***Predicting Churn Out Customer Analysis:***

**“Akshay Verma”**

**--22 December 2018—**

**Problem Statement** : Churn (loss of customers to competition) is a problem for companies because it is more expensive to acquire a new customer than to keep your existing one from leaving. This problem statement is targeted at enabling churn reduction using analytics concepts.

**Data**: Our task is to check what customers are churning out from a plan and who are not churning out and in order to find out this we will do multiple analysis to filter out the data and to extract meaningful information out of it..

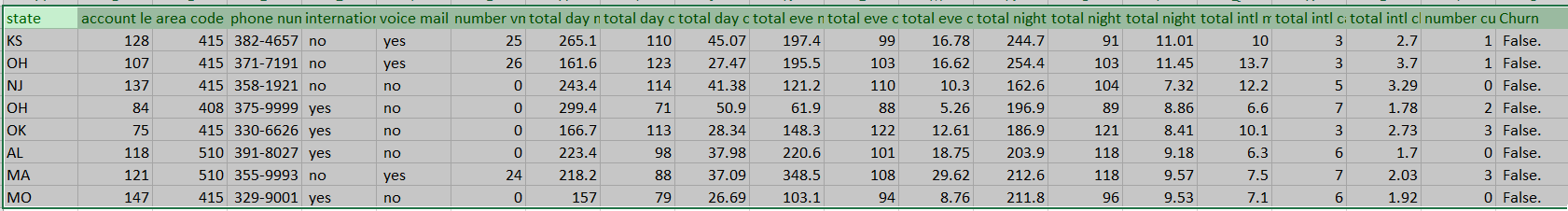
We will start from some pre – processing techniques and these are below..

1. Missing value Analysis
2. Outlier Analysis
3. Feature selection
4. Feature scaling

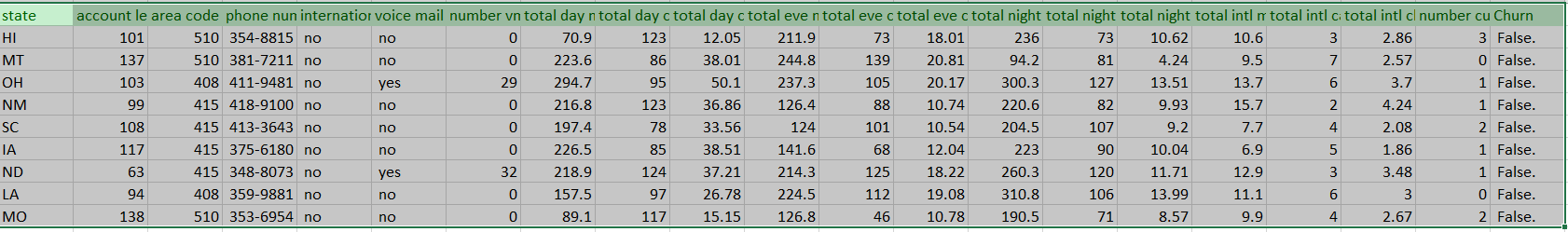
And after that we will apply the best model to predict the further data. To make the prediction we will divide the data into train and test, here in our project the data is already divided into train and test..

***“Note”*** : The codes written in R and Python will be separately attached..

Below is the 1st few observation of train data and we will perform above pre – processing techniques on top of this..



Test data :



**Predictor variables** :

* account length
* international plan
* voicemail plan
* number of voicemail messages
* total day minutes used
* day calls made
* total day charge
* total evening minutes
* total evening calls
* total evening charge
* total night minutes
* total night calls
* total night charge
* total international minutes used
* total international calls made
* total international charge
* number of customer service calls made

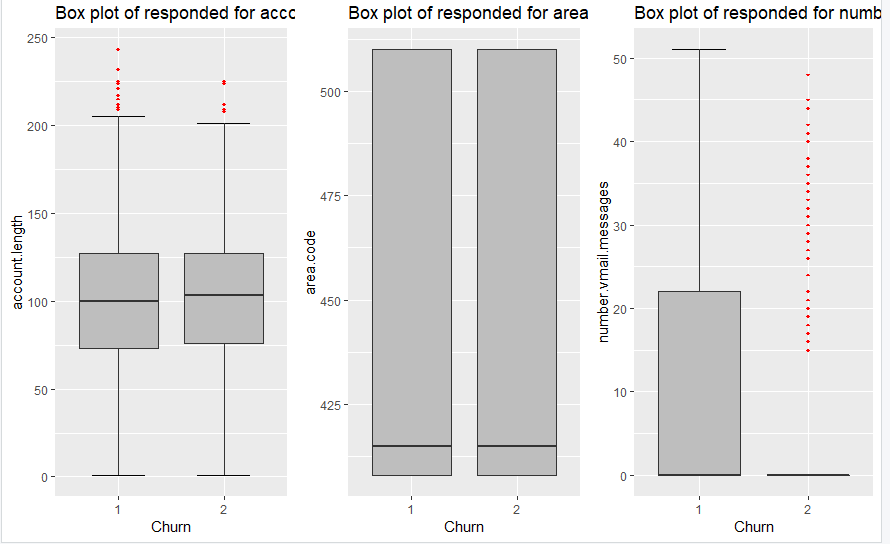
**Target Variable** :

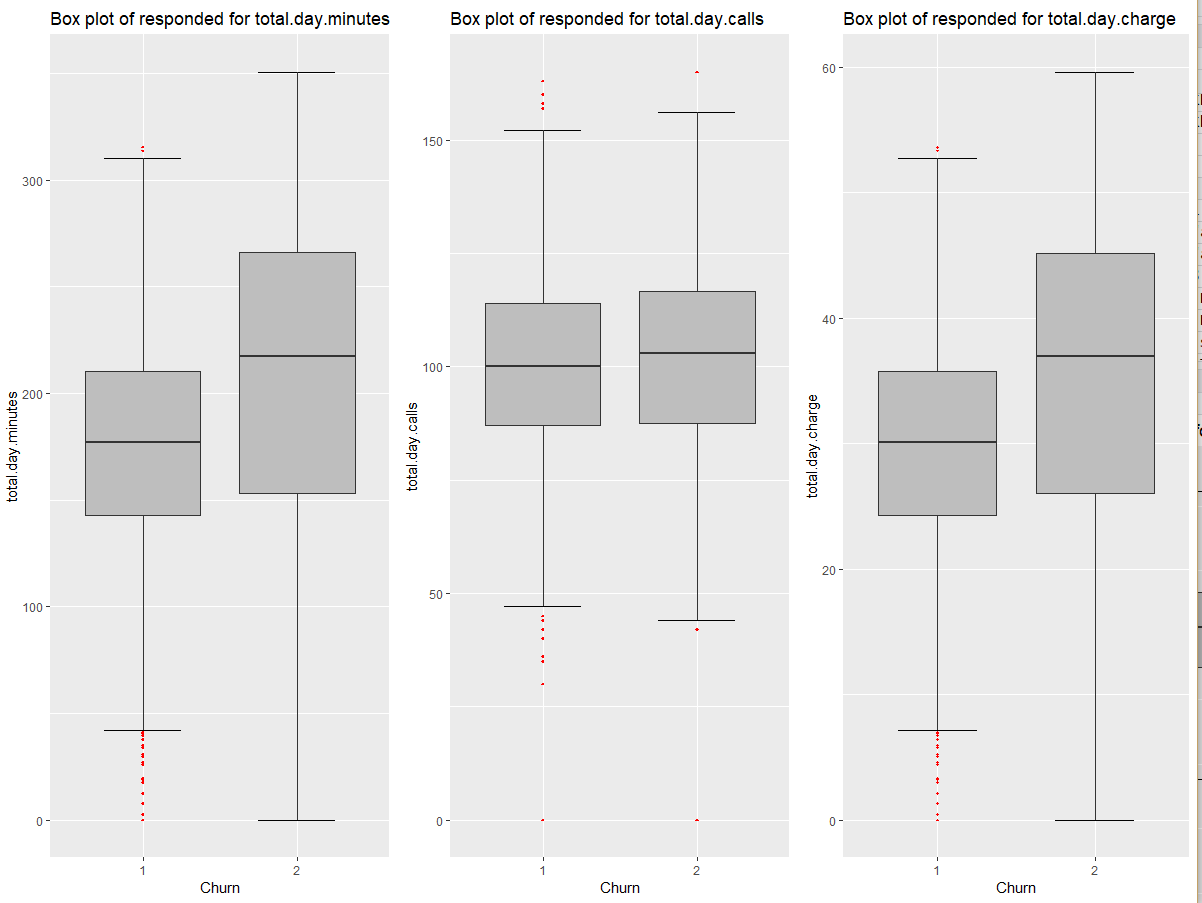
* Churn

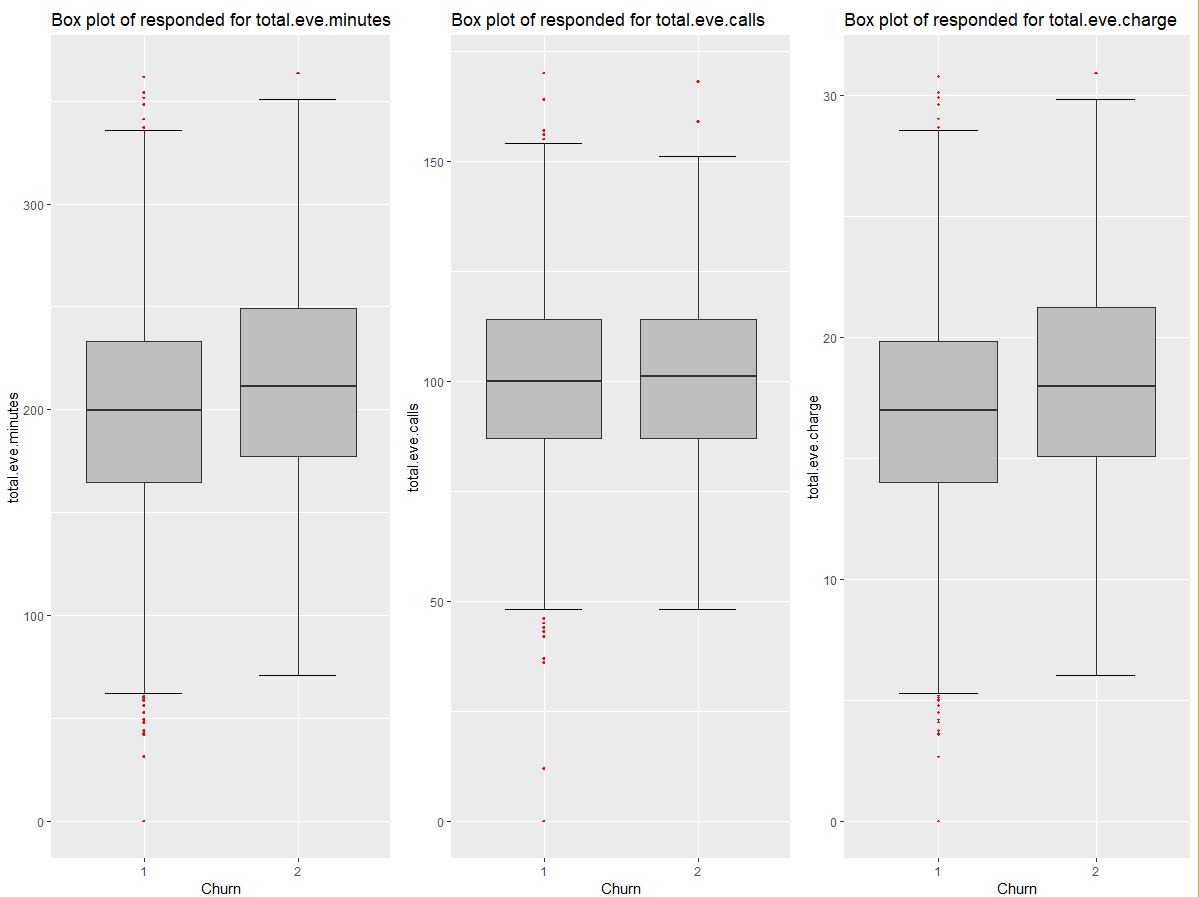
Now we will explain the pre – processing steps one by one and will apply the codes in R and Python..

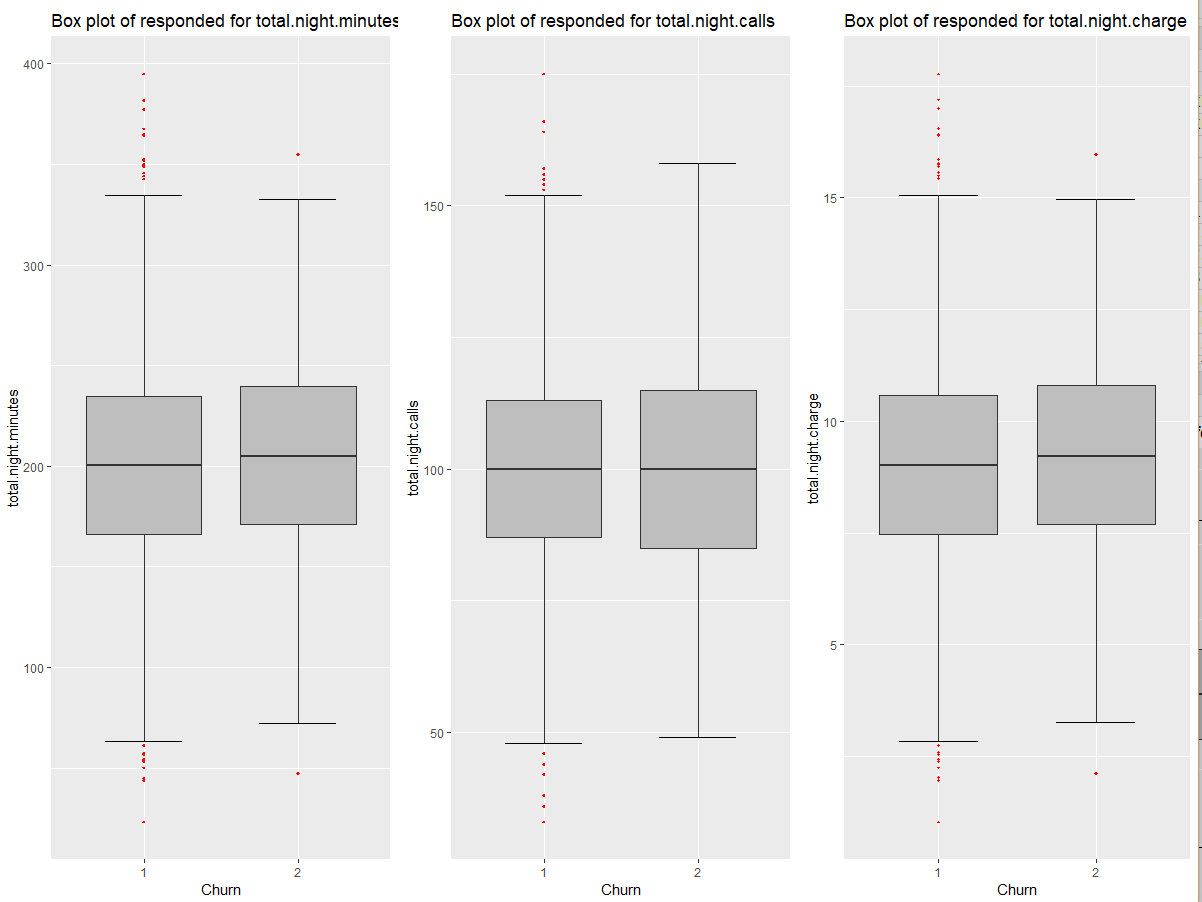
1. **Outlier Analysis**: In statistics, an **outlier** is an observation point that is distant from other observations. An **outlier** may be due to variability in the measurement or it may indicate experimental error; the latter are sometimes excluded from the data set. An **outlier** can cause serious problems in statistical **analysis.**

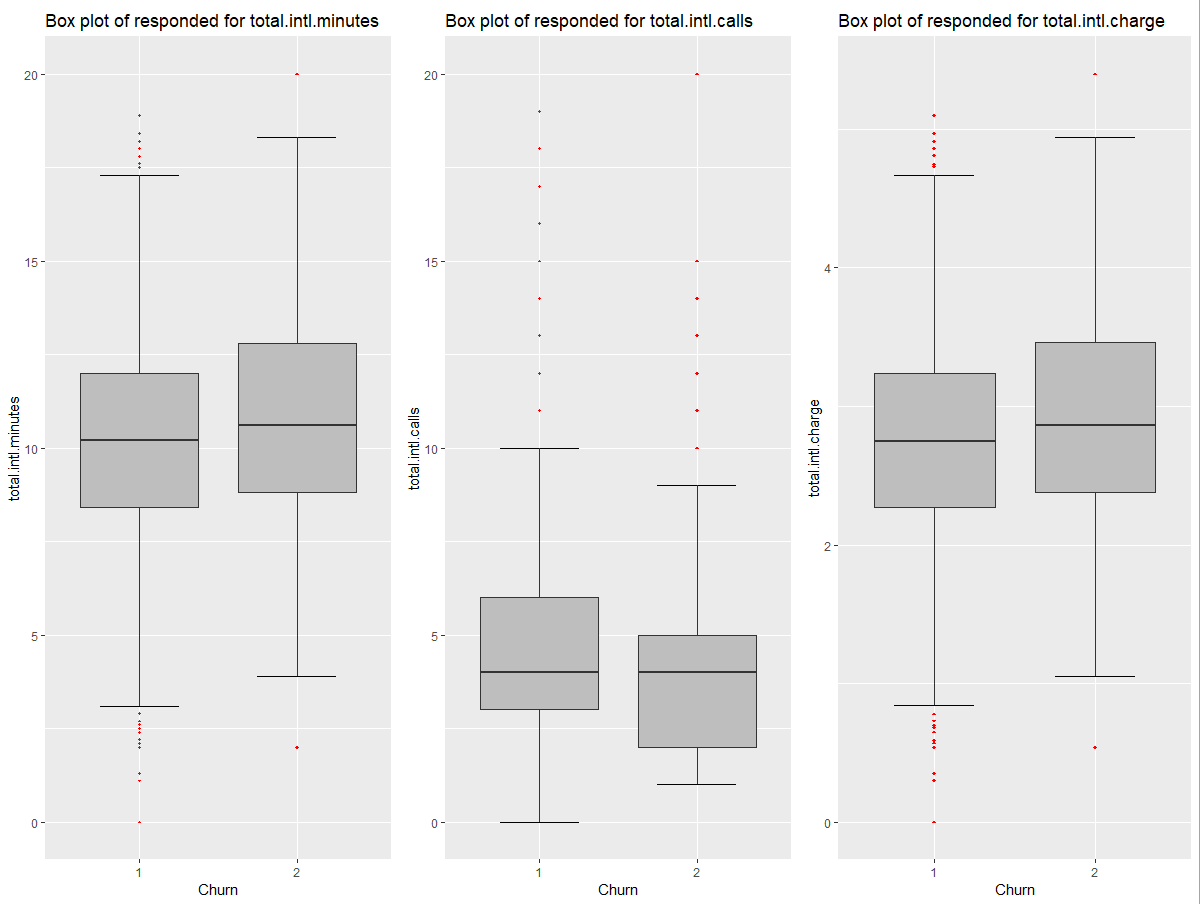
So , below are the outlier plots of each numerical variable using R..

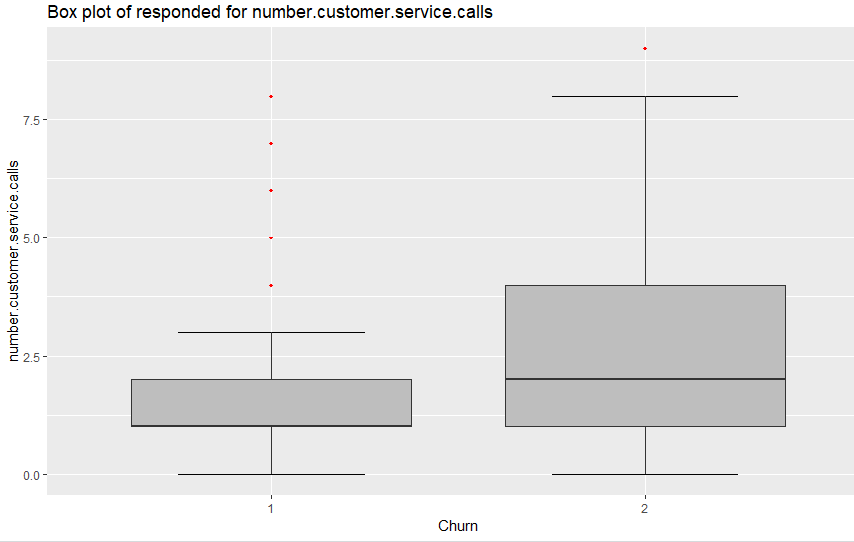












Now, we will remove the outliers using R and python and that is explained in R and python doc file..

1. **Missing Value Analysis** : In every huge data, there is a possibility that for some variables values gets missed and it may be due to many reasons. So if we do not correct those values or if we do not do pre processing in the data, the result may get biased and it will impact the prediction and accuracy.

In this data ( both train and test) there are no missing values. So we will move further..

1. **Feature Selection** : Feature selection is a pre – processing technique which help us to identify redundant variable as there is no point to keep 2 variables having same information. For numerical variable , we use correlation technique and for categorical variable we use chi square test.

After applying Feature selection , we have removed few of the variable in test and train data and this is explained in the codes written in R and python..

1. **Feature Scaling**: In order to set data in same range and to avoid biasing of the output we use feature scaling pre – processing technique. There are two methods of feature scaling i.e “***Normalisation”*** and other is “***standardization”***

If the data is normally distributed then we will use standardization and if data is not normally distributed then we will use Normalisation..

**Model Development**: After doing all the pre – processing techniques, we will select the model to do our analysis. Here we have used ***Decision tree .***For both R and python..

The commands are explained in both structure of R and python. The rules are extracted and we will take one reference decision tree rule to explain few lines of it. (See the attached file i.e c50Rules.txt) below is one rule which we extracted from c5.0 library.

Rule 0/3: (65, lift 9.1)

voice.mail.plan = 1

total.day.minutes > 0.7945927

total.eve.minutes > 0.5

total.night.minutes > 0.2334573

-> class 2 [0.985]

There are 3 parameters that will explain the authentication of model.

* ***Support***
* ***Confidence***
* ***Lift***

In above example, class 2 [0.985] here 0.985 is the confidence or we can say 98.5 % is the confidence

And Lift is 9.1

So it is said that, confidence should be greater than 80 % lift should be greater than 1 and support should be greater than 20 %

So there are other parameters also which explain the true nature of a model..

So we will do all the analysis in train data and then we will apply the same model in test data keeping dependent variable aside and then compare the value of predicted one to test one to check the accuracy and other parameters.

***“Notes”*** : The codes are attached separately in R and python format along with one c50Rules.txt file..

Thanks ,

Akshay Verma.