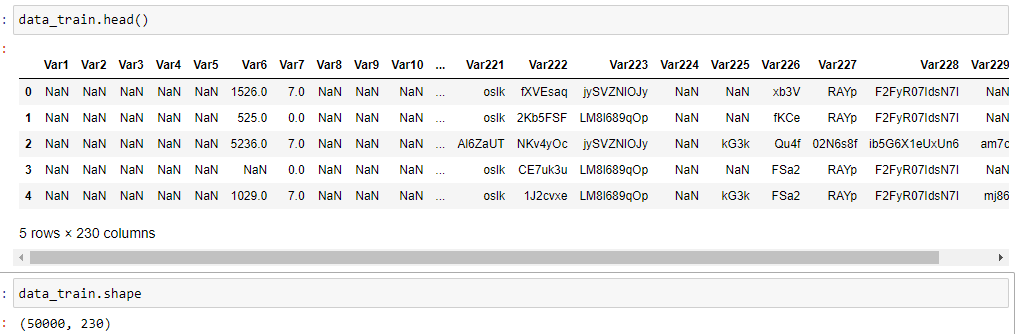
Introduction

Customer Relationship Management (CRM) is a key element of modern marketing strategies. The KDD Cup 2009 offers the opportunity to work on large marketing databases from the French Telecom company Orange to predict the propensity of customers to switch provider (churn).

The most practical way, in a CRM system, to build knowledge on customer is to produce scores. A score (the output of a model) is an evaluation for all instances of a target variable to explain (i.e. churn). Tools which produce scores allow to project, on a given population, quantifiable information.

Exploratory data analysis

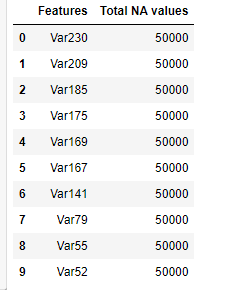
Our data consists of 50k rows and 230 columns as we see below



From our first look on the data we found that the features consist of categorical and numerical we will analyze them in data cleaning process.

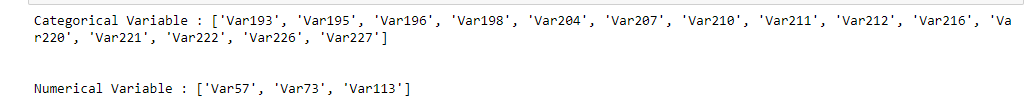
We can see also that there is a lot of features that contains Nan values,

This is a plot of top features that has more than 40k rows Nan values.

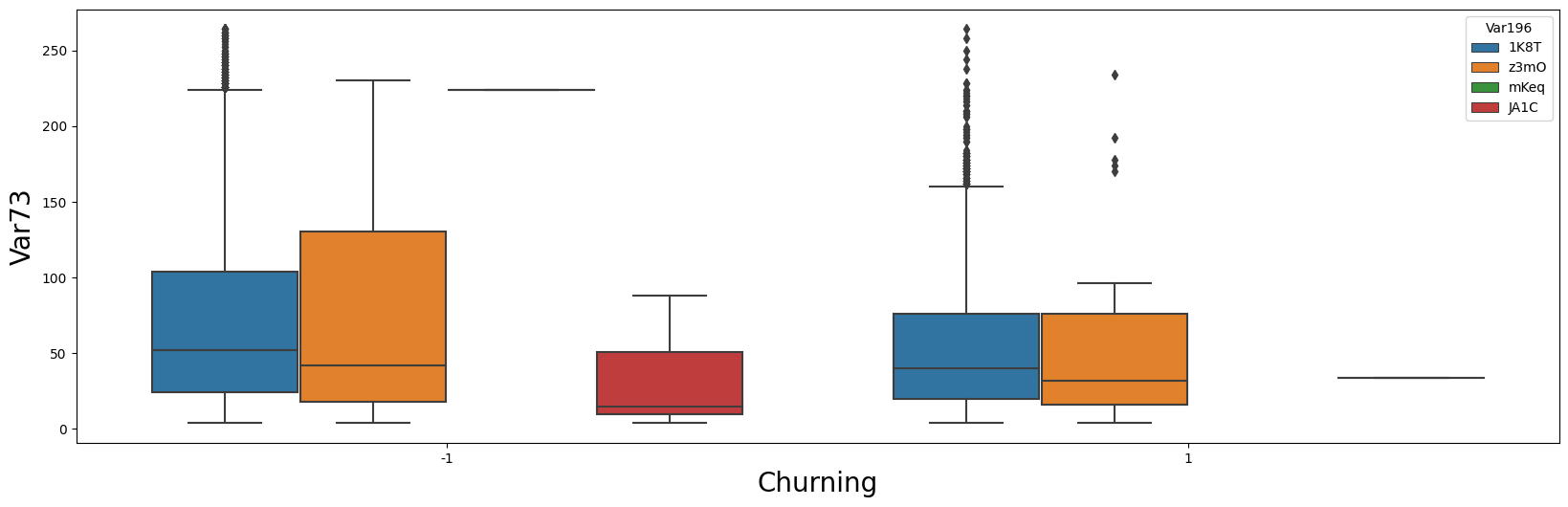


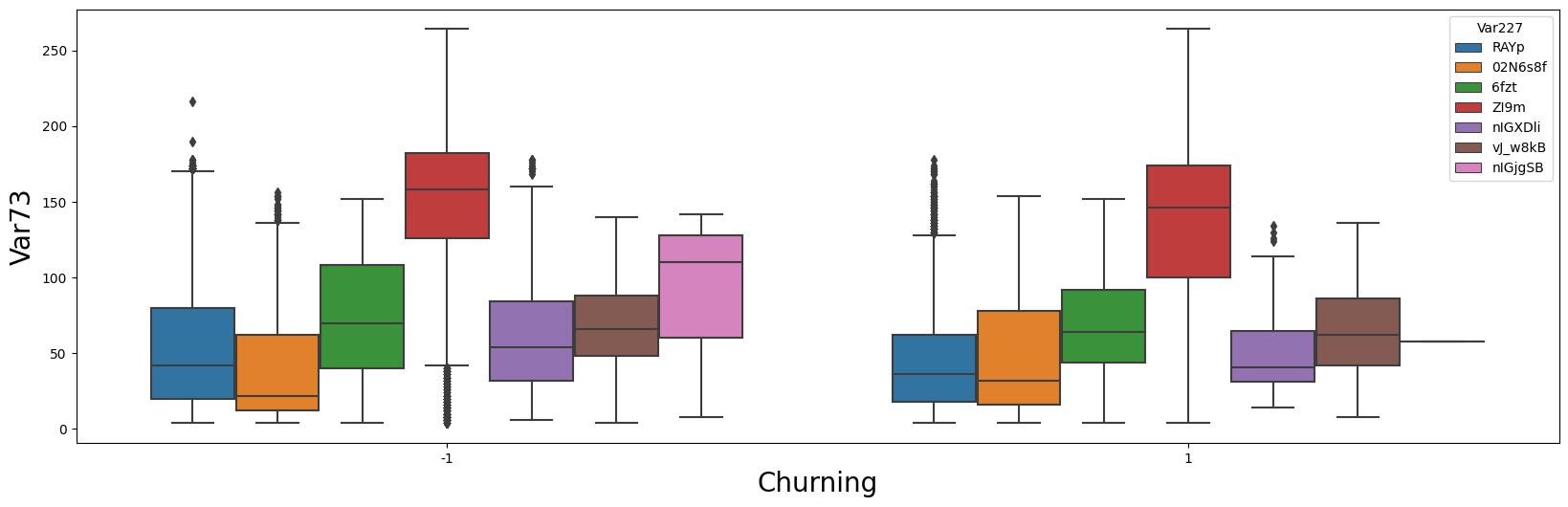
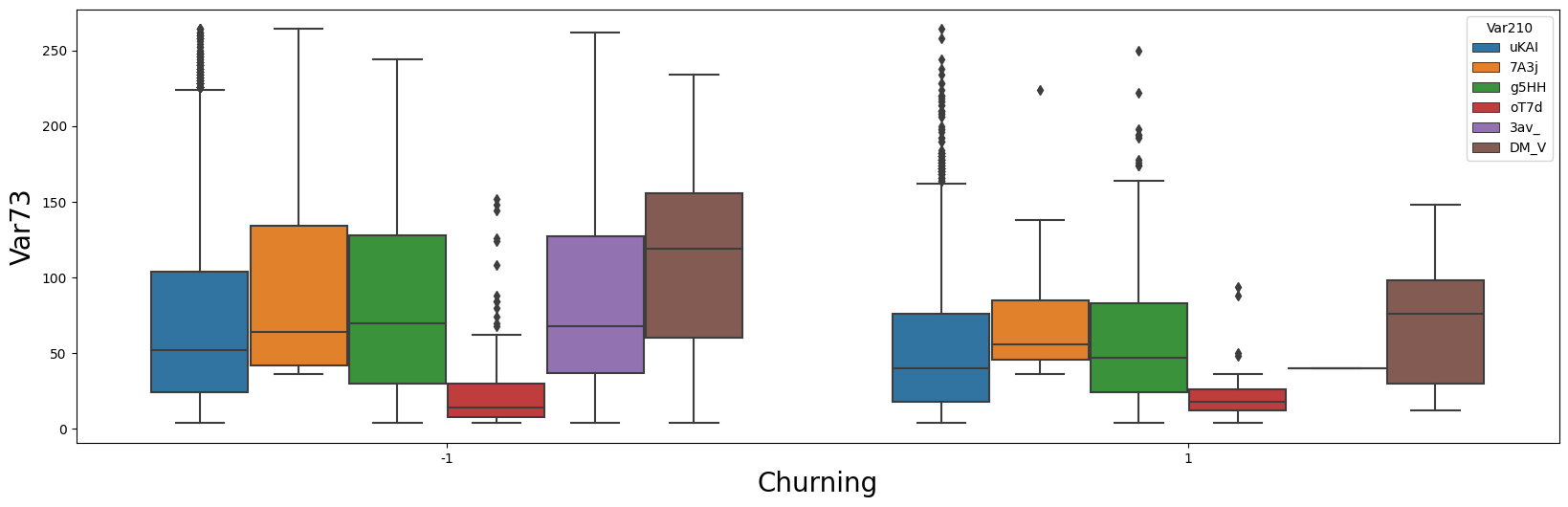
In order to proceed further in our exploratory analysis we will take the feature has no Nan values, as those are the most important features also we can plot those features easier.

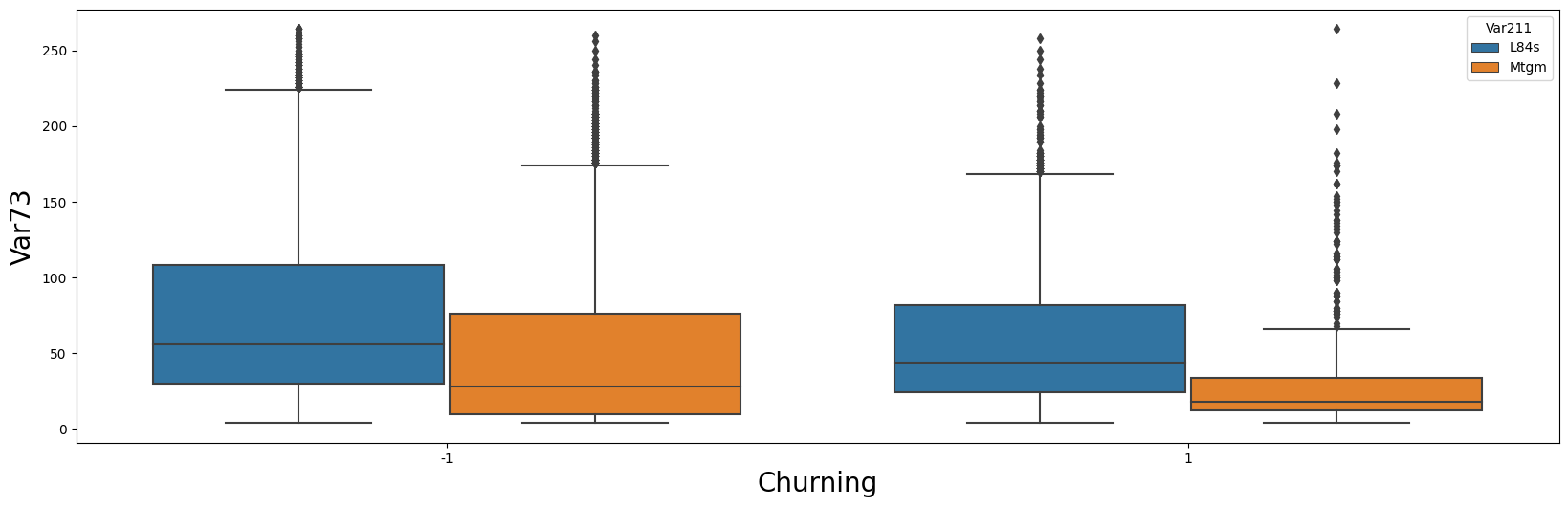
Those are the features we got Categorical and Numerical.



Here we are comparing the distribution of the categorical features values against one of the numerical one.



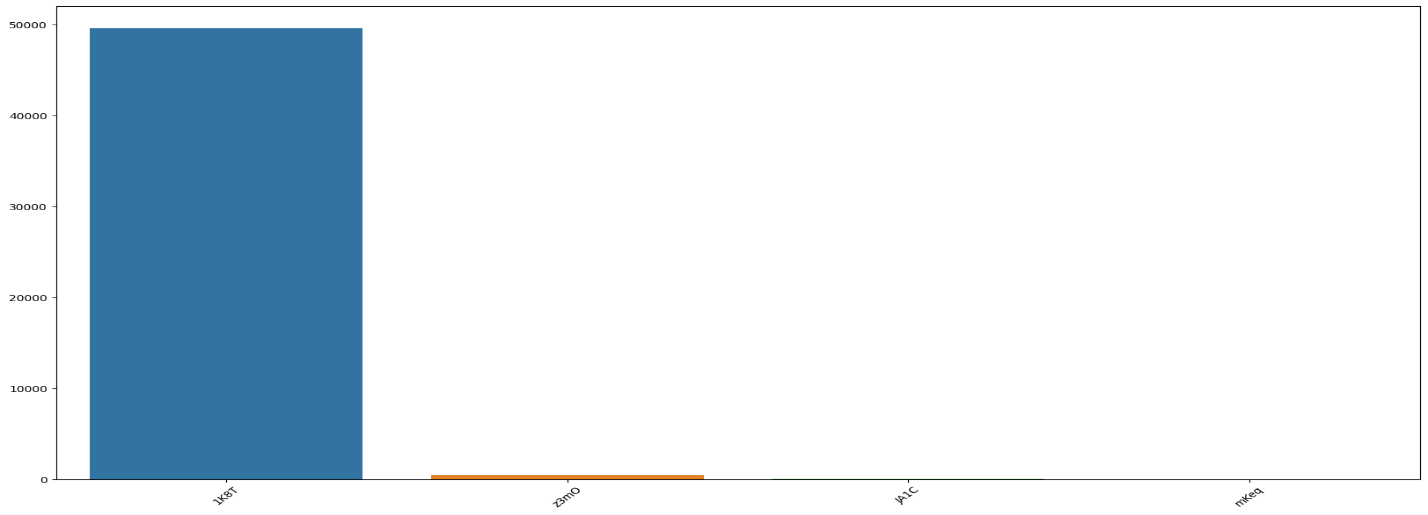
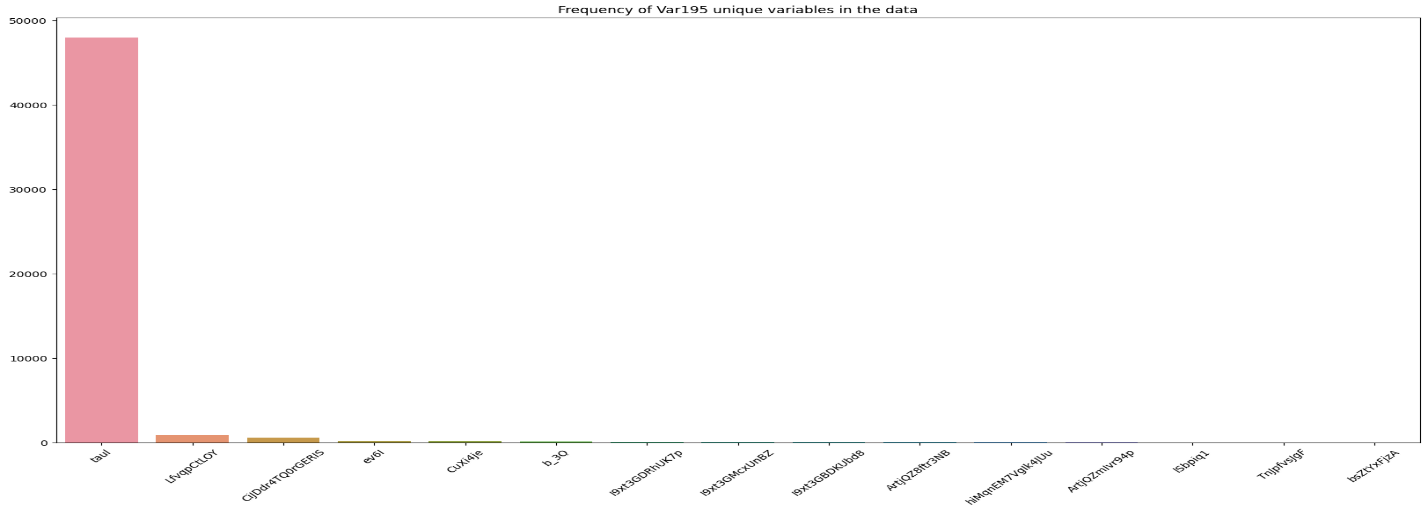


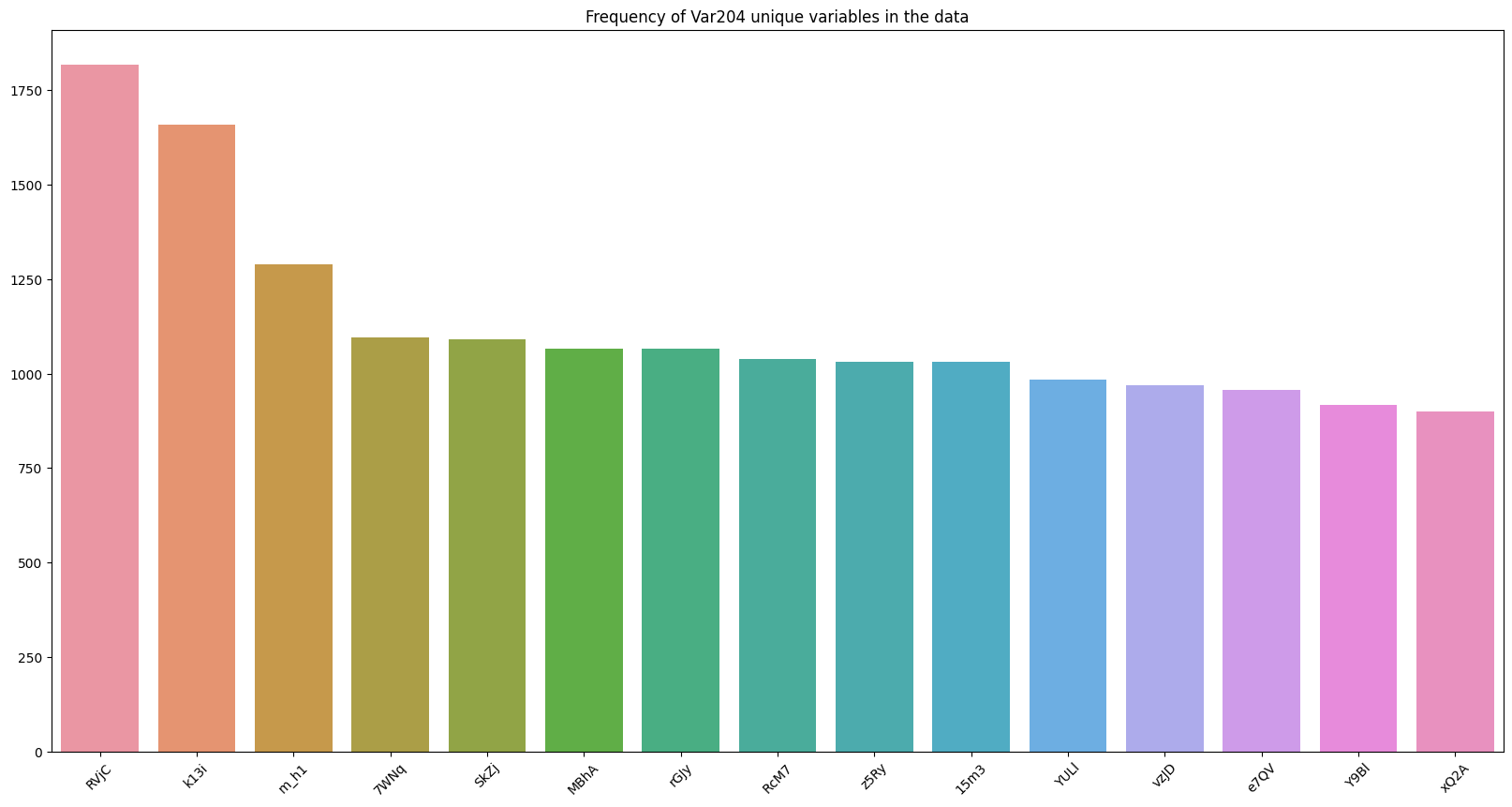
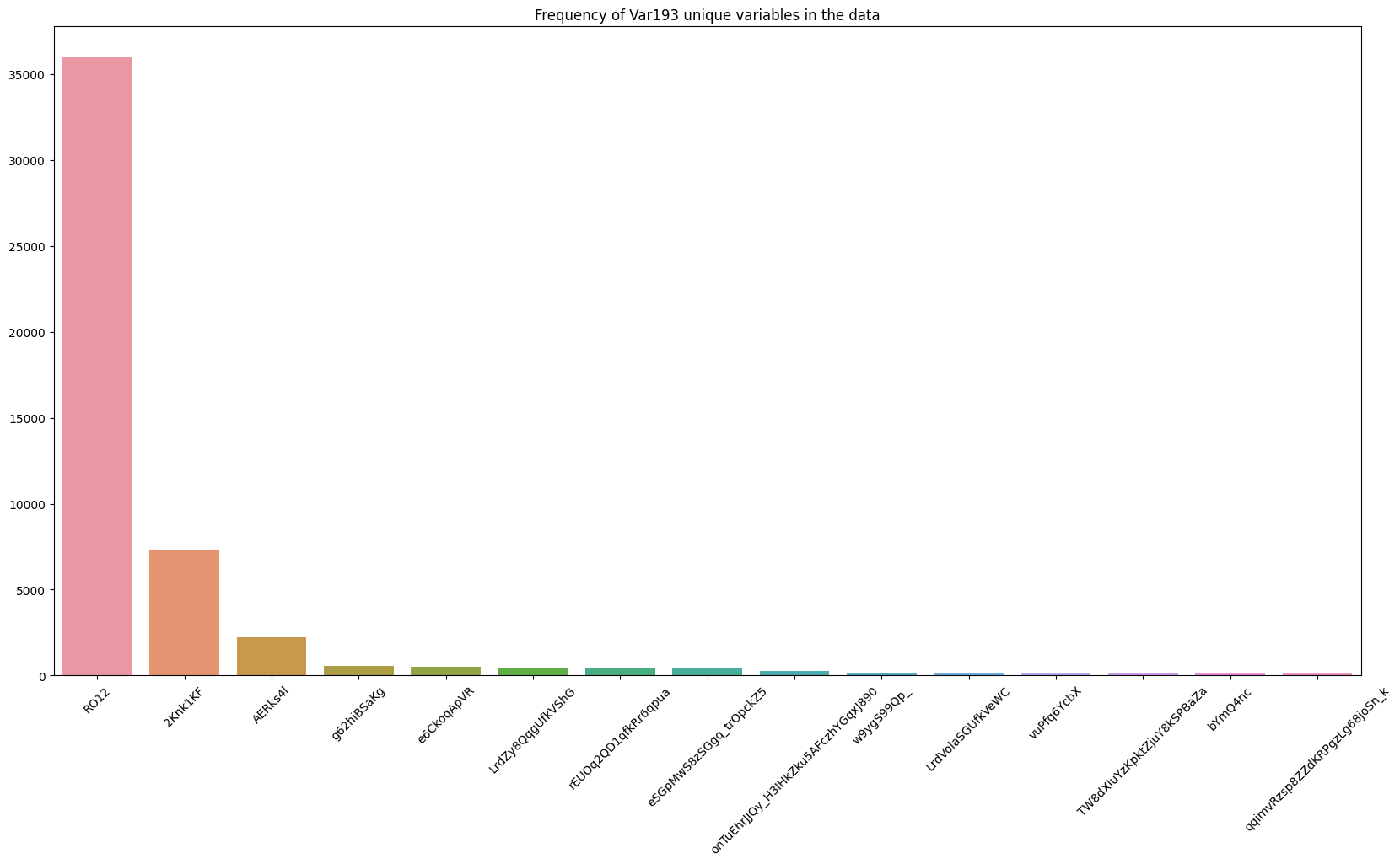


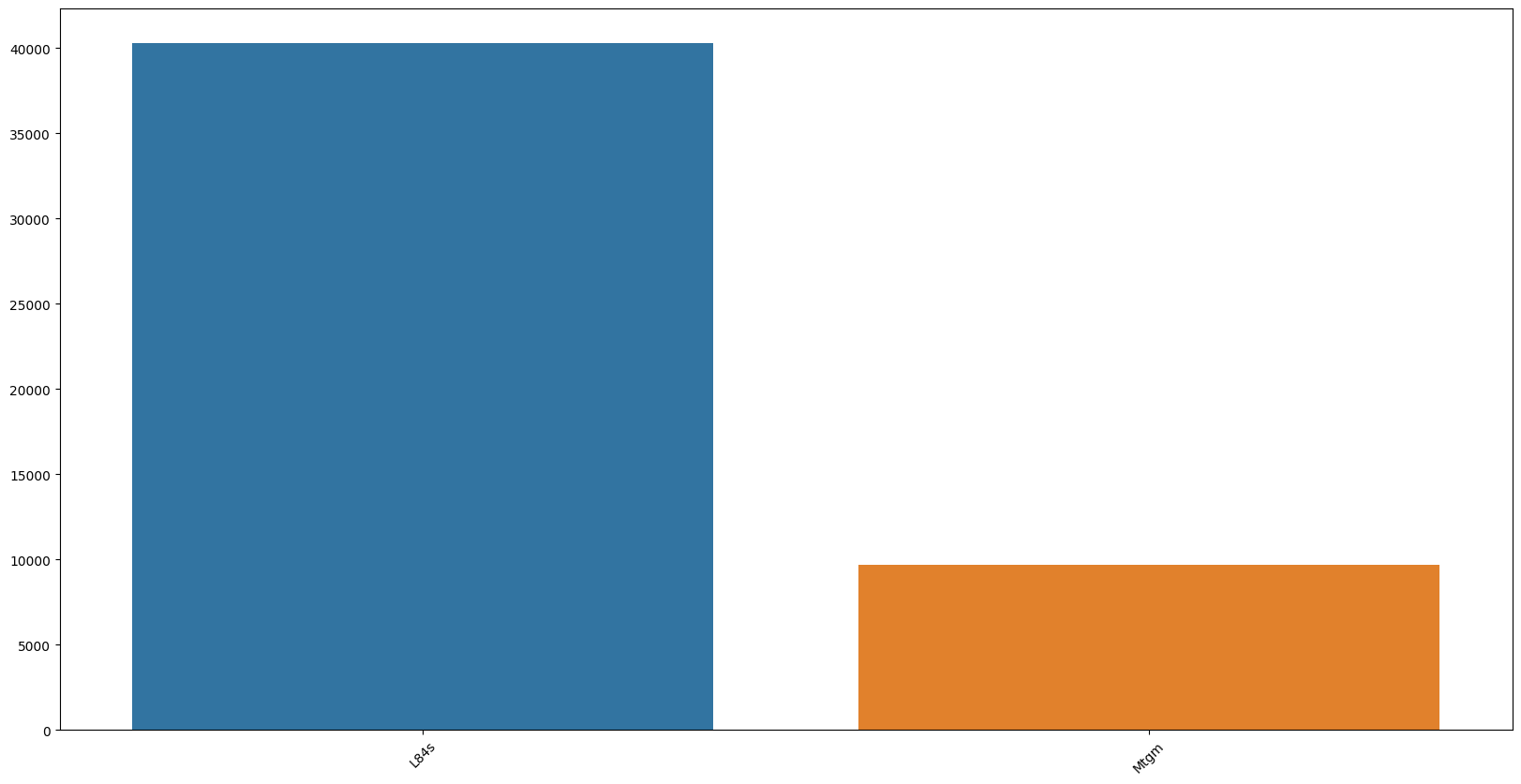
But by looking at boxplot hardly we can say anything as many of variable for both class label -1 and 1 have high proportion of collision.

And also there are some variable in features which are only present for class label -1 and not for 1.

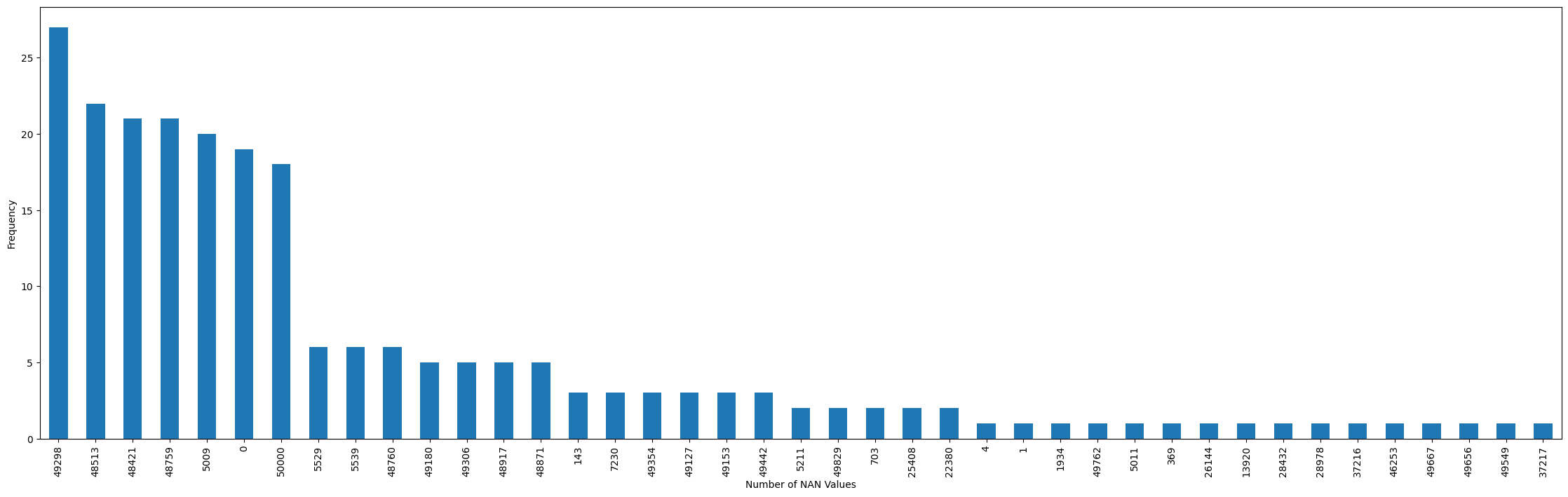
Plotting the frequency of values of some of our categorical Features:



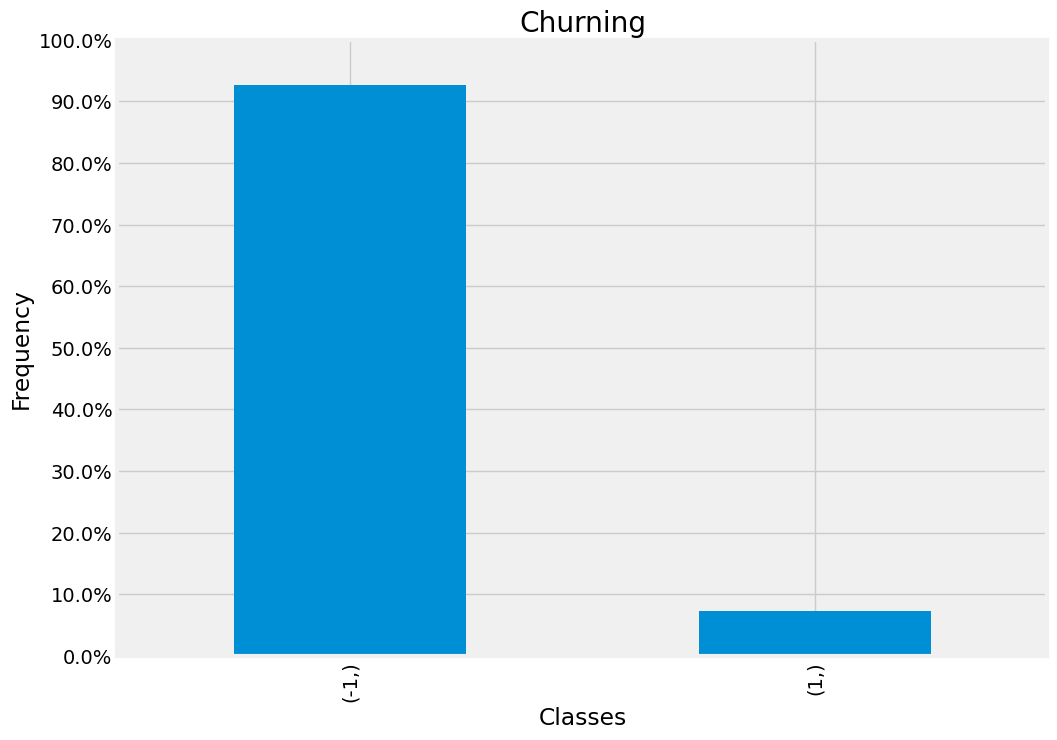




Plotting the top 50 variable that contains Nan values more than 40k rows.



Plotting the top 50 variable that contains Nan values more than 40k rows.



We can see in the distribution that data has highly unbalanced targets.

Data Cleaning

With the first look to our data we can see that the data needs a lot of cleaning, only by getting head() of the data we can see that there is a lot of Nan values .

First thing we have done is to search for columns that all of its rows contains Nan values , we found 18 columns and that means that they are useless features as it’s considered empty columns so we dropped these columns.

After that we left with the columns mixed with values and Nans.

Every column should represent a feature which has distinctive meaning to our goal which is to predict churn customers, so we decided to drop those columns that its values contains 70 % of nan values .

Now we have two steps which are handling categorical Nan values and handling numerical Nan values.

We handled the categorical Nan values by replacing the Nan with another categorical value like “others”.

Then we will encode the categorical values to use them in the model.

Handling the numerical Nan values will be as follow , we will substitute the Nan with the mean value of the column containing the values , but we have to split the data to train and test first before doing this step.

We split the data before calculating the means in order to avoid data leakage problem if we calculated mean before splitting our test data would have information about the train which is well known as data leakage problem.

Feature selection

In this section we used a built in method in sklearn library which called selectKBest which get the best number of features that works with this model we specified 20 features.

In our example selectkBest are based on mutual info regression that measure the mutual information between a matrix containing a set of feature vectors and the target.

 The mutual information can be used to gain some understanding on how good of a predictor a feature may be.

The mutual information between two variables, measures how much a given feature can explain another (target), or more technically, how much information about the target will variable will be obtained by having observed a feature.

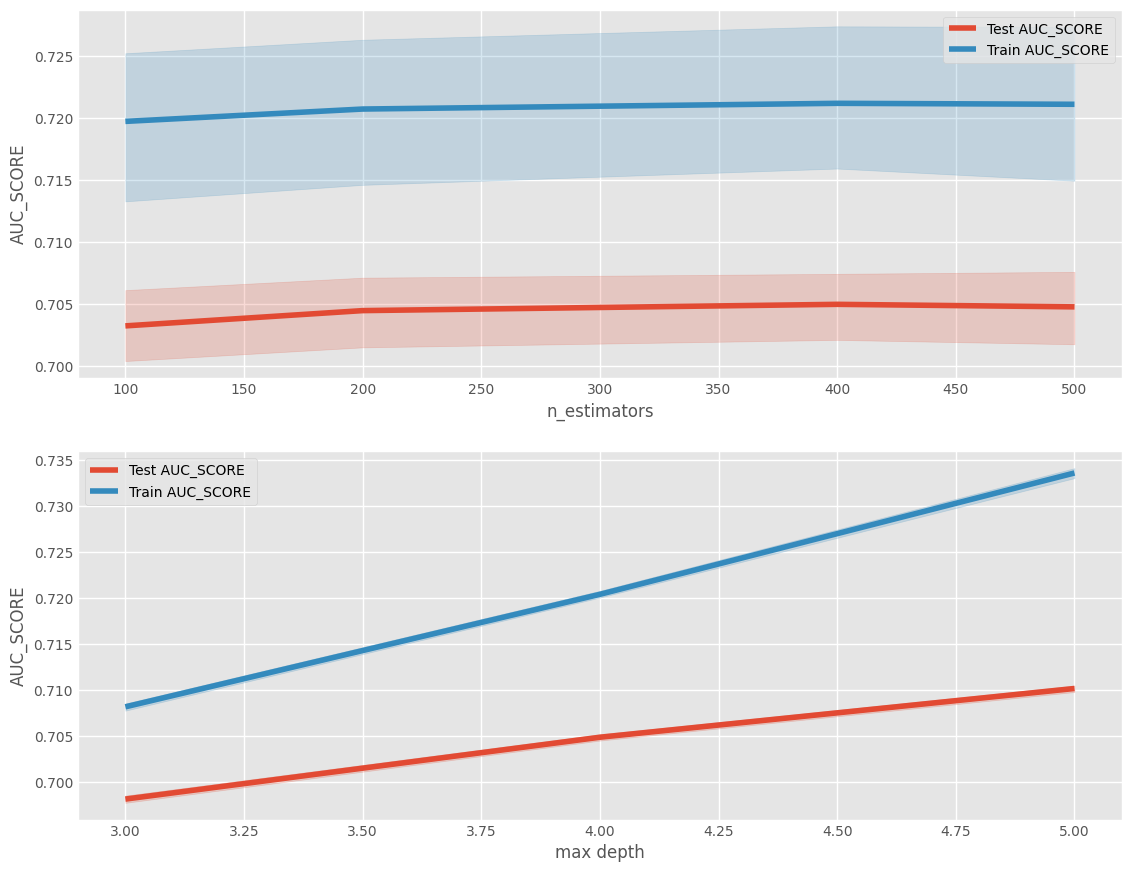
Second Step we normalized our features using min max scalar transformer to ensure all values lie between 0 and 1.

Model

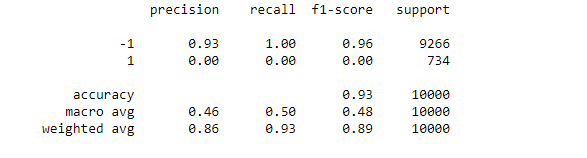
We selected Random Forest Classifier as suggested with cross validation technique using 5 folds.

We used these list of parameters using cross validation :  
n\_estimators:[100,200,400,500] , max\_depth : [3,4,5]

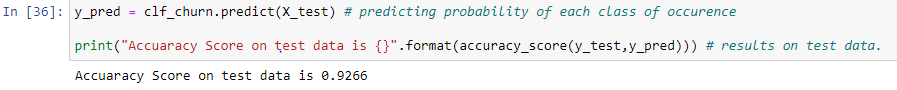
We had a graph across train auc score and test auc score to ensure that there is no over fitting as the scores are close .



This is our classification report results:



Accuracy score:



Conclusion

As expected Random forest got us very reasonable results with the help of cross validation we found that number of estimators has slightly effect of the model but max\_depth has huge influence on the model , we can see in the above graphs that max depth of 5 got us the best results which is around 72 % with auc score where it calibrates the trad-off between sensitivity and specificity at the best threshold and we got 92 % Accuracy Score on the test Data.