Telecom Churn

Matteo Carucci, Alessandro Natoli, Tommaso Agudio, Lorenzo Ciampana

2023-05-07

1 Final Project: Identifying Telecom Churn Clients

Matteo Carucci, Alessandro Natoli, Tommaso Agudio and Lorenzo Ciampana.

***Important Disclaimer: We understand the importance of providing clear and smart code, however we prioritized the importance of our findings. To check all our elaborations and techniques tried, please refer to the complete Rscript code - Some important chunks have been omitted from this report due to their size**

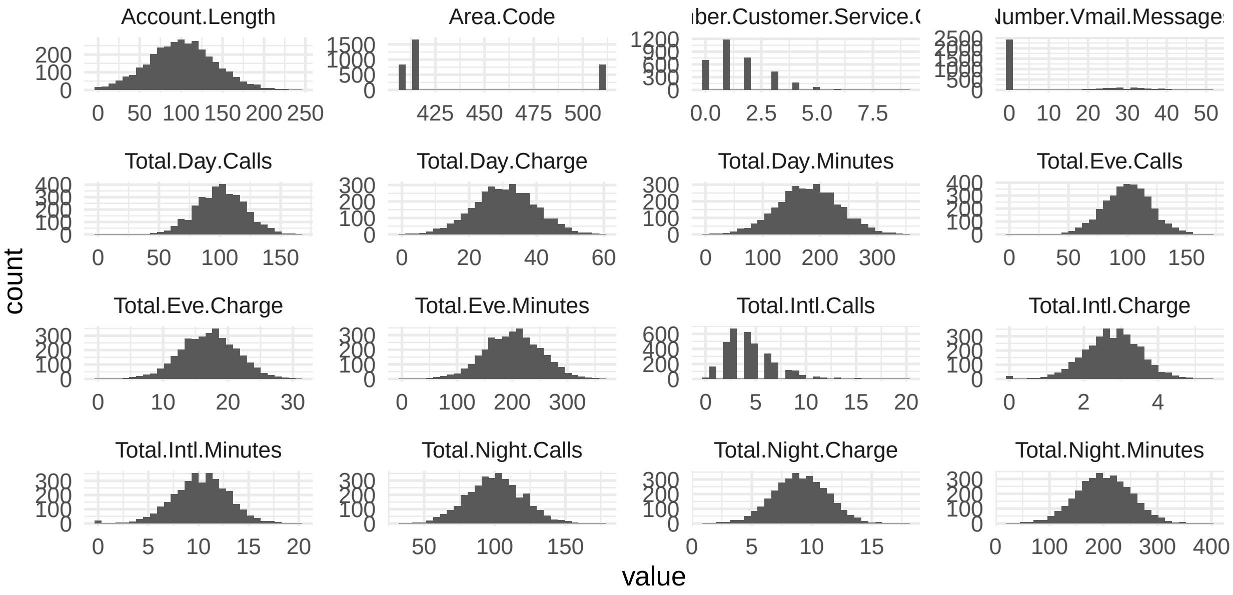
1.1

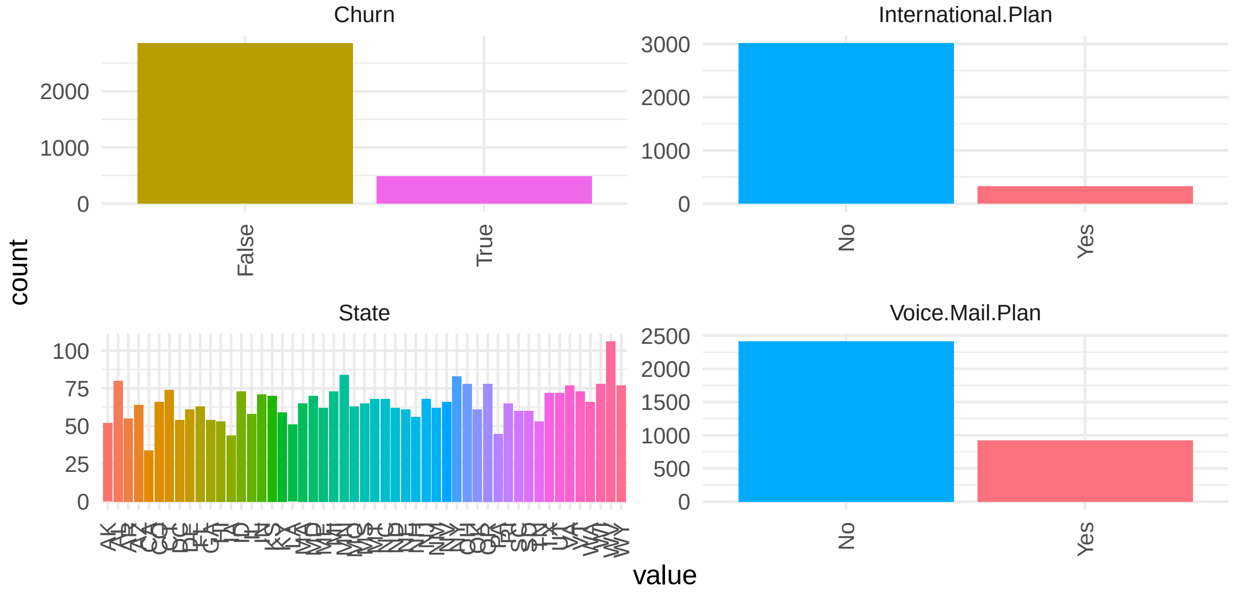
Importing all the necessary libraries for the data analysis and model creation

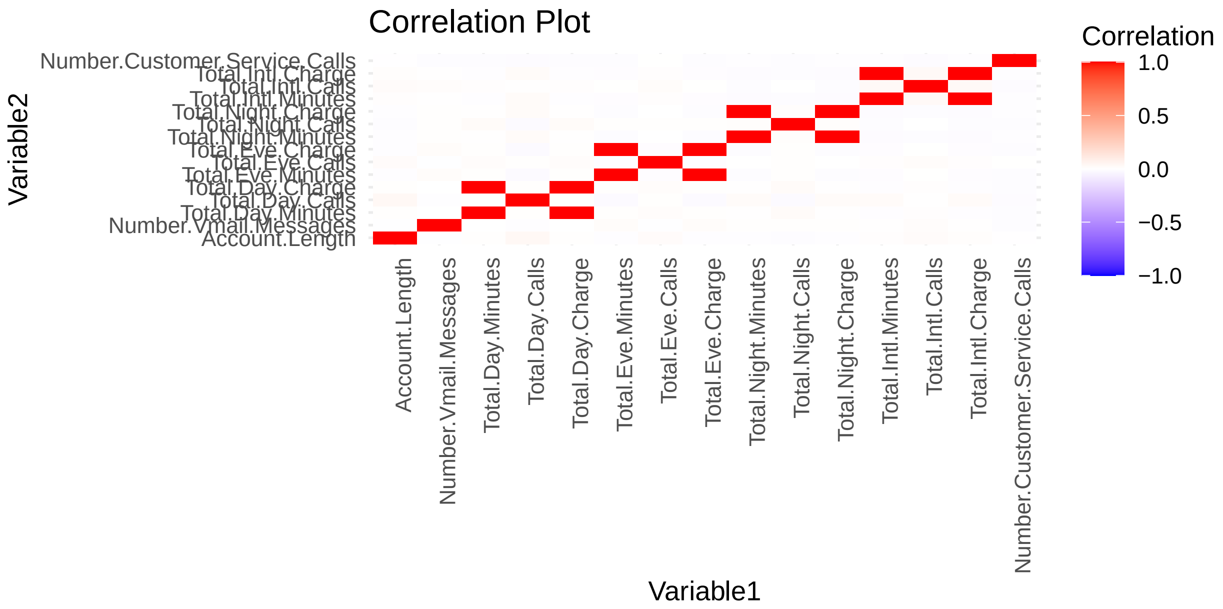
First we will load the dataset, rename to columns and convert all the categorical features into factors

2 Exploratory Data Analysis

We perform some EDA, in order to better understand the dataset we are dealing with and eventually some of the behaviors of the customers.





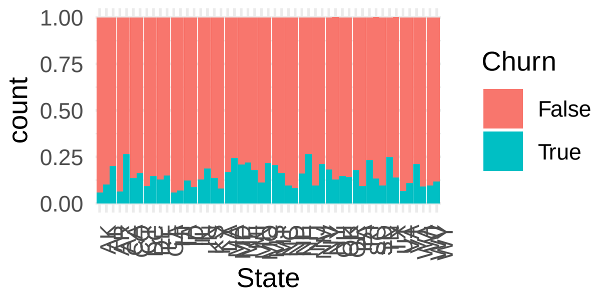


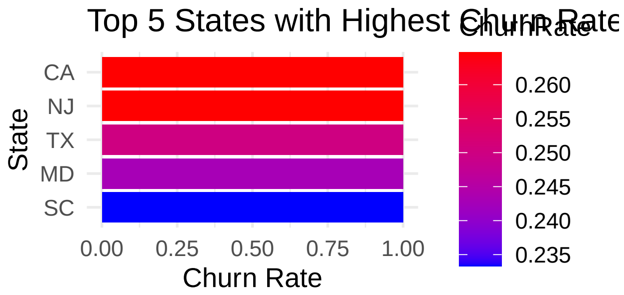
1. For the first graph we see a summary of the whole dataset, and we noticed that most of the people do not have a Voice mail plan and don't have an International plan. The churn rate is around 16%. The interquantile ranges seem rather normal and there is nothing that might be unsettling apart from the 3rd quantile for Total.Int.Calls and the Max.

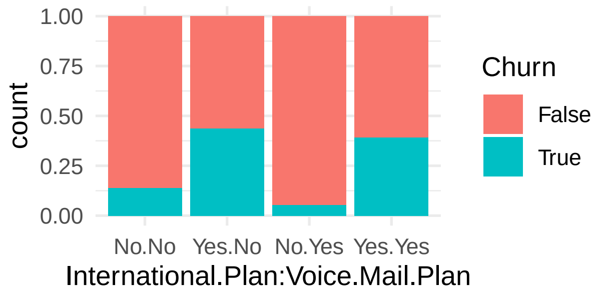
2. In the second graph, we plotted various distributions of the numerical variables and we can see that most of the variables have a normal distribution, while on the other hand we can clearly see that "Total international calls" and "Number of customer service calls" are right skewed. We can also say that "Area Code" and "Number Vmail Messages" have a strange distribution (we will later see if they influence or not).

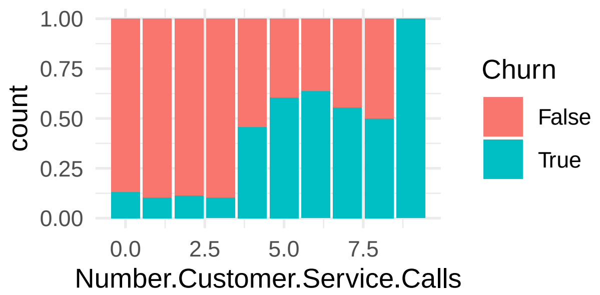
3. In the third graph we plotted the categorical variables and we can confirm what we said about the categorical variables before, while we have a better view of the states and how many customers belong to each state. We cannot state anything significant except for a state that has a slight higher number of customers (only 20 more than the second biggest).

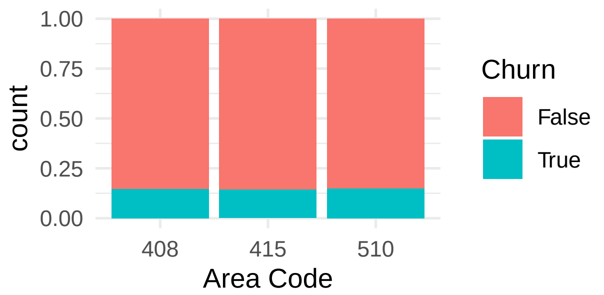
4. In the fourth graph we plotted the correlations of the variables and we can say that all of them are almost not correlated at all (there are slight colors but the intensity is not significant). The only variables that are correlated are “Total Day Minutes”-“Total Day Charge”, “Total Eve Minutes”-“Total Eve Charge”, “Total night charge”-“Total night minutes” and “Total international minutes”-“Total international Charge”. They are correlated for the fact that the more you call the more you spend.

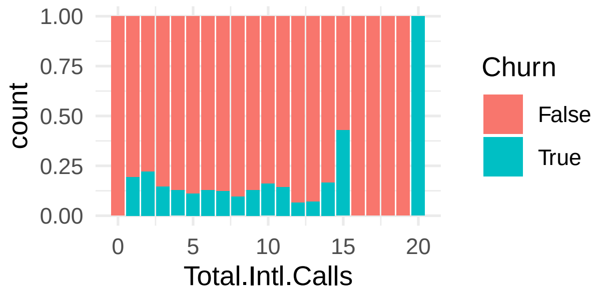


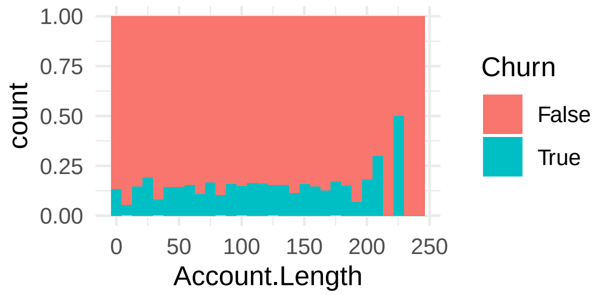












1. In the first graph and the second we plotted the churn rate based on the states. We can see that the maximum churn rate is 26% for California. The lowest one has around 5% of churn rate. In general the churn rate is not that significant due to the fact that we have states that have large population which will clearly have a larger churn rate compared to states with a smaller population (ie California 32.6 million and Iowa 3.1 million).

2. In the third graph we see a combination of 'International Plan' with The 'voice Mail Plan'. We can clearly see that the combination yes:no and yes:yes have a churn rate that is slightly less than 50% which is rather significant, implying that potential additional costs for international plans or the voice mail plan do not satisfy the customers needs.

3. In the fourth graph we can see that the more a customer calls the customer service the more likely he will churn. For instance after 8 calls the customer will almost certainly churn.

4. In the sixth graph we can say that area code doesn't influence the churn rate.

5. In the seventh graph we noticed something interesting, between 0 and 14 calls the churn rate is rather low, while for 15 calls we have a spike increase of the churn rate of around 45%, between 16 and 19 we have no churn rate (maybe due to the small amount of observations) and at 20 calls we have a churn rate of 100%, but if we look at the dataset in depth we can see that from 16 to 20 calls there are only 7 observation, therefore this range shouldn't be considered significant.

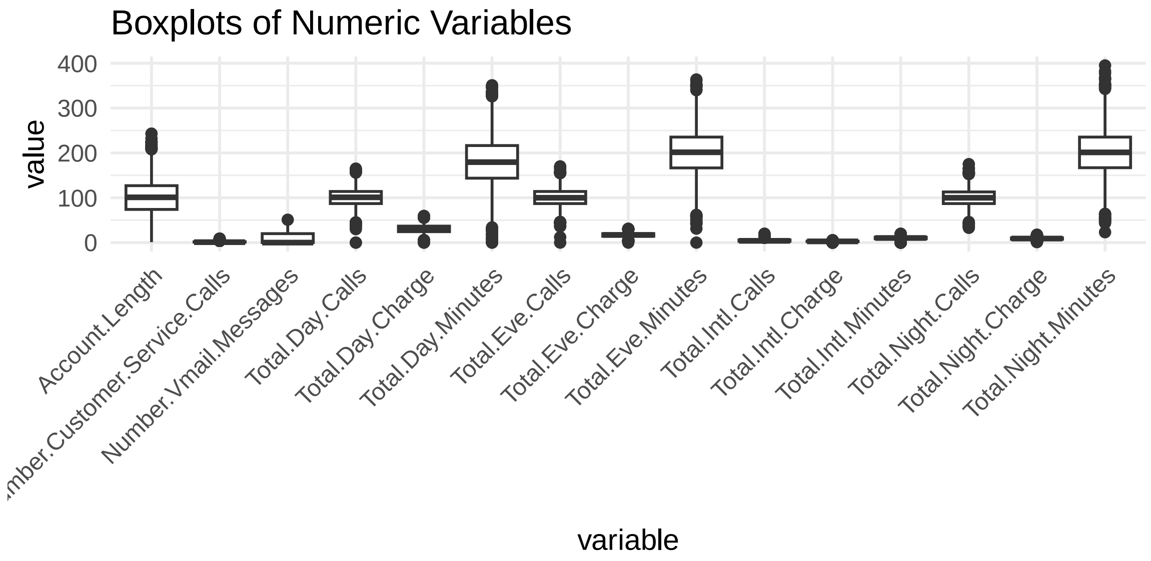
6. In the last graph we can state that the account length doesn't influence that much the churn rate. But in the range 225-250 there are only 3 observations, meaning that the range cannot be taken into consideration.

3 Data Cleaning

Now we want to see if the dataset is clean and if there is a significant amount of outliers.

```
## [1] "No duplicates found."
```

```
## [1] "No NaN values found in any column."
```



The dataset doesn't have null values nor duplicate values, so it doesn't need any sort of cleaning. For what concerns the outliers it seems reasonable to eliminate only "Total International Calls" based on its distribution. We've seen also that "Number of customer service calls" is right skewed, but we will keep it has it is because it is most likely that a person that calls a lot the customer service will churn, since he might be facing problems with the service.

4 Model Implementation

Now we will create, fit and test the accuracy of a lasso logistic regression model. The goal of the model is to predict whether a customer will churn or not. The primary objective is to identify if a customer will churn or not, we can probably guess that there is an intrinsic relationship between churning and the number of customer service calls. Our objective is to notice if other features could influence the churn rate, for example the international plan and the voice mail plan, and maybe see if it is related to the price charged of the customer.

Now, let us see if by excluding the Number Customer Service Calls, which is clearly a key feature when considering churning rate, will give us a worse model for the classification predictions.

```
##                                Model         Precision               AUC
## 1 LLR without Customer Service Calls 0.833333333333333 0.749141081871345
## 2     LLR with Cutomer Service Calls 0.538461538461538 0.847916666666667
##                  F1             Recall          Accuracy
## 1 0.0980392156862745 0.0520833333333333 0.861861861861862
## 2  0.128440366972477 0.0729166666666667 0.857357357357357
```

From the provided table, we can see the following insights:

1. **Precision**: The model excluding “Customer Service Calls” (0.8333) has a higher precision compared to the model including “Customer Service Calls” (0.5385). This means that when the first model predicts a customer will churn, it is correct more often than the second model.

2. **AUC**: The model including “Customer Service Calls” (0.8479) has a higher AUC-ROC compared to the model excluding “Customer Service Calls” (0.7491). This indicates that the model including “Customer Service Calls” is better at distinguishing between the classes (churn vs no churn) across different threshold levels.

3. **F1 Score**: The model including “Customer Service Calls” (0.1284) also has a higher F1 Score than the model excluding “Customer Service Calls” (0.0980). The F1 score is a measure of a test’s accuracy that considers both the precision and the recall.

4. **Recall**: The model including “Customer Service Calls” (0.0729) has a higher recall compared to the model excluding “Customer Service Calls” (0.0521). This means that the model including “Customer Service Calls” is better at identifying positive instances (churned customers).

The model excluding “Customer Service Calls” has higher precision, but the model including “Customer Service Calls” outperforms it in terms of AUC, F1 Score, and Recall. If you want to be more certain when predicting a positive (a churned customer), then higher precision (and the model excluding “Customer Service Calls”) is better. If you want to be more comprehensive in finding positive cases, higher recall (and the model including “Customer Service Calls”) is better. Moreover, if you want a balance between precision and recall, you should consider the F1 score. Lastly, if you want the model with better overall class separation ability, you might prefer the one with higher AUC.

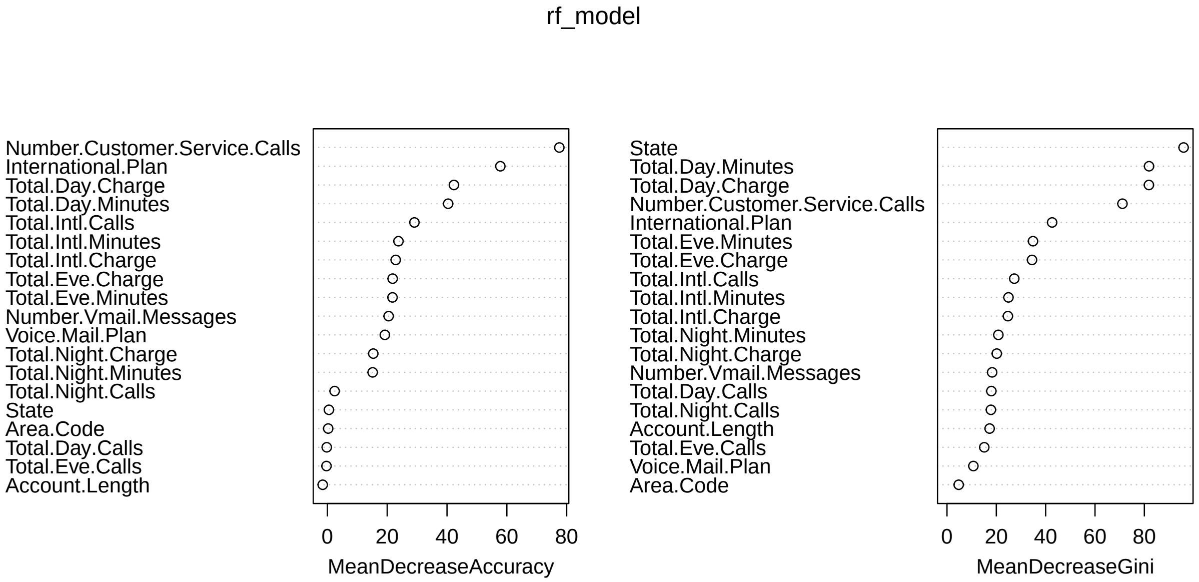
Since we are looking to predict customers that will churn, higher precision might be better, thus the model excluding the Number of Customer Service Calls would fit our needs.

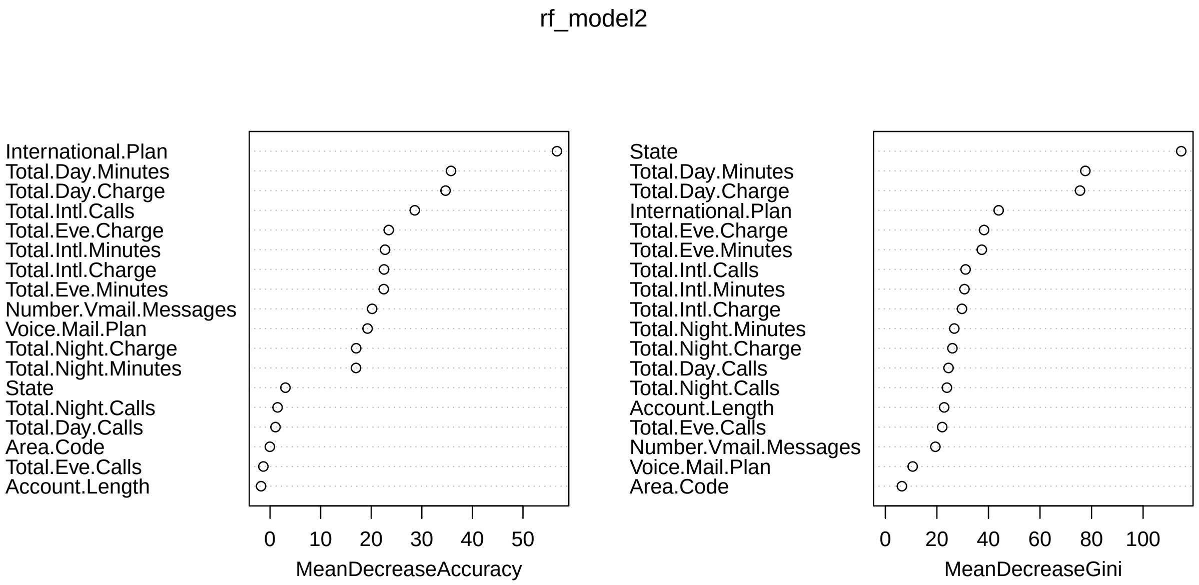
4.1 Non Linear Models (Random Forest)

The results of the lasso logistic regression are somewhat good. But by using such method we are not taking into consideration possible non linear relations between the variables, so we now create a random forest model.

Random Forest: we will also in this case analyse the model in two, the first will contain the feature Number Customer Service Calls, while the second will not, this to remain coherent with what we have done previously.

Now we will test the Random Forest model without the Number of Customer Service Calls





Looking at rf_model (with number of customer service call) and rf_model2 (without number of customer service call), we can see that the features that have the top 3 MeanDecreaseAccuracy are Number.Customer.Service.Calls (for rf_model), International.Plan, Total.Day.Minutes and Total.Day.Charge. While if we look at the MeanDecreaseGini, the top 3 stay the same, with State, Total.Day.Minutes and Total.Day.Charge.

```
##                                         Model         Precision
## 1           LLR without Customer Service Calls 0.833333333333333
## 2               LLR with Cutomer Service Calls 0.538461538461538
## 3    Random Forest with customer service calls 0.923076923076923
## 4 Random Forest without customer service calls 0.928571428571429
##                  AUC                 F1             Recall          Accuracy
## 1 0.749141081871345 0.0980392156862745 0.0520833333333333 0.861861861861862
## 2 0.847916666666667  0.128440366972477 0.0729166666666667 0.857357357357357
## 3 0.926909722222222  0.827586206896552               0.75 0.923076923076923
## 4 0.800986842105263  0.684210526315789  0.541666666666667 0.928571428571429
```

3. Random Forest with customer service calls: The Random Forest model with customer service calls is performing better across all metrics. *Precision* is 0.923, *Recall* is 0.75, *AUC* is 0.926, and *Accuracy* is 0.923. The high *F1* score (0.827) suggests a good balance between precision and recall. This model appears to perform well overall and considerably better than either Lasso Logistic Regression model with and without Number Customer Service Calls.

4.Random Forest without customer service calls: Removing customer service calls from the Random Forest model increases *Precision* (0.928) and *Accuracy* (0.928), but decreases *Recall* (0.541), *AUC* (0.800), and *F1* (0.684). This suggests that, while the model makes fewer false positive predictions without the customer service calls feature, it also misses a larger number of actual positive instances.

Overall, the Random Forest model with customer service calls seems to provide the best performance, considering all the metrics. Even if we consider the Random Forest without the Customer Service Calls, the precision and accuracy is just slightly better than the other model implemented, while the AUC, F1 score and recall are significantly worse. This goes to show that Number Customer Service Calls is an important feature.

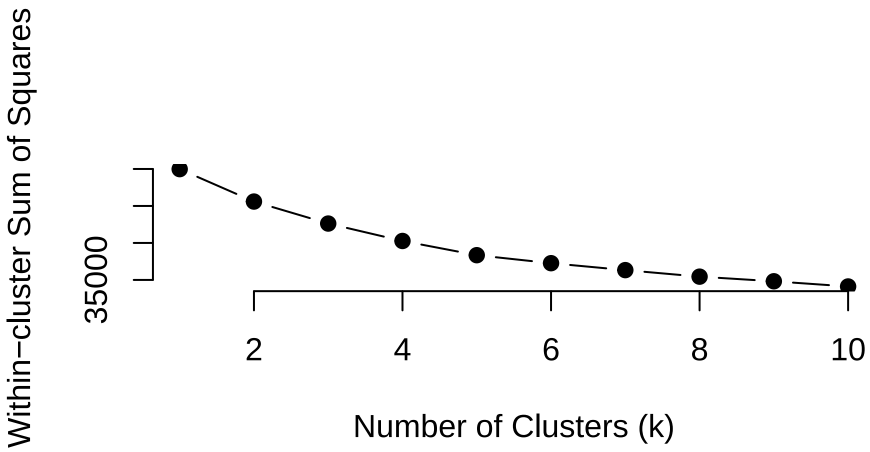
Now we get into clustering the telecom's customers. First let's see the how Kmeans clusters with outliers.

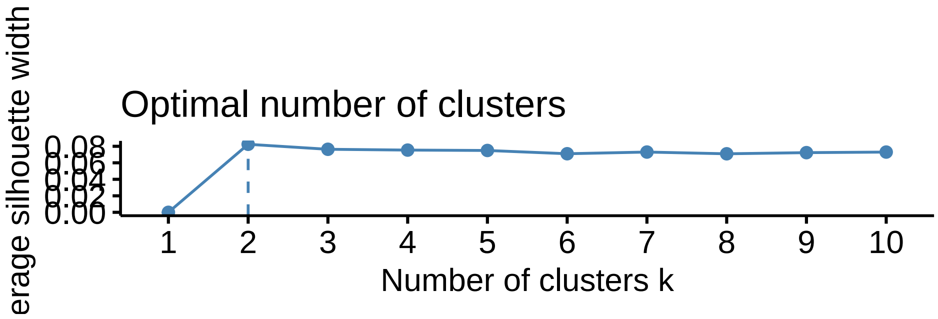
5 Clustering

We now want to cluster the customers according to their behavior and then we will check if the clusters have been grouped together well based on the Churn rate.

5.1 Kmeans

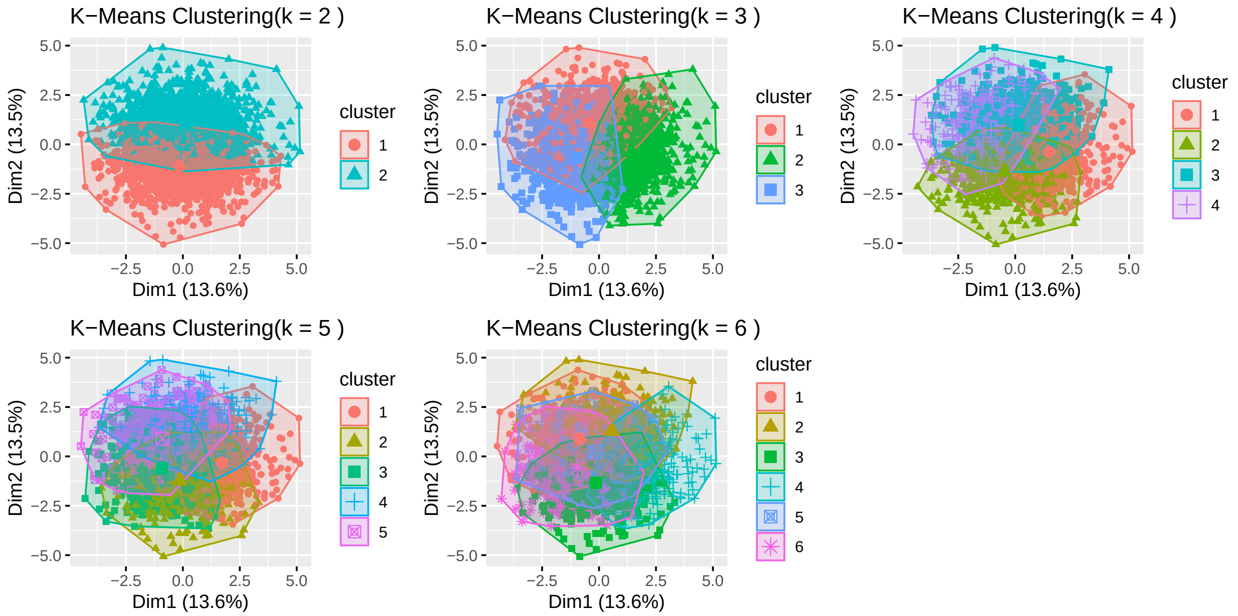
We will now first calculate the WSS (elbow method) and the silhouette score to better decide the number of clusters to implement. Afterwards we will see how well the method clustered the different customers and analyze the potential criterion for this partition.





From the WSS method, we can see that the potential number of clusters ranges from 5 to maybe 6, but if we only look at the silhouette score, we see that the optimal number of clusters is 2. We thus decided to see the potential clusters ranging from 2 to 6.

Indeed the silhouette score by k, measures the quality of clustering, the higher it is, the better, as it indicates that they are more distinguishable and separate. On the other hand the within sum of squares indicated in the elbow plot indicates roughly "how close" data points are in the cluster, that is, how segregated they are (the higher, the better, stronger community).



We can clearly see that in fact, the silhouette score was correct, the optimal amount of clusters is in fact 2. We will now see if kmeans has done a good job in clustering the customers also based on the churn rate.

5.2 Hierarchical Clustering

We will use Euclidean distance and 1-row correlation to see which is better at clustering the data.

The process is similar to what we did for k-mean, we calculate the silhouette and then based on the resulting k choose the number of clusters.

```
## Silhouette scores for euclidean distance: 0.0498352 0.03169852 0.03670976 0.03050948 0.02095289
```

```
## Silhouette scores for 1 - row correlation: 0.09467995 0.07526763 0.08150353 0.06556275 0.05300178
```

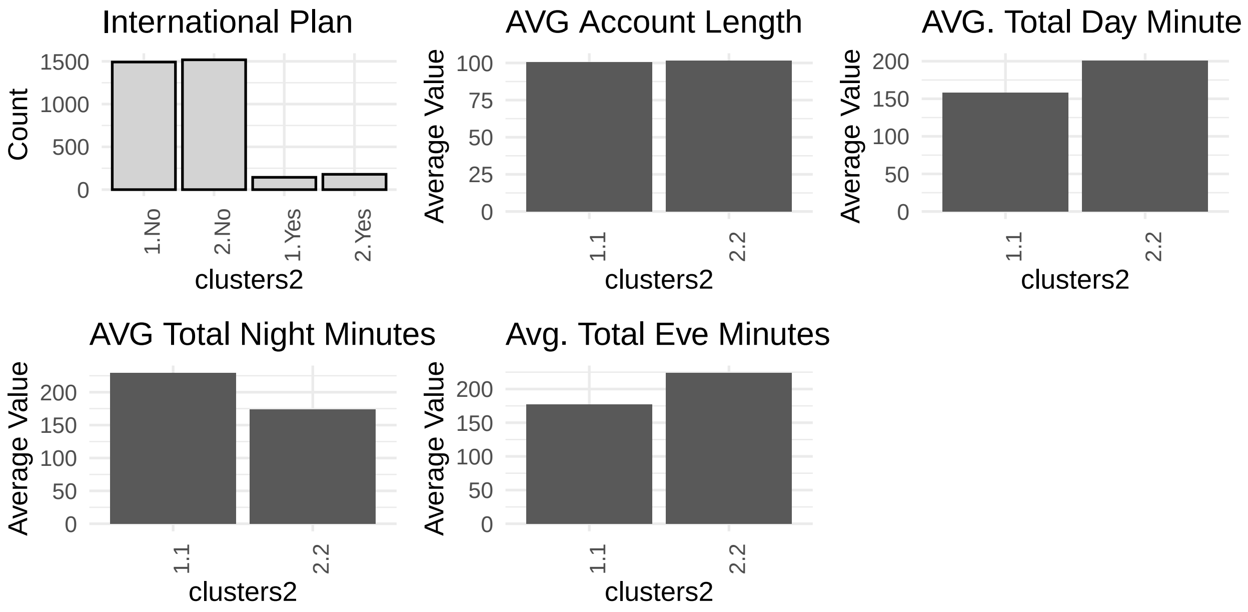
We see that the best score is with k=2 or k=3 for both types of distance measures. Now let us see how the clusters look to better understand the groups.

The results obtained with Hierarchical clustering, using both measures is rather disappointing, the clusters overlap. We can clearly say that K-means did a better job at identifying the clusters both graphically and with a better silhouette score.

5.3 Deeper dive into the clusters

Now we shall have a deeper look at the two clusters obtained using the K-means and what they actually represent. We will focus on International.Plan, Account.Length, Total.Day.Minutes, Total.Eve.Minutes, Total.Night.Minutes which were important for the loadings of the second component of the PCA.

```
## # A tibble: 2 x 2
##   cluster churn_rate
##     <int>      <dbl>
## 1       1      0.114
## 2       2      0.174
```



6 Final Considerations

Looking at the 2 clusters and the bar-plots above, we can deduce the following information. Using the information below from the PCA loadings, we found out that Kmeans presumably clustered based on the total call minutes during the different periods of the day - The features loadings have significant positive/negative correlation with the second component, and as we can see the clusters are divided based almost solely on it (we are just taking 27% if variability but still quite relevant).

There were not other substantial differences in customers belonging to the 2 different clusters; It is noticeable that the second cluster has an higher churn rate than the first with a 17% vs 11% of the first.

The general churn rate for all customers is of 16.95%, suggesting that the primary cause of churning might be that the price charge during the day and the evening is not adequate to retain customers in the long run, whilst those who call most during the night are more satisfied with the call price they are charged.

These assumptions are based on the results of the clustering and on the bar-plots above - On average clients in the second cluster call more during the day and the evening, whilst in cluster 1, people tent to call much more during the night.

```
##                                                   variable          PC2
## Account.Length                               Account.Length  0.001376639
## Number.Vmail.Messages                 Number.Vmail.Messages  0.011567548
## Total.Day.Minutes                         Total.Day.Minutes  0.109128344
## Total.Day.Calls                             Total.Day.Calls -0.040764763
## Total.Day.Charge                           Total.Day.Charge  0.109131652
## Total.Eve.Minutes                         Total.Eve.Minutes  0.517795596
## Total.Eve.Calls                             Total.Eve.Calls -0.006282107
## Total.Eve.Charge                           Total.Eve.Charge  0.517798003
## Total.Night.Minutes                     Total.Night.Minutes -0.467867886
## Total.Night.Calls                         Total.Night.Calls  0.003066367
## Total.Night.Charge                       Total.Night.Charge -0.467874968
## Total.Intl.Minutes                       Total.Intl.Minutes  0.006180220
## Total.Intl.Calls                           Total.Intl.Calls  0.013131633
## Total.Intl.Charge                         Total.Intl.Charge  0.006156827
## Number.Customer.Service.Calls Number.Customer.Service.Calls -0.007212042
```

```
## 22.22205 for Night calls
```

```
## 11.76101 for Evening calls
```

```
## 5.878711 for Day calls
```

Now let us see if the churn rate of the various features corresponds to our hypothesis.

```
## 0.1122786 Churn rate for Day calls
```

```
## 0.1014406 Churn rate for Evening calls
```

```
## 0.0960096 Churn rate for Night calls
```

Here we can clearly see that Kmeans and as a consequence the loading of the second PCA, have in fact correctly divided the clusters. The second cluster had in fact more customers that called during the day and evening and also had a higher churn rate at 17%. We can see now in fact that the churn rate for the Day (11%) and Evening customers (10%) is slightly higher than the Night customers (9%), suggesting that maybe the price for the Day (5.8) and Evening customers (11.8), even if it's lower than the Night price (22.2), doesn't keep the customers in the long run. It might be due to potential technical issue (i.e server overcrowding) or maybe there are more competitive Day and Evening prices among the competition.