Project -4 (Predictive Modeling)

Telecom Customer Churn Prediction Assessment

By Saleesh Satheeshchandran
PGP-BABI 2019-20 (G-6)

Objective

Customer Churn is a burning problem for Telecom companies. In this project, we simulate one such case of customer churn where we work on a data of post-paid customers with a contract. The data has information about the customer usage behaviour, contract details and the payment details. The data also indicates which were the customers who cancelled their service. Based on this past data, we need to build a model which can predict whether a customer will cancel their service in the future or not.

Assumptions

There are no particular assumptions.

Tool used for the analysis

RStudio Version 1.2.1335
R Version 3.6.0

Input Data

The data is available in a spreadsheet format with .xisx file extension.
The same is uploaded into the R Studio with the help of the function “read-xlsx.

Steps involved in the analysis

- Exploratory Data Analysis
- Univariate Analysis
- Bivariate Analysis
- Missing values
- Outliers and their treatment
- Multicollinearity
- Logistics Regression
- KNN
- Naïve Bayes
- Model Comparison

Exploratory Data Analysis

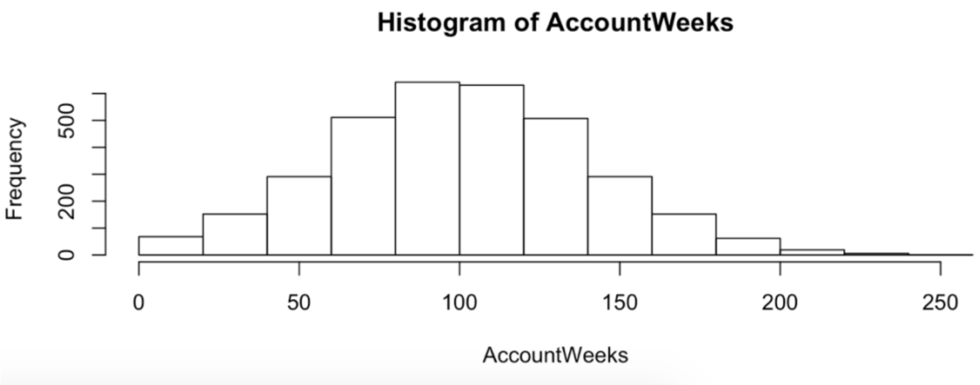
- All the variables are numeric in nature
- Continuous variables – Account weeks, Data usage, Day Mins, Day Calls, Monthly Charge, Overage Fee, RoamMins.
- Category variables – Churn, Contract Renewal, Data Plan, CustServCalls,
- The target variable, "Churn" is a binary variable.
- Monthly charge is highly skewed towards the left.

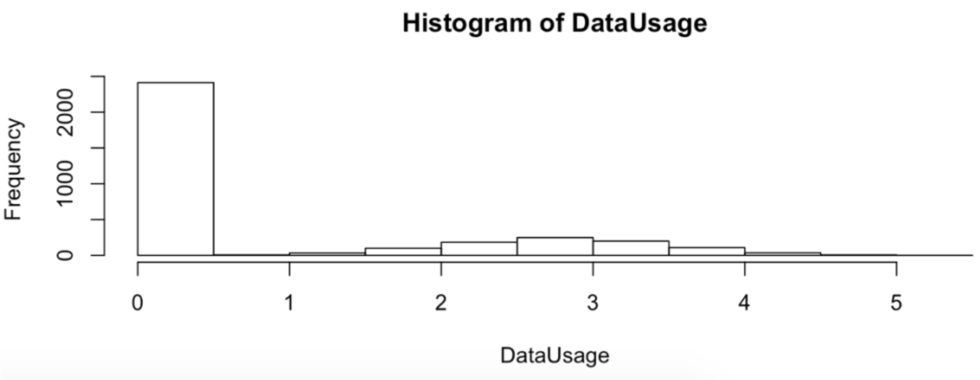
```
> str(Cellphone)
Classes 'tbl_df', 'tbl' and 'data.frame':       3333 obs. of  11 variables:
 $ Churn          : num  0 0 0 0 0 0 0 0 0 0 ...
 $ AccountWeeks   : num  128 107 137 84 75 118 121 147 117 141 ...
 $ ContractRenewal: num  1 1 1 0 0 0 1 0 1 0 ...
 $ DataPlan       : num  1 1 0 0 0 0 1 0 0 1 ...
 $ DataUsage      : num  2.7 3.7 0 0 0 0 2.03 0 0.19 3.02 ...
 $ CustServCalls  : num  1 1 0 2 3 0 3 0 1 0 ...
 $ DayMins        : num  265 162 243 299 167 ...
 $ DayCalls       : num  110 123 114 71 113 98 88 79 97 84 ...
 $ MonthlyCharge  : num  89 82 52 57 41 57 87.3 36 63.9 93.2 ...
 $ OverageFee     : num  9.87 9.78 6.06 3.1 7.42 ...
 $ RoamMins       : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
> 
```

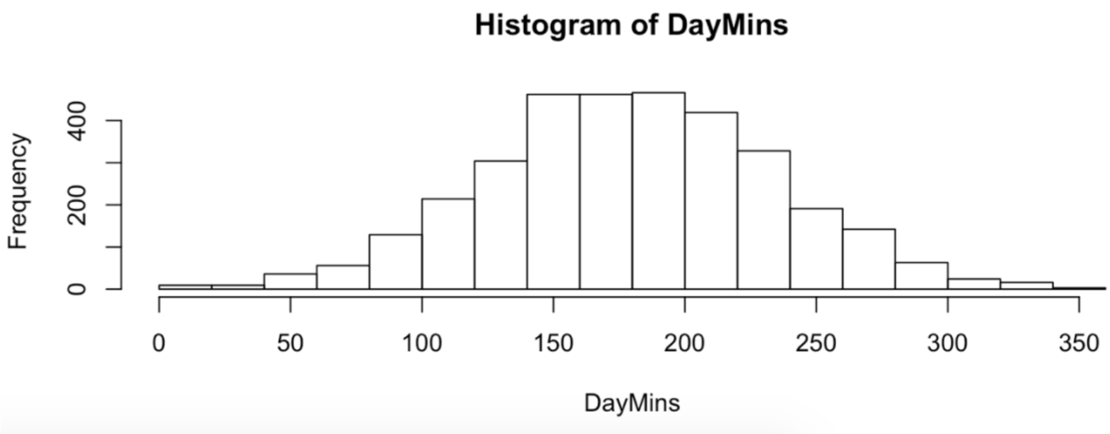
```
> head(Cellphone)
  Churn AccountWeeks ContractRenewal DataPlan DataUsage CustServCalls DayMins DayCalls
1     0          128               1        1       2.7             1   265.1      110
2     0          107               1        1       3.7             1   161.6      123
3     0          137               1        0       0.0             0   243.4      114
4     0           84               0        0       0.0             2   299.4       71
5     0           75               0        0       0.0             3   166.7      113
6     0          118               0        0       0.0             0   223.4       98
  MonthlyCharge OverageFee RoamMins
1            89       9.87     10.0
2            82       9.78     13.7
3            52       6.06     12.2
4            57       3.10      6.6
5            41       7.42     10.1
6            57      11.03      6.3
> 
```

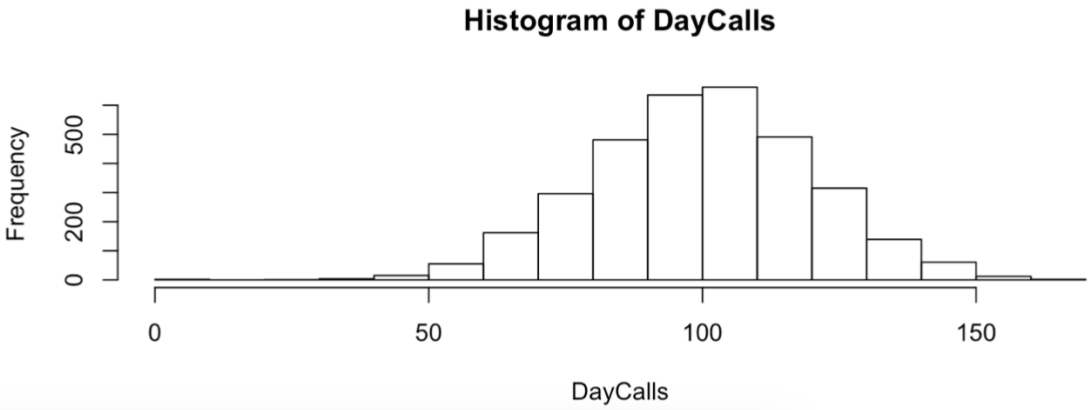
```
> summary(Cellphone)
     Churn         AccountWeeks   ContractRenewal     DataPlan        DataUsage      CustServCalls
 Min.   :0.0000   Min.   :  1.0   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
 1st Qu.:0.0000   1st Qu.: 74.0   1st Qu.:1.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.000
 Median :0.0000   Median :101.0   Median :1.0000   Median :0.0000   Median :0.0000   Median :1.000
 Mean   :0.1449   Mean   :101.1   Mean   :0.9031   Mean   :0.2766   Mean   :0.8165   Mean   :1.563
 3rd Qu.:0.0000   3rd Qu.:127.0   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.7800   3rd Qu.:2.000
 Max.   :1.0000   Max.   :243.0   Max.   :1.0000   Max.   :1.0000   Max.   :5.4000   Max.   :9.000
    DayMins         DayCalls     MonthlyCharge      OverageFee       RoamMins
 Min.   :  0.0   Min.   :  0.0   Min.   : 14.00   Min.   : 0.00   Min.   : 0.00
 1st Qu.:143.7   1st Qu.: 87.0   1st Qu.: 45.00   1st Qu.: 8.33   1st Qu.: 8.50
 Median :179.4   Median :101.0   Median : 53.50   Median :10.07   Median :10.30
 Mean   :179.8   Mean   :100.4   Mean   : 56.31   Mean   :10.05   Mean   :10.24
 3rd Qu.:216.4   3rd Qu.:114.0   3rd Qu.: 66.20   3rd Qu.:11.77   3rd Qu.:12.10
 Max.   :350.8   Max.   :165.0   Max.   :111.30   Max.   :18.19   Max.   :20.00
> 
```

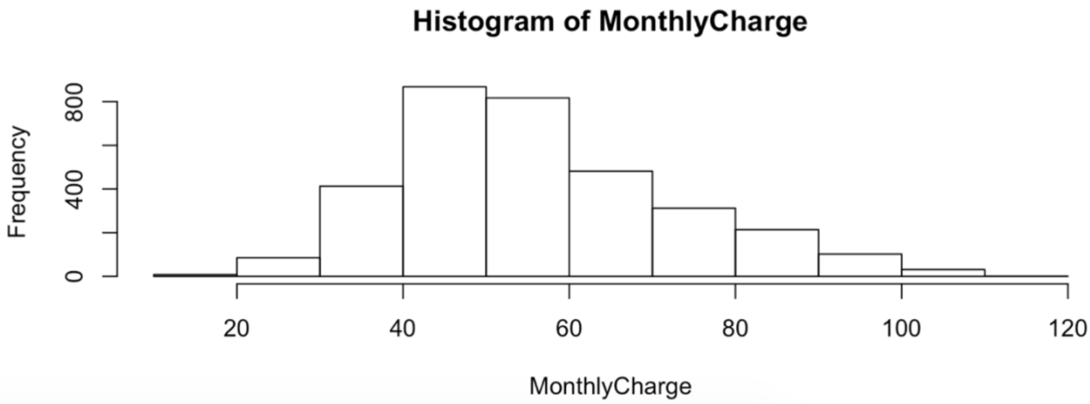
- **Histograms**
  All the continuous variables were tested with histograms.
  Data usage does not show a curve nature.

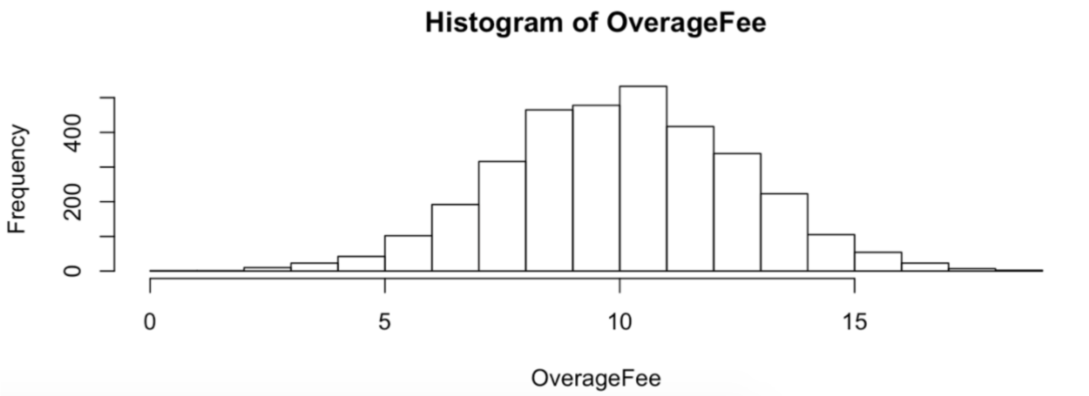


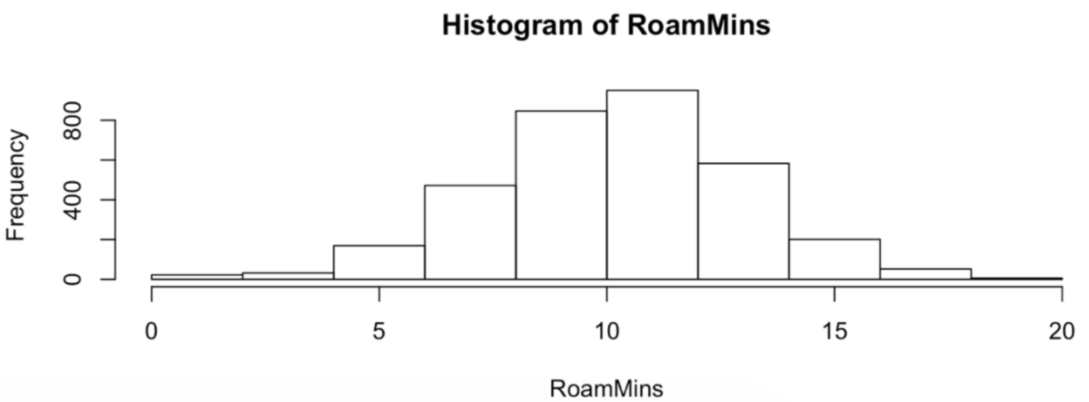




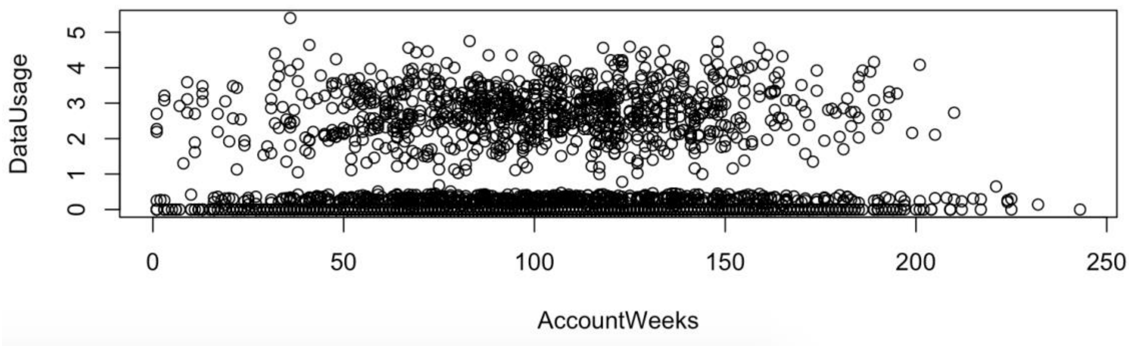

Histogram of DayCalls
Frequency
0
200
500
0
50
100
150
DayCalls

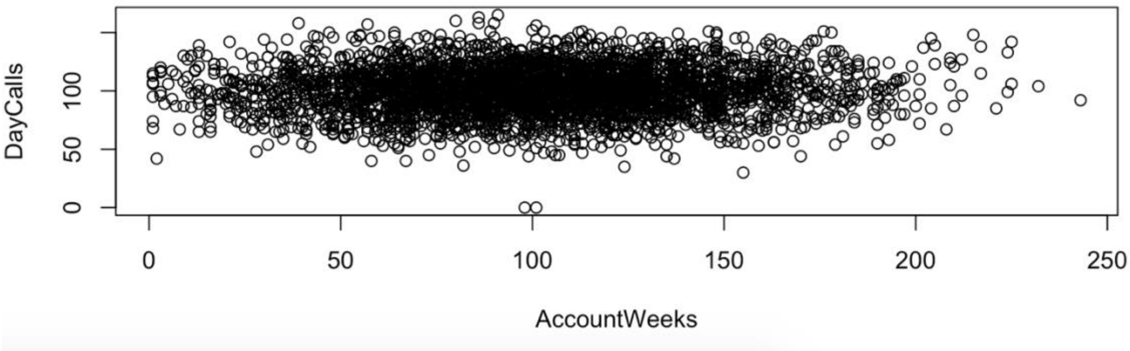

Histogram of MonthlyCharge
Frequency
0
400
800
20
40
60
80
100
120
MonthlyCharge

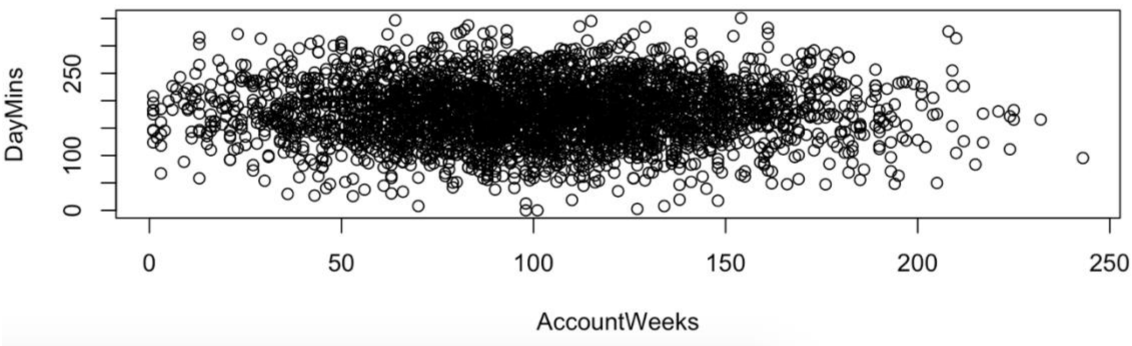

Histogram of OverageFee
Frequency
0
200
400
0
5
10
15
OverageFee

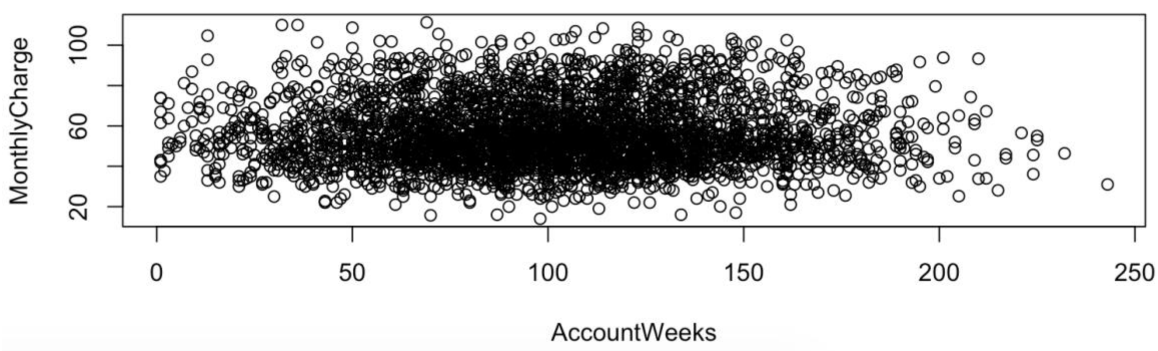

Histogram of RoamMins
Frequency
0
400
800
0
5
10
15
20
RoamMins

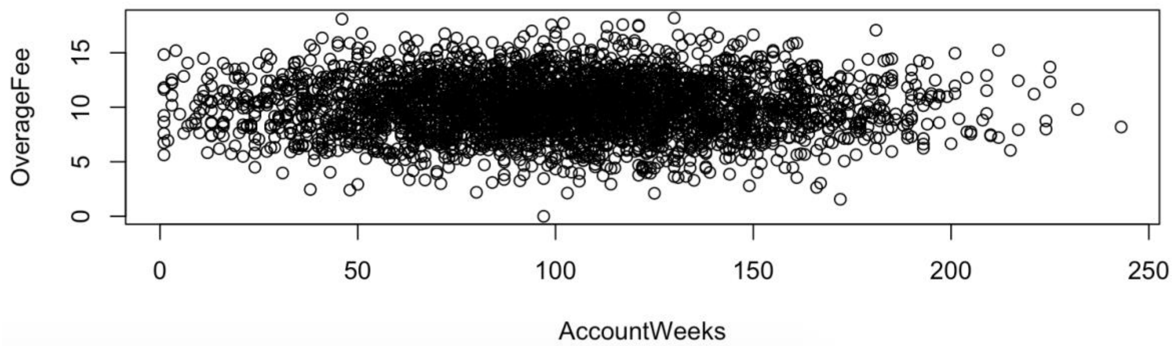
Bivariate analysis

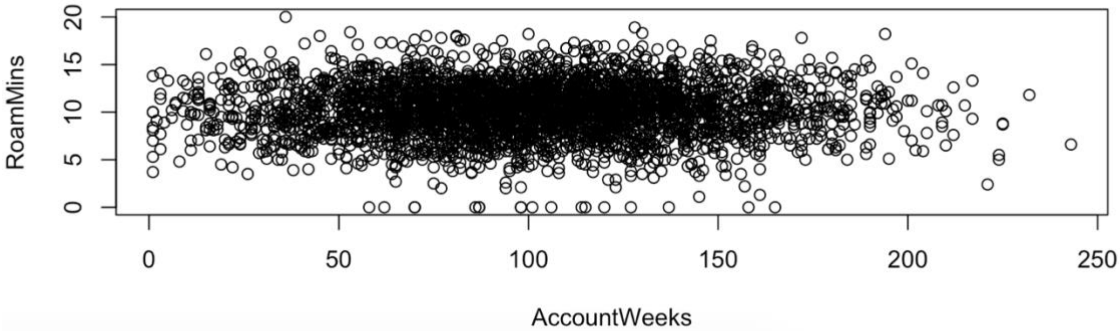


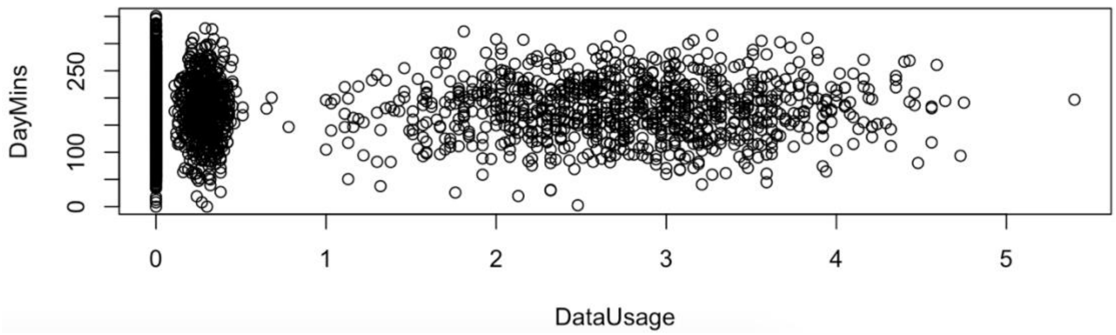


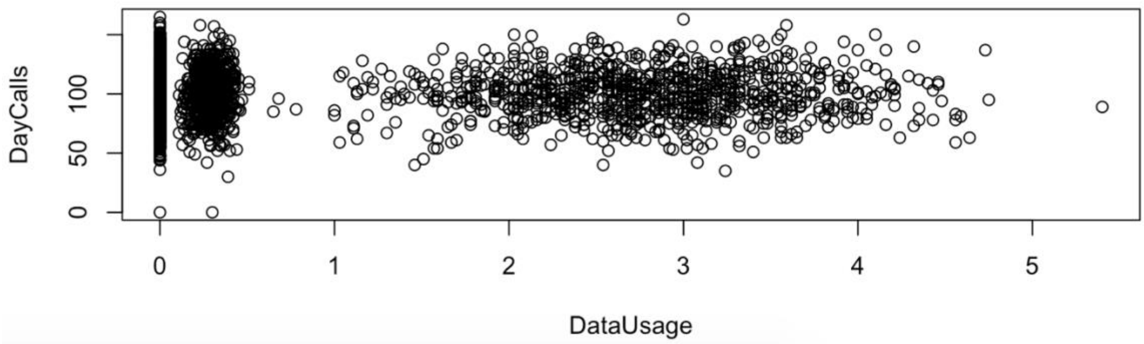


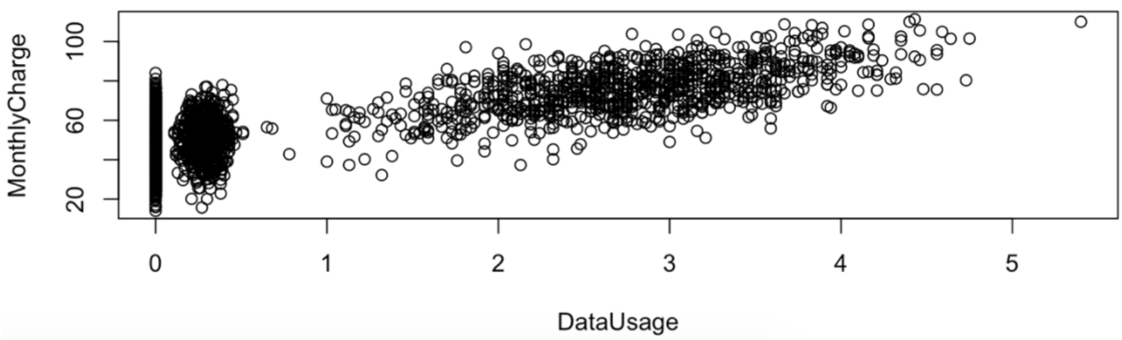


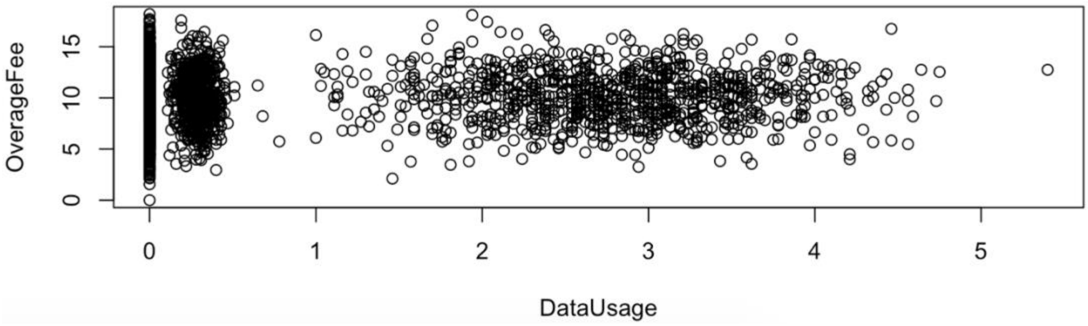

OverageFee
0
5
10
15
0
50
100
150
200
250
AccountWeeks

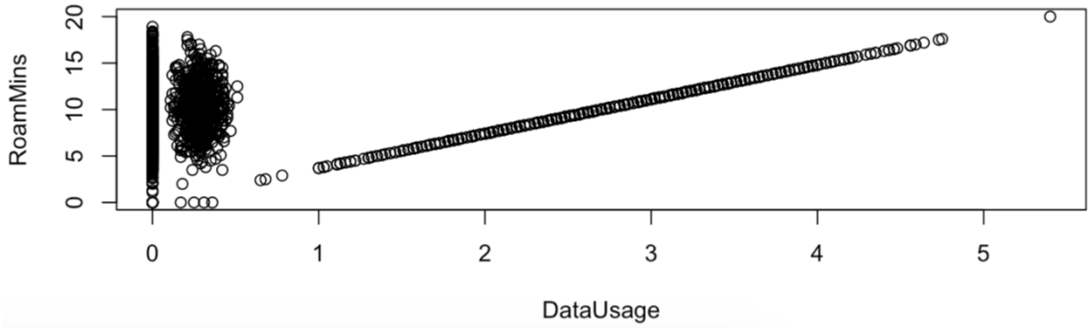

RoamMins
0
5
10
15
20
0
50
100
150
200
250
AccountWeeks

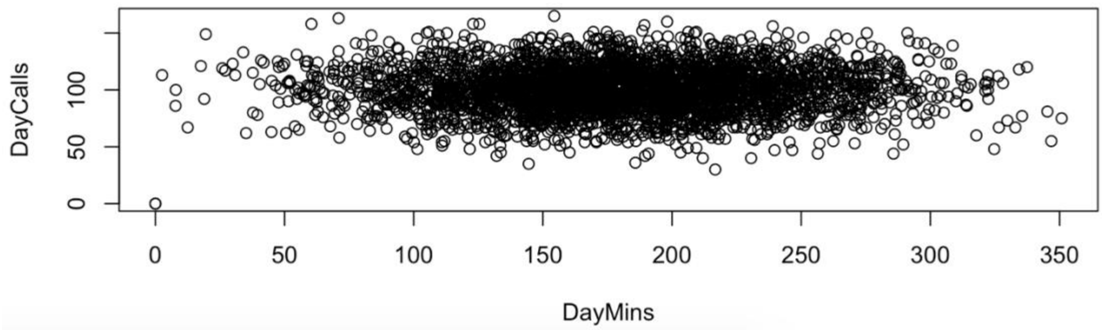

DayMins
0
100
250
0
1
2
3
4
5
DataUsage

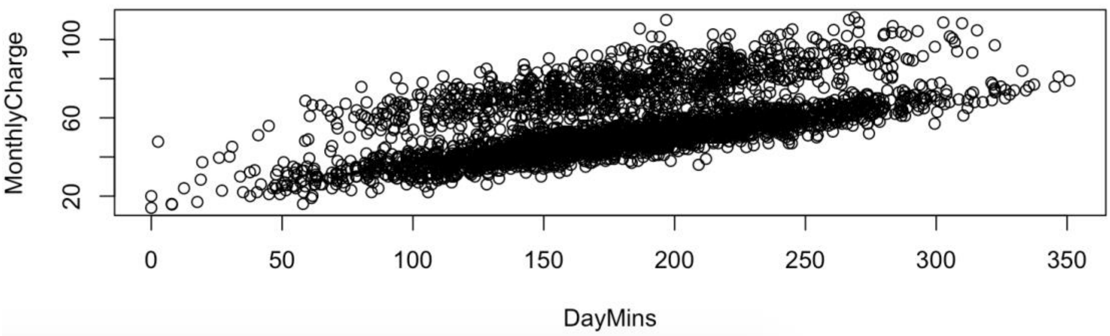

DayCalls
0
50
100
0
1
2
3
4
5
DataUsage

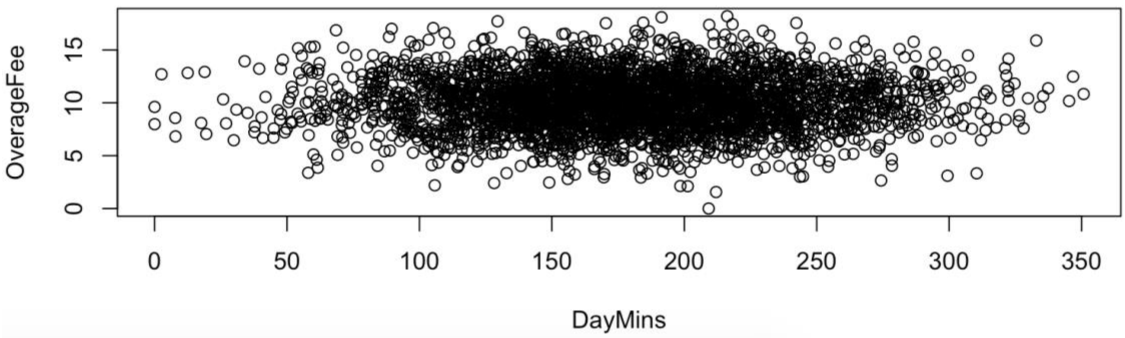

MonthlyCharge
20
60
100
0
1
2
3
4
5
DataUsage

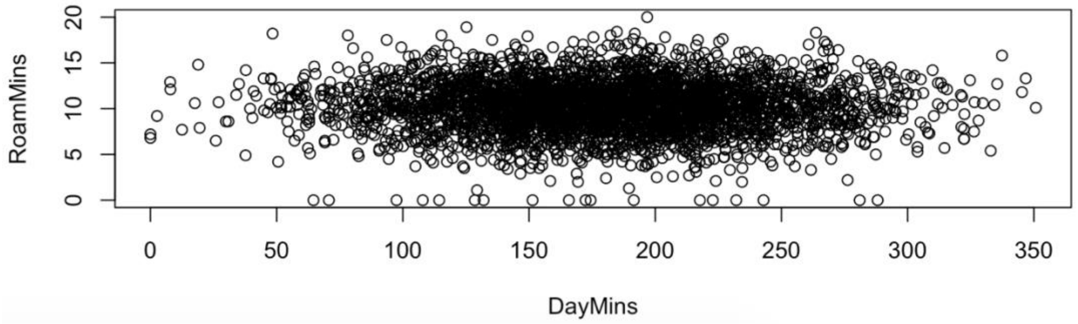

OverageFee
0
5
10
15
0
1
2
3
4
5
DataUsage

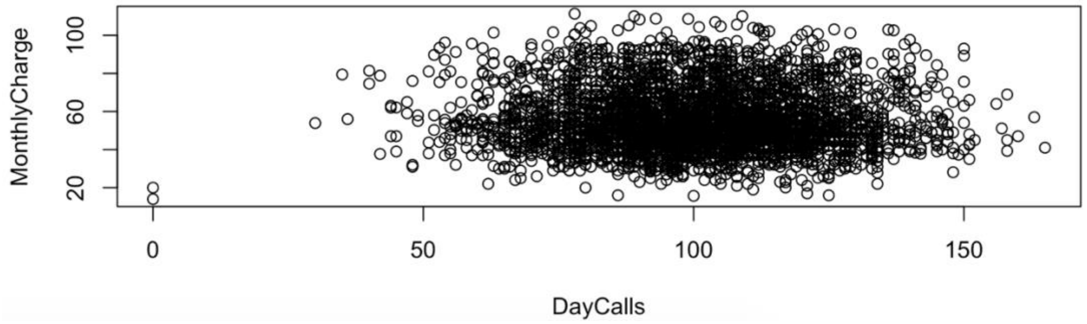

RoamMins
0
5
10
15
20
0
1
2
3
4
5
DataUsage

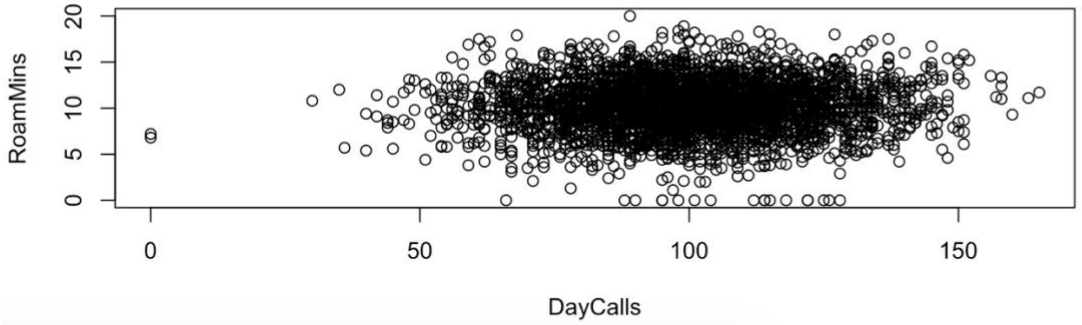

DayCalls
0
50
100
0
50
100
150
200
250
300
350
DayMins

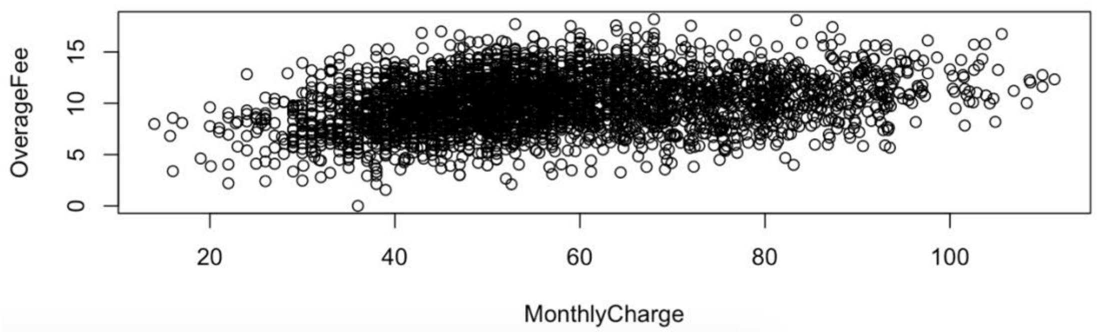

MonthlyCharge
20
60
100
0
50
100
150
200
250
300
350
DayMins

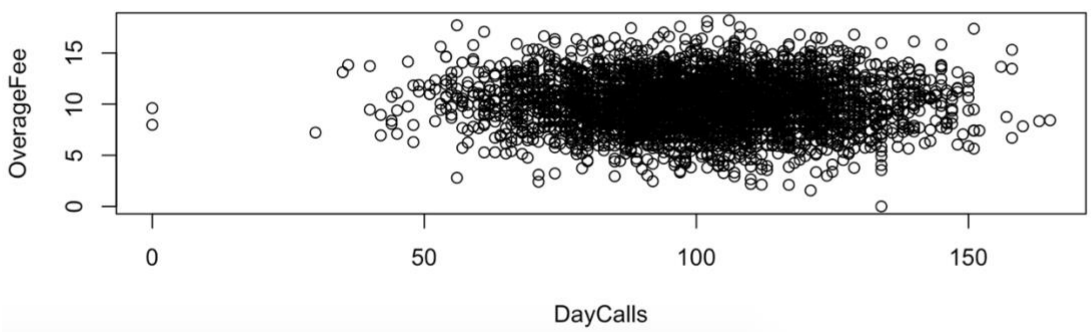

OverageFee
0
5
10
15
0
50
100
150
200
250
300
350
DayMins

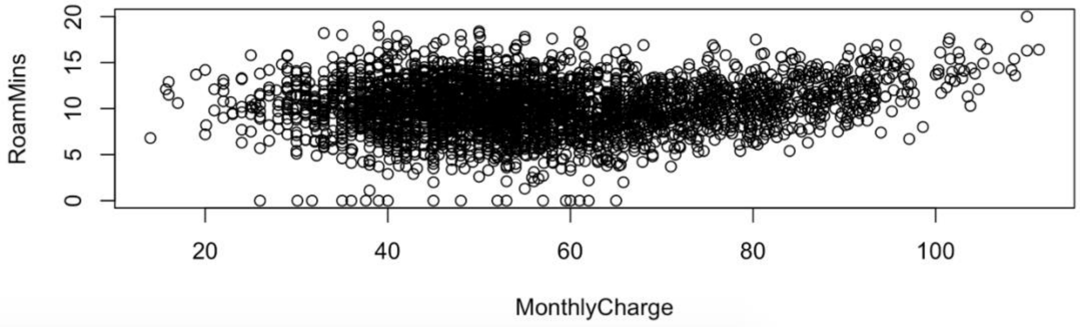

RoamMins
0 5 10 15 20
0 50 100 150 200 250 300 350
DayMins

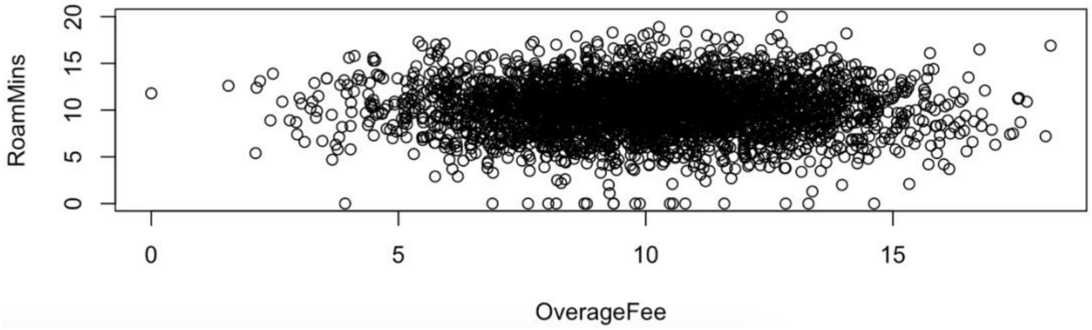

MonthlyCharge
20 60 100
0 50 100 150
DayCalls


RoamMins
0 5 10 15 20
0 50 100 150
DayCalls


OverageFee
0 5 10 15
20 40 60 80 100
MonthlyCharge


OverageFee
0 5 10 15
0 50 100 150
DayCalls



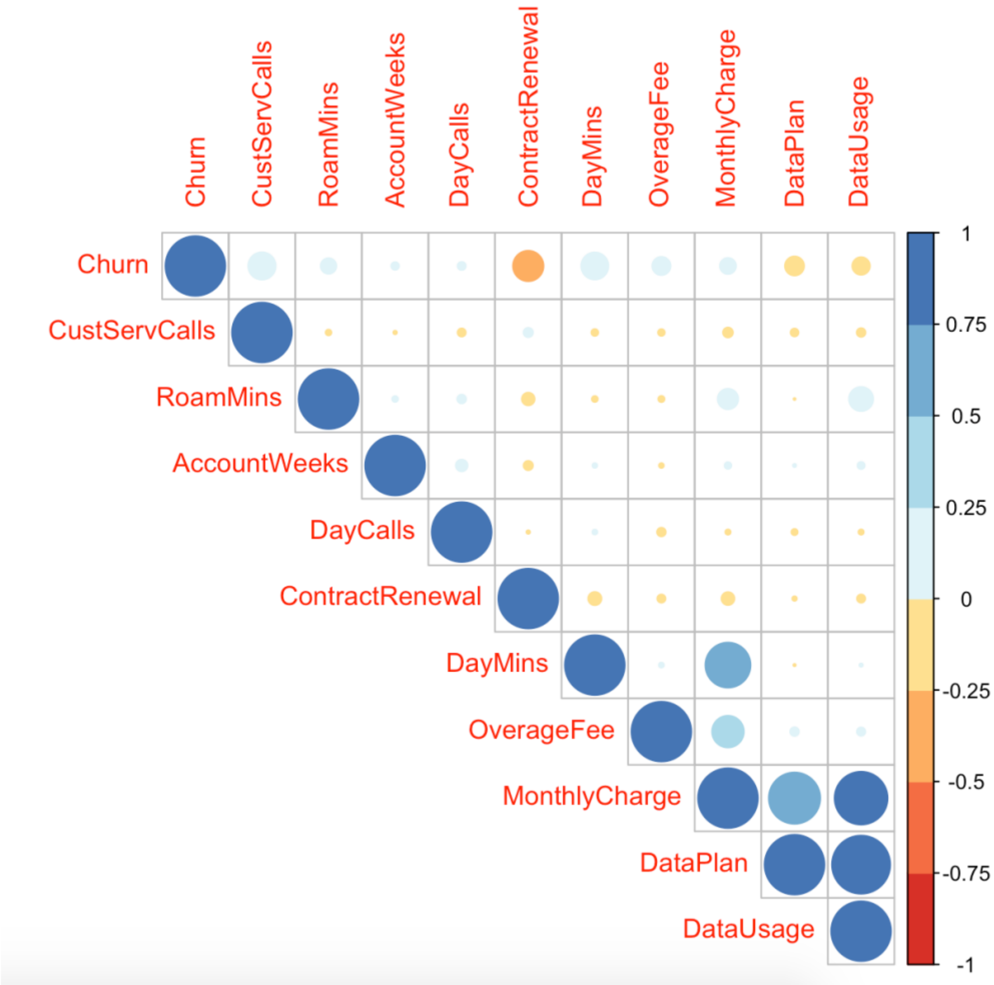


Checking for correlation

There is a significant correlation between the below pairs of variables.

- Data Plan Vs Data usage
- Monthly Charge Vs Day Mins
- Monthly Charge Vs Data Plan
- Monthly Plan Vs Data Usage

```
> CP
                     Churn AccountWeeks ContractRenewal     DataPlan    DataUsage CustServCalls
Churn           1.00000000  0.016540742    -0.259851847 -0.102148141 -0.087194509   0.208749999
AccountWeeks    0.01654074  1.000000000    -0.024734655  0.002918409  0.014390757  -0.003795939
ContractRenewal -0.25985185 -0.024734655    1.000000000 -0.006006371 -0.019222913   0.024521956
DataPlan        -0.10214814  0.002918409   -0.006006371  1.000000000  0.945981734  -0.017823944
DataUsage       -0.08719451  0.014390757   -0.019222913  0.945981734  1.000000000  -0.021722518
CustServCalls    0.20875000 -0.003795939    0.024521956 -0.017823944 -0.021722518   1.000000000
DayMins          0.20515083  0.006216021   -0.049395824 -0.001684069  0.003175951  -0.013423186
DayCalls         0.01845931  0.038469882   -0.003754626 -0.011085902 -0.007962079  -0.018941930
MonthlyCharge    0.07231271  0.012580670   -0.047291399  0.737489653  0.781660429  -0.028016853
OverageFee       0.09281243 -0.006749462   -0.019104644  0.021525559  0.019637372  -0.012964219
RoamMins         0.06823878  0.009513902   -0.045870743 -0.001317871  0.162745576  -0.009639680
                     DayMins     DayCalls MonthlyCharge   OverageFee     RoamMins
Churn            0.205150829  0.018459312   0.072312711  0.092812426  0.068238776
AccountWeeks     0.006216021  0.038469882   0.012580670 -0.006749462  0.009513902
ContractRenewal -0.049395824 -0.003754626  -0.047291399 -0.019104644 -0.045870743
DataPlan        -0.001684069 -0.011085902   0.737489653  0.021525559 -0.001317871
DataUsage        0.003175951 -0.007962079   0.781660429  0.019637372  0.162745576
CustServCalls   -0.013423186 -0.018941930  -0.028016853 -0.012964219 -0.009639680
DayMins          1.000000000  0.006750414   0.567967924  0.007038214 -0.010154586
DayCalls         0.006750414  1.000000000  -0.007963218 -0.021448602  0.021564794
MonthlyCharge    0.567967924 -0.007963218   1.000000000  0.281766048  0.117432607
OverageFee       0.007038214 -0.021448602   0.281766048  1.000000000 -0.011023336
RoamMins        -0.010154586  0.021564794   0.117432607 -0.011023336  1.000000000
>
```



Checking Multicollinearity and treating it

The below process is followed to check multicollinearity.

1. Create a linear regression model with all variables.
2. Check the variable inflation factor.
3. Check the summary and identify variables showing considerable inflation.

```
> lm.churn.full=lm(Churn~AccountWeeks+ContractRenewal+DataPlan+DataUsage+CustServCalls+DayMin
s+DayCalls+MonthlyCharge+OverageFee+RoamMins)
> vif(lm.churn.full)
   AccountWeeks ContractRenewal        DataPlan       DataUsage   CustServCalls
       1.003791        1.007216       12.473470     1964.800207        1.001945
        DayMins        DayCalls   MonthlyCharge      OverageFee        RoamMins
    1031.490608        1.002935     3243.300555      224.639750        1.346583
> summary(lm.churn.full)

Call:
lm(formula = Churn ~ AccountWeeks + ContractRenewal + DataPlan +
    DataUsage + CustServCalls + DayMins + DayCalls + MonthlyCharge +
    OverageFee + RoamMins)

Residuals:
     Min       1Q   Median       3Q      Max
-0.66572 -0.16629 -0.08236  0.02060  1.08844

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -1.433e-01  5.363e-02  -2.672 0.007580 **
AccountWeeks     8.888e-05  1.396e-04   0.637 0.524402
ContractRenewal -2.993e-01  1.882e-02 -15.904  < 2e-16 ***
DataPlan        -4.175e-02  4.381e-02  -0.953 0.340650
DataUsage       -2.835e-02  1.933e-01  -0.147 0.883401
CustServCalls    5.829e-02  4.222e-03  13.804  < 2e-16 ***
DayMins          1.021e-03  3.272e-03   0.312 0.754936
DayCalls         3.409e-04  2.769e-04   1.231 0.218433
MonthlyCharge    1.428e-03  1.924e-02   0.074 0.940838
OverageFee       1.046e-02  3.280e-02   0.319 0.749780
RoamMins         8.765e-03  2.307e-03   3.800 0.000147 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3203 on 3322 degrees of freedom
Multiple R-squared:  0.1747,	Adjusted R-squared:  0.1722
F-statistic: 70.31 on 10 and 3322 DF,  p-value: < 2.2e-16

>
```

The following variables show high **contribution to multicollinearity.**

- Dataplan
- Data usage
- Daymins
- Monthly Charge
- Overage Fee

Multiple combinations of linear model are tried by removing variables.

Final combination of significant variables are arrived at as per below.

```
> lm.churn.partial=lm(Churn~AccountWeeks+ContractRenewal+CustServCalls+DayMins+DayCalls+Month
lyCharge+OverageFee+RoamMins)
> vif(lm.churn.partial)
   AccountWeeks ContractRenewal   CustServCalls         DayMins        DayCalls 
       1.002383        1.006214        1.001760        1.552928        1.002924 
  MonthlyCharge      OverageFee        RoamMins 
       1.712815        1.134234        1.030120 
> summary(lm.churn.partial)

Call:
lm(formula = Churn ~ AccountWeeks + ContractRenewal + CustServCalls + 
    DayMins + DayCalls + MonthlyCharge + OverageFee + RoamMins)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.66777 -0.16753 -0.08162  0.01926  1.09167 

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)    
(Intercept)     -1.525e-01  5.148e-02  -2.962  0.00308 ** 
AccountWeeks     9.278e-05  1.395e-04   0.665  0.50604    
ContractRenewal -2.995e-01  1.881e-02 -15.924  < 2e-16 ***
CustServCalls    5.825e-02  4.221e-03  13.799  < 2e-16 ***
DayMins          1.748e-03  1.269e-04  13.767  < 2e-16 ***
DayCalls         3.405e-04  2.769e-04   1.230  0.21884    
MonthlyCharge   -2.832e-03  4.421e-04  -6.405 1.71e-10 ***
OverageFee       1.770e-02  2.330e-03   7.596 3.94e-14 ***
RoamMins         9.828e-03  2.017e-03   4.872 1.15e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3203 on 3324 degrees of freedom
Multiple R-squared:  0.1744,	Adjusted R-squared:  0.1725 
F-statistic:  87.8 on 8 and 3324 DF,  p-value: < 2.2e-16

> 
```

The insignificant variables are removed from the data.

Identify Missing values.

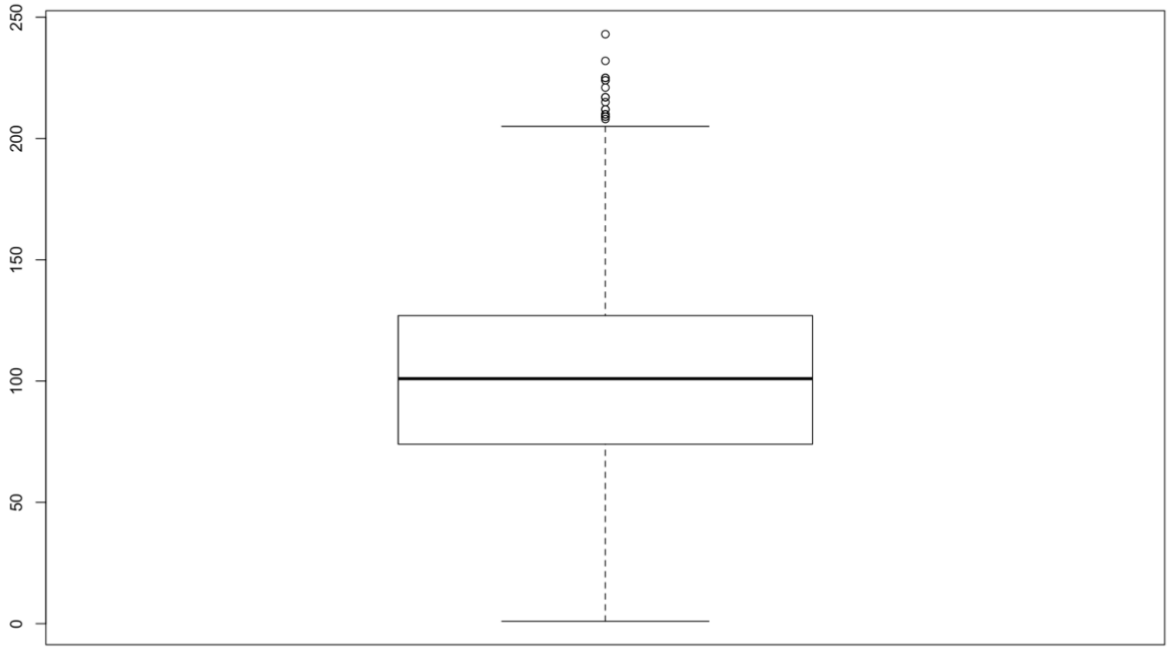
There are no missing values in the significant variables.

```
> summary(is.na(CellphoneData))
   Churn          AccountWeeks    ContractRenewal CustServCalls    DayMins       
 Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical  
 FALSE:3333      FALSE:3333      FALSE:3333      FALSE:3333      FALSE:3333     
  DayCalls       MonthlyCharge   OverageFee       RoamMins      
 Mode :logical   Mode :logical   Mode :logical   Mode :logical  
 FALSE:3333      FALSE:3333      FALSE:3333      FALSE:3333     
> 
```

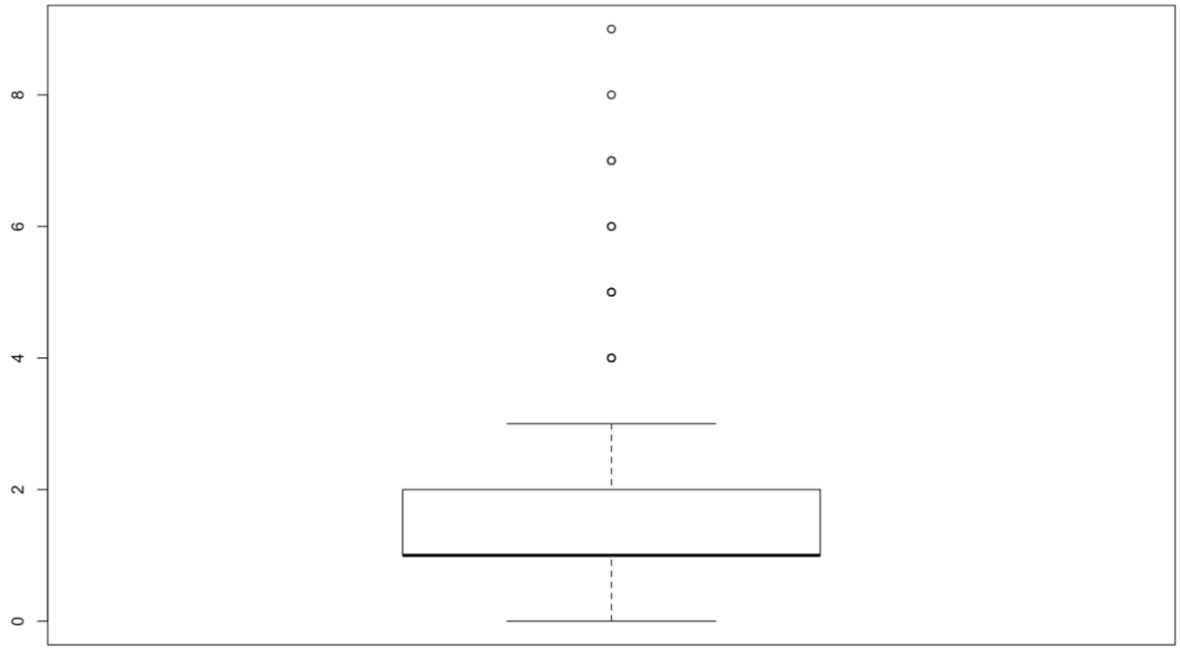
Identify Outliers.

Boxplot method is used to identify outliers.

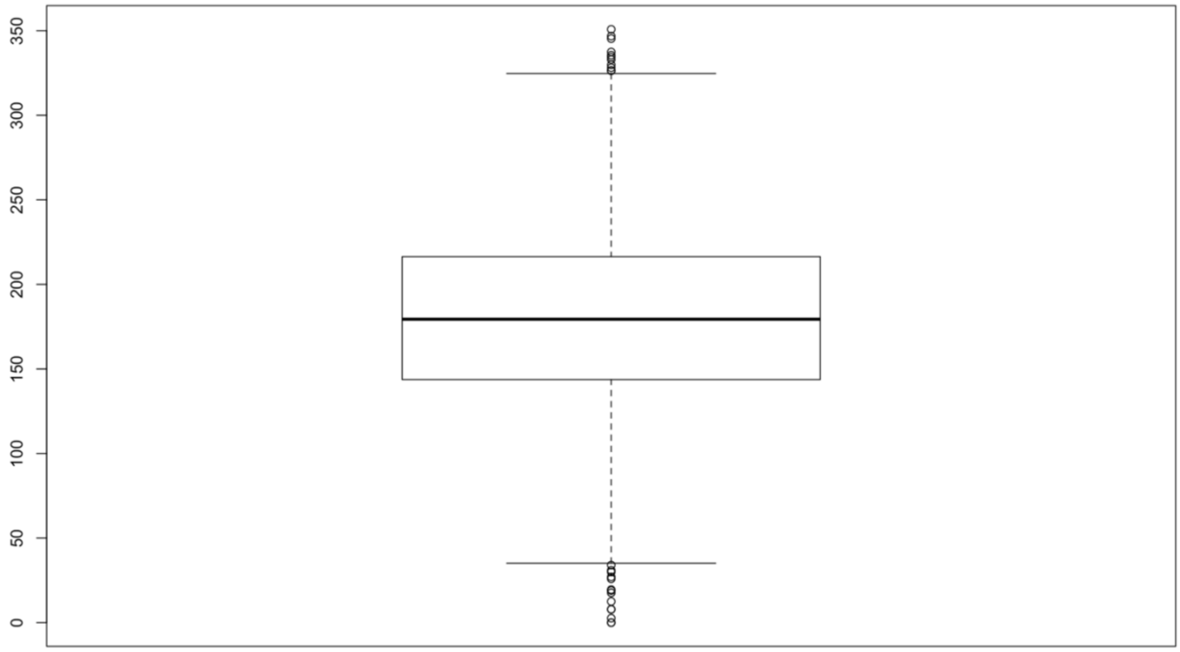
- Account weeks show between 10 to 15 outliers towards the upper limit. These are required in the data.



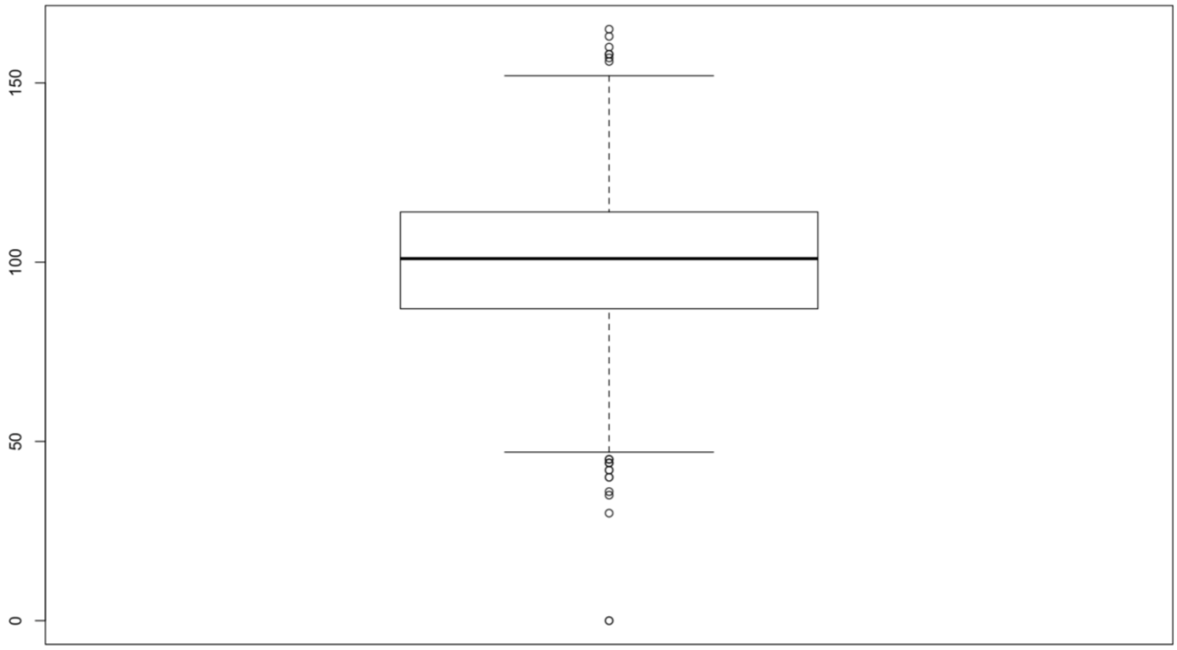
- Cust service calls shows about 6 outliers. These are required in the data.



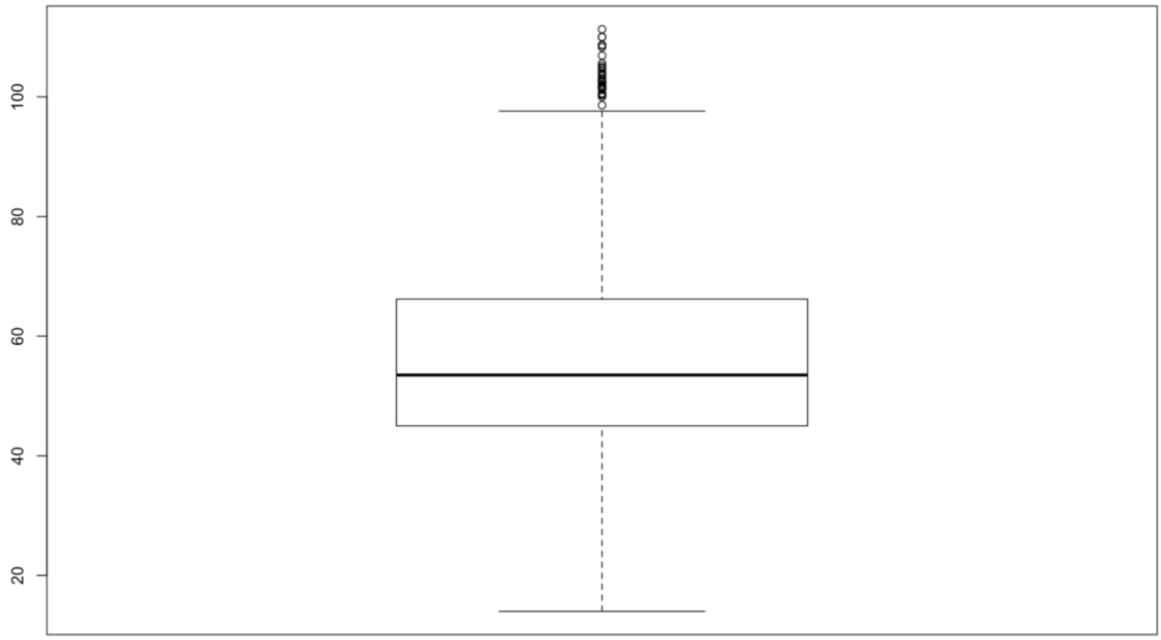
- Daymins



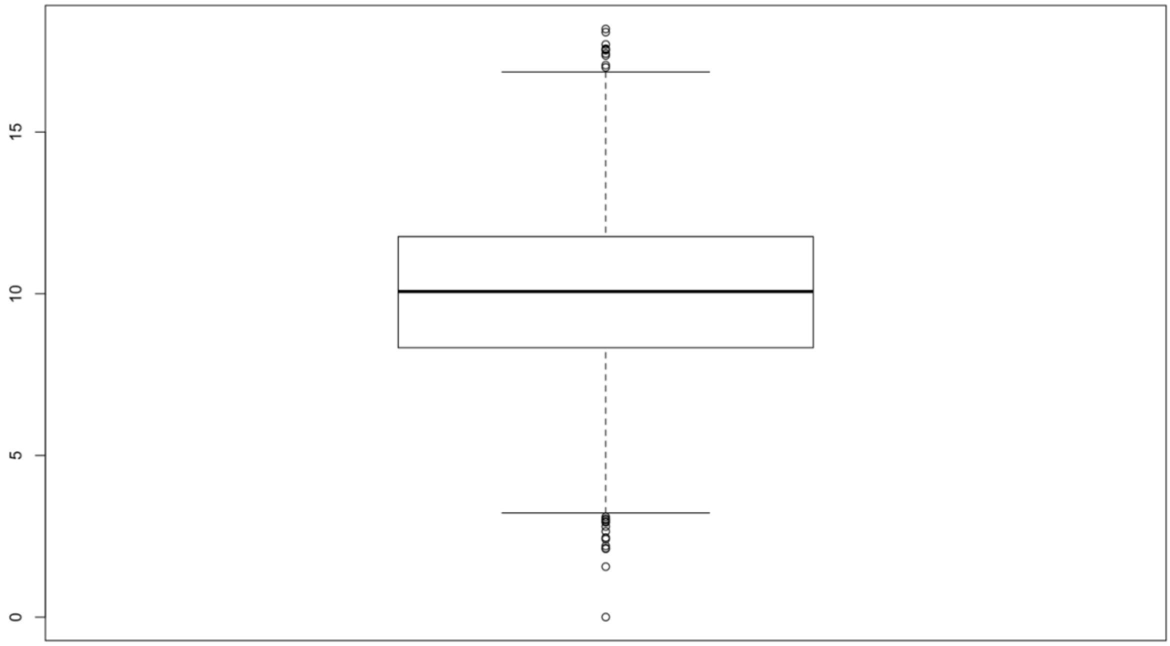
- Daycalls show 10 each of outliers in both the upper and the lower limit. These are also included in the model.
-



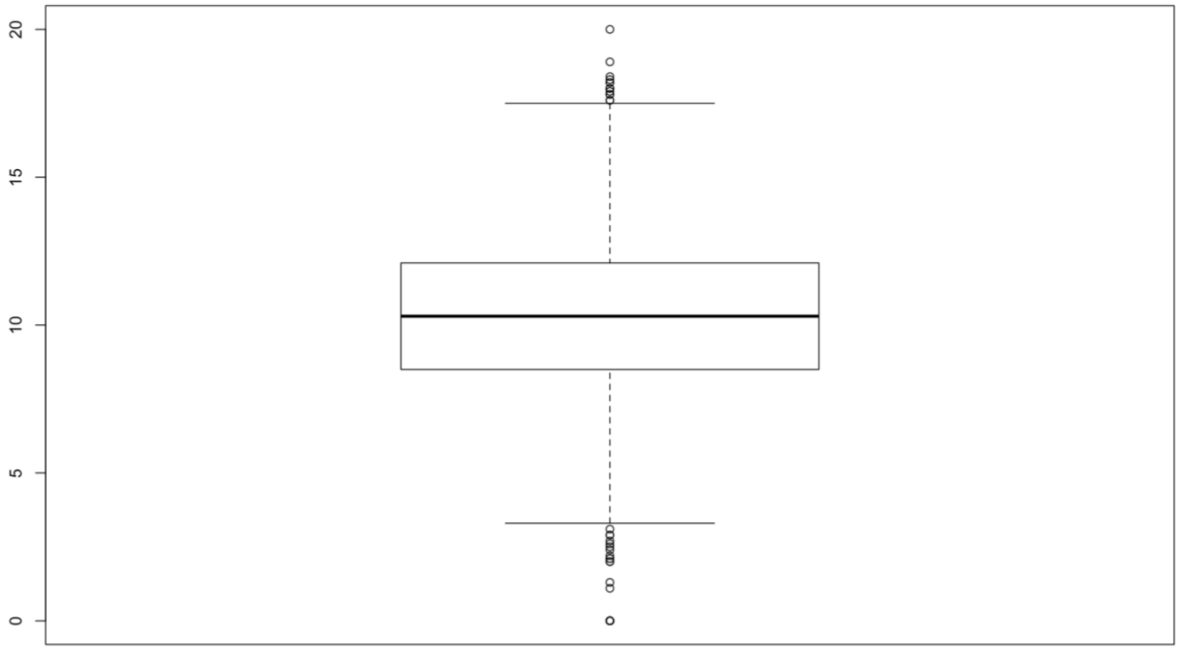
- Monthly Charges show a large number of outliers towards the upper side. These are also considered in the model.



- Overage Fee shows a large number of outliers on both sides. These are considered in the model.



- RoamMins shows a large number of outliers on both sides. These are considered in the model.



Summary

1. Variables are numeric in nature.
2. There are 7 continuous variables and 4 categoric variables.
3. Target variable is binary in nature
4. Histogram of all continuous variables except data usage are bell shaped.
5. Bivariate analysis graphs show some levels of correlation between the variables.
6. Multicollinearity with VIF shows some variables need to be removed.
7. There are no missing values in the significant variables.
8. Outliers have been identified in some of the variables. But they need not be removed.

Logistic Regression

One of the best models for predictions when the variable to be determined is binary in nature is the logistic regression model.
Like all regression analyses, the logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Firstly Summary of the data is taken to determine the % of Churn rate in this model.

```
> Churn=as.factor(Churn)
> summary(Churn)
   0    1
2850  483
>
```

```
483/(2850+483)#Churn rate=14.5%
```

It is found that Churn rate of Churn is 14%

A logistic regression model is run with all the significant variables that are available.

```
> glm(Churn~AccountWeeks+CustServCalls+DayMins+DayCalls+MonthlyCharge+OverageFee+RoamMins+Mon
thlyCharge,data=CellphoneData,family="binomial")

Call:  glm(formula = Churn ~ AccountWeeks + CustServCalls + DayMins + 
    DayCalls + MonthlyCharge + OverageFee + RoamMins + MonthlyCharge, 
    family = "binomial", data = CellphoneData)

Coefficients:
  (Intercept)   AccountWeeks  CustServCalls        DayMins       DayCalls  
    -7.638707       0.001246       0.448785       0.017104       0.003160  
MonthlyCharge     OverageFee       RoamMins  
    -0.027163       0.177339       0.102513  

Degrees of Freedom: 3332 Total (i.e. Null);  3325 Residual
Null Deviance:	    2758 
Residual Deviance: 2377 	AIC: 2393
>
```

```
Call:
glm(formula = Churn ~ AccountWeeks + CustServCalls + DayMins +
    DayCalls + MonthlyCharge + OverageFee + RoamMins + MonthlyCharge,
    family = "binomial", data = CellphoneData)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7540  -0.5628  -0.4041  -0.2585   2.9934

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   -7.638707   0.512938 -14.892  < 2e-16 ***
AccountWeeks   0.001246   0.001331   0.936    0.349
CustServCalls  0.448785   0.036932  12.152  < 2e-16 ***
DayMins        0.017104   0.001334  12.824  < 2e-16 ***
DayCalls       0.003160   0.002642   1.196    0.232
MonthlyCharge -0.027163   0.004706  -5.772 7.83e-09 ***
OverageFee     0.177339   0.023482   7.552 4.28e-14 ***
RoamMins       0.102513   0.019607   5.228 1.71e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2758.3  on 3332  degrees of freedom
Residual deviance: 2377.0  on 3325  degrees of freedom
AIC: 2393

Number of Fisher Scoring iterations: 5

>
```

```
> vif(glm(Churn~AccountWeeks+CustServCalls+DayMins+DayCalls+MonthlyCharge+OverageFee+RoamMins+MonthlyChar
ge,data=CellphoneData,family="binomial"))
 AccountWeeks CustServCalls       DayMins      DayCalls MonthlyCharge    OverageFee      RoamMins
     1.001109      1.033396      1.720974      1.003296      1.837854      1.194900      1.017910
```

Two of the variables as per below are identified to be non-significant in the model.

- Accountweeks
- Day Calls

The mentioned variables are removed and the logistic regression model is run again.

```
> Logistic.Churn=glm(Churn~CustServCalls+DayMins+MonthlyCharge+OverageFee+RoamMins+MonthlyCharge,data=CellphoneData,family="binomial")
> summary(Logistic.Churn)

Call:
glm(formula = Churn ~ CustServCalls + DayMins + MonthlyCharge + 
    OverageFee + RoamMins + MonthlyCharge, family = "binomial", 
    data = CellphoneData)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.7153  -0.5640  -0.4046  -0.2578   2.9358  

Coefficients:
               Estimate Std. Error z value Pr(>|z|)    
(Intercept)   -7.186736   0.411240  -17.48  < 2e-16 ***
CustServCalls  0.447519   0.036857   12.14  < 2e-16 ***
DayMins        0.017113   0.001333   12.84  < 2e-16 ***
MonthlyCharge -0.027129   0.004702   -5.77 7.94e-09 ***
OverageFee     0.176280   0.023473    7.51 5.92e-14 ***
RoamMins       0.102768   0.019575    5.25 1.52e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2758.3  on 3332  degrees of freedom
Residual deviance: 2379.3  on 3327  degrees of freedom
AIC: 2391.3

Number of Fisher Scoring iterations: 5

> 
```

The variables are showing very high P values. Which means all the variables in the model are significant.

K Nearest Neighbour (KNN)

k nearest neighbors is a simple algorithm that stores all available cases and classifies new cases by a majority vote of its k neighbors. This algorithms segregates unlabeled data points into well defined groups.

The kNN algorithm is applied to the training data set and the results are verified on the test data set.

Total number of rows in the data is 3333.
This is getting split so that 2/3 of the data is train data and 1/3 becomes test data.
Rows in the train data – 2222
Rows in the test data – 1111

```
> knn(CPData.train[,c(4,8)],CPData.test[,c(4,8)],CPData.train$Churn,k=5)
   [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  [47] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  [93] 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
 [139] 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0
 [185] 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [231] 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0
 [277] 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [323] 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0
 [369] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [415] 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [461] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
 [507] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0
 [553] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [599] 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
 [645] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 [691] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 [737] 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1 0 1
 [783] 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0
 [829] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [875] 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0
 [921] 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0
 [967] 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 [ reached getOption("max.print") -- omitted 111 entries ]
Levels: 0 1
> summary(KNN.Churn)
   0    1 
1066   45 
> 
```

KNN Model does not have a formula like regression models.
It just classifies
Multiple values of K was used and experimented with the model.
When the value of K was 2, it seemed to predict the values much closer to the initial % of the Churn rate.

Naïve Bayes

Naïve Bayes is a classification model which was initially established with a condition that all the variables are supposed to be categorical and not continuous.
But in practical situations such a condition is not possible as we would be always getting a mix of continuous and categorical variables.
We can try building a Naïve Bayes model data using the library (e1071) in R.

```
> naiveBayes(Churn~ContractRenewal+MonthlyCharge,data=CPData.train)

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
        0         1 
0.8537354 0.1462646 

Conditional probabilities:
   ContractRenewal
Y        [,1]      [,2]
  0 0.9309436 0.2536169
  1 0.7046154 0.4569190

   MonthlyCharge
Y       [,1]     [,2]
  0 56.27580 16.68927
  1 59.76338 16.27097
```

The model is able to predict Churn quite closer to the actual Churn date of the data (14%)
It also gives the conditional probabilities of 2 of the variables Contract renewal and MonthlyCharges.

```
> NB.Churn=naiveBayes(Churn~ContractRenewal+MonthlyCharge,data=CPData.train)
> predNB=predict(NB.Churn,type="raw",newdata=CPData.test)
> predNB
                 0          1
   [1,] 0.89947341 0.10052659
   [2,] 0.90424893 0.09575107
   [3,] 0.03890750 0.96109250
   [4,] 0.93050470 0.06949530
   [5,] 0.92958957 0.07041043
   [6,] 0.93132912 0.06867088
   [7,] 0.94514643 0.05485357
   [8,] 0.94505739 0.05494261
   [9,] 0.93600358 0.06399642
  [10,] 0.93956294 0.06043706
  [11,] 0.95203589 0.04796411
  [12,] 0.92594496 0.07405504
```

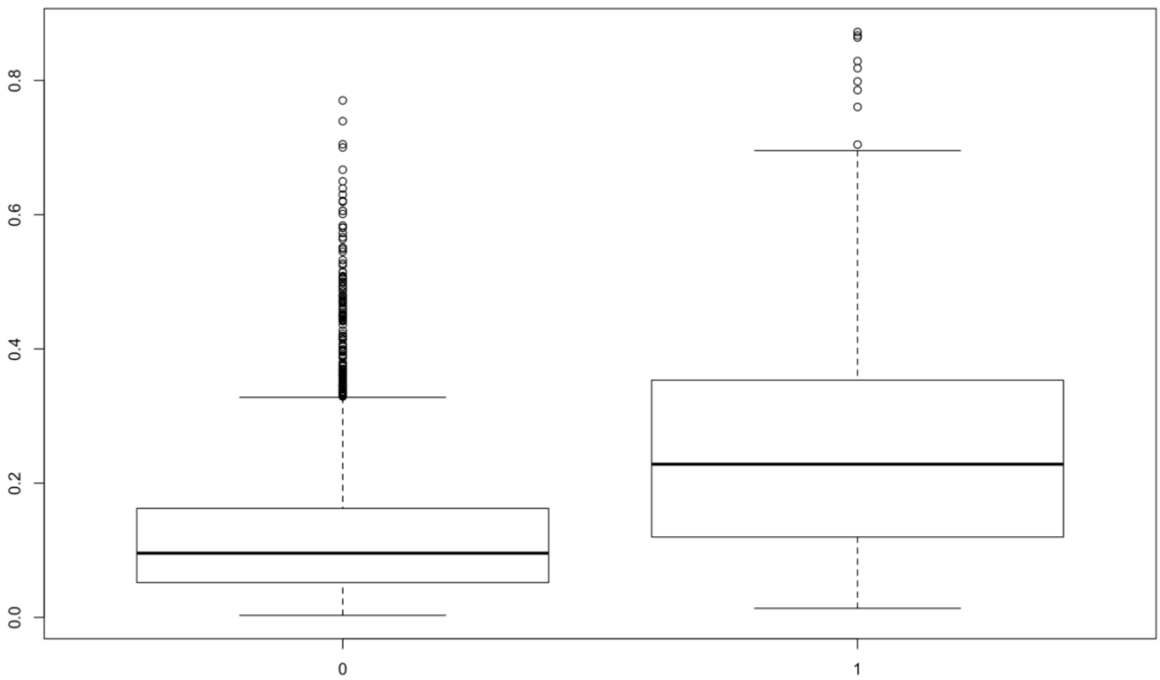
Model performance Metrics

1. Logistic Regression

A confusion matrix and ROC curve are used to measure the performance of the logistic regression model achieved.

```
Logistic.Churn$fitted.values
plot(Churn,Logistic.Churn$fitted.values)
Churn.predicted=ifelse(Logistic.Churn$fitted.values>0.9,"Churn","No")
table(Churn,Churn.predicted)
roc(Churn,Logistic.Churn$fitted.values)
plot.roc(Churn,Logistic.Churn$fitted.values)
```

Plotting Fitted Values



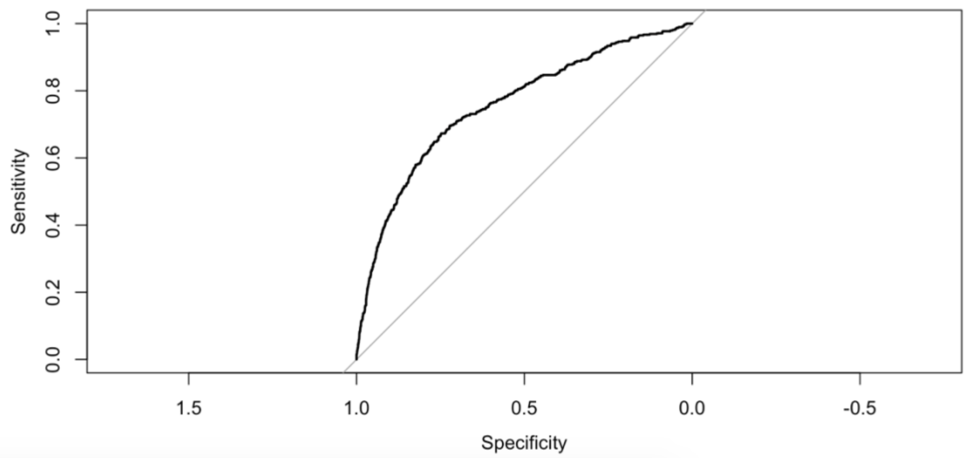

```
      Churn.predicted
Churn    No
    0  2850
    1   483
```

Churn rate from the model shows 483/2850% = 16.9%

The ROC curve has an area under the curve of 0.7566

```
Call:
roc.default(response = Churn, predictor = Logistic.Churn$fitted.values)

Data: Logistic.Churn$fitted.values in 2850 controls (Churn 0) < 483 cases (Churn 1).
Area under the curve: 0.7566
```



2. KNN Model

The following confusion matrix is obtained.

```
  KNN.Churn
      0   1
  0 853 100
  1 115  43
```

Measure	Value	Derivations
Sensitivity	0.8800	TPR = TP / (TP + FN)
Specificity	0.3007	SPC = TN / (FP + TN)
Precision	0.8940	PPV = TP / (TP + FP)
Negative Predictive Value	0.2722	NPV = TN / (TN + FN)
False Positive Rate	0.6993	FPR = FP / (FP + TN)
False Discovery Rate	0.1060	FDR = FP / (FP + TP)
False Negative Rate	0.1200	FNR = FN / (FN + TP)
Accuracy	0.8047	ACC = (TP + TN) / (P + N)
F1 Score	0.8869	F1 = 2TP / (2TP + FP + FN)

Actionable Insights

From the various analysis and modelling done on this project, logistic regression is identified to be the best model.
The model can predict the Churn probability of a customer with about 75% accuracy.
It is quite clear from the models tried that 2 of the variables are very critical in deciding the Churn rate.

1. Contract Renewal and
2. Monthly Charges.

This shows that if the company focuses to improve these 2 variables, they can significantly improve the Churn rate of their customers.

By adapting a predictive modelling based on logistic regression, the company would be able to predict the tendency of a customer to move away from itself to the competition. Hence, immediately they can do a timely intervention with offers to keep the customer