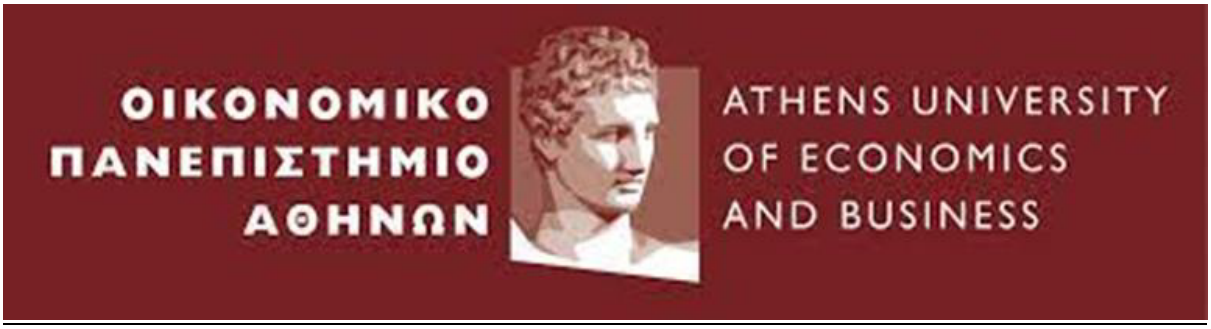


ATHENS UNIVERSITY OF ECONOMICS & BUSINESS
DEPARTMENT OF MANAGEMENT, SCIENCE & TECHNOLOGY
MSc BUSINESS ANALYTICS

Statistics for Business Analytics II

"To Churn or not to Churn"

Full Name: ATHANASIOS ALEXANDRIS
Register Number:p2822202

ATHENS, 2023

TABLE OF CONTENTS

1.INTRODUCTION

1.1 Description of the problem

In this report we had to work on the telecommunications company that we have worked on, in the first assignment. This report focuses on two different parts.

In the first part of this project we are assigned to make a model for predicting the customers of the company that are about to churn, so these customers to be offered a better deal in order to avoid their churn. So our model will classify the customers, to the categories "churn" or "not to churn" (pre-defined classes). For this task we used many the classifications methods 'Decision Tree', 'Random Forest', 'Support Vector Machines' and 'Naive Bayes'. The metrics that drove us to select the most appropriate one were the accuracy, Adjusted Rand Index and the ROC Curve, which they are all explained in the first part.

In the second part, we had to cluster the company's users based on their usage characteristics, so the marketing department to use different strategies for each cluster. For this task we had first to select the appropriate variables that we had to take into consideration, and then to use the Hierarchical Clustering method and the K-Means method. The metric that drove us to select the most appropriate one, was the average silhouette width, of each clustering we made.

1.2 DATASET

The dataset that we used for our study is the same with first assignment's project. The dataset contains 3.333 records and 21 variables. The response variable for the first task is "churn" that indicates whether the customer leaves the company or not. There are also 20 independent variables, which have been taken into consideration for both classification and clustering parts.

2. PART 1 - CLASSIFICATION

2.1 VARIABLES SELECTION

Before we proceed with the various classifications methods, it is important to isolate only the variables of our dataset, that are important for our predictive models. For the selection process, we first created a model through logistic regression method, that has the following form:

logit(p) = b0 + b1X1 + b2X2 + ... + bnXn

logit(p)= ln(p/(1-p))

p = The probability of churn for a customer

X1, X2,….,Xn = Independent variables

With the implementation of the GLM function in R, we constructed our first model, which involves all variables of our dataset. The summary of the first model indicates a great number of variables that are not statistically significant (P-value> 0.05 for Ho: b1=0, b2 =0 ... bn = 0). The ratio $\frac{Residual\ Deviance}{\text{Degrees of freedom}}$ = 0.6355 and the AIC = 2210.9.

The next step of our procedure, was to implement the LASSO method, exactly they way we have done it in the first project, as it performs well in large data sets and removing all the unnecessary variables without important loss of information.

Following exactly the same steps, we ended up to the variables presented in Table 1, which we used to as input to our second model.

Table 1

Day.Mins	Eve.Mins	Night.Mins	Intl.Mins	CustServ.Calls
Int.l.Plan	VMail.Plan	Day.Charge	Eve.Charge	Night.Charge
Intl.Calls	Intl.Charge	State		

The model created had ratio $\frac{Residual\ Deviance}{\text{Degrees of freedom}}$ = 0.6505 and AIC = 2204.2, with 13 variables. (Steps explained in the first assignment)

Consequently, we decided to create a third model, using the AIC method on the full model, which is more appropriate for predictions, in order to end up to the most effective model. We ended up with a model with 9 variables, where the AIC has its minimum value (2181.6). The variables selected are presented in the Table 2.

Table 2

VMail.Message	Int.l.Plan	Night.Charge
Eve.Mins	VMail.Plan	Intl.Calls
CustServ.Calls	Day.Charge	Intl.Charge

The model created had ratio $\frac{Residual\ Deviance}{\text{Degrees of freedom}}$ = 0.6505 and AIC = 2181.6, with 9 variables. (Steps explained in the first assignment).

Comparing the two models, we observe that the AIC model has both lower AIC and ratio Res. Deviance/ Deg. of freedom more close to 1. Thus the variables that we select to use as input to the various classification methods that we perform in the next steps, are the variables of the AIC model which are presented on Table 2.

2.2 SPLIT DATASET – K FOLD CROSS VALIDATION

Before the implementation of each classification method, we split our dataset in the 'Train' dataset, which were used to train our models, and the 'Test' dataset, which were used in order to evaluate each model. More particularly, the data of the 'Test' were used in order to predict the response variable, which in our case is whether the customer will churn or not. The whole process of splitting the dataset in two parts, training the models with the 'Train' dataset and testing them through the 'Test' dataset, was repeated six times, as we decided to use the K-Fold cross validation method (K=6). The idea is to split the data into K equally sized subsets, and then train the model K times, each time using a different fold as the validation set and the remaining folds as the training set. The performance of the model is then evaluated by averaging the performance metrics (accuracy, Adjusted Rand Index) obtained from the K runs.

2.3 PERFORMANCE METRICS

The metrics we used to evaluate the performance of the classifications methods that we implemented are the accuracy, the Adjusted Rand Index and the ROC (Receiver Operating Characteristic) curve.

Accuracy, is the metric that shows us the percentage of predictions that are correct. In our case we calculate the accuracy of every method for each one of the six folds, and as final accuracy we consider the average accuracy of the six folds.

In the context of Adjusted Rand Index, the predicted labels are treated as clusters, and the true labels are treated as the ground truth clusters. The ARI is then computed to measure the similarity between the predicted clusters and the ground truth clusters.

The ROC curve is a graphical representation of the performance of a binary classifier. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at different classification thresholds. The TPR is the fraction of positive instances that are correctly identified by the classifier, while the FPR is the fraction of negative instances that are incorrectly classified as positive by the classifier. What we are interested in, from the ROC curve, is the area under the curve (AUC), which represents the overall performance of the model. A high AUC value close to 1 indicates that the model has high predictive power and accuracy. So for the condition for the AUC is the closer to 1 the better model.

2.4 CLASSIFICATION METHODS

METHOD 1 - DECISION TREE

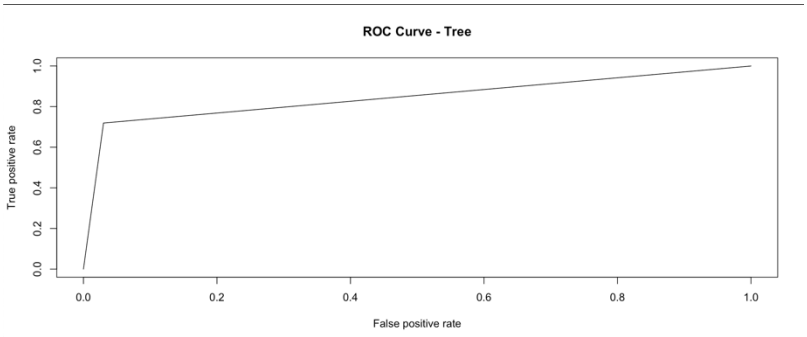
The second method we used, is the 'Decision Tree' method. A decision tree is a type of supervised learning algorithm used to make predictions or decisions based on data. The basic idea is to use a tree-like structure to model a set of possible decisions and their likely outcomes. To build a decision tree, the algorithm starts with a single node, which represents the entire dataset. It then looks at each of the input variables and finds the one that best separates the data into different groups based on the target variable. The decision tree can then be used to make predictions for new data points by following the path from the top of the tree to the appropriate leaf node. One of the nice things about decision trees is that they can be very intuitive and easy to understand, since they mirror the way we might make decisions in our everyday lives. However, it can be prone to overfitting, where the model is too complex and fits the training data too closely, resulting in poor generalization performance on new data.

We have created a decision tree using the data from our training dataset.

Regarding the predicting ability of the Decision Tree model the average accuracy of the 6 folds is 93.15% , and the Adjusted Rand Index is 63.63% .

In the graph below (Figure 1) we have designed the ROC Curve for the Decision Tree model, with AUC equals to 0.8445 indicating that the classifier has high predictive power and accuracy.

Figure 1 – ROC Curve Decision Tree



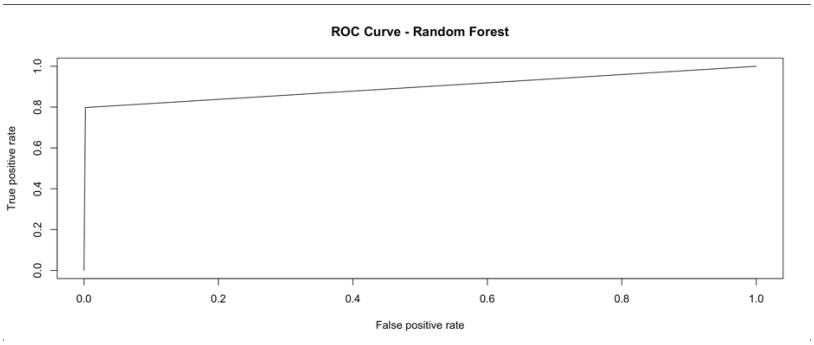
METHOD 2 – RANDOM FOREST

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. Each decision tree in a Random Forest is constructed using a random subset of the training data and a random subset of the input features. This randomness helps to reduce overfitting and increase the accuracy of the model. During prediction, the class predicted by each decision tree is tallied, and the class with the most votes is assigned as the final prediction. Random Forest is designed to reduce overfitting, which is a common problem with decision trees. By constructing multiple trees using random subsets of the data and features, Random Forest can produce accurate predictions without relying too heavily on any single decision tree.

Regarding the predicting ability of the Random Forest model the average accuracy of the 6 folds is 95.65% , and the Adjusted Rand Index is 76.22% .

In the graph below (Figure 2) we have designed the ROC Curve for the Random Forest model, with AUC equals to 0.8978 indicating that the classifier has high predictive power and accuracy.

Figure 2 – ROC Curve Random Forest



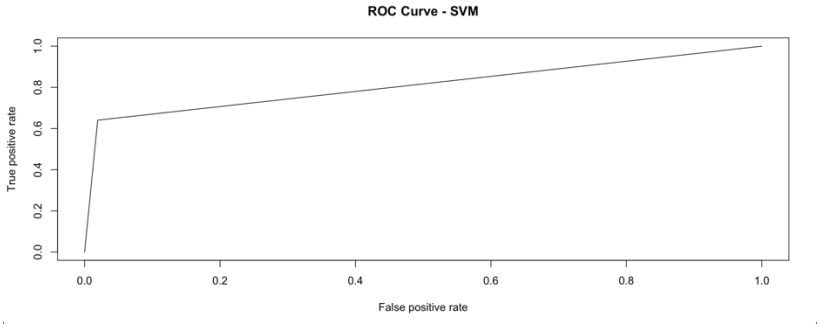
METHOD 3 - SUPPORT VECTOR MACHINES

SVMs are a type of machine learning algorithm that can be used to classify data into different categories. The goal of SVMs is to find the best way to draw a line or boundary (called the hyperplane) between different groups of data. This line or boundary should maximize the distance between the closest points in each group of data (margin), which in our case are the customers that will churn and the customers that will not churn. The SVM algorithm tries to find the hyperplane that has the largest margin.

Regarding the predicting ability of the SVM model the average accuracy of the 6 folds is 93.03% , and the Adjusted Rand Index is 60.35% .

In the graph below (Figure 3) we have designed the ROC Curve for the SVM model, with AUC equals to 0.8106 indicating that the classifier has high predictive power and accuracy.

Figure 3 – ROC SVM



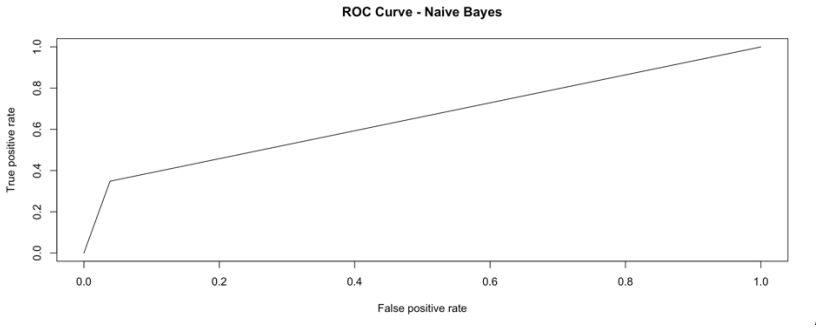
METHOD 4 – NAIVE BAYES

Naive Bayes is a probabilistic algorithm that uses the probabilities of features and their relationship to predict the class labels of new data points. The algorithm works by calculating the probability of each class given the values of the features, and then selecting the class with the highest probability as the predicted class. The probabilities are calculated using Bayes' theorem, which states that the probability of a hypothesis (in this case, the class label) given the evidence (the feature values) is proportional to the probability of the evidence given the hypothesis, multiplied by the prior probability of the hypothesis. The simplicity of its algorithm that allows it to be trained quickly, making it a good choice for large datasets. The probabilities calculated by Naive Bayes can be easily interpreted and used to gain insights into the data.

Regarding the predicting ability of the Naive Bayes model the average accuracy of the 6 folds is 87.23% , and the Adjusted Rand Index is 29.11% .

In the graph below (Figure 4) we have designed the ROC Curve for the Naive Bayes model, with AUC equals to 0.6548 indicating that the classifier has not that good predictive power and accuracy.

Figure 4 – ROC Curve Decision Naïve Bayes



2.5 CLASSIFICATION METHOD SELECTION

In order to decide the most appropriate classification method for our case, we had to compare the values of the three metrics discussed in the beginning of the chapter, which are the accuracy, the Adjusted Rand Index, and the ROC curve. The table presented below (Table 3) , indicates the average accuracy and Adjusted Rand Index for each method. As we can observe from the Table 3 the most efficient method is the Random Forest, as it has the highest values in both Accuracy and Adjusted Rand Index. This is also obvious, from the box plots of the two metrics (Figure 5 & Figure 6).

Table 3

Method	Accuracy	Adjusted Rand Index
Decision Tree	93.15%	63.63%
Random Forest	95.65%	76.22%
Support Vector Machines	93.03%	60.35%
Naive Bayes	87.23%	29.11%

Figure 5 – Accuracy Boxplot

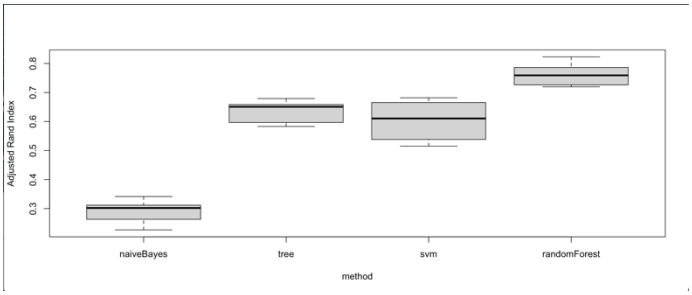
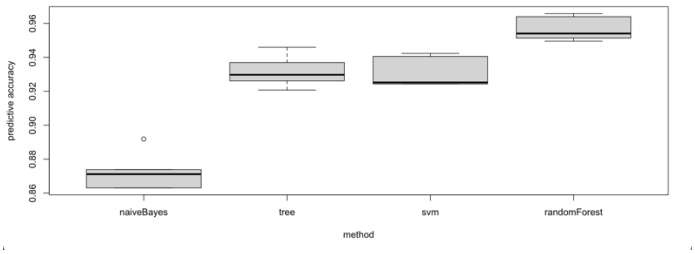
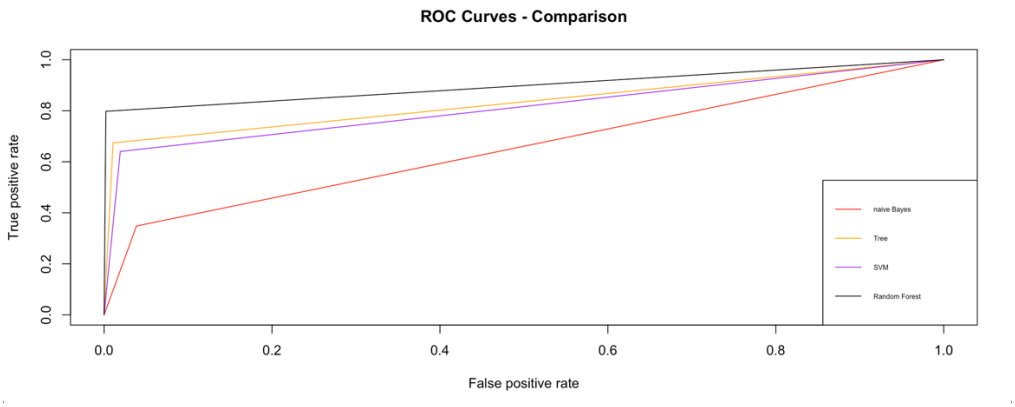


Figure 6 – Adjusted Rand Index Boxplot



Additionally, we designed a plot for the ROC curves of all four methods that we implemented, which once more indicates that the Random Forest is the most efficient method, with AUC equals to 0.8978 indicating that the classifier has high predictive power and accuracy. The methods Decision Tree and SVM have also good curves and high AUC, but smaller than Random Forest's.

Figure 7 – ROC Curves Comparison



3. PART 2 – CLUSTERING

3.1 INTRODUCTION

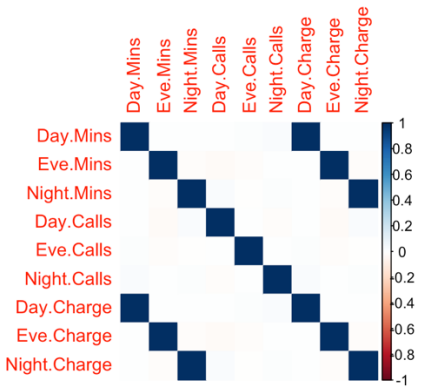
In the second part of this project, we had to cluster the customers of the company based on their usage behavior. For this task we transformed our dataset appropriately and then used the K-Means method

3.2 DATASET TRANSFORMATION

In the second part of this project, we had to cluster the customers of the company based on their usage behavior. Hence, we had to isolate only the variables that are related to usage. These variables were, the calling minutes, number of calls, and their respective charge, for customer separated to the hour zone (day, evening and night).

The next step of our procedure was to find any correlation between the variables we selected. Creating a correlation plot presented below (Figure 8), we identified that the charge variables have high correlation with the number of calls variables. Thus, we decided to remove the charge variables, in order not to have any correlation in our dataset, since correlation between variables can affect the similarity metric used to cluster the data points, and thus the resulting clusters.

Figure 8 – Correlation Plot



As our final dataset contained only the variables related to the minutes and to the number of various calls, we decided to transform our dataset keeping only two variables, that contained the sum of minutes and number of calls accordingly. Actually, we tried many different transformations to our dataset but this one was performing better on the various clustering methods that we implemented.

Finally, in order to be able to use the Euclidean distance we had to scale the two variables of our dataset. Scaling ensures that all variables are on the same scale, making them comparable and consistent in their influence on the clustering result. Without scaling, variables with larger scales will have a greater influence on the clustering result, regardless of their actual importance. Clustering algorithms are based on distance measures, and variables that are on

different scales can have a disproportionate influence on the clustering result. Scaling can also help to normalize the data, making it easier to compare and interpret the clustering results. Practically, what we did is to subtract the mean of each variable to each data point and divide by the standard deviation ($z = (x - \mu) / \sigma$).

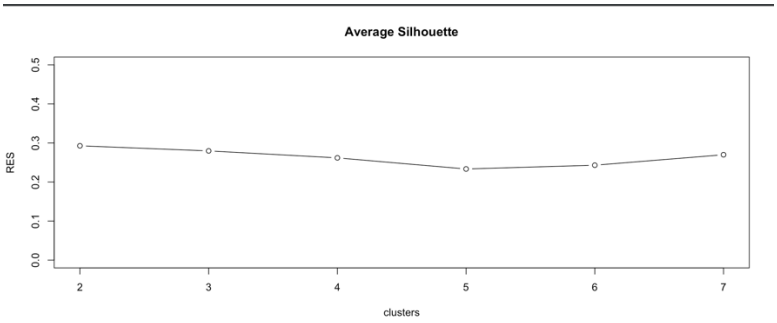
3.3 HIERARCHICAL CLUSTERING

Hierarchical clustering is a clustering algorithm that groups similar objects into clusters based on their pairwise similarities or distances. The algorithm starts by treating each object as its own cluster, and then iteratively merges the two closest clusters together until all objects belong to a single cluster. The types of hierarchical clustering are agglomerative and divisive. Agglomerative hierarchical clustering is the more commonly used type, and it starts by treating each object as a separate cluster and then iteratively merges the two closest clusters together until all objects belong to a single cluster. Divisive hierarchical clustering, on the other hand, starts with all objects in a single cluster and then iteratively splits the cluster into smaller clusters until each object is in its own cluster.

In our case we implemented agglomerative clustering with 'Ward method', that aims to minimize the sum of squared distances within each cluster, which tends to produce compact, spherical clusters.

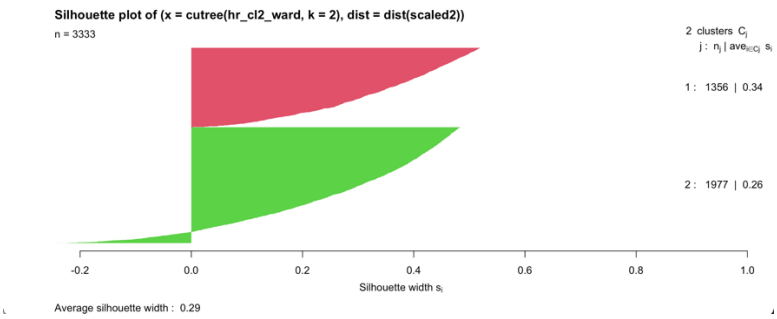
In order to decide the optimal number of clusters, we implemented average silhouette plot (Figure 9). By observing that plot we understand that the optimal number of clusters is two.

Figure 9 – Average Silhouette



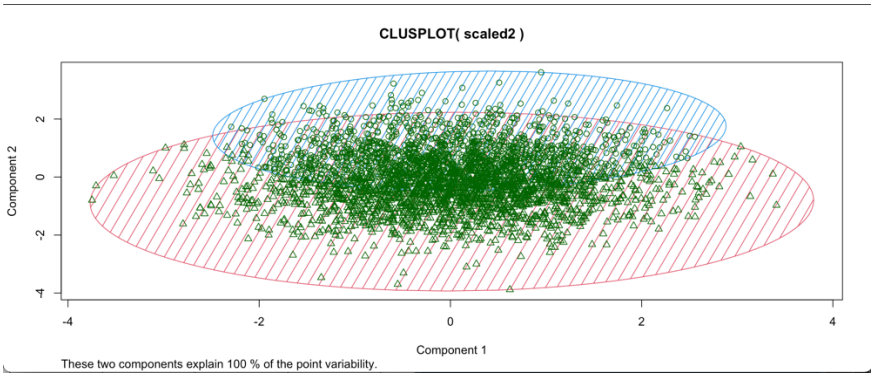
Then we plot the Silhouette values for the two clusters (Figure 10). The average silhouette width is 0.29. The first cluster has 1356 observations and average silhouette width 0.34, and the second has 1977 observations and average silhouette width 0.26 . It must be noted that the second cluster has the some observations with negative silhouette values and is the cluster with the lowest average silhouette width. This could mean that this cluster is not so "clear" and many observations belonging to it should not be there.

Figure 10 – Silhouette widths of the 2 clusters' observation



Finally we design a plot with the two clusters (Figure 11). As we see the majority of the observations are gathered in the center so it was very difficult to avoid the overlapping between the clusters.

Figure 11



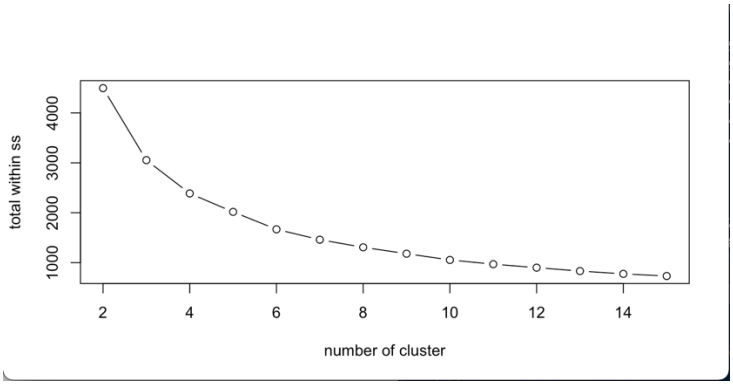
3.4 K-MEANS

The K-means algorithm is an unsupervised learning algorithm, which means it is used to analyze data that doesn't have predefined labels or categories. The goal of the algorithm is to group similar data points together into clusters based on their similarity. The algorithm starts by randomly selecting K centroids from the data points, where K is the number of clusters we want to create. Each data point is then assigned to the nearest centroid based on the Euclidean distance between the data point and the centroid. Once each data point is assigned to a centroid, the algorithm recalculates the centroids as the mean of all the data points assigned to each centroid. Then the algorithm repeats the assignment and update steps until the centroids no longer move significantly or the maximum number of iterations is reached. At this point, the algorithm has converged, and the final centroids represent the center of each cluster.

In order to decide the optimal number of clusters, we implemented the elbow method. The method involves plotting (Figure 12) the within-cluster sum of squares (WCSS) against the number of clusters, and identifying the "elbow" point on the plot where the rate of decrease in WCSS starts to level off.

By observing the Elbow plot below, we can not identify at which point we have an elbow. Actually we observe two elbow points in 3 and 4.

Figure 12 – Elbow



In order to be sure for the optimal number of clusters we did the plots of the Silhouette values, for three and four clusters, to see which one have better average Silhouette width.

The silhouette value for a single observation is a measure of how similar it is to its own cluster compared to other clusters in the dataset. The silhouette value ranges from -1 to 1, where:

- A value of -1 indicates that the observation is probably assigned to the wrong cluster.
- A value of 0 indicates that the observation is on the boundary between two clusters.
- A value of 1 indicates that the observation is well-clustered and clearly belongs to its own cluster.

The average silhouette value across all observations in a cluster is used as a measure of the quality of the clustering results. A high average silhouette value indicates that the clustering is good, while a low value indicates that the clustering may not be optimal.

In our case, by observing the silhouette plots for three and four clusters (Figure13 & Figure14), we observe that the three clusters, have higher average silhouette width, so we consider this as the optimal number of clusters.

As we see from (Figure13), the first cluster consists of 1152 observations and has an average silhouette width of 0.32, making it the cluster with the lowest average silhouette width and the biggest cluster. The second cluster consists of 1138 observations, and has an average silhouette width of 0.34, which also happens to be the highest out of all clusters. The third cluster consists of 1043 observations and has an average silhouette width of 0.33. We see that none of the clusters has negative silhouette values.

Figure 13 – Silhouette widths of the 3 clusters' observation

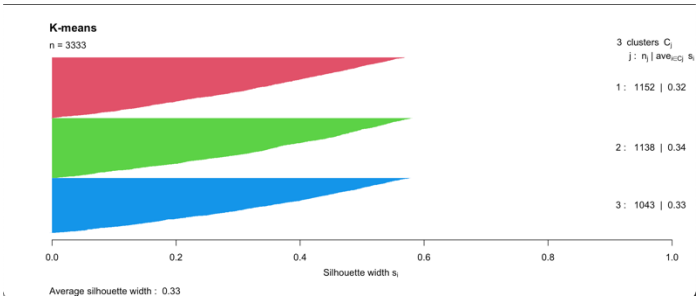
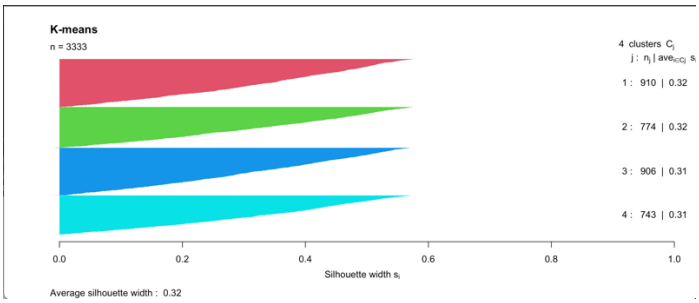
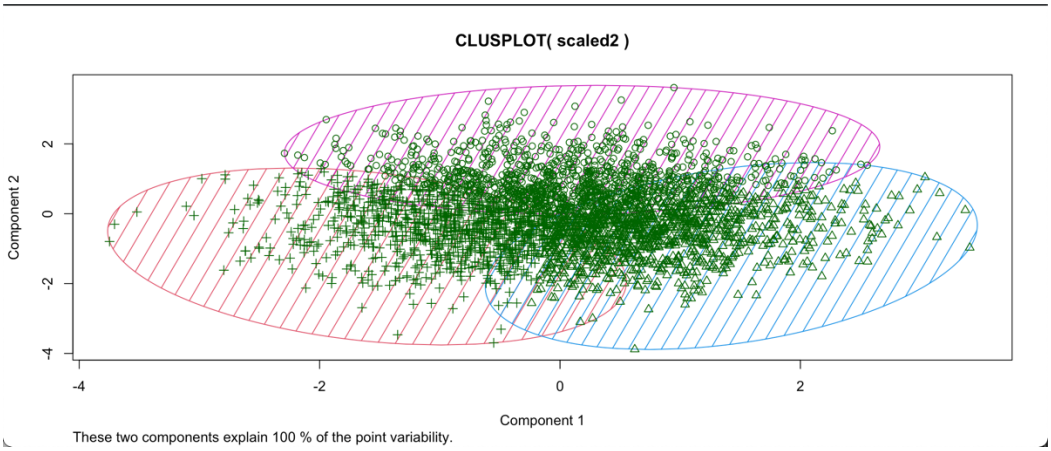


Figure 14 – Silhouette widths of the 4 clusters' observation



Finally we design a plot with the three clusters (Figure15). As we see the majority of the observations are gathered in the center so it was very difficult to avoid the overlapping between the clusters.

Figure 15



3.5 INTERPRETATION

In order to have an understanding of the characteristics of our clusters, we found the observations of each cluster, and we matched this with the unscaled dataset, to have a more clear view for the data. However, given that the dataset used for the clustering had been transformed in a way that includes the total calls and the total minutes, we can have a view only for the total numbers. In the first cluster the average sum of minutes is equal with 650.60 and the average of calls is equal to 278.01 For the second cluster 595.68 and 335.39 respectively, and for the third 490.10 and 287.78. We could say that in the first cluster, customers tend to have a very high number of minutes and a slight small number of calls , for the second cluster, customers have fewer minutes and a very high number of calls, and for the third cluster, customers have a usual number of calls and a low number of minutes.

4. CONCLUSION

In the classification part, after we cleaned our dataset and we selected the most appropriate variables of our dataset, we implemented the classifications methods 'Decision Tree', 'Random Forest', 'Support Vector Machines' and 'Naive Bayes'. Our final decision is to use the Random Forest, as it combines the highest values of accuracy. Adjusted Rand Index, and AUC (Area Under the Curve).

In the clustering part, after we selected the appropriate variables that are related to the usage of the customers, we implemented the Hierarchical clustering and the K-Means methods. From the two methods we believe that we have a clearer view of the clusters with K-Means, as clusters have higher average silhouette width and there almost nonnegative silhouette values. We believe that the company should implement different strategies for the customers that make more calls, but they do not speak for long time, for customers that have fewer calls but the talk longer, and different for the customers that have both usual number of calls and talk for usual number of minutes on average.