Data Mining Project

MASTER DEGREE PROGRAM IN DATA SCIENCE AND ADVANCED ANALYTICS

XYZ Sports Company

Group 56

Diogo Silva number: 20201643

Hugo Miguel number: 20230745

January, 2024

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

NOVA IMS

Table of Contents

1. Introduction

To expand any business, one has to learn everything about current customers, by doing this the company can group individuals by their needs, interests, or even preferences to the products or services the business provides. With a detailed study on this subject, not only will the customers be better served, but also improve targeting of prospective ones.
The goal of this project is exactly this, to perform market segmentation on a client company's ('XYZ Sports Company') dataset provided by their ERP system, using several perspectives and approaches. XYZ is a well-established fitness facility up and running for several years, whose goals with this endorsement are to enhance its strategies, improve customer engagement, and tailor its services.
The development of this project consisted of 5 main steps: (1) Data Exploration, (2) Data Pre-Processing, (3) Clustering, (4) Cluster Analysis, and (5) Conclusion.
We started by importing and analyzing the data (**Data Exploration** phase), secondly, we began the **Data Pre-Processing** phase, which is an extensive phase that consists of several steps with extreme importance to the outcome, and finally, we started testing **Clustering** algorithms, to understand which would perform better according to our data's characteristics. To wrap up, we performed **Cluster Analysis** and reached our final **Conclusions.**

2. Data Exploration

Data exploration is a fundamental phase in a data mining project where the primary goal is to comprehensively understand the dataset at hand. It involves examining and analyzing the data to uncover patterns, trends, anomalies, and relationships that may exist within the data.

2.1. Initial Analysis

We started the initial analysis by checking the first rows of the dataset and analyzing the data's descriptive statistics (depicted in Figures 7 and 8), as we immediately set column *'ID'* to index, and normalized it to start at 0, since these customers are a population on their own and their actual IDs aren't relevant to our study. Furthermore, we checked feature data types and other initial information like unique values to have a grasp of how the data is kept, we also performed pandas profiling which gave us an extensive report on important feature details, where we concluded that features *'DanceActivities'* and *'NatureActivities'* had only one value (0), and features such as *'OtherActivities'*, *'HasReferences'* and *'NumberOfReferences'* showed extremely low variability and showed potential for removal, duplicates were also analyzed and removed (only 1 row found). to conclude some discrepancies like *'Age'* having zeros were found and some data types that should be altered, but more on that later.

2.2. Visual Exploration

This subset of data exploration was mostly to aid the initial analysis with visual representations of our data, and conclusions are the same as previously mentioned. Before moving on, metric and non-metric features were defined and lists for each were created, furthermore, the features that represented date-times were also separated into a different list.

3. Data Pre-Processing

A critical stage in a data mining project that involves cleaning, transforming, and preparing raw data to ensure its quality, relevance, and suitability for analysis. We define it as the most extensive in our project as it required an extremely deep understanding of the data in our set and any modifications made during its execution would certainly influence the project outcome a great deal.

3.1. Initial Engineering

After Data Exploration, some initial engineering was obvious and was performed before any other pre-processing steps. The features that previously showed potential for removal were dropped, feature *'Gender'* was relabeled to binary (0-female and 1-male) and renamed to *'Male'*, features that represented date-time values were respectively changed to date-time data type (object initially), it was also observed that features 'NumberOfFrequencies' and both 'AllowedWeeklyVisitsBySLA' and 'AllowedNumberOfVisitsBySLA' were float data types and should be int, however since they still presented missing values, we made this change later on.

3.2. Incoherence Checking

During this step, we tried to create all (or at least the most relevant) arguments, that if true, would be considered incoherent in our data, for which we concluded that would be considered incoherences records where:

01. ***'Age'* having zeros**
    a. assumed it as a mistake
    b. results only showed 19 records that matched, because the number is very small we removed them.
02. ***'Income'* > 0 and *'Age'* < 16**
    a. Since we assumed below this age people don't legally discount their earnings (if they even have any).
    b. results only showed 20 records that matched, because the number is very small we removed them.
03. ***'EnrollmentFinish'* < *'EnrollmentStart'***

a. Since these are respective sequential dates, it wouldn't make sense for this to happen.
b. no records match.

04. ***'LastPeriodFinish' < 'LastPeriodStart'***
    a. following the same logic as argument 2, this wouldn't make sense.
    b. no records match.
05. ***'EnrollmentFinish' < 'LastPeriodStart'***
    a. Although it is possible for a customer to have recently renewed his contract and as such his *'EnrollmentFinish'* value hasn't changed since he hasn't yet finished his new enrollment, we assumed this attribute to automatically change based on the end of his recent renewed contract, making this argument incoherent if true.
    b. 2211 records match.
06. ***'DateLastVisit'* compared to enrollments and periods**
    a. This check was split into 4 phases, where we checked the feature in relation to both the start and finish of enrollment and period attributes.
    b. again a significant number of records were found, about 3100 together (~20% of the data).
07. **'RealNumberOfVisits' > 'AllowedNumberOfVisitsBySLA'**
    a. following the logic that his actual number of visits to the facility in the last active period shouldn't be superior to how many he is allowed.
    b. results only showed 48 records that matched, because the number is very small we removed them.

Concluding, most incoherences were resolved well, however, major unsolved incoherences that presented a large number of records had to do with features: *'EnrollmentFinish'* and both *'LastPeriodStart'* and *'LastPeriodFinish'*. Which in some cases had dates that didn't make sense, and the length of the represented periods wasn't constant, showing values that ranged further than 6 months. Overall for the purpose of the project, we thought these attributes weren't contributing much, and it was decided for them to be removed. In the end, **0.5%** of total data was removed.

3.3. Missing Values

For this step we started by checking which columns had missing values. By doing so, it was possible to see that 11 had missing records so we checked the percentage of NaNs for each of them.

01. '***Athletics', 'Water', 'Fitness', 'Team', 'Racket', 'Combat'* and *'SpecialActivites'***

- Given that the missing values for these features accounted for less than 1% of the total data of each column respectively, we opted to remove those records, as it would result in minimal data loss.

02. ***'Income', 'AllowedWeeklyVisitsBySLA', 'NumberOfFrequencies'*****
    - Since the missing values for these features represented more data loss if deleted, we opted to use the KNNImputer to replace the NaNs' with its closest 5 neighbors.

03. ***'Age'***
    - since age had too many unique values and they were all scattered, we decided to use the median to replace the missing values of these features.

3.4. Outliers

For the outliers we started by checking the percentage of outliers for each metric feature. Some features like ***'AttendedClasses'*** and ***'DaysWithoutFrequencies'*** had significant percentages of outliers (20.45% and 10.23% respectively) and the only feature that had no of outliers was ***'AllowedWeeklyVisitsBySLA'***

3.4.1 Outliers Removal.

We chose to manually remove outliers by inspecting boxplots (figure 9). To identify and exclude outliers, we established thresholds for most of them based on observations where outliers appeared to start getting more isolated and exhibited extreme values.

For the ***'DaysWithoutFrequencies'*** feature we opted to remove every record that hadn't visited the gym in 1025 days (3 years).

3.5. Feature Engineering

As for further feature engineering, we came to the conclusion that after eliminating both attributes that directly represented the last active periods, some features that referenced these periods like *'NumberOfFrequencies'* didn't have a proper reference to it, and so came up with a new feature we called *'LastActivePeriod'*, that would represent the last active semester which *'DateLastVisit'* pointed to, this way there was a direct reference to the last active period, in case it became relevant. Furthermore, binary features were changed to data type Int, instead of boolean.

Since most engineering was performed earlier, no further changes were made in this section.

3.6. Transformations

This stage involves preparing the data for analysis by applying scaling and encoding techniques. These transformations are crucial to ensure that the data is in a format suitable for models and other analytical methods.

3.6.1. Scaling

As for scaling, we analyzed each metric feature's distribution (Figure 9) and did research to determine which method would be most suitable for our data, we initially thought Min-Max would fit better, however couldn't make it work perfectly with cluster methods down the line, and as such decided to use the Standard scaler, which even improved some results during cluster method testing.

3.6.2. Encoding

After that, we moved on to a crucial step in our project – converting our categorical variables into dummy variables. We used a method called One-Hot-Encoding, which is essential for dealing with categorical data. It turns each category into its own column, representing it with binary values, 0s and 1s. Since most of our categorical data was already binary, we only had to apply this method to the ***'LastActivePeriod'*** variable.

3.7. Feature Selection

A crucial step that involves identifying and choosing the most relevant and informative features. The primary goal is to reduce the dimensionality of the data by selecting a subset of features that contribute the most to further modeling or analysis while excluding redundant or irrelevant ones.

During feature selection, we performed a correlation matrix using Pearson's method for both metric and non-metric features (now scaled and encoded respectively). For the metric features (Figure 5), we concluded that features *'Income'*, and *'Age'* were extremely correlated (0.86 correlation), and that feature *'LifetimeValue'* had a high correlation with 3 other attributes. We decided to remove both *'Age'* and *'LifetimeValue'*. For the non-metric features (Figure 6), the matrix didn't show any extreme values that were significant enough to drop any feature.

3.8. PCA

Principal Component Analysis is a helpful dimensionality reduction technique when you have lots of features that might be connected in some way. It finds new values for our data and splits those that are not connected or related, making things simpler.
After applying the model there were 3 principal components that had Eigenvalues superior to 1 and they managed to retain 68% of the overall variance of our data, which is a good value.
After the PCA Analysis was completed, we decided not to use the method, because of its complexity and difficult interpretability and since we don't have too many features, dimensionality reduction techniques are not extremely necessary.

4. Clustering

4.1. Segmentation

Before starting clustering our data, we decided to split our data into different perspectives. Looking at our data we opted for 2 different perspectives:

- Engagement Perspective: Features that focus on client engagement. Included features: '***AttendedClasses***', '***RealNumberOfVisits***', '***NumberOfFrequencies***' and '***DaysWithoutFrequency***'
- Service Perspective: Perspective centered on service-related metrics. Included Features: '***AllowedWeeklyVisitsBySLA***', '***AllowedNumberOfVisitsBySLA***' and '***NumberOfRenewals***'

Instead of using clustering methods using all the data, we implemented them to each perspective respectively and merged the perspectives for the results.

4.2. Methods Tested

4.2.1 Hierarchical clustering

Hierarchical clustering is a widely used method for grouping data points. Specifically, agglomerative hierarchical clustering begins with each data point as its own cluster. In each step, the algorithm combines two clusters based on how similar they are, and this process continues until all data points belong to a cluster. For the distance measure we used the Euclidean Distance and for the linkage we tested the complete, single, average, and ward methods and decided to use the ward method since it was the one with the best results.
This method resulted in the partition of the data into 6 clusters for the engagement perspective and 5 clusters for the service perspective.

4.2.2 KMeans Clustering

In K-means, each data point is assigned to the closest centroid, essentially placing it in a specific cluster if its distance to that cluster's center is shorter than to any other center.
For the engagement perspective, the silhouette score was better for 2 clusters but we wanted to have a bit more partition in our data so we decided to test the K-means method using 3 and 4 clusters, and the optimal result between the silhouette and r2 scores was achieved using 4 clusters
The silhouette for the service perspective gave better results for 10 clusters but we deemed 10 clusters as too many, so we decided to test the method using 4 and 5 clusters and in the end, the optimal number of clusters according to the silhouette and r2 scores was 5.

4.2.3 Mean Shift

Mean-Shift Clustering dynamically assigns data points to clusters by iteratively shifting them towards the mode, which represents the central tendency within a region.
For this approach at start we used quantile = 0.06 but the results gave 102 clusters for the engagement perspective and 64 clusters for the service perspective. So we did some research and we used the default value (0.3) and it gave us better results, so we decided to use the default quantile value. For the engagement perspective, the Mean Shift Method still gave us 22 clusters as the optimal number, whereas for the service product, it gave us 3 clusters.

4.2.4 DBSCAN

Unlike traditional clustering algorithms, DBSCAN does not require the number of clusters to be predefined. The method works by considering two important parameters: epsilon (eps) and minimum points (minPts). We started by drawing the graph that shows eps values in relation to neighbor distances and determined the right eps value to be 1.0 for the Engagement Segment (Figure 11) and 0.4 for the Service Segment (Figure 12). Unfortunately, DBSCAN was one of our worst-performing methods.

4.2.5 Gaussian Mixture Model

In a GMM, each cluster is represented by a Gaussian distribution with its own mean and variance parameters. Firstly we drew a graph depicting BIC and AIC values in relation to the number of components to be used in the method (both BIC and AIC are used to prevent overfitting by penalizing models with a larger number of parameters), and concluded that for our Engagement Segment, the optimal number of components was 3 (Figure 13) and 2 for Service Segment (Figure 14). GMM's results weren't the best and hence weren't used for the final model.

4.3. Merging Segments

In this stage, testing was already made in both segments created previously, for each clustering method, and the results are as follows:

Cluster Method	Silhouette	R^2
Hierarchical	0.28	0.64
Kmeans	0.51	0.55
Mean shift	0.54	0.53
Dbscan	0.70	0.09
GMM	0.13	0.21

Table 1: Cluster Method results (Engagement Segment)

Cluster Method	Silhouette	R^2
Hierarchical	0.53	0.81
Kmeans	0.55	0.82
Mean shift	0.42	0.48
Dbscan	0.30	0.82
GMM	0.49	0.46

Table 2: Cluster Method results (Service Segment)

After analyzing the results, we can see that for **Service** (Table 2), the clustering method that presented best results was Kmeans, and for **Engagement** (Table 1), although initially Mean Shift looks slightly better, it displays an optimal number of clusters of 22, which we assumed was too many and hence hard to interpret for later conclusions, thus going for Kmeans as well. With the optimal clustering methods decided, we then executed them for each segment, with the respective optimal number of clusters (4 for engagement, 5 for service), obtaining finally, the respective labels.
As for the type of merging, we tried both manual and hierarchical, for which we came to the conclusion that hierarchical performed better for our data as the merging looked more even across labels and easier to analyze. After applying the hierarchical

merging, the now merged labels were formed and attached to our final data frame, ready for final cluster analysis.

5. Cluster Analysis

Finally, we arrived at cluster analysis, in this stage, we performed profiling for each cluster, and analyzed which number of clusters was optimal for our final segmentation. For our base number of clusters, we used 4 (as depicted in Figure 1), and its results are shown in Figure 2, but we wanted to try with 3 and 5, whose results are shown in Figure 3 and 4 respectively. We ended up identifying clustering with 3 as being the most prominent for the context of our project, as we can see in the other examples (with 4 and 5) the clustering ends up piling in mostly 3 clusters, leaving the rest with barely any data. In terms of cluster characterization, we were able to describe all 3 clusters as following:

- **Cluster 1 - 'The Overconfidents'**
  - Represents most of the customers.
  - People who attend the least amount of classes, however, are somewhat active, and request services with most allowed visits.
- **Cluster 2 - 'Mr. Consistents**
  - Those who attend the most amount of classes, not very periodically active, however consistently attending the facility, low nr of weekly visits, and the ones who renew their membership the most.
- **Cluster 3 - ‘'The "Tried Once, didn't like it"'’**
  - Least engagement in the facility, and request services with a low number of weekly visits.

These descriptions of each cluster depict their distribution among each feature that can be seen in Figure 3. We believe these descriptions not only segment well the customers in XYZ Sports, but also the reality in such an industry, were we have people who get excited, and go to the gym, however, don’t last very long, people who are very consistent and are constantly renewing their contracts, and finally people who try it a few times but never quite end up frequenting the facility consistently.

6. Conclusion

In conclusion, this data mining project is a game-changer for XYZ Sports company. It helps understand their clients better by putting them into groups based on their behaviors, patterns and preferences. This means they can now offer personalized services, create targeted promotions, and make their gym experience more enjoyable for everyone, as well as making them more competitive and attractive in this business environment.

Understanding your customers and finding a way to connect with them and attend to their needs is a must for every successful company, a happy customer often equals a profitable company!

7. Bibliography

Python | Box-Cox Transformation. (2020, June 5). GeeksforGeeks. https://www.geeksforgeeks.org/box-cox-transformation-using-python/

Studio, V. D. (2021, July 27). Data Transformation and Feature Engineering in Python. Visual Design.https://www.visual-design.net/post/data-transformation-and-feature-engineering-in-python

Gusarova, M. (2023, February 5). Data Scaling and Skewness Handling. Medium. https://medium.com/@data.science.enthusiast/data-scaling-and-skewness-handling-c3800f7c9b0a

Clustering in Data Mining. (2020, October 13). GeeksforGeeks. https://www.geeksforgeeks.org/clustering-in-data-mining/

sklearn.cluster.estimate_bandwidth. (n.d.). Scikit-Learn. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.estimate_bandwidth.html

Anakin. (2020, November 7). Clustering with Python — Mean Shift. Medium. https://anakin297.medium.com/clustering-with-python-mean-shift-ec311e9698a

How Density-based Clustering works—ArcGIS Pro | Documentation. (n.d.). Pro.arcgis.com. Retrieved January 6, 2024, from https://pro.arcgis.com/en/pro-app/3.1/tool-reference/spatial-statistics/how-density-based-clustering-works.htm

Linoff, G. S., & Berry, M. J. A. (2011). Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management. In Google Books. John Wiley & Sons. https://books.google.pt/books?hl=pt-PT&lr=&id=AyQfVTDJypUC&oi=fnd&pg=PR37&dq=data+mining+best+practices&ots=KZFtwrQYyK&sig=QPFt28AGL741spifis51TyjX7tU&redir_esc=y#v=onepage&q=data%20mining%20best%20practices&f=false

Sharma, R. (2020, January 20). Cluster Analysis in Data Mining: Applications, Methods & Requirements. UpGrad Blog. https://www.upgrad.com/blog/cluster-analysis-data-mining/

Hayslett, M. (n.d.). LibGuides: Text and Data Mining: Best Practices. Guides.lib.unc.edu. Retrieved January 6, 2024, from https://guides.lib.unc.edu/tdm/best_practices

Lopez Yse, D. (n.d.). *Introduction to K-Means Clustering*. Pinecone. Retrieved January 7, 2024, from https://www.pinecone.io/learn/k-means-clustering/

Jain, S. (2023, January 23). *ML | Mean-Shift Clustering*. GeeksforGeeks. Retrieved January 7, 2024, from https://www.geeksforgeeks.org/ml-mean-shift-clustering/

8. Annex

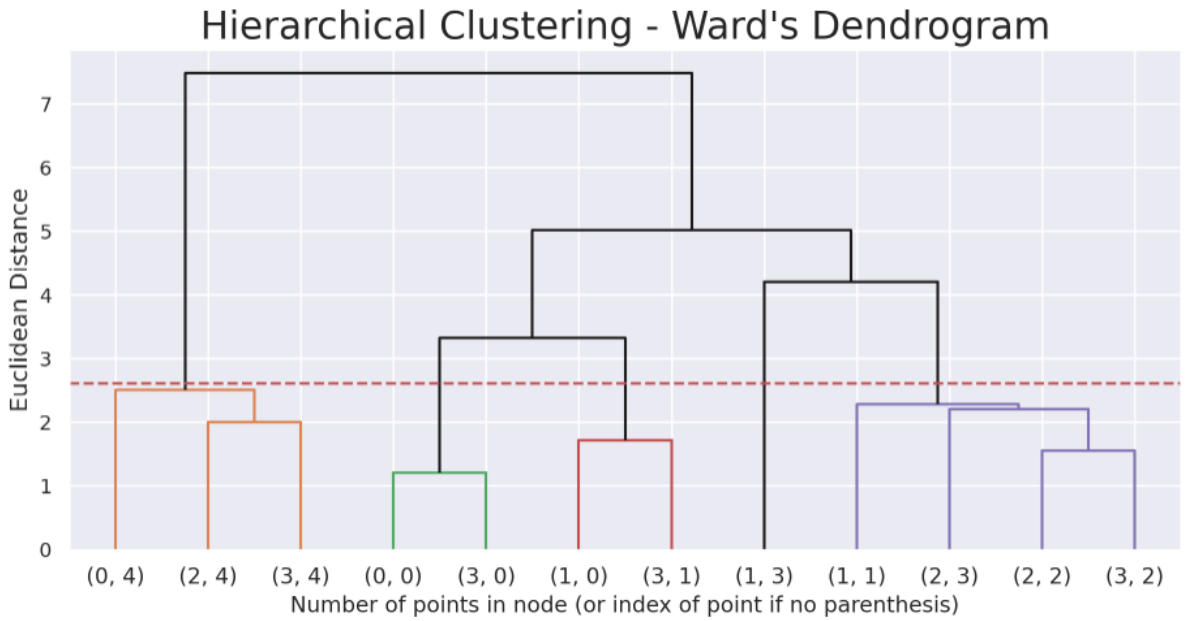

Figure 1: Ward's Dendogram of final segment merging

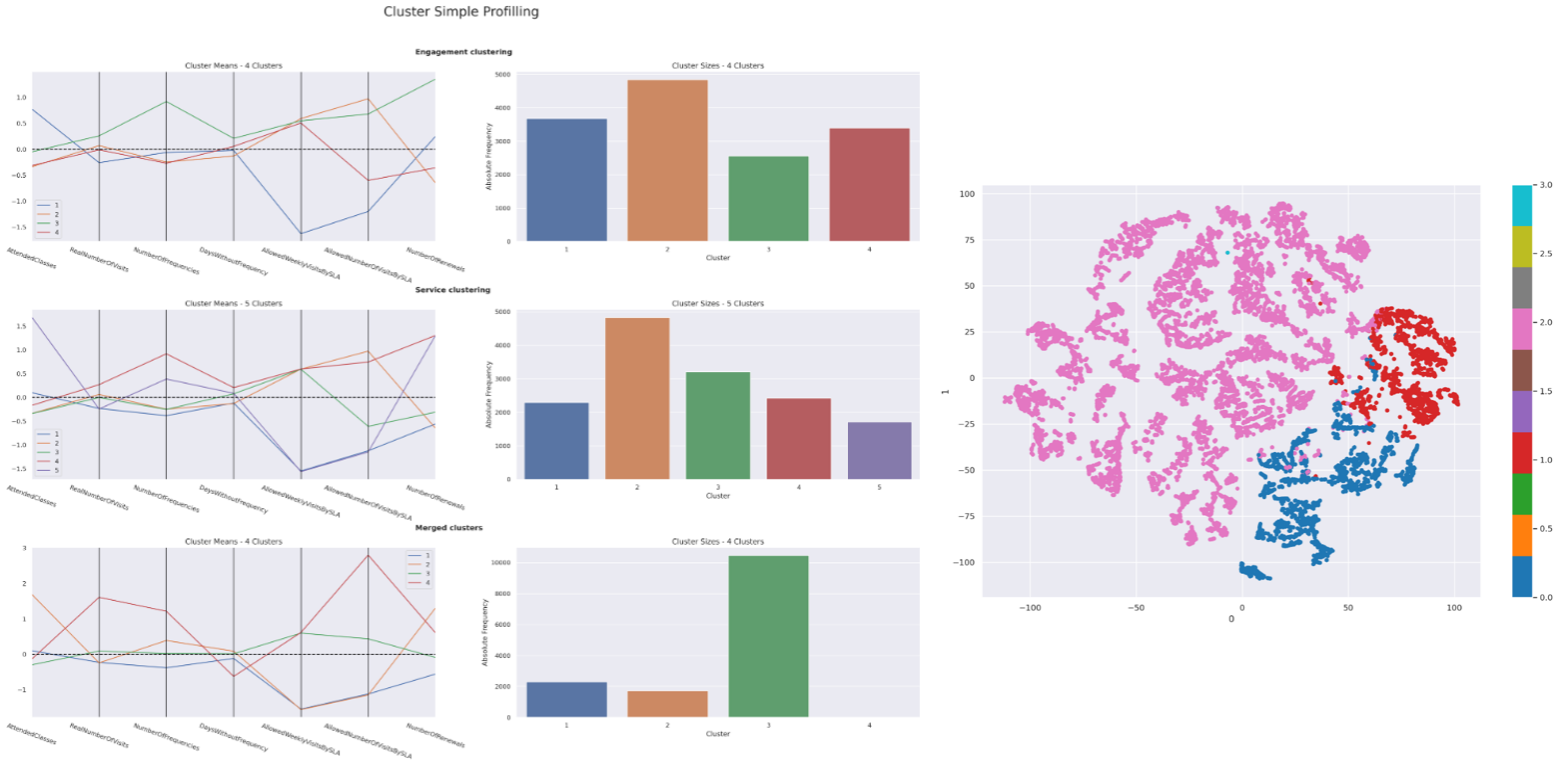

Figure 2: Profiling and t-SNE with 4 clusters

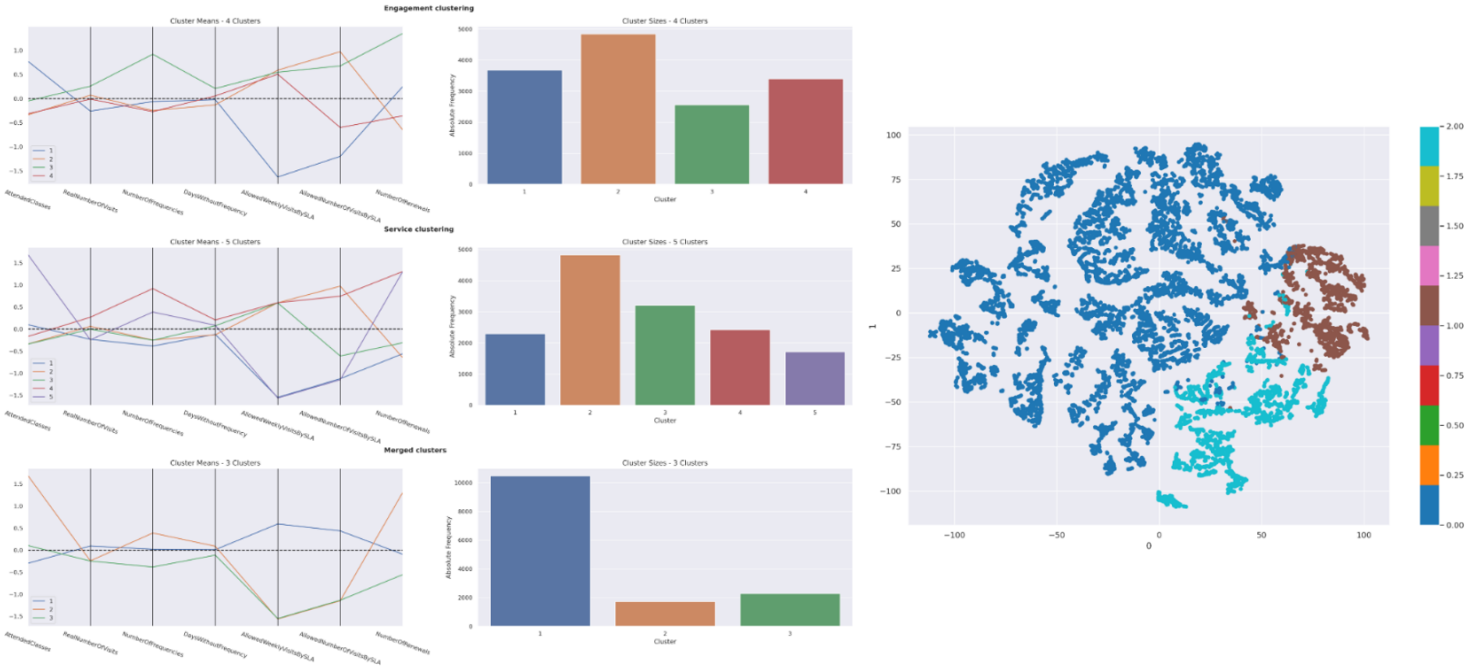

Figure 3: Profiling and t-SNE with 3 clusters

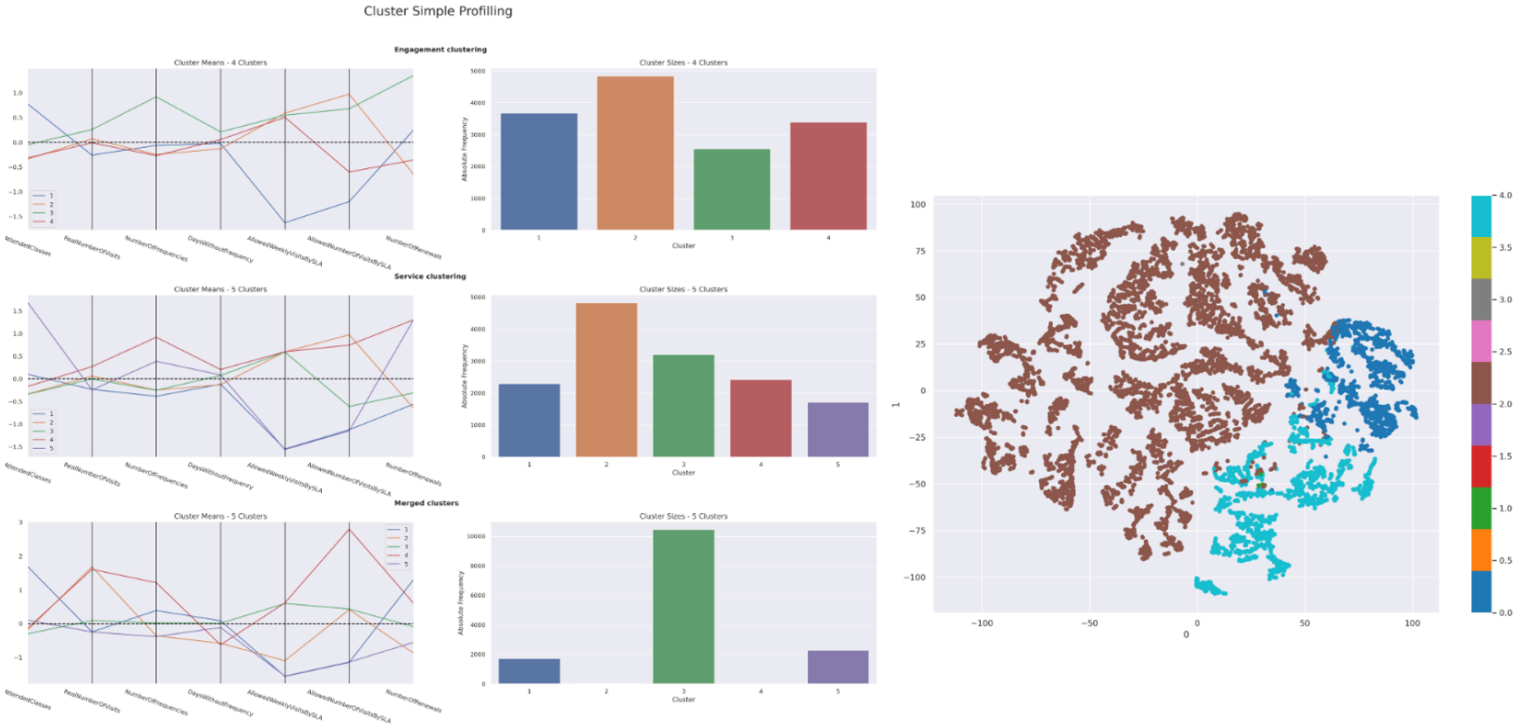

Figure 4: Profiling and t-SNE with 5 clusters

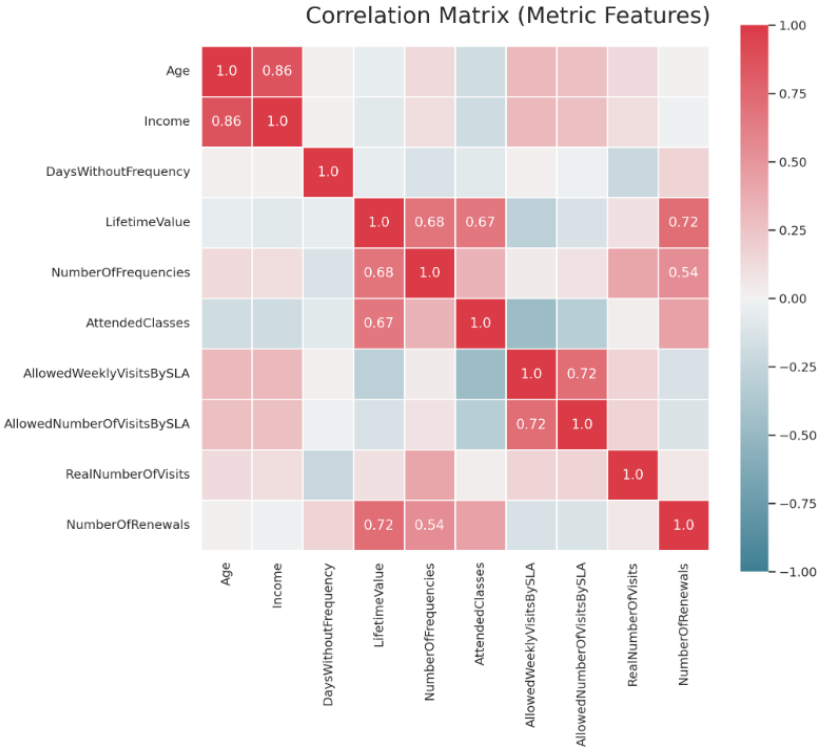

Figure 5: Correlation Matrix (Metric Features)

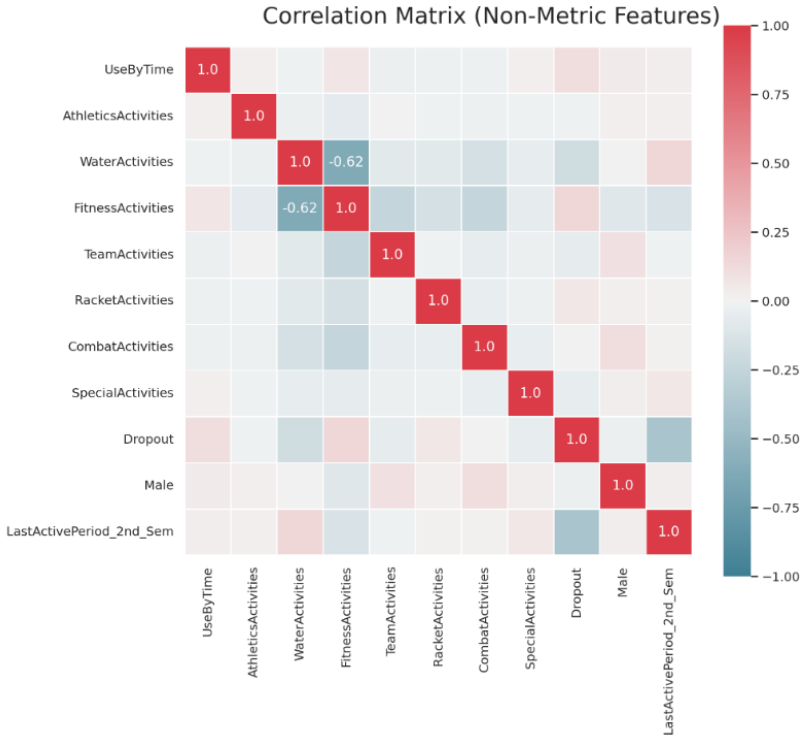

Figure 6: Correlation Matrix (Non-Metric Features)

	count	mean	std	min	25%	50%	75%	max
Age	14941.0	26.016732	14.156592	0.00	19.00	23.00	31.00	87.00
Income	14446.0	2230.970511	1566.471988	0.00	1470.00	1990.00	2790.00	10890.00
DaysWithoutFrequency	14941.0	81.227629	144.204026	0.00	13.00	41.00	84.00	1745.00
LifetimeValue	14941.0	302.577212	364.326932	0.00	83.60	166.20	355.10	6727.80
UseByTime	14941.0	0.047119	0.211900	0.00	0.00	0.00	0.00	1.00
AthleticsActivities	14905.0	0.007380	0.085593	0.00	0.00	0.00	0.00	1.00
WaterActivities	14904.0	0.296162	0.456579	0.00	0.00	0.00	1.00	1.00
FitnessActivities	14906.0	0.576077	0.494195	0.00	0.00	1.00	1.00	1.00
DanceActivities	14905.0	0.000000	0.000000	0.00	0.00	0.00	0.00	0.00
TeamActivities	14906.0	0.055548	0.229055	0.00	0.00	0.00	0.00	1.00
RacketActivities	14904.0	0.023417	0.151227	0.00	0.00	0.00	0.00	1.00
CombatActivities	14908.0	0.107929	0.310301	0.00	0.00	0.00	0.00	1.00
NatureActivities	14894.0	0.000000	0.000000	0.00	0.00	0.00	0.00	0.00
SpecialActivities	14897.0	0.026515	0.160668	0.00	0.00	0.00	0.00	1.00
OtherActivities	14906.0	0.001878	0.043302	0.00	0.00	0.00	0.00	1.00
NumberOfFrequencies	14915.0	40.122293	65.468305	1.00	7.00	18.00	45.00	1031.00
AttendedClasses	14941.0	10.152667	29.155167	0.00	0.00	0.00	3.00	581.00
AllowedWeeklyVisitsBySLA	14406.0	5.759614	2.118931	1.00	4.00	7.00	7.00	7.00
AllowedNumberOfVisitsBySLA	14941.0	41.636121	21.066860	0.56	25.72	38.99	60.97	240.03
RealNumberOfVisits	14941.0	5.320394	6.333055	0.00	1.00	4.00	7.00	84.00
NumberOfRenewals	14941.0	1.205274	1.381350	0.00	0.00	1.00	2.00	6.00
HasReferences	14929.0	0.019894	0.139641	0.00	0.00	0.00	0.00	1.00
NumberOfReferences	14941.0	0.022288	0.166783	0.00	0.00	0.00	0.00	3.00
Dropout	14941.0	0.800950	0.399299	0.00	1.00	1.00	1.00	1.00

Figure 7: Descriptive Statistics numeric variables

	count	unique	top	freq
Gender	14941	2	Female	8930
EnrollmentStart	14941	1490	2015-03-02	92
EnrollmentFinish	14941	1300	2015-09-16	1684
LastPeriodStart	14941	12	2019-07-01	3171
LastPeriodFinish	14941	11	2019-12-31	3693
DateLastVisit	14941	1384	2019-10-31	475

Figure 8: Descriptive Statistics datetime and gender variables

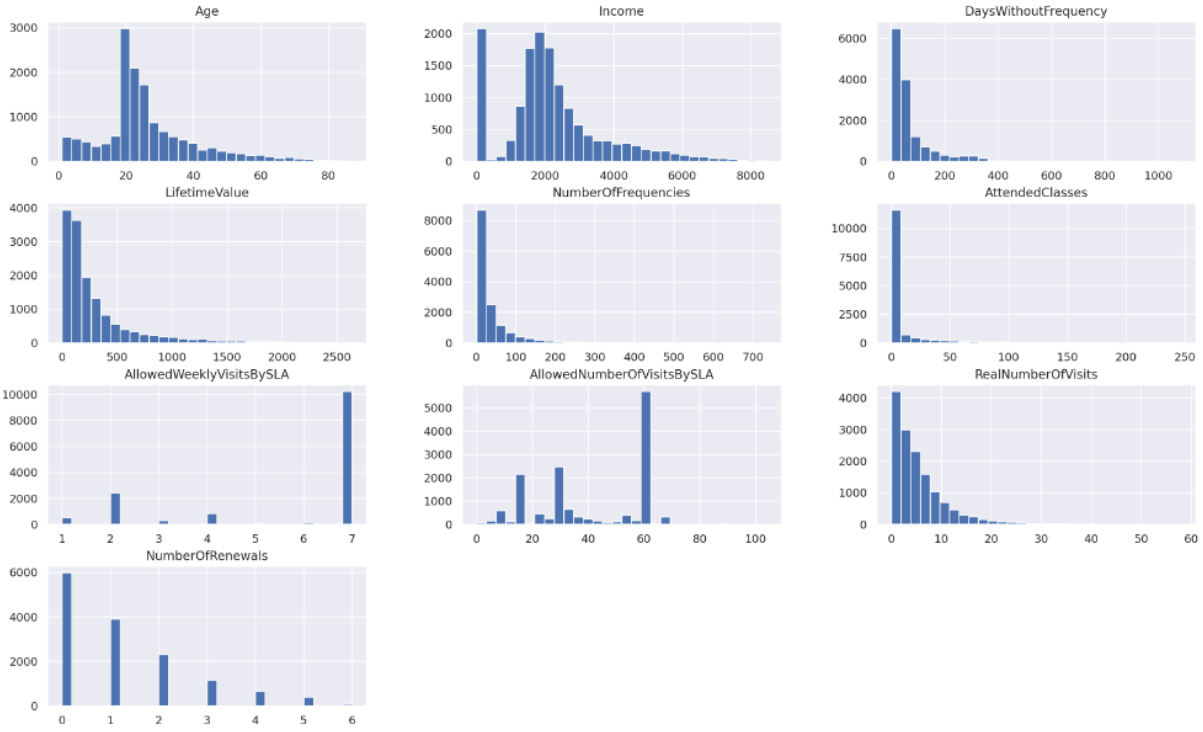

Figure 9: Numeric variables distributions

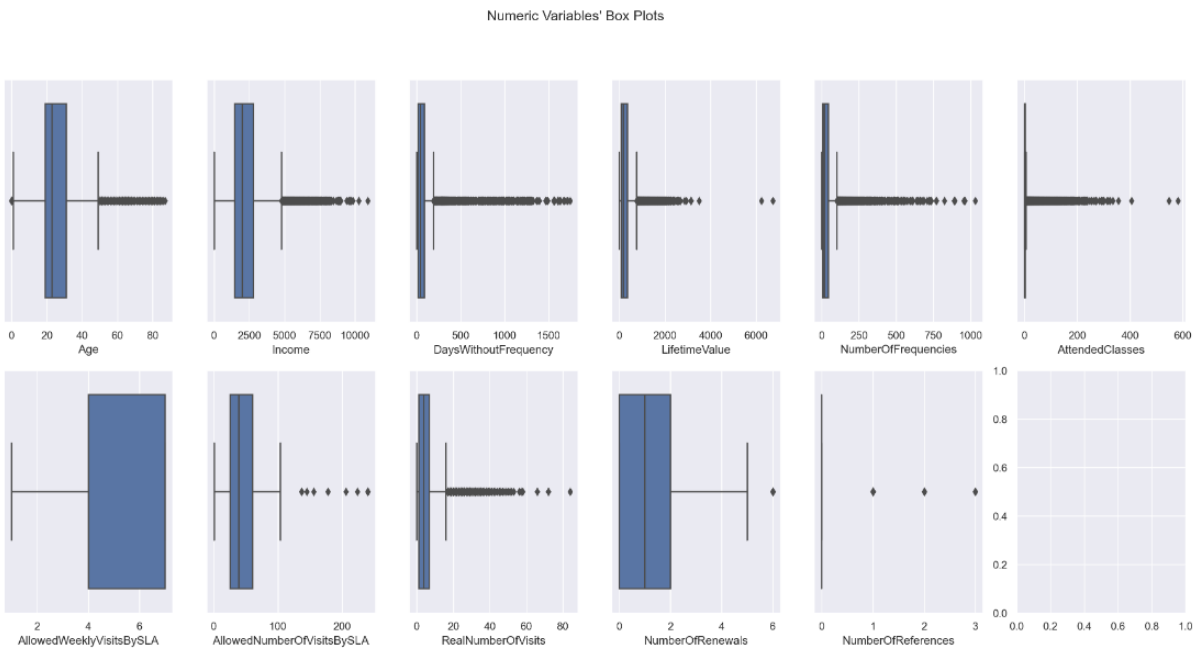

Figure 10: Numeric Variables' Box Plots

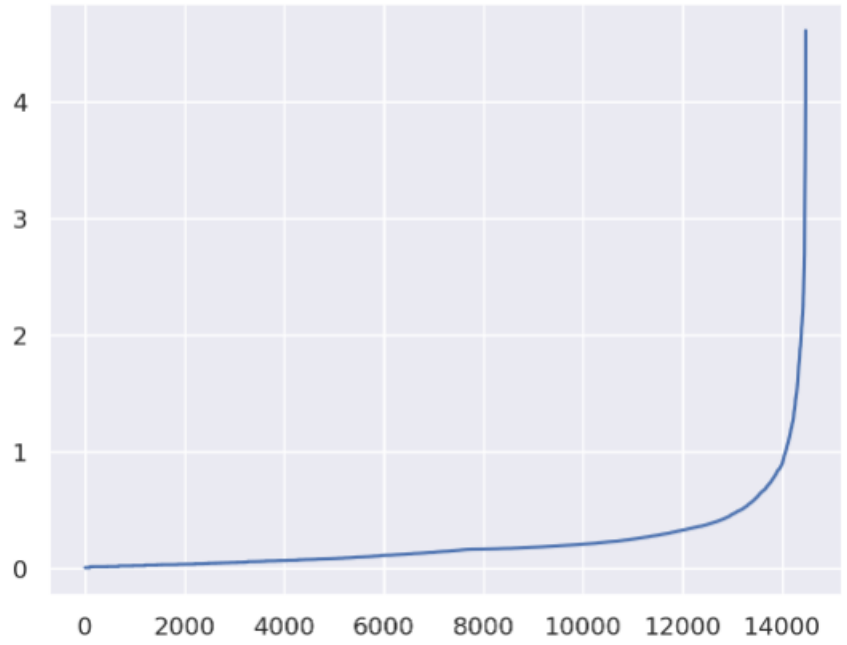

Figure 11: DBSCAN eps value in relation to neighbour distance for Engagement Perspective

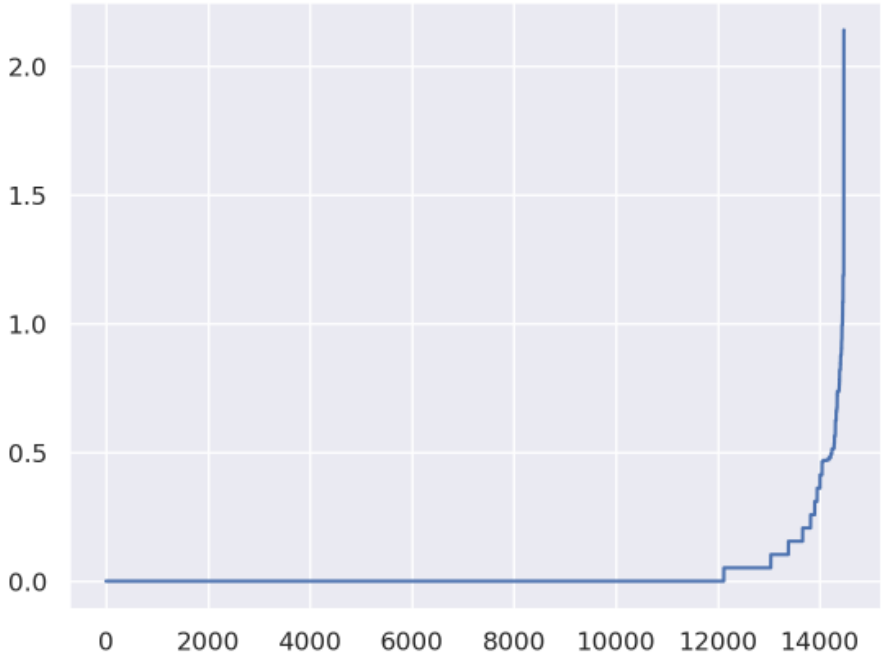

Figure 12: DBSCAN eps value in relation to neighbour distance for Service Perspective

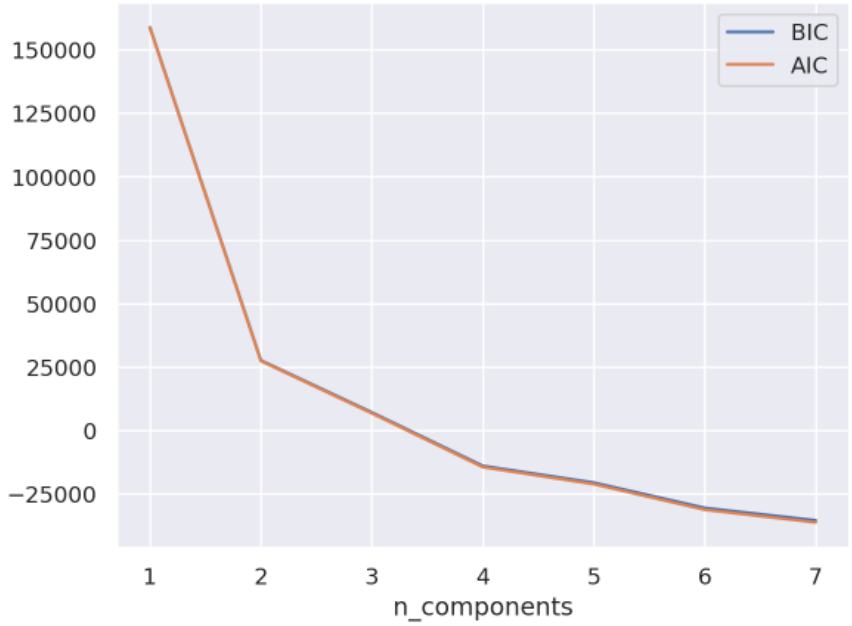

Figure 13: Gaussian mixture model for Engagement Perspective

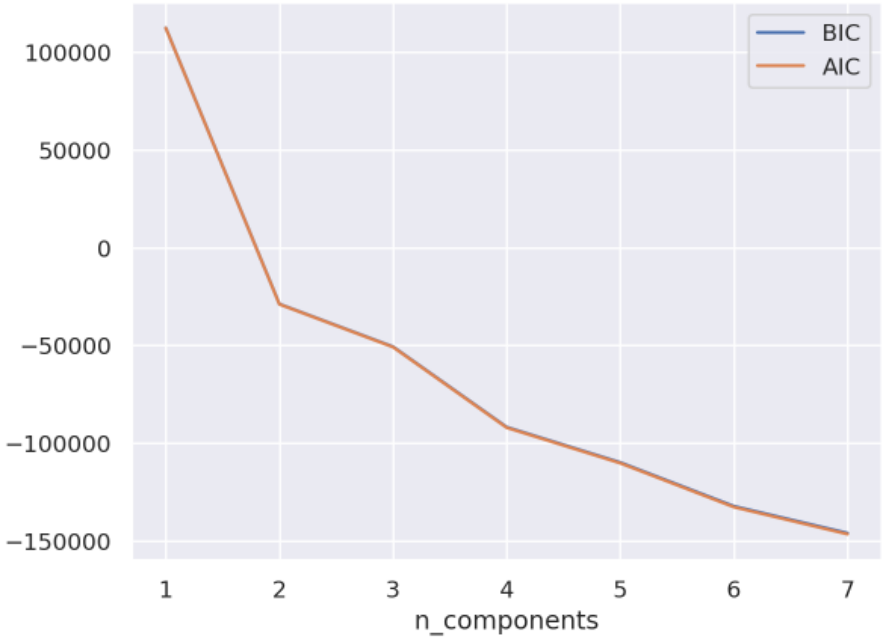

Figure 14: Gaussian mixture model for Service Perspective