# **To-do:**

* Include our environment in the hand in
* Include Henrique’s text

# **Project Report for**

# **Data Mining Group Project**

# 

## **Master’s in Data Science and Advanced Analytics at NOVA IMS, Lisbon**

## **Group Information**

Group H

Group members:

Henrique Vaz

Philipp Metzger

Link to GitHub repository:

<https://github.com/ph1001/NOVA-Data-Mining-Project>

**Abstract**

This report describes the step-wise creation of a customer segmentation on a dataset provided by Paralyzed Veterans of America (PVA). In chapter I, information necessary for understanding the problem at hand is presented. In chapter II, some background knowledge on clustering is conveyed. In chapter III, the methodology of this project is described and in chapter IV, the results are presented and discussed. Finally in chapter V, conclusions from the findings are drawn and presented.

**I. Introduction**

The goal of this project is to develop a customer segmentation based on a dataset provided by the non-profit organization Paralyzed Veterans of America (PVA). PVA provides programs and services for US veterans with spinal cord injuries or disease. The dataset contains information on individuals that have donated to PVA and that are classified as “Lapsed” donors[[1]](#footnote-1), meaning that they made their last donation to PVA 13 to 24 months ago.

**II. Background**

In chapter 1 of [1], the goal of “clustering” is described as the discovery of groups of similar examples within the data. The customer segmentation in this project was done by clustering the observations contained in the dataset provided by PVA, resulting in subgroups of similar observations, to which then through interpretation of the characteristics of the different clusters, different marketing approaches were assigned.

In chapter 9.1 of [1], the intuition of clustering is described as the effort to find subgroups in the dataset at hand, such that the inter-point distances between points of the same cluster are small in comparison to their distances to points from different clusters.

There are several clustering techniques. As an example of a clustering algorithm, k-means, which is one of the most popular clustering algorithms, is described:

The goal of k-means is to partition the data into k subgroups, where every datapoint is allocated to one and only one subgroup. The process of k-means clustering can be summarised in the following way: k vectors from the same vector space as the points from the dataset are chosen in an appropriate manner. These k vectors are called centroids. For all points in the dataset, the nearest centroid is identified, using a distance measure such as the Euclidian distance, and the point is assigned to it. Then, all k centroids are updated in such a way that their new position represents the centre of the subgroup of points that were assigned to each one of them. Then, the assignment of every point to the nearest centroid is repeated and after that, the position of the k centroids is updated accordingly. These steps are repeated until a stopping criterion is fulfilled. The resulting k centroids represent the centres of k clusters in the dataset, which are comprised of the points that are nearer to the respective centroid than to any other one.

**III. Methodology**

**III.1 Materials and Software**

The materials used in this project are the dataset provided by PVA and the related metadata file, which describes that variables contained in the dataset. The dataset consists of 95412 observations and 475 variables.

The software that is used in order to complete the project is Python and more precisely Anaconda and Jupyter Notebook. In the latter, the code for this project is created.

In the following part, the steps conducted in our Jupyter notebook are described. The order of these descriptions is the same as the order of actions undertaken in the Jupyter notebook that has been handed in alongside with this report. The results and findings as well as their implications will be presented and discussed in the following chapter IV.

**III.1 Imports and Organisation of Libraries and Data**

As a first step, the necessary libraries were imported, and the dataset was loaded into main memory from the csv file provided.

Next, using the file “pva\_metadata.txt”, the features obtained from the imported dataset were split into metric and non-metric features.

**III.2 Data Cleaning**

In this step, again using the metadata file, the existence of cells was assessed, where spaces (“ “) carry a meaning, such as for the feature ‘MAILCODE’, where a space means that the address of this individual “is OK”, whereas the value “B” means that the address “is bad”. Other features where something like this is the case are the features ‘NOEXCH’, 'RECINHSE', 'RECP3', 'RECPGVG', 'RECSWEEP' and ‘MAJOR’. For these features, the spaces were replaced by a meaningful string, such as “Address is OK”.

In the next step, the existence of duplicated observation was assessed and all remaining spaces in the dataset were replaced by NaN values[[2]](#footnote-2). It was also checked if the dataset contains any empty strings.

In the next part of the code, the percentage of missing values contained in each feature was assessed. Features that have more then 40 % missing values were discarded.

**III.3 Data Transformation**

In this section of the Jupyter notebook, all features containing time related information such as dates were transformed from their original string format to the datatype datetime.date[[3]](#footnote-3). For this purpose, using the metadata file, a list ‘date\_features’ was defined, which contains all the features that are to be changed in this step. Each column represented by an element of this list, is then sent through a pipeline consisting of three functions that were defined in this step. Part of the functionality of this pipeline is to ensure that NaN values remain unchanged.

**III.4 Further Data Cleaning**

In this section of the code, the distributions of some features, where anomalies had caught the authors’ eyes, were first visualised in order to then remove the unusual patterns that are likely to be errors stemming from a faulty process of data collection. Then, the unusual values were replaced by NaN values.

**III.5 Feature Selection**

As a first feature selection step, correlations between all metric features were assessed. Of feature pairs that were highly correlated, one was discarded, and one was kept in the dataset. Also, metric features that only contain a very small number of distinct values and thus carry little information for our analysis as well as features containing mostly zeros were discarded.

**III.6 Further Feature Transformation**

In this step, the date features whose transformation was described in III.3 were further transformed to integers representing their distance in days to the reference date stored in each observation’s value of the variable ‘ADATE\_2’[[4]](#footnote-4). For better traceability, the resulting features were renamed in the following format: “<<original feature name>>>\_rel\_in\_days”.

**III.7 Excluding features with very low Gini coefficient**

In the following step, the Gini coefficient was assessed for every column in the dataset. In order to reduce dimensionality, all columns were discarded that had a Gini coefficient lower than , being the average of all columns’ Gini coefficients. The reasoning behind this step was that columns with a low Gini coefficient contain values that are quite similar to each other, thus containing little information and potential for clustering.

**III.8. Filling the missing values**

For filling the missing values in the dataset, two approaches were considered: The IQR method and KNN imputation. After testing both approaches, the authors came to the conclusion that IQR does not work so well, since even with very high values for the IQR multiplier, too many features were excluded. Because of this, the decision was made to use KNN imputation only.

**III.9 Removal of outliers**

For outlier removal, DBSCAN was chosen.This density-based algorithm, which can also be used for clustering, pools observations, that share densely populated spaces with a sufficient number of other observations and marks the rest of the observations as outliers. As the value for the first parameter minPts, the approximate square root of the number of observations was chosen and in order to find a good value for the second parameter Eps, a k-distance graph was created. This graph shows the sorted distances of the points in the dataset to their kth (k = minPts) neighbor. With its help it can be seen which distance needs to be chosen in order to exclude the points that are quite far away from their kth neighbor. These points are regarded as potential outliers and not included in the process of clustering, but instead later added to the final result, by allocating them to their nearest cluster.

**III.10 Data normalisation**

For the step of feature scaling, normalisation was chosen as the technique applied. All columns corresponding to metric features were normalised.

**III.11 Clustering**

For the clustering, two approaches were followed:

**III.11.i Approach 1**

Henrique’s approach

**III.11.ii Approach 2**

Summary of approach 2:

K-means clustering on different feature sets, representing different perspectives on the data with subsequent combination of those perspectives and hierarchical clustering on the found centroids in order to determine the perspective combination that leads to the best result.

In the following, the second approach will be described in detail:

First, the feature sets representing the different perspectives on the data were assessed again, drastically reducing their number and their number of features per perspective. For this, the component planes from approach 1 were used to identify the most promising features from the features used in approach 1. As an example, the component planes for the perspective ‘donor\_info’ are presented in Figure 1. From these four features, three were selected to be used in this approach: ‘ODATEDW\_rel\_in\_days’, ‘INCOME’ and ‘NUMPROM’, since these three features show clear and distinguishable patterns in the component planes.

A picture containing graphical user interface

Description automatically generated

Figure 1: Component planes that were created in clustering approach 1

Then, on each of these reduced feature sets, k-means clustering was done iteratively with values of k ranging from 2 to to 9, each time saving the R2 score of the resulting clustering solution. Like this, the optimal value for k was identified for each perspective. Figure 2 shows the resulting R2 scores for each perspective and for varying values for k.

Chart, line chart

Description automatically generated

Figure 2: R2 scores for the different perspectives and varying values of k (numbers of clusters)

Then, with the optimal value for k for each perspective, k-means was run again once more on each perspective, yielding cluster labels for each perspective. In the next step, pairwise combinations of all perspectives were created and the centroids for each of the the resulting cluster label pairs were computed. Each of the resulting arrays of centroids were then the basis for hierarchical clustering, with the purpose of assessing which of the cluster pair combinations could be combined into merged, larger clusters. For this ‘AgglommerativeClustering’ was used on each result of each pair of perspectives and then a dendrogram for each fitted instance was created, in order to visually assess the clustering solution.

The plan was then to choose the most promising few of these clustering solutions and their perspectives pairs in order to then assess, which was would be the best for the final clustering. But, taking into consideration all the result, the authors came to the conclusion to use only one feature perspective for the final clustering. The reasons for this and the results are further described in IV.11.ii. After the final clustering and interpretation described in IV.11.ii, the outliers whose removal is described in III.9 were each allocated to their closes cluster.

**IV. Results** **and Discussion**

**IV.1 Results and discussion: Imports and Organisation of Libraries and Data**

We identified 398 metric and 77 non-metric features in the dataset.

**IV.2 Results and discussion: Data Cleaning**

In this step it was found out, that no duplicated observations or empty strings exist in the dataset. Furthermore, 3011889 spaces were converted to NaN values.

67 metric and 30 non-metric features were discarded from the dataset, due to them having a percentage of missing values higher than 40 %.

**IV.3 Results and discussion: Data Transformation**

The result of this step is the updated dataset, where the features defined in the list ‘date\_features’ have been changed to objects of the type datetime.date. This facilitated their further processing and enabled us to do calculations such as addition or subtraction of different date columns.

**IV.4 Results and discussion: Further Data Cleaning**

The variables with the names ‘AGE90x’, x ∈ {1, 2, 3, 4, 5, 6, 7} serve as a good example for the process of removing values that are likely to be faulty. Figure 1 shows the distributions of these features before any values were removed. It is apparent, that there are values similar to zero, whose frequencies don’t integrate well with the rest of the frequencies of these distributions. These values are indeed the value zero. After all of these zeros were replaces by NaN, the distributions were checked again. The resulting distributions, visualised as histograms, are presented in Figure 2.

Graphical user interface, diagram

Description automatically generated

Figure 3: The distributions of the variables ‘AGE90x’, x ∈ {1, 2, 3, 4, 5, 6, 7} before the removal of the value zero

**A picture containing chart

Description automatically generated**

Figure 4: The distributions of the variables ‘AGE90x’, x ∈ {1, 2, 3, 4, 5, 6, 7} after the removal of the value zero

**IV.5 Results and discussion: Feature Selection**

In this step, 92 features were discarded. The number of metric features was decreased from 331 to 249 and the number of non-metric features dropped from 47 to 37.

**IV.6 Results and discussion: Further Feature Transformation**

The values in days resulting from this step range from -122 days (= approximately -0.33 years) to 31928 days (= approximately 87 years). After the discovery that negative values exist, they were located in the dataset. They belonged to the feature ‘DOB’, or more precisely to the newly created feature ‘DOB\_rel\_in\_days’. There were five observations present in the dataset that had values equal to or smaller than zero in this column[[5]](#footnote-5). It was decided to remove these observations, since it is highly unlikely that any promotion was mailed to an individual that wasn’t born yet or that was born on the day of the mailing.

**IV.7 Results and discussion: Excluding features with very low Gini coefficient**

In this step, 71 features were excluded that all have a Gini coefficient lower than 0.1956. The authors decided that dropping these features is adequate and helpful for the later task of making the final feature selection for clustering. The remaining numbers of features after this step were 249 metric and 178 non-metric features.

**IV.8. Results and discussion: Filling the missing values**

Since KNN imputation was used on the whole set of metric features, the computation time when running it is very long. Because of this, the authors decided to save a copy of the data that is the result of this step in a newly created csv file called 'donors\_after\_KNN\_imputation\_with\_date\_features.csv'. The resulting dataset contains no more NaN values in any of the metric features.

**IV.9 Results and discussion: Removal of outliers**

Figure 5 shows the k-distance graph created for this application. Just by looking at it, it can be seen that that there are approximately 10000 observations that are much further away from their kth neighbour than the other observations. These observations have an approximate distance of 400 to 500 to their kth neighbour. To be conservative, 500 was chosen in order to be sure to exclude most outliers. Running DBSCAN with minPts = 300 and Eps = 500 resulted in the removal of 7547 observation, which accounts for 7.91 % of the dataset’s observations.

**IV.10 Results and discussion: Data normalisation**

As the result of this step, all metric features were normalised.

Chart

Description automatically generated

Figure 5: k-distance graph used for determining a good value for the DBSCAN parameter Eps, which was used for outlier detection

**IV.11 Results and discussion: Clustering**

**IV.11.i Results and discussion: Approach 1**

**IV11.ii.1 Results and discussion: Approach 2 - description of the results**

The result of the feature perspectives selection process, conducted by assessing the component planes from approach 1 resulted in seven perspectives that are as follows:

* donor\_info\_features\_2: 'ODATEDW\_rel\_in\_days', 'INCOME', 'NUMPROM',
* gifts\_given\_features\_2: ’RAMNTALL', 'NGIFTALL',
* population\_features\_2: 'POP90C1', 'POP90C2', 'POP90C3'
* provenance\_features\_2: ‘ETH2', 'ETH5',
* housing\_price\_features\_2: 'HVP3', 'HVP5', 'RP3’,
* housing\_quantity\_features\_2: ‘HU4', 'HU5’,
* income\_features\_2:’ HHD4', 'IC6'.

As an example, the dendogram resulting from pairing donor\_info\_features\_2 and gifts\_given\_features\_2 is presented in Figure 6.

**Chart, histogram

Description automatically generated**

Figure 6: Dendogram for pairing the feature sets of donor\_info\_features\_2 and gifts\_given\_features\_2

The feature perspective pairs that the authors deemed as promising for further investigation are (donor\_info\_features\_2, gifts\_given\_features\_2), (gifts\_given\_features\_2, housing\_price\_features\_2), and (gifts\_given\_features\_2, income\_features\_2). The analysis of the result yielded though, that none of the results of these feature perspective pairs merged by hierarchical clustering were satisfying. Instead, it was noticed, that the features in donor\_info\_features\_2 yielded quite good results on their own. In Figure 7, the profiling of the clustering solution that was acquired by using only the features from ‘donor\_info\_features\_2’ is presented. It is apparent that the four different clusters have distinct characteristics, which will be further described later on.

Chart, line chart

Description automatically generated

Figure 7: Profiling of the clustering on the features from 'donor\_info\_features\_2'

In a next step, as a basis for interpretation, more features were included in the visualisation presented in Figure 7. These features are POP90C1, HVP3, HVP5, RP4, and EC7. These features, again, were chosen on the basis of the component planes from clustering approach 1. The result of the visualisation with these new features for interpretation included is presented in Figure 8. It can be seen that also for these features that were not used for this clustering, the clusters show a clear pattern.

Chart, line chart

Description automatically generated

Figure 8: Visualisation of the clustering solution created with the features from 'donors\_info\_features\_2', extended by more features for interpretation purposes

Next, the absolute frequencies per cluster were visualised. The result is presented in Figure 9. It can be seen that the clusters’ frequencies are all in a similar range and no cluster has very little observations allocated to it.

Chart, bar chart

Description automatically generated

Figure 9: Visualisation of the absolute frequencies of observations in the clusters created with the features from 'donors\_info\_features\_2'

Next, a categorical feature was included in the interpretational analysis. The features chosen is ‘HOMEOWNR’, which represents a flag stating wether or not it is knows that the respective observation owns a home or not. A visualisation of the absolute frequencies of this variable in the four clusters was created. The result is displayed in Figure 10. It is apparent, that in the clusters 0 and 4, significantly less individuals are known to own a home than in the clusters 1 and 2.

Chart, bar chart

Description automatically generated

Figure 10: Absolute frequencies of the variable 'HOMEOWNR' for the clusterin solution obtained by using the features from 'donors\_info\_features\_2'

**IV.11.ii.2 Results and discussion: Approach 2 – interpretation**

Interpreting Figure 8 and Figure 10, the following characterisations of and appropriate marketing strategies for the four clusters can be derived:

Cluster 0:

Characterisation:

The average individual in this cluster is younger than the average. It has a below-average household income and has not donated much in the past, but has not received many promotions either. The ratio is higher than in cluster 1 and 3, meaning that the effect of the promotions is higher in this cluster than in the other two mentioned. The average individual of this cluster lives in a not-so-urbanised area that has below-average home values and rents. Is not so likely to hold a bachelor’s degree and is a little bit more likely to be homeowner than not to be one.

Marketing strategy:

Even though the average individual in this cluster has little money, the effect of promotions is higher than in clusters 1 and 3. **Increase the number of promotions sent per time span, but with caution!**

Cluster 1:

Characterisation:

The average individual in this cluster is older than the average. It has an above-average household income, has received a lot of promotions and has also donated more than the average person, but the ratio is lower for this cluster than in clusters 0 and 2, meaning that the effect per promotion is not as high in this cluster as it is in two clusters mentioned. The average individual of this cluster lives in a rather urbanised area that has above-average home values and rents. Is likely to hold a bachelor’s degree and is very likely to own a home.

Marketing strategy:

**Keep the number of promotions sent steady or even consider reducing the amount of promotions sent, but with caution!**

Cluster 2:

Characterisation:

The average individual in this cluster is younger than the average. It has an above-average household income, has not donated much in the past, but has received very little promotions as well. The ratio is the highest in this cluster, meaning that promotions seem to be quite effective in comparison to the other clusters. The average individual of this cluster lives in rather urbanised area that has above average home values and rents. Is likely to hold a bachelor’s degree and it is very likely to own a home.

Marketing strategy:

**Send more promotions!**

Cluster 3:

Characterisation:

The average individual in this cluster is older than the average. Even though it has a below-average household income, it donates significantly more than the other clusters’ average individuals. The average individual of this cluster lives in a not-so-urbanised area that has below-average home values and rents. It is less likely to hold a bachelor’s degree and is a little bit more likely to be homeowner than not to be one.

Marketing strategy:

**Keep sending the same volume of promotions. They are likely to be rewarded with donations.**

**V. Conclusions**

Conclusions Henrique’s approach

The clustering solution presented in III.11.ii and IV.11.ii yield results that can be interpreted in an intuitive way. It can be concluded that it is possible to segement the dataset at hand into four cluster that carry a meaning in terms of their characterisation as well as their appropriate marketing strategy. The features used for this clustering were ‘ODATEDW\_rel\_in\_days’, which is a transformation of the original variable ‘ODATEDW’, ‘INCOME’, ‘NUMPROM’ and ‘RAMTALL’ and the features used for the interpretation of the results obtained were ‘POP90C1’, ‘HVP3’, ‘RP3’ and ‘EC7’. Marketing approaches were defined that range from the recommendation to cautiously decrease the number of promotions mailed in a time frame to the increase of the promotiuons mailed in order to leverage the effectivity of the promotions in certain clusters.

**VI. References**

|  |  |
| --- | --- |
| [1] | C. Bishop, Pattern recognition and machine learning, Springer, 2006. |
| [2] | [Online]. Available: https://docs.python.org/3/library/datetime.html. [Accessed 30 Dec. 2020]. |

1. In the dataset’s metadata file also denoted as “Lapsing” donor [↑](#footnote-ref-1)
2. NaN, short for “not a number”, commonly denotes a missing value in a dataset. [↑](#footnote-ref-2)
3. For the documentation of the Python library “datetime”, see [2] [↑](#footnote-ref-3)
4. ‘ADATE\_2’ represents the dates on which the most recent promotion was sent to each individual. This variable was chosen to be the reference time for each observation. [↑](#footnote-ref-4)
5. These observations have the following indices: 22984, 40565, 52252, 60753, and 94452. [↑](#footnote-ref-5)