.720000	0.980000	1.620000
PremMotor	10115.0	299.526876	136.755161	-4.110000	196.15000	301.500000	409.190000	585.220000
PremHousehold	10115.0	203.003762	229.359918	-75.000000	48.90000	132.250000	285.050000	1544.750000
PremHealth	10115.0	167.977643	74.826794	-2.110000	111.91000	163.030000	219.930000	442.860000
PremLife	10115.0	39.454589	42.806902	-7.000000	9.78000	24.560000	55.900000	230.820000
PremWork	10115.0	39.011000	42.714620	-12.000000	9.89000	25.340000	54.790000	242.600000
CustomerYears	10115.0	29.985017	6.596894	18.000000	24.00000	30.000000	36.000000	42.000000
AnnualSal	10115.0	35211.623806	13640.003918	4662.000000	24353.00000	35350.000000	46102.000000	70294.000000
PremTotal	10115.0	748.973870	189.198937	0.000000	625.95500	696.010000	820.660000	1809.690000
CustMonVal_Year	10115.0	7.651264	9.350578	-16.629091	-0.29621	6.242439	13.485415	76.094444

Table 4 – All Variables' Descriptive Statistics After Feature Engineering

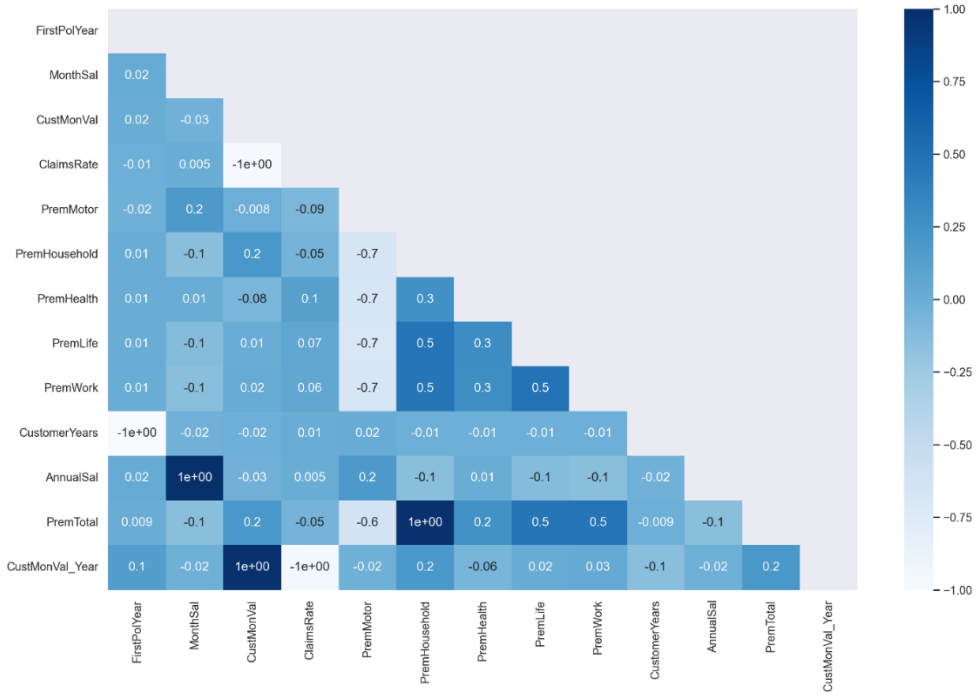

Figure 8 - Metric Variables' Spearman Correlation Matrix

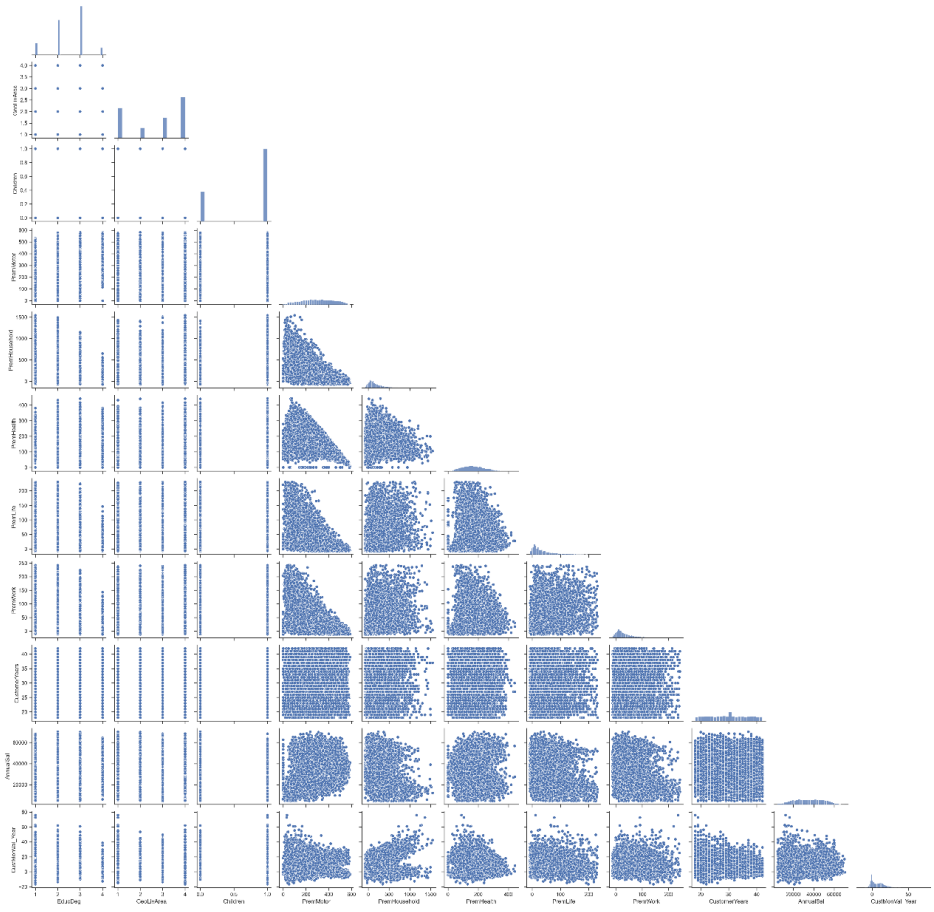

Figure 9 – Pairwise Relationship of All Variables

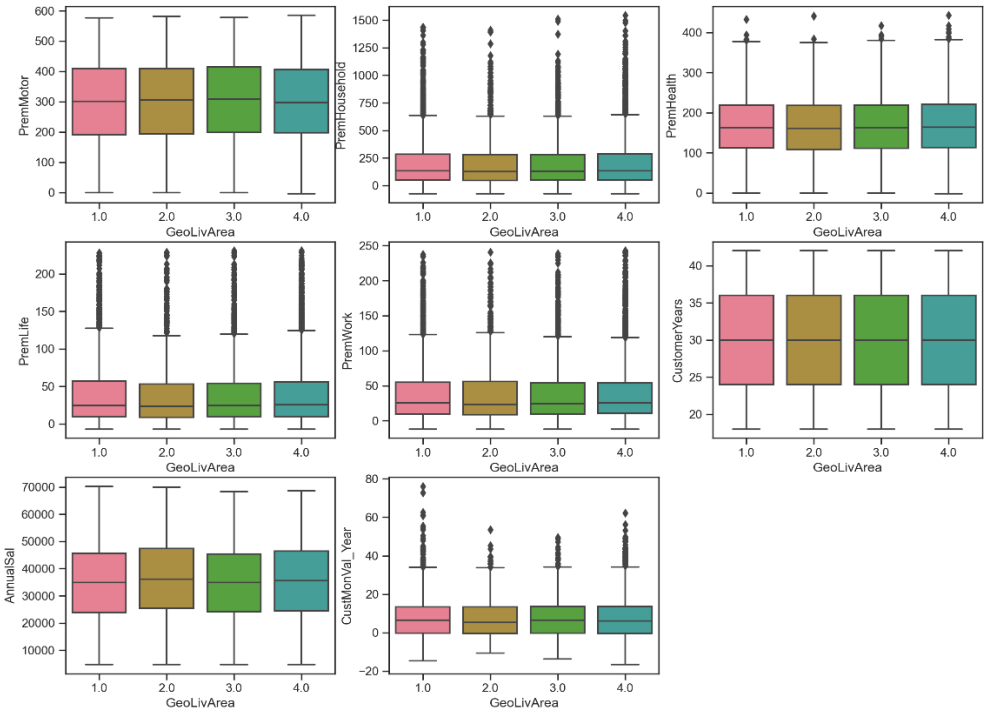

Figure 10 – Box Plots Displaying the Impact of *GeoLivArea* on other features

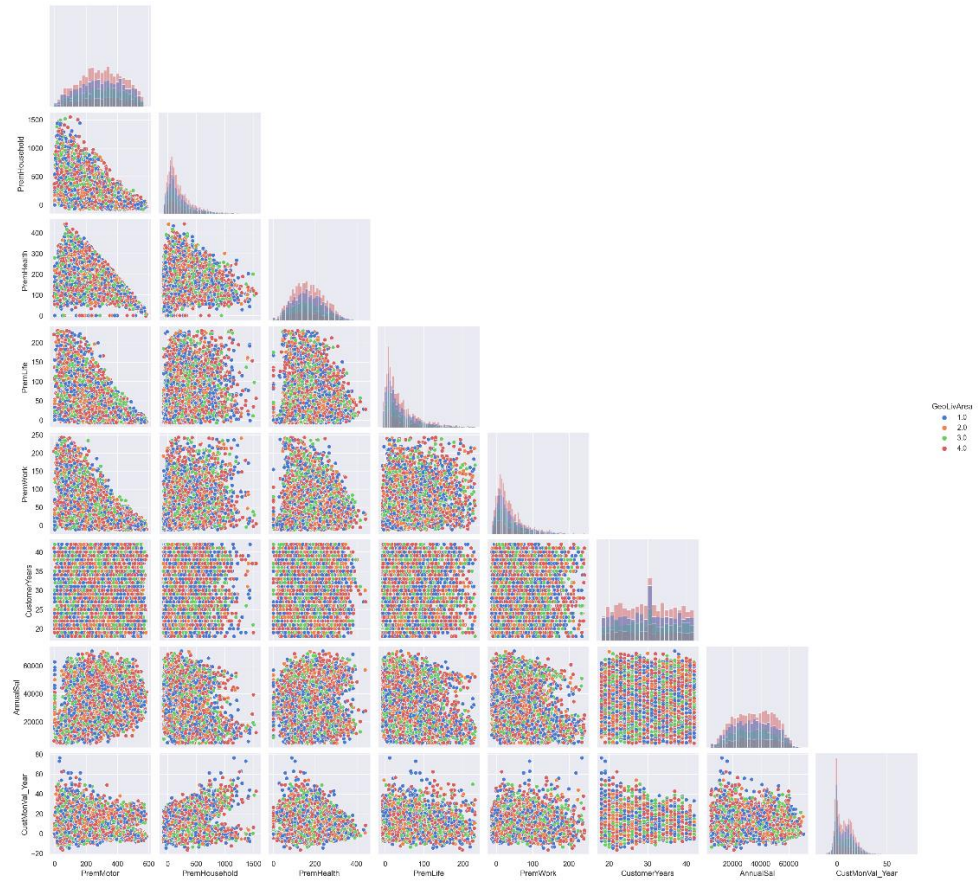

Figure 11 – Pairwise Relationship of Metric Variables Using *GeoLivArea* as Hue

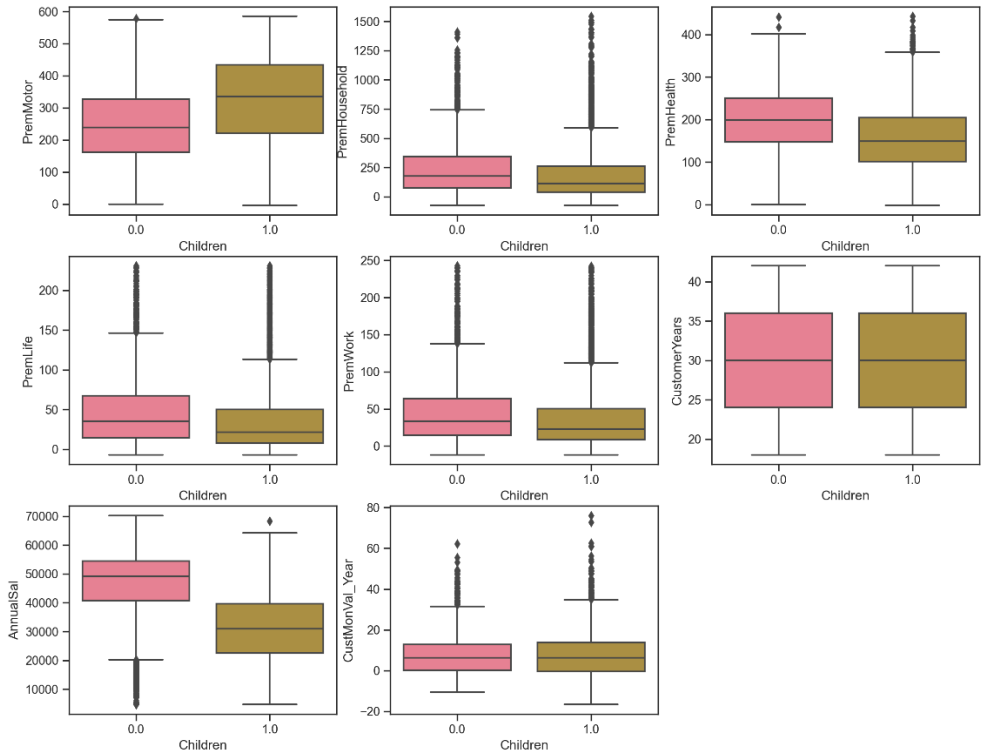

Figure 12 – Box Plots Displaying the Impact of *Children* on other features

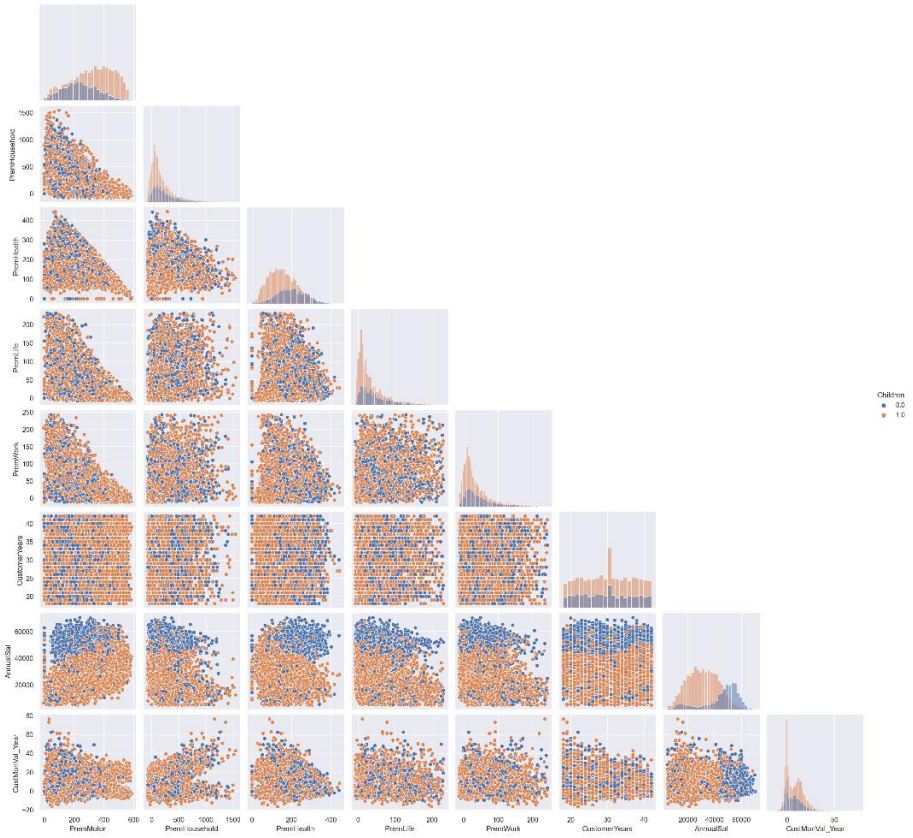

Figure 13 – Pairwise Relationship of Metric Variables Using *Children* as Hue

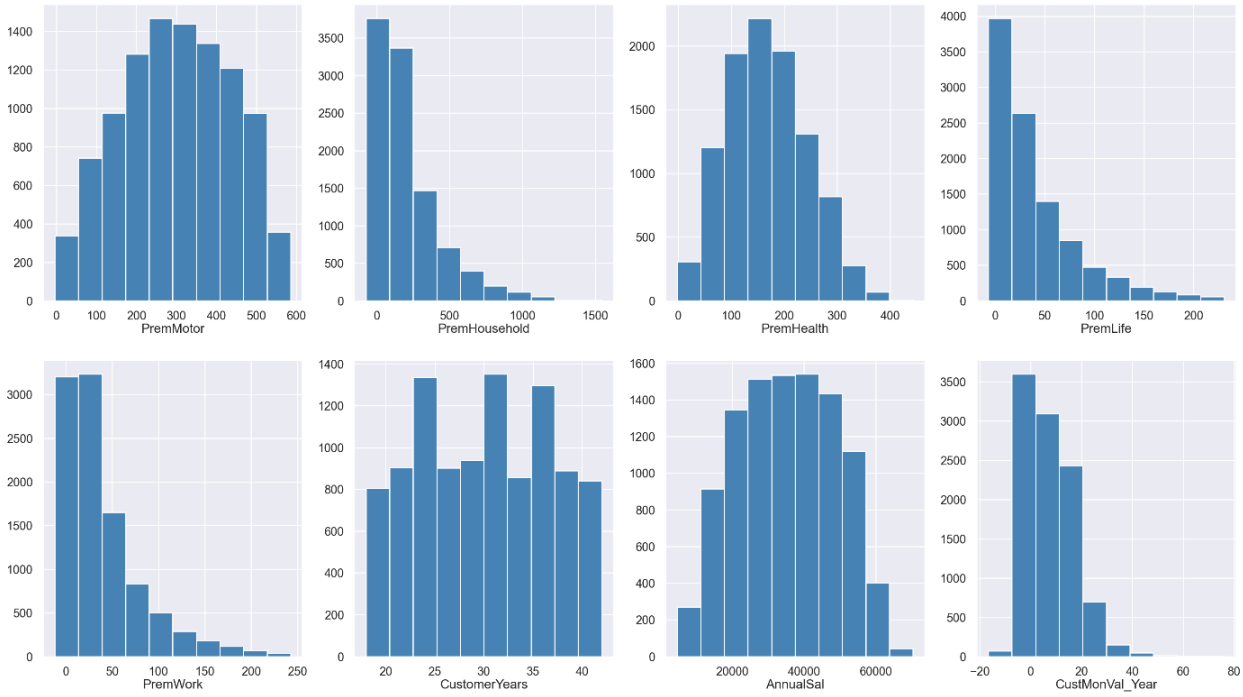

Figure 14 – Metric Variables' Histograms After Feature Engineering and Dimensionality Reduction

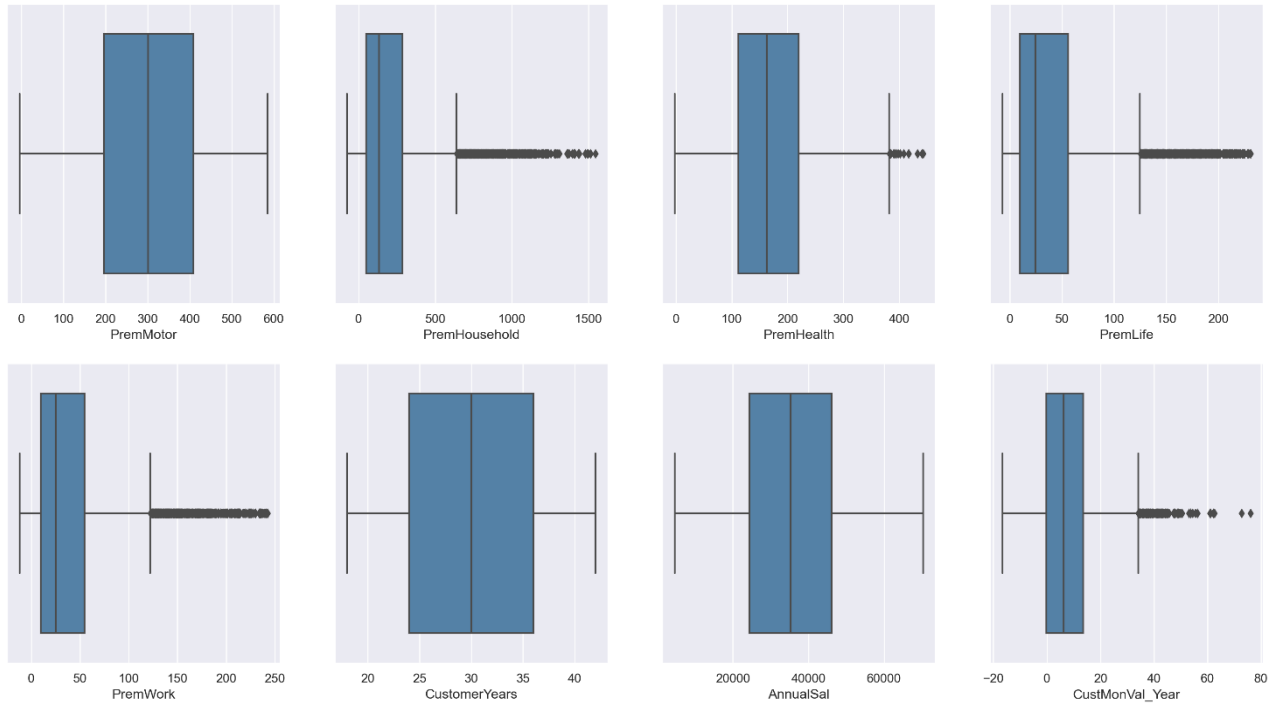

Figure 15 – Metric Variables' Box Plots After Feature Engineering and Dimensionality Reduction

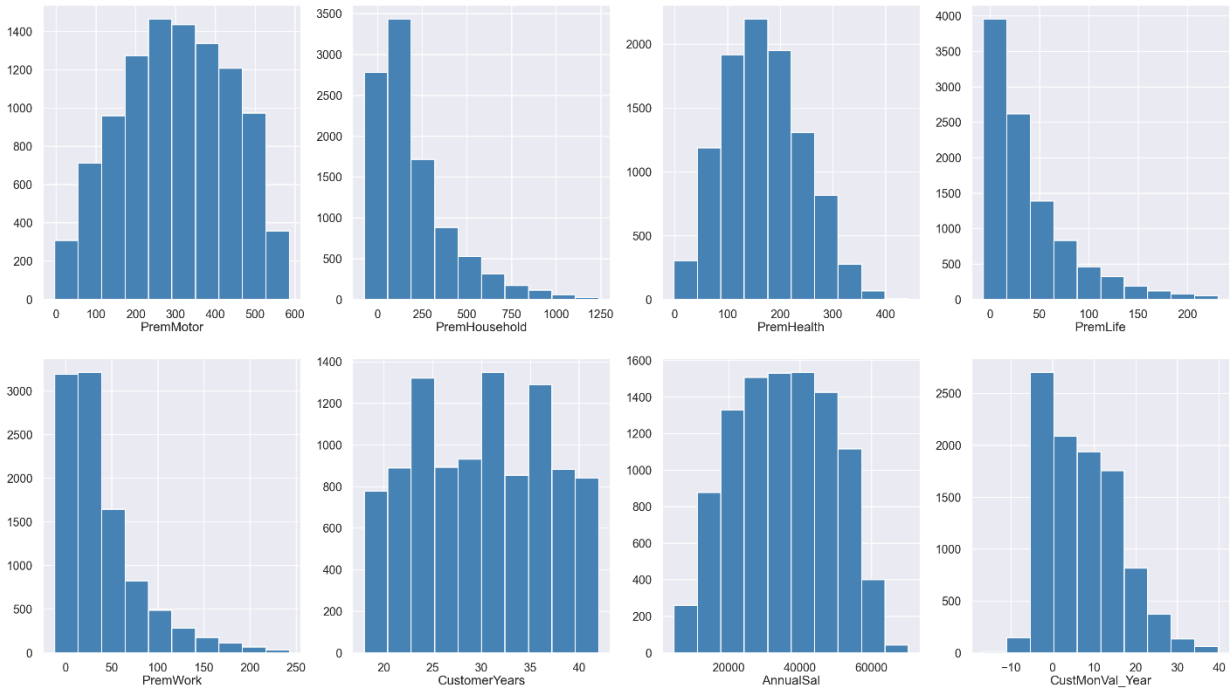

Figure 16 – Metric Variables' Histograms After Second Outlier Removal (Manual Filter)

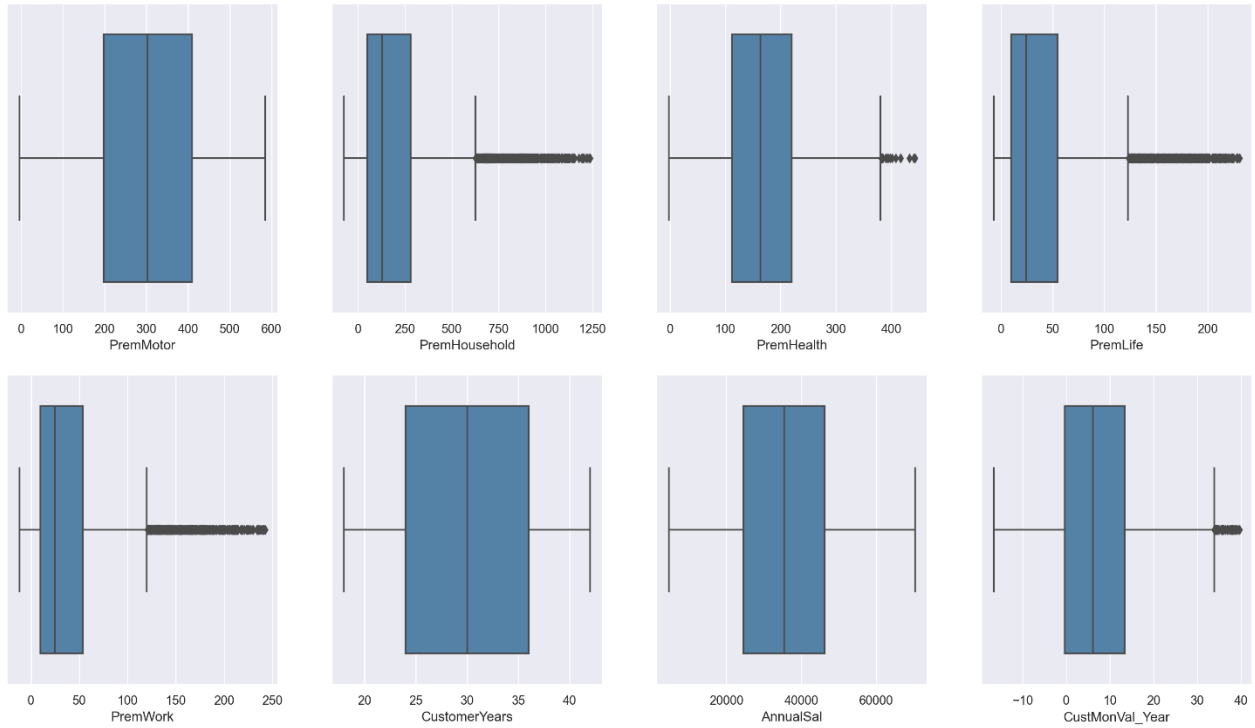

Figure 17 – Metric Variables' Box Plots After Second Outlier Removal (Manual Filter)

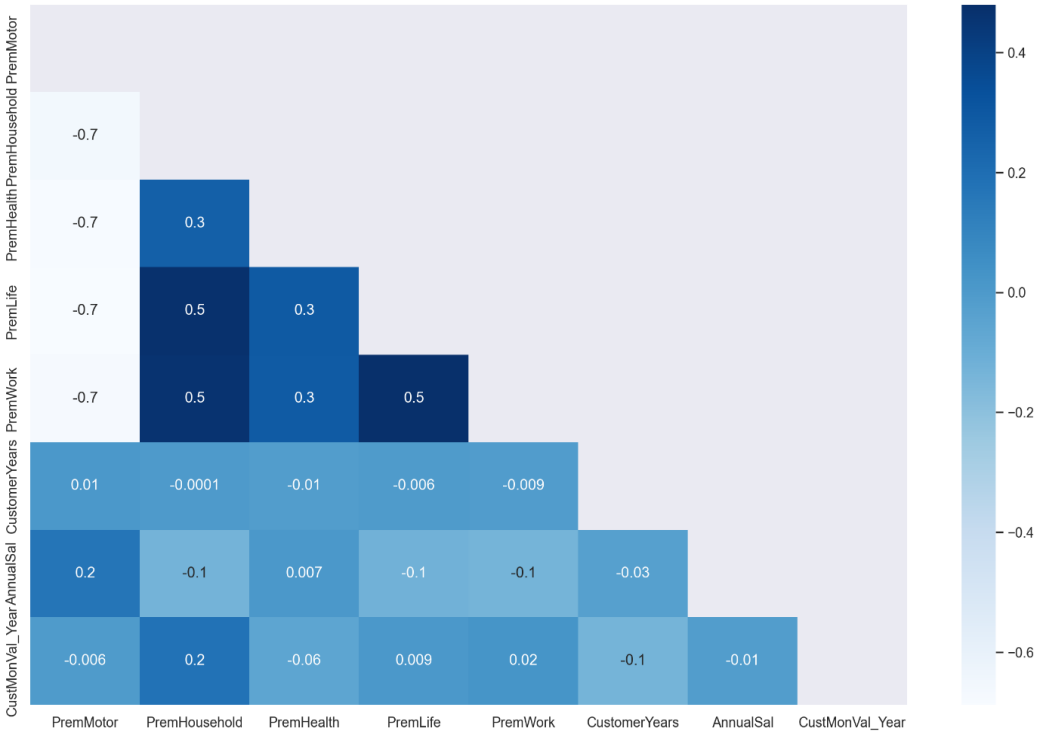

Figure 18 – Metric Variables' Spearman Correlation Matrix when Redoing Data Exploration

	count	mean	std	min	25%	50%	75%	max
PremMotor	10033.0	1.820089e-16	1.00005	-2.246788	-0.756806	0.015192	0.796977	2.089894
PremHousehold	10033.0	-1.303098e-16	1.00005	-1.255688	-0.682027	-0.311625	0.388204	4.823540
PremHealth	10033.0	-2.128158e-16	1.00005	-2.274273	-0.750645	-0.059251	0.689947	3.666049
PremLife	10033.0	1.701464e-16	1.00005	-1.084784	-0.690288	-0.342812	0.373064	4.506338
PremWork	10033.0	1.473068e-16	1.00005	-1.196711	-0.679977	-0.331081	0.361519	4.813368
AnnualSal	10033.0	-2.202520e-16	1.00005	-2.255770	-0.793140	0.008216	0.798241	2.572967
CustMonVal_Year	10033.0	-1.260606e-16	1.00005	-2.735166	-0.878529	-0.142591	0.678480	3.677016

Table 5 – Metric Variables' Descriptive Statistics (Standard Scaler)

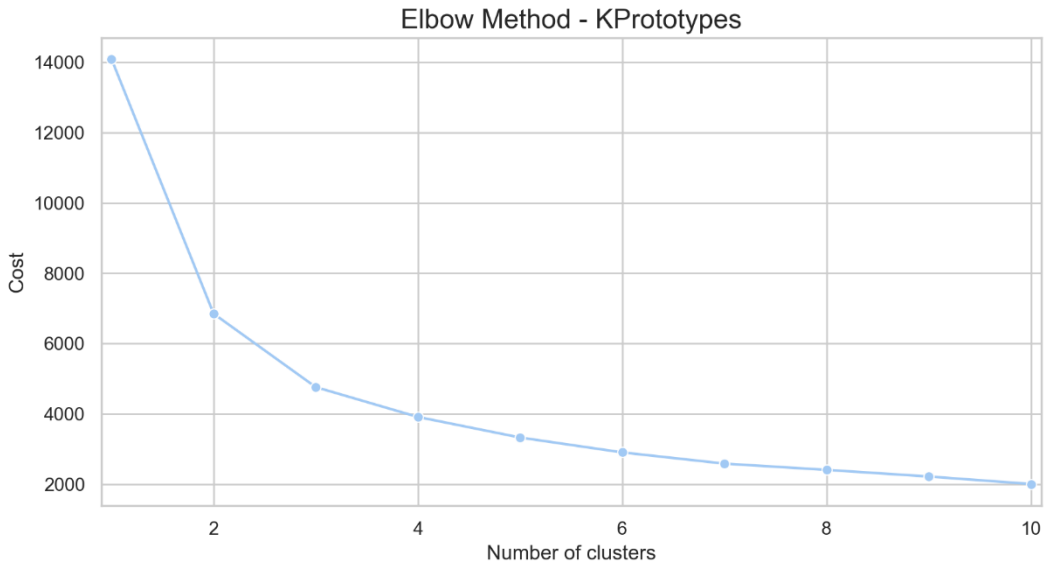

Figure 19 – Cost Value for Each Number of Clusters (K-Prototypes)

kproto3_labels	HighSchool	BSc/MSc	PhD	Children	AnnualSal
0	0.544622	0.201222	0.062776	0.870716	-1.122108
1	0.154642	0.716572	0.059837	0.918247	-0.054158
2	0.394711	0.423802	0.088595	0.269091	1.165880

Table 6 – Cluster's Centroids: K-Prototypes

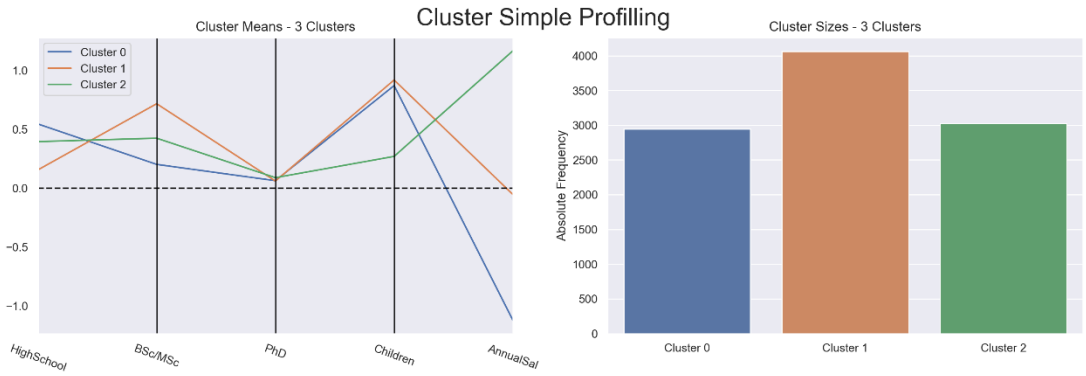

Figure 20 – Cluster Profiling: K-Prototypes with k=3

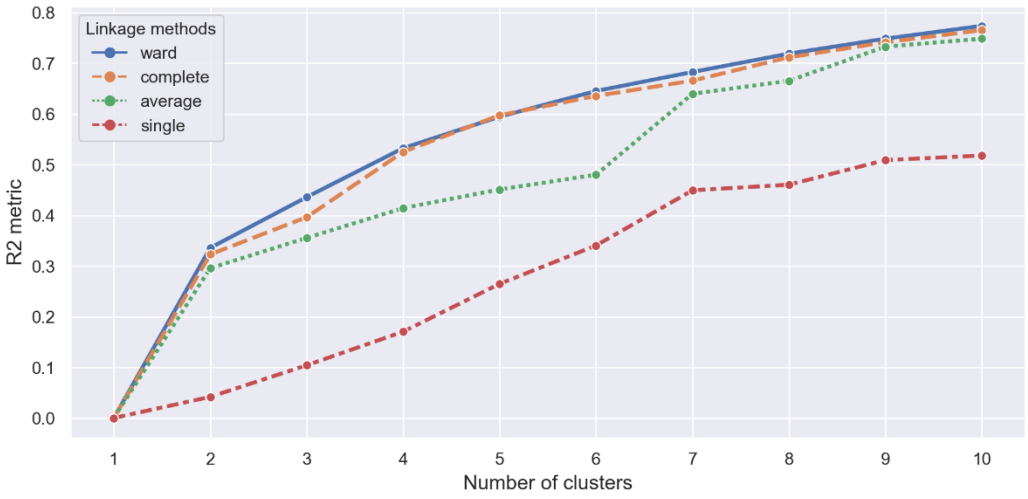

Figure 21 – R^2 Plot for Various Linkage Methods and Numbers of Clusters (K-Means + Hierarchical)

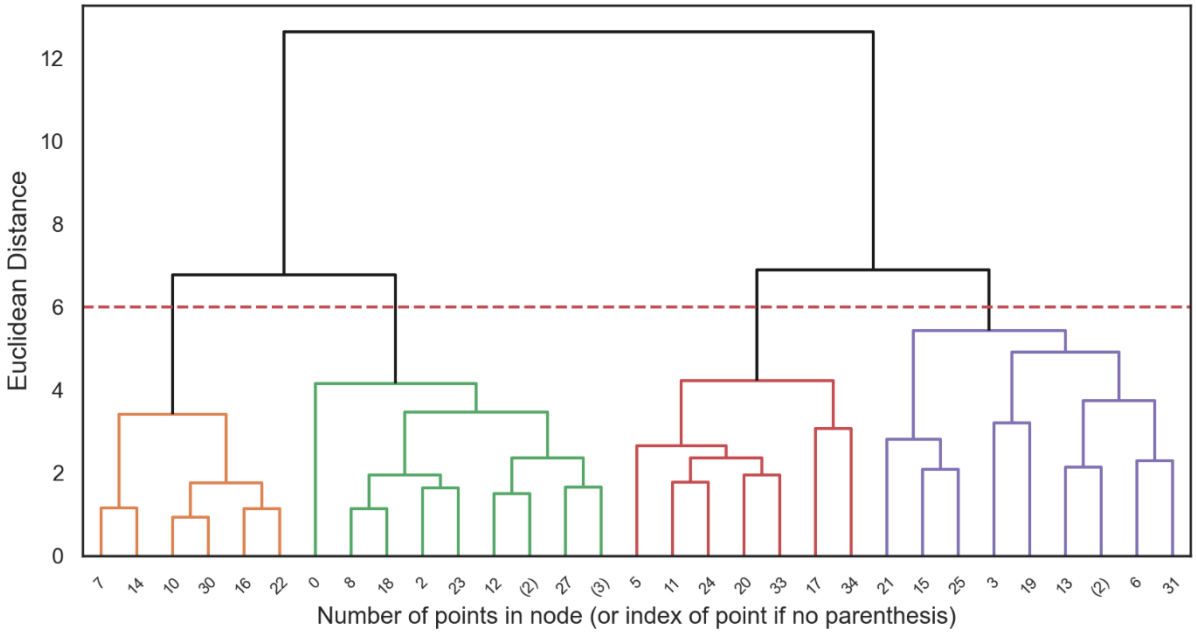

Figure 22 – Dendrogram: K-Means + Hierarchical

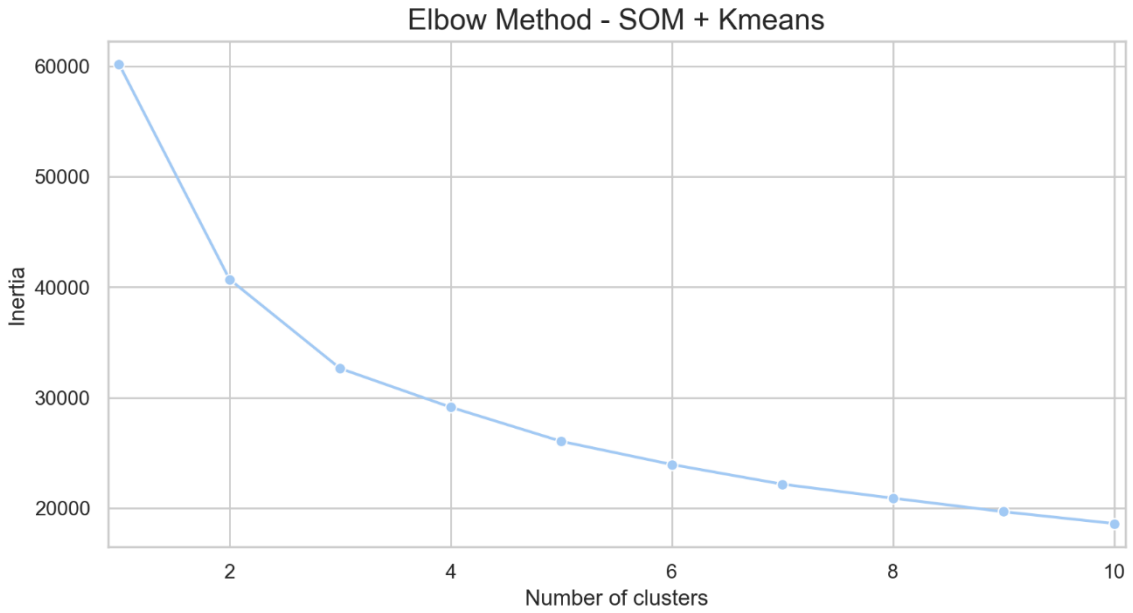

Figure 23 – Inertia for Each Number of Clusters (SOM + K-Means)

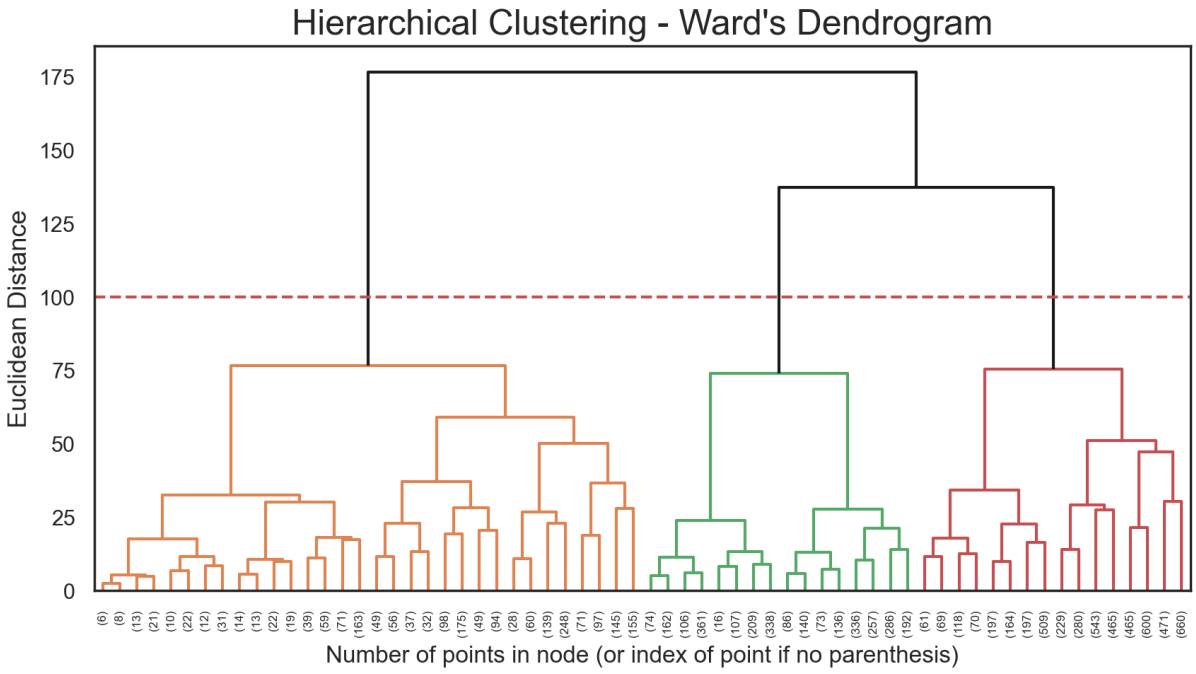

Figure 24 – Dendrogram: Hierarchical Clustering + SOM

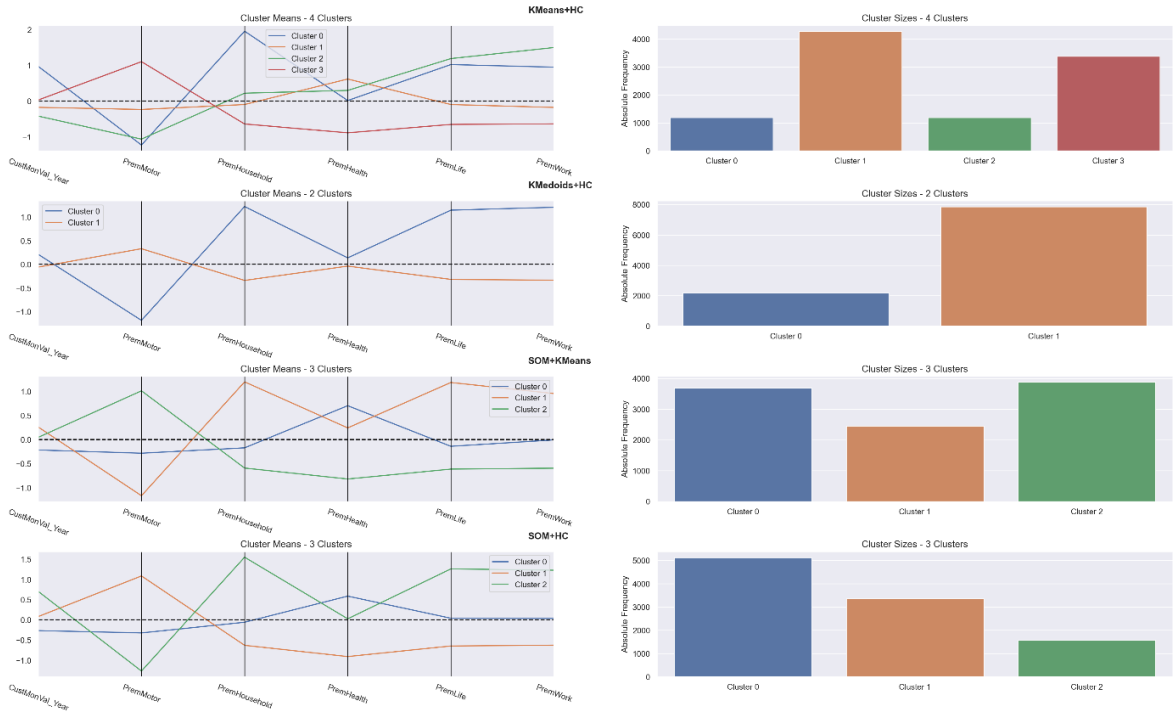

Figure 25 – Simple Profiling Comparison: Product Perspective

som_k_label	CustMonVal_Year	PremMotor	PremHousehold	PremHealth	PremLife	PremWork
0	-0.218400	-0.284988	-0.173030	0.701781	-0.141640	-0.008889
1	0.257122	-1.167787	1.197021	0.240582	1.184006	0.954690
2	0.045031	1.010865	-0.593431	-0.820337	-0.615065	-0.596184

Table 7 – Clusters' Centroids: SOM + K-Means

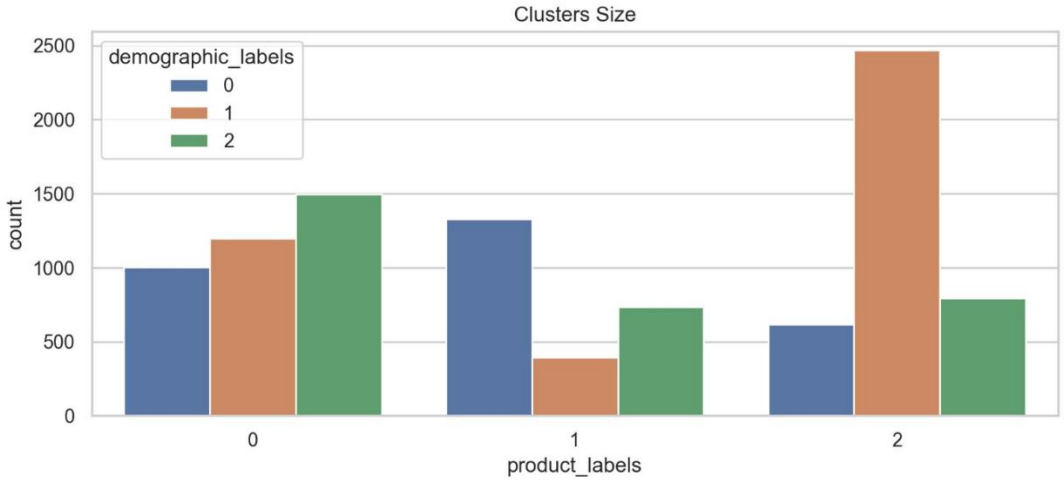

Figure 26 – Clusters' Sizes (Semi-Final Solution of 9 Clusters)

product_labels	0	1	2
demographic_labels			
0	1002	1328	617
1	1197	394	2470
2	1495	736	794

Table 8 – Contingency Table (9 Clusters)

merged_labels	HighSchool	BSc/MSc	PhD	Children	AnnualSal	CustMonVal_Year	PremMotor	PremHousehold	PremHealth	PremLife	PremWork
2	0.511295	0.134036	0.003012	0.782380	-1.335504	0.223828	-1.356338	1.360448	0.189391	1.397025	1.267444
4	0.338336	0.533879	0.050023	0.916326	-0.554067	-0.211370	-0.254637	-0.173054	0.645135	-0.146948	-0.000037
6	0.242979	0.599330	0.120072	0.851585	0.098156	0.045031	1.010865	-0.593431	-0.820337	-0.615065	-0.596184
7	0.353177	0.512375	0.070903	0.177258	1.214314	-0.228741	-0.329632	-0.172994	0.785101	-0.133833	-0.021911
8	0.471681	0.301770	0.008850	0.429204	0.704072	0.296249	-0.946197	1.004959	0.300744	0.933662	0.587133

Table 9 – Final Clusters' Centroids

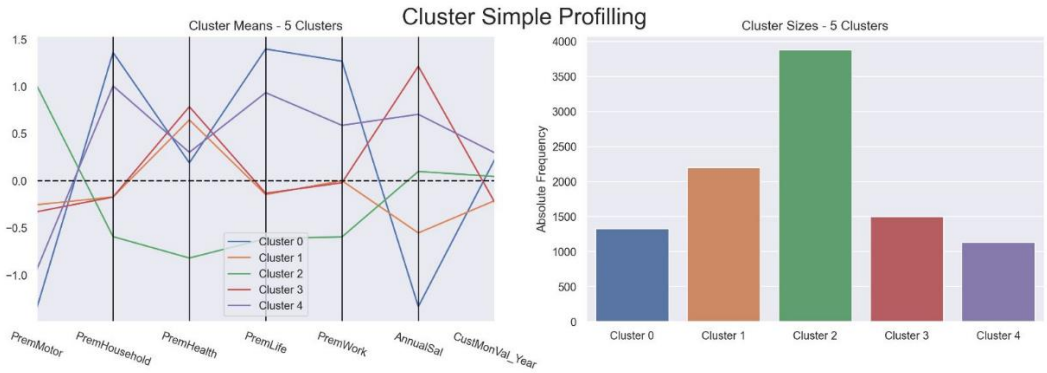

Figure 27 – Cluster Profiling: Final Solution

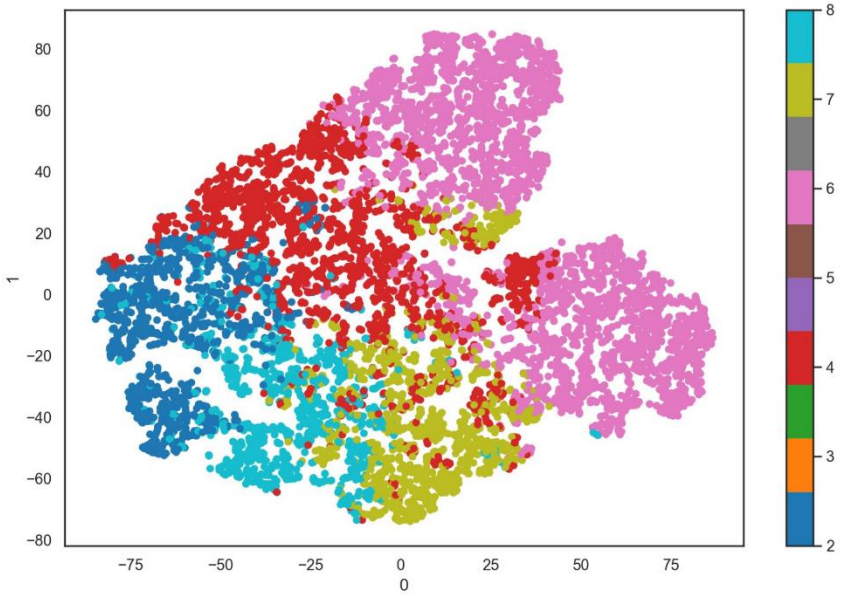

Figure 28 – T-SNE Cluster Visualization