# MINI PROJECT
# - CellPhone Churn

***Model Report***


***By – Nikhil Rawal***

GREAT LAKES
INSTITUTE OF MANAGEMENT

## Project Overview

Logistic regression is a predictive modelling algorithm that is used when the Y variable is binary categorical. That is, it can take only two values like 1 or 0. The goal is to determine a mathematical equation that can be used to predict the probability of event 1. Once the equation is established, it can be used to predict the Y when only the X’s are known.

The given dataset of Cellphone from Cellphone Company, whose objective is to find or predict the Customers who are likely to churn, on the basis of variables like Data Usage, Contact Renewal, Day Calls, Monthly Bills, etc.

## Project Approach

- DATA EXPLORATION
- DATA VISUALISATION
- DATA PARTITION
- LOGISTIC REGRESSION MODEL
- LIKELIHOOD & MCFADEN
- CONFUSION MATRIX

- Data Exploration

#set working directory

- getwd()

```
"F:/r cellphone"
```

- Read Input File

```
celldata= read.csv("cellphone.csv")
```

- Head(celldata)

| Churn | AccountWeeks | ContractRenewal | DataPlan | DataUsage | CustServCalls |
|---|---|---|---|---|---|
| 0 | 128 | 1 | 1 | 2.7 | 1 |
| 0 | 107 | 1 | 1 | 3.7 | 1 |
| 0 | 137 | 1 | 0 | 0.0 | 0 |
| 0 | 84 | 0 | 0 | 0.0 | 2 |
| 0 | 75 | 0 | 0 | 0.0 | 3 |
| 0 | 118 | 0 | 0 | 0.0 | 0 |

| DayMins | DayCalls | MonthlyCharge | OverageFee | RoamMins |
|---|---|---|---|---|
| 265.1 | 110 | 89 | 9.87 | 10.0 |
| 161.6 | 123 | 82 | 9.78 | 13.7 |
| 243.4 | 114 | 52 | 6.06 | 12.2 |
| 299.4 | 71 | 57 | 3.10 | 6.6 |
| 166.7 | 113 | 41 | 7.42 | 10.1 |
| 223.4 | 98 | 57 | 11.03 | 6.3 |

- Str(celldata)

```
'data.frame':    3333 obs. of  11 variables:
 $ Churn          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ AccountWeeks   : int  128 107 137 84 75 118 121 147 117 141 ...
 $ ContractRenewal: int  1 1 1 0 0 0 1 0 1 0 ...
 $ DataPlan       : int  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  : int  1 1 0 2 3 0 3 0 1 0 ...
 $ DayMins        : num  265 162 243 299 167 ...
 $ DayCalls       : int  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 ...
```

- Summary of data

| | Churn | AccountWeeks | ContractRenewal | DataPlan |
|---|---|---|---|---|
| | Min. :0.0000 | Min. : 1.0 | Min. :0.0000 | Min. :0.0000 |
| | 1st Qu.:0.0000 | 1st Qu.: 74.0 | 1st Qu.:1.0000 | 1st Qu.:0.0000 |
| | Median :0.0000 | Median :101.0 | Median :1.0000 | Median :0.0000 |
| | Mean :0.1449 | Mean :101.1 | Mean :0.9031 | Mean :0.2766 |
| | 3rd Qu.:0.0000 | 3rd Qu.:127.0 | 3rd Qu.:1.0000 | 3rd Qu.:1.0000 |
| | Max. :1.0000 | Max. :243.0 | Max. :1.0000 | Max. :1.0000 |

| DataUsage | CustServCalls | DayMins | DayCalls |
|---|---|---|---|
| Min. :0.0000 | Min. :0.000 | Min. : 0.0 | Min. : 0.0 |
| 1st Qu.:0.0000 | 1st Qu.:1.000 | 1st Qu.:143.7 | 1st Qu.: 87.0 |
| Median :0.0000 | Median :1.000 | Median :179.4 | Median :101.0 |
| Mean :0.8165 | Mean :1.563 | Mean :179.8 | Mean :100.4 |
| 3rd Qu.:1.7800 | 3rd Qu.:2.000 | 3rd Qu.:216.4 | 3rd Qu.:114.0 |
| Max. :5.4000 | Max. :9.000 | Max. :350.8 | Max. :165.0 |

| MonthlyCharge | OverageFee | RoamMins |
|---|---|---|
| Min. : 14.00 | Min. : 0.00 | Min. : 0.00 |
| 1st Qu.: 45.00 | 1st Qu.: 8.33 | 1st Qu.: 8.50 |
| Median : 53.50 | Median :10.07 | Median :10.30 |
| Mean : 56.31 | Mean :10.05 | Mean :10.24 |
| 3rd Qu.: 66.20 | 3rd Qu.:11.77 | 3rd Qu.:12.10 |
| Max. :111.30 | Max. :18.19 | Max. :20.00 |

- Names of the Variables of the data

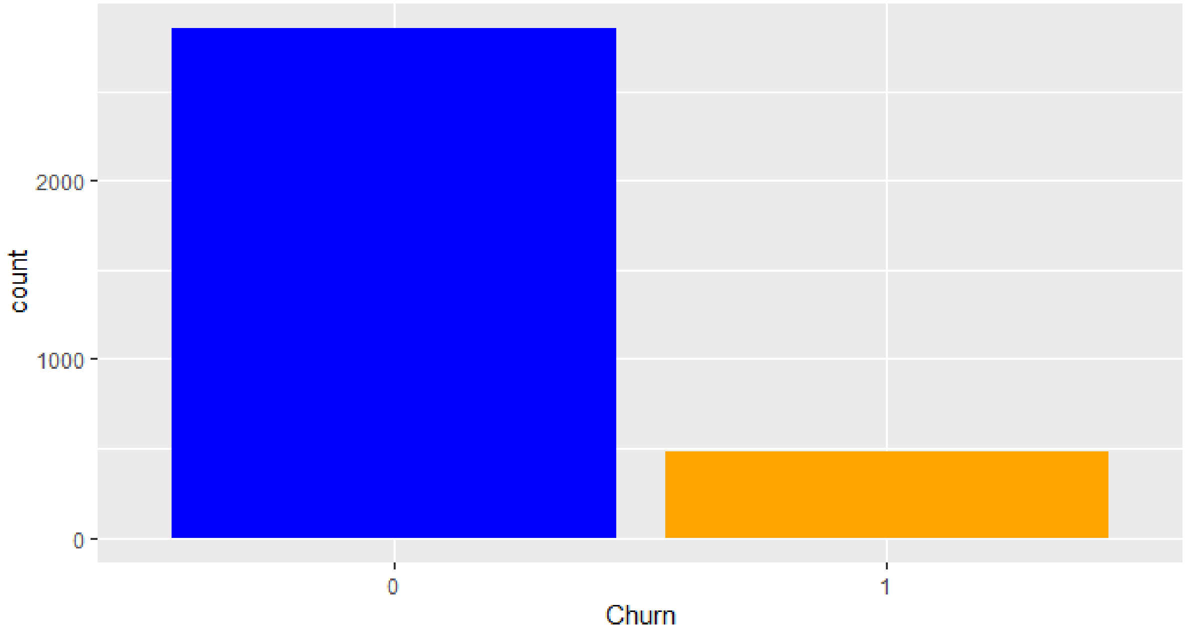
```
"Churn"         "AccountWeeks"    "ContractRenewal" "DataPlan"
 "DataUsage"      "CustServCalls"   "DayMins"        "DayCalls"
 "MonthlyCharge"   "OverageFee"      "RoamMins"
```

- Dimension of the data

```
3333  11
```

- Data Visualisation

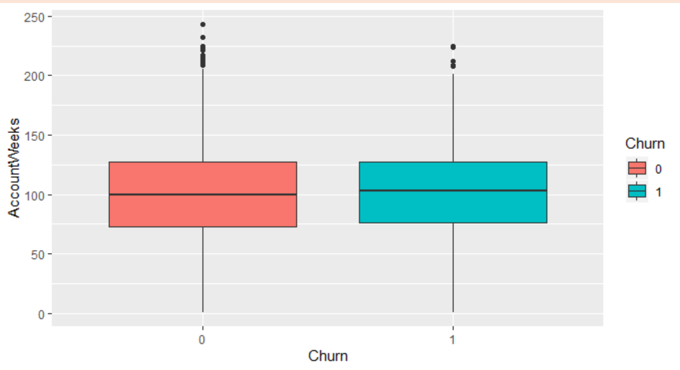
  - Churn Count

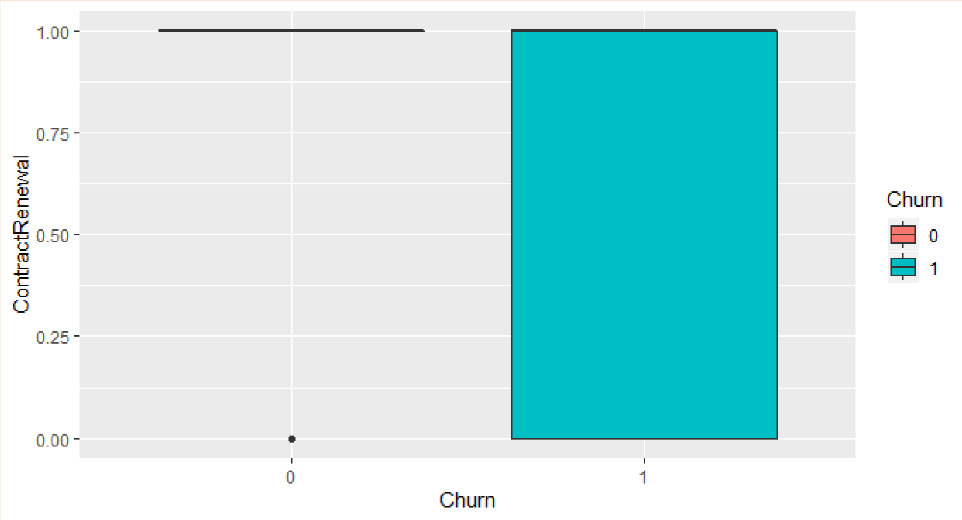


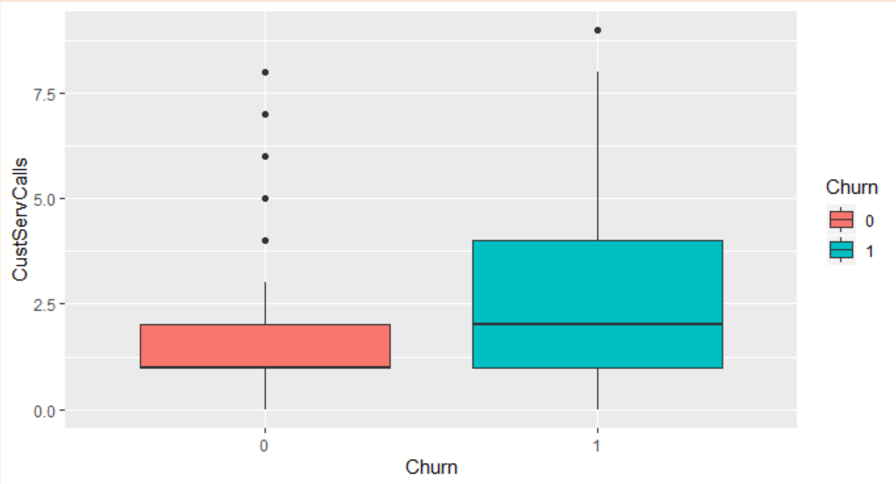

Churn : 1 – if Customer cancelled Service ; 0 – if not
1: 483 ; 0:2850

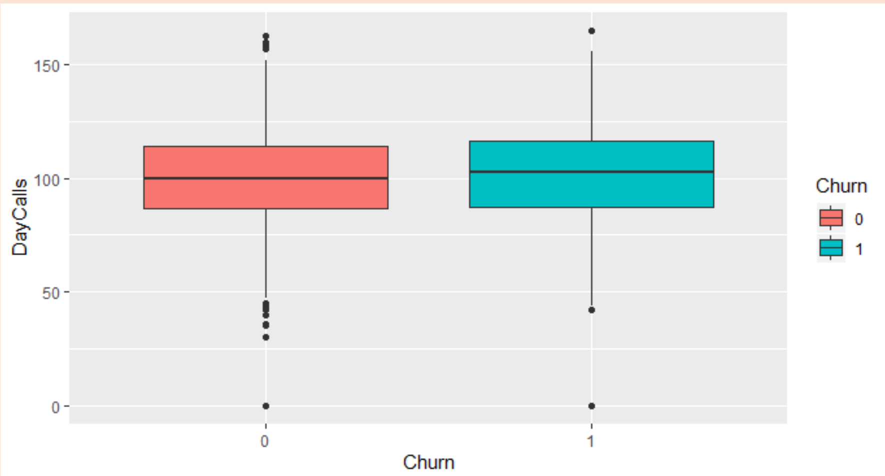
Hence, We can see the data that Customer Cancelling the Service are low as 483, but nos had to be taken seriously, and must find out the reason and base of the customer who used to churn their cellphone service.

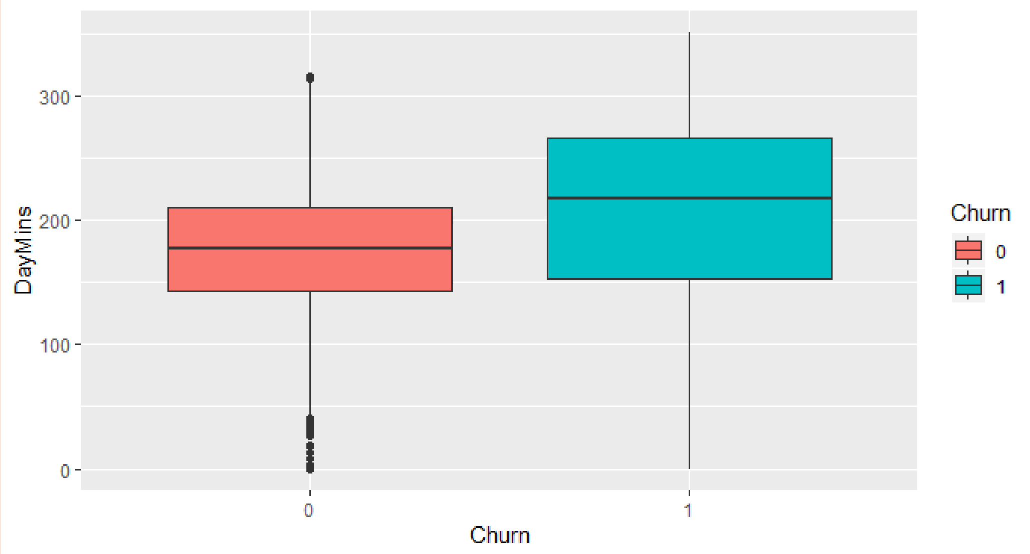
  - BoxPlot of All Variable

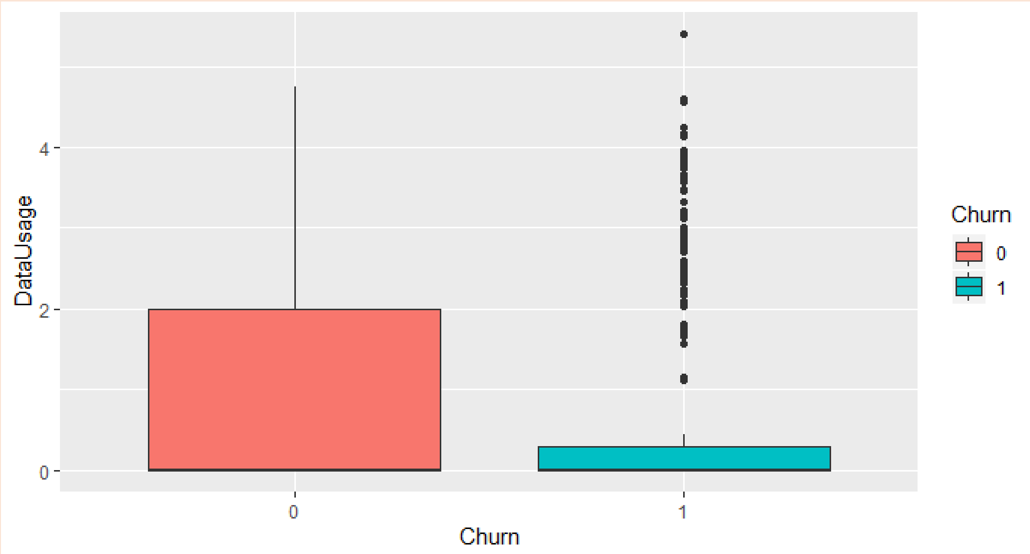


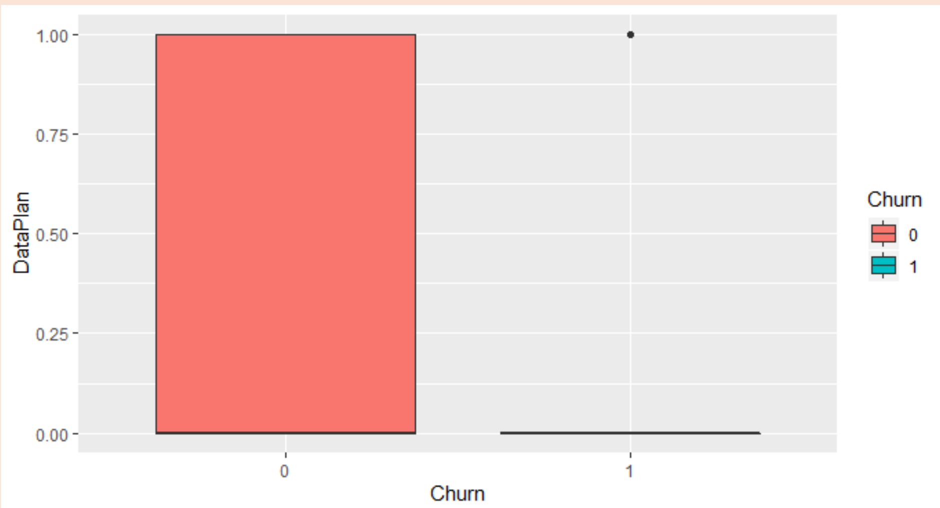

1.00
0.75
0.50
0.25
0.00
ContractRenewal
0
1
Churn
Churn
0
1

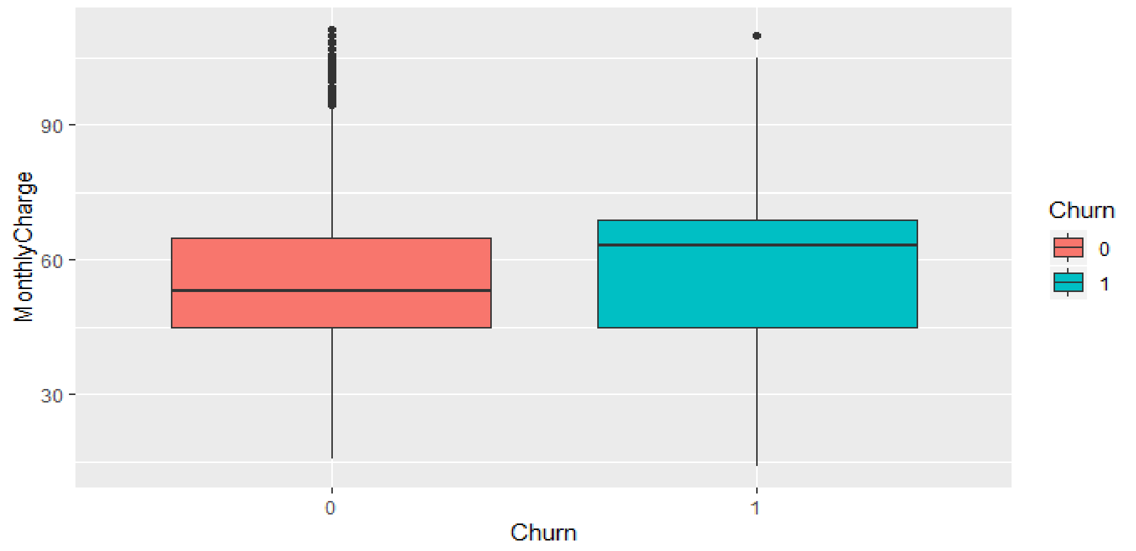

7.5
5.0
2.5
0.0
CustServCalls
0
1
Churn
Churn
0
1


150
100
50
0
DayCalls
0
1
Churn
Churn
0
1

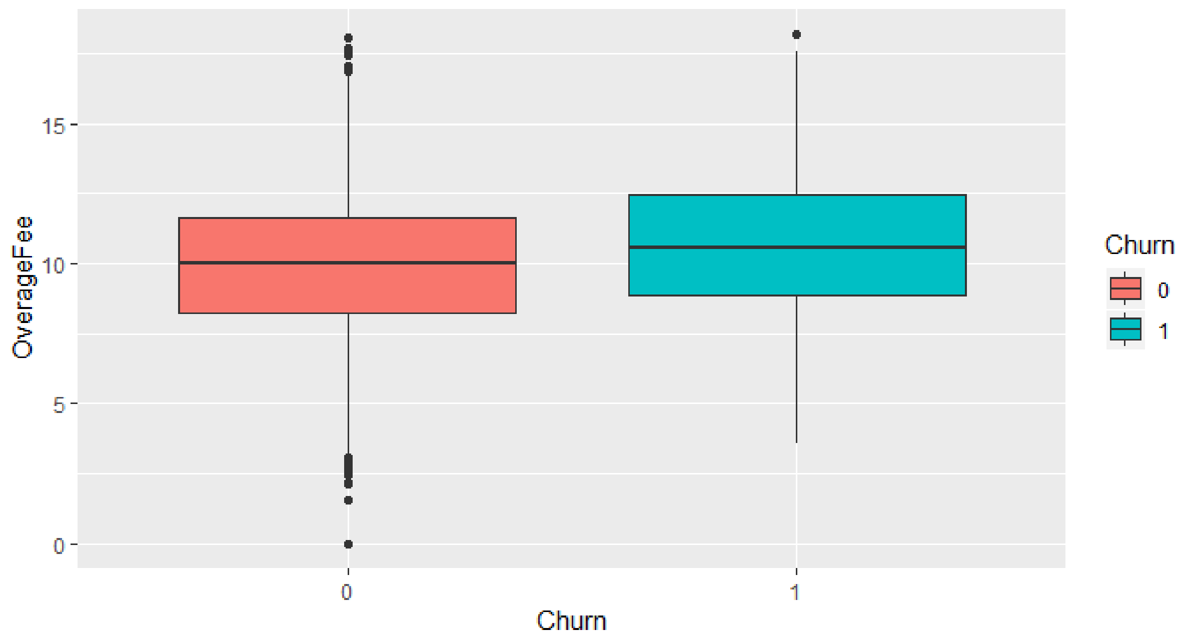

DayMins
300
200
100
0
0
1
Churn
Churn
0
1


DataUsage
4
2
0
0
1
Churn
Churn
0
1

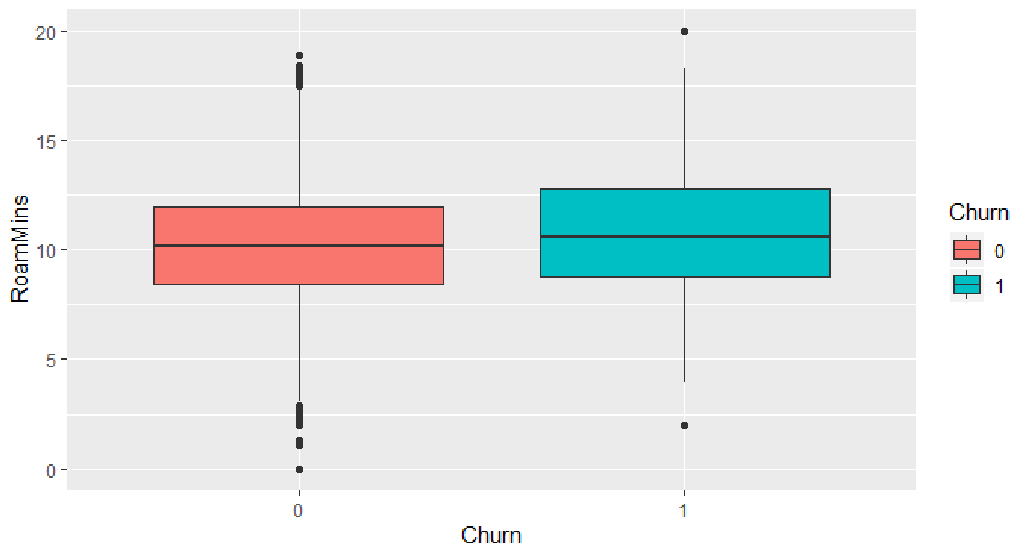

DataPlan
1.00
0.75
0.50
0.25
0.00
0
1
Churn
Churn
0
1



MonthlyCharge
90
60
30
0
1
Churn
Churn
0
1

OverageFee
15
10
5
0
0
1
Churn
Churn
0
1

RoamMins
20
15
10
5
0
0
1
Churn
Churn
0
1

- Data Partition

- Splitting of data in 70:30 ratio for train and test data

```
set.seed(111)
spindex<-createDataPartition(celldata$Churn, p=0.7, list = FALSE)
traindata<-celldata[spindex,]
testdata<-celldata[-spindex,]
```

Hence, After Splitting of data

#dim(testdata)

```
  999  11
```

#dim(traindata)

```
 2334   11
```

- Checking of Partition of data

#table(traindata$Churn)

```
   0    1
2000  334
```

table(testdata$Churn)

```
  0   1
850 149
```

Hence, the distribution of Partition of data is Correct.

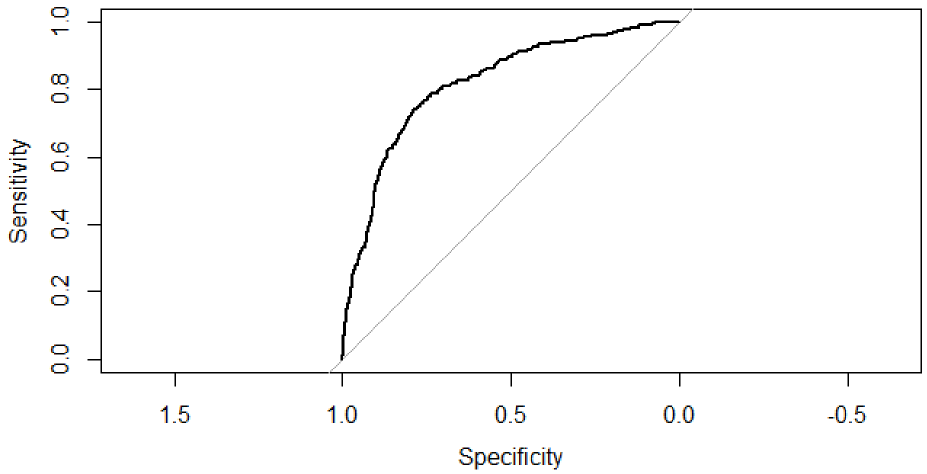
- Logistic Regression Model

- Logistic Regreesion Model of Train data

```
model1 <- glm(Churn ~., family = "binomial", data = traindata)
Summary
```

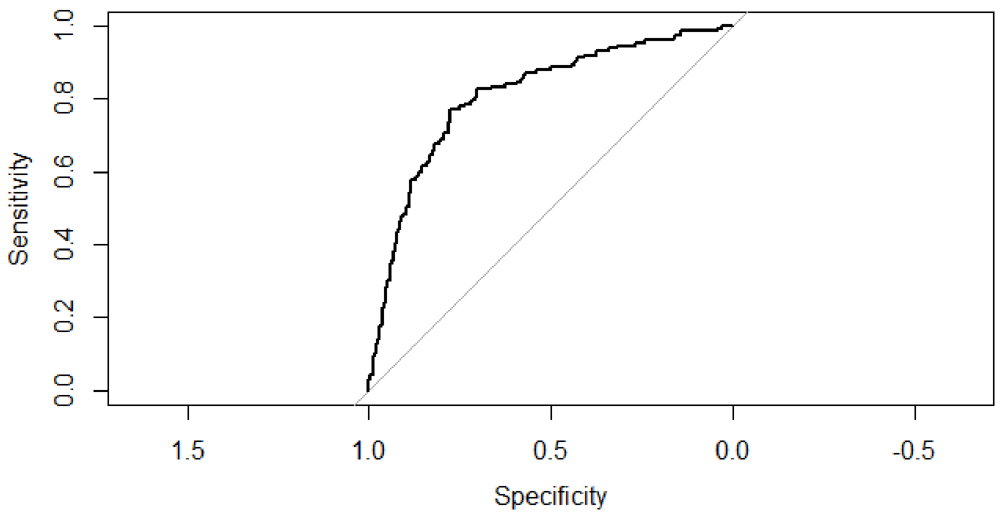
```
Call:
glm(formula = Churn ~ ., family = "binomial", data = traindata)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9831  -0.5084  -0.3456  -0.2016   2.8958

Coefficients:
                 Estimate Std.    Error        z value    Pr(>|z|)
(Intercept)       -5.7367307     0.6681880    -8.586     <2e-16 ***
AccountWeeks     -0.0001389     0.0016927    -0.082      0.9346
ContractRenewal -2.0196231      0.1690085   -11.950     <2e-16 ***
DataPlan          -1.5656002     0.6458028   -2.424      0.0153 *
DataUsage         0.4224692      2.3432681    0.180      0.8569
CustServCalls     0.4946874      0.0484476   10.211     <2e-16 ***
DayMins           0.0166227      0.0395244    0.421      0.6741
DayCalls          0.0010363      0.0033290    0.311      0.7556
MonthlyCharge    -0.0191252      0.2322938   -0.082      0.9344
OverageFee        0.1870373      0.3959652    0.472      0.6367
RoamMins          0.0668978      0.0268691    2.490      0.0128 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1916.5  on 2333  degrees of freedom
Residual deviance: 1511.0  on 2323  degrees of freedom
AIC: 1533

Number of Fisher Scoring iterations: 6
```

- Plot ROC of traindata




Area under the curve: 0.817

- Plot ROC of testdata




Area under the curve: 0.8102

## *Significance of the Logistic regression model to test applicability*

- Likelihood Test

- lrtest(model1)

```
Likelihood ratio test

Model 1: Churn ~ AccountWeeks + ContractRenewal + DataPlan + DataUsage +
    CustServCalls + DayMins + DayCalls + MonthlyCharge + OverageFee +
    RoamMins
Model 2: Churn ~ 1
  #Df  LogLik  Df  Chisq Pr(>Chisq)
1  11 -755.52
2   1 -958.23 -10 405.42  < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

- Mcfaden or Pseudo $r^2$ test

- pR2(model1)

```
          llh       llhNull            G2      McFadden          r2ML
-755.5233015  -958.2347784   405.4229539     0.2115468     0.1594536
         r2CU
   0.2847096
```

- Confusion Matrix

- Predicting the Outcome to Compute Confusion Matrix

```
#predict(model1, type = "response",data=testdata)
```

- Confusion Matrix

```
# confusionMatrix(predict_response,testdata$Churn)
```

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 841 119
         1  14  25

               Accuracy : 0.8669
                 95% CI : (0.8442, 0.8873)
    No Information Rate : 0.8559
    P-Value [Acc > NIR] : 0.1724

                  Kappa : 0.2256

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.9836
            Specificity : 0.1736
         Pos Pred Value : 0.8760
         Neg Pred Value : 0.6410
             Prevalence : 0.8559
         Detection Rate : 0.8418
   Detection Prevalence : 0.9610
      Balanced Accuracy : 0.5786

       'Positive' Class : 0
```

- Odds Ratio

```
# exp(cbind(OR=coef(model1),confint(model1)))
```

| | OR | 2.5 % | 97.5 % |
|---|---|---|---|
| (Intercept) | 0.00269272 | 0.0007299897 | 9.592316e-03 |
| AccountWeeks | 1.00044399 | 0.9971889961 | 1.003710e+00 |
| ContractRenewal | 0.14390967 | 0.1034223482 | 1.999374e-01 |
| DataPlan | 0.27971676 | 0.0751911457 | 9.946476e-01 |
| DataUsage | 4.01151284 | 0.0435696997 | 3.729683e+02 |
| CustServCalls | 1.67801185 | 1.5343267716 | 1.838185e+00 |
| DayMins | 1.03511304 | 0.9590035346 | 1.117510e+00 |
| DayCalls | 1.00474098 | 0.9983079944 | 1.011234e+00 |
| MonthlyCharge | 0.87788145 | 0.5599474409 | 1.375038e+00 |
| OverageFee | 1.40360171 | 0.6524148495 | 3.025790e+00 |
| RoamMins | 1.10020878 | 1.0442535208 | 1.160086e+00 |

# Source Code

```
------------------------------------------------
--------------------------------------------------------------------------------------
------------------------------------------------
library(readr)
library(ggplot2)
library(dplyr)
library(tidyr)
library(corrplot)
library(caret)
library(rms)
library(MASS)
library(e1071)
library(ROCR)
library(gplots)
library(pROC)
library(rpart)
library(randomForest)
library(ggpubr)
library(car)
library(rpart.plot)
```

```
#Data Exploration
setwd("F:/r cellphone")
getwd()
celldata= read.csv("cellphone.csv")
celldata
head(celldata)
tail(celldata)
str(celldata)
summary(celldata)
celldata$Churn<-factor(celldata$Churn)
names(celldata)
attach(celldata)
dim(celldata)

#data visualisation


ggplot(celldata, aes(x = Churn))+
  geom_histogram(stat = "count", fill = c("blue", "orange"))
table(celldata$Churn)
```

```r
# Data Visualization boxplot

ggplot(data = mydata, aes(x=Churn, y=AccountWeeks,
fill=Churn)) + geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=ContractRenewal,
fill=Churn)) + geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=DataPlan, fill=Churn))
+ geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=DataUsage,
fill=Churn)) + geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=CustServCalls,
fill=Churn)) + geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=DayMins, fill=Churn))
+ geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=DayCalls, fill=Churn))
+ geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=MonthlyCharge,
fill=Churn)) + geom_boxplot()
```

```
ggplot(data = mydata, aes(x=Churn, y=OverageFee, fill=Churn)) + geom_boxplot()

ggplot(data = mydata, aes(x=Churn, y=RoamMins, fill=Churn)) + geom_boxplot()

#Split data

set.seed(111)
spindex<-createDataPartition(celldata$Churn, p=0.7, list = FALSE)
traindata<-celldata[spindex,]
testdata<-celldata[-spindex,]

dim(testdata)
dim(traindata)

table(traindata$Churn)
table(testdata$Churn)
```

```
#Logitic Model train
model1 <- glm(Churn ~., family = "binomial", data =
traindata)
summary(model1)

#plot roc train data
P_train = predict(model1, newdata = traindata, type =
"response")
rocplottrain <- plot(roc(traindata$Churn, P_train))
auc(rocplot)

#plot roc test data
P_test = predict(model1, newdata = testdata, type =
"response")
rocplottest <- plot(roc(testdata$Churn, P_test))
auc(rocplottest)

#likelihood test
library(lmtest)
lrtest(model1)

#McFaden  or pseudo r^2 and interpretation
library(pscl)
pR2(model1)
```

```
#predict
predict(model1, type = "response",data=testdata)

predictprob<-predict(model1,testdata[,2:11], type="response")
predict_response<-ifelse(predictprob>0.5,1,0)
predict_response<-as.factor(predict_response)

##Confusion Matrix
confusionMatrix(predict_response,testdata$Churn)

### odds ratio
exp(cbind(OR=coef(model1),confint(model1)))
```

---------------------------------------------------------------------------------------------

Thank You

---------------------------------------------------------------------------------------------