MINI PROJECT – LOGISTIC REGRESSION

Submitted By

Pratheeba R

Contents

1. Project Objective

The given dataset is a cell phone data set which contains variables like Churn, account weeks, contract renewal, data plan, data usage, customer service calls, daytime minutes, daytime calls, monthly charge, overage fee and average number of roaming minutes. The objective is to find the probability of a customer cancelling the service provider depending on the other varibles like data plan used by the customer, average monthly bill, monthly data usage etc by building a logistic regression model and interpret the result. Make sure you partition the data set by allocating 70% -for training data and 30% -for validating the results.

2. Exploratory Data Analysis

3.1 Environment Setup and Data import

3.1.1 Install necessary packages

The package needed to be installed is

- library(SDMTools)
- library(pROC)
- library(Hmisc)
- library(ggplot2)
- library(DataExplorer)
- library(PerformanceAnalytics)
- library(car)
- library(nFactors)
- library(psych)
- library(dplyr)
- library(corrplot)
- library(lmtest)
- library(pscl)

3.1.2 Setup working directory

Set the working directory to the location where you have the dataset and code files using setwd() function. When a working directory is set, you don't have to mention the whole path of the file while importing a dataset which minimizes the time and probability of unwanted errors.

3.1.3 Import and read the dataset

The dataset under study is an csv file and so we can use read.csv() function to read the dataset and frame it as a data frame for our study.

3.2 Variable Identification

To understand the structure of the dataset, the following functions are being used,

FUNCTION	PURPOSE
dim(dataframe)	Number of columns and rows
str(dataframe)	Examine each column separately (the data type of each column and their sample values)
introduce(dataframe)	Displays number of rows, columns, discrete columns, continuous columns, missing columns, total missing values, complete rows, total observations and memory usage
summary(dataframe)	Data summary
mean(datacolumn)	Sample mean of the column
sd(datacolumn)	Standard deviation of column
table(datacolumn)	Frequency of datapoints for a particular column in a dataframe
anyNA(dataframe/attribute)	Returns a Boolean value indicating the presence.
plot(x)	Produces a scatterplot of the given attribute

3.2.1 Inferences

The given dataset contains 3333 rows of 11 columns. All the values except are numeric values. which represent the rating of the customer satisfaction.

```
dim(mydata)
[1] 3333   11
```

```
str(mydata)
'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:

```
summary(mydata)
     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
```

Missing Values:

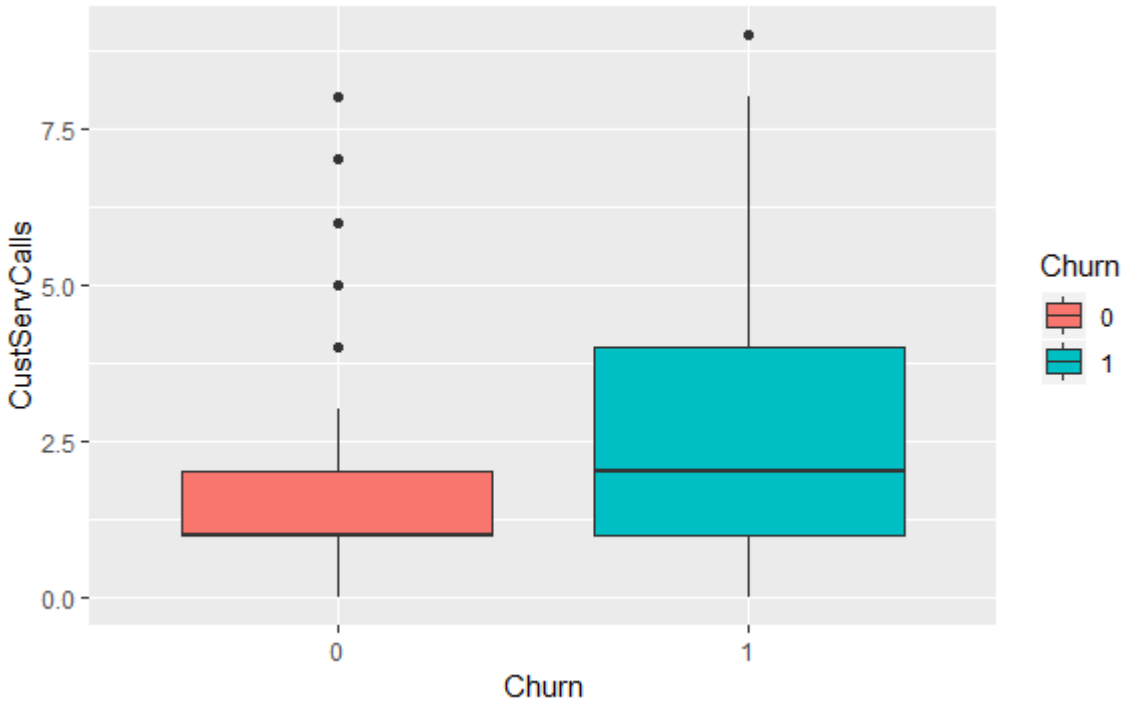
It can be seen that there are no missing values.

Baseline Proportion:

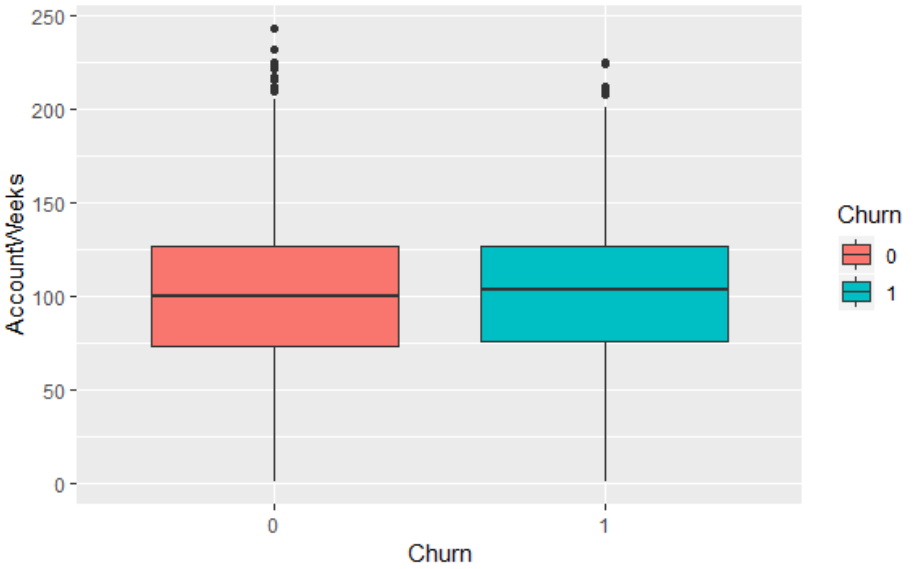
```
table(Churn)
Churn
   0    1
2850  483

prop.table(table(Churn))
Churn
        0         1
0.8550855 0.1449145
```

From the baseline proportion, it is visible that the predictor variable is 85% and 15% of Churn yes/no respectively. It indicates that the dataset is an imbalanced dataset. Hence we cannot be accurate about the predictions made.

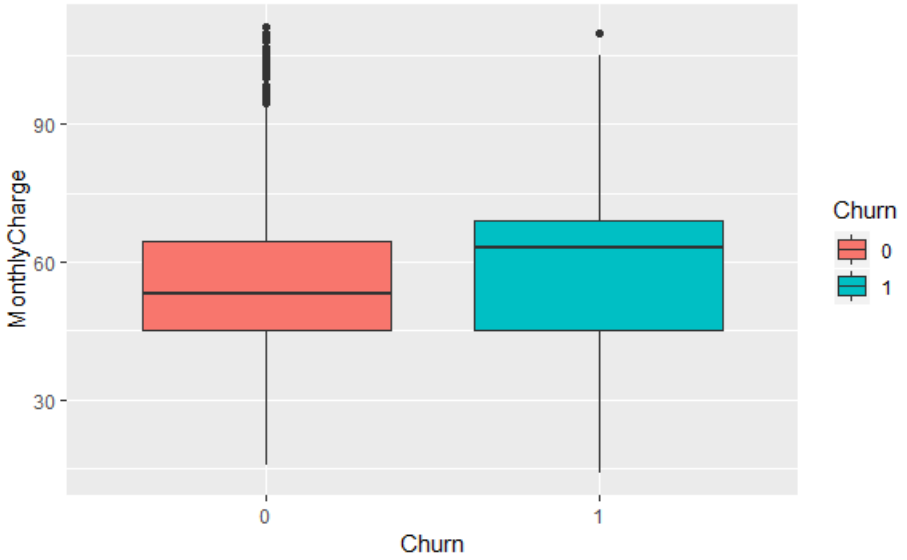
3. Data Visualization



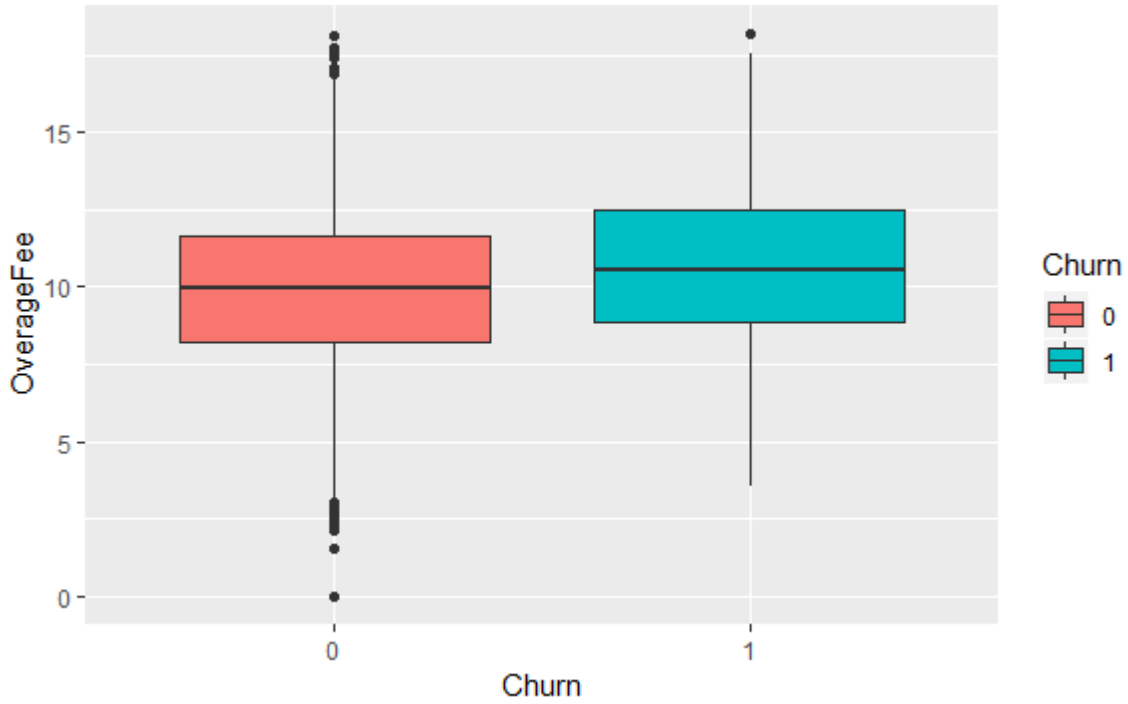
By the boxplot of Churn and customer service calls, we can roughly say that when the number of customer service calls are low, customer do not churn.



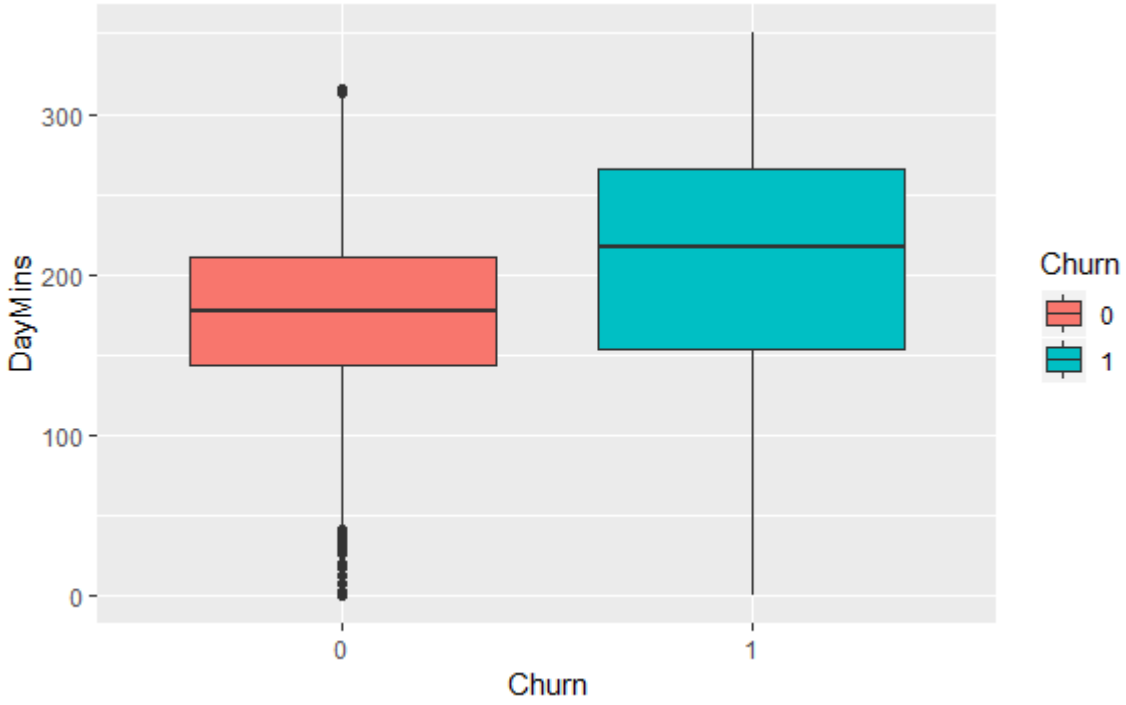
By the above boxplot, we can infer that account weeks does not have an impact on Churn ratio.



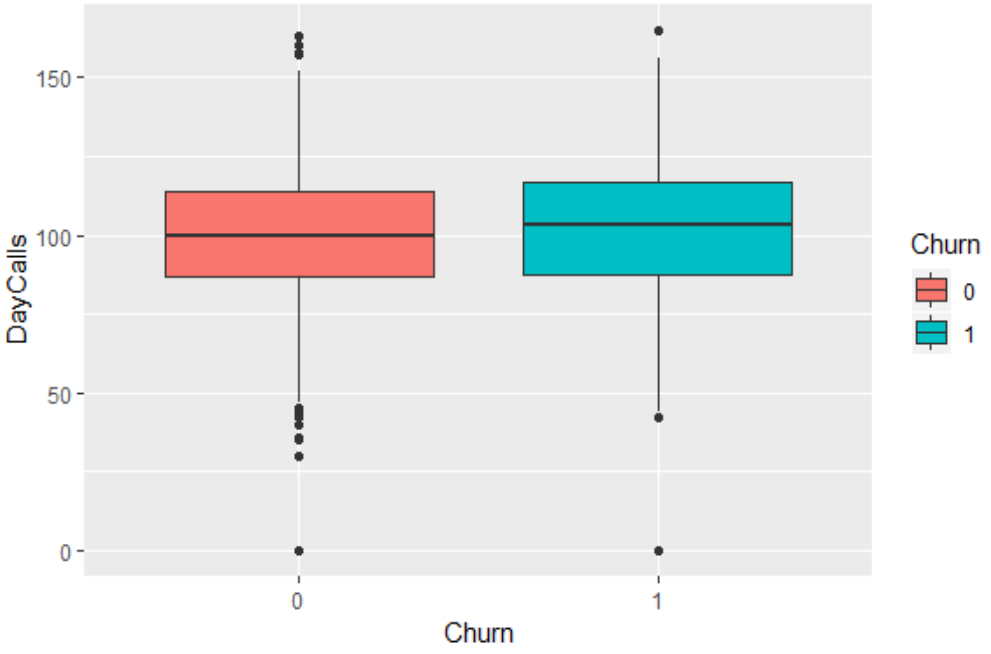
By the above boxplot, we can infer that the ones who has monthly charge more is likely to Churn. As the customers who do not Churn has low monthly charge.



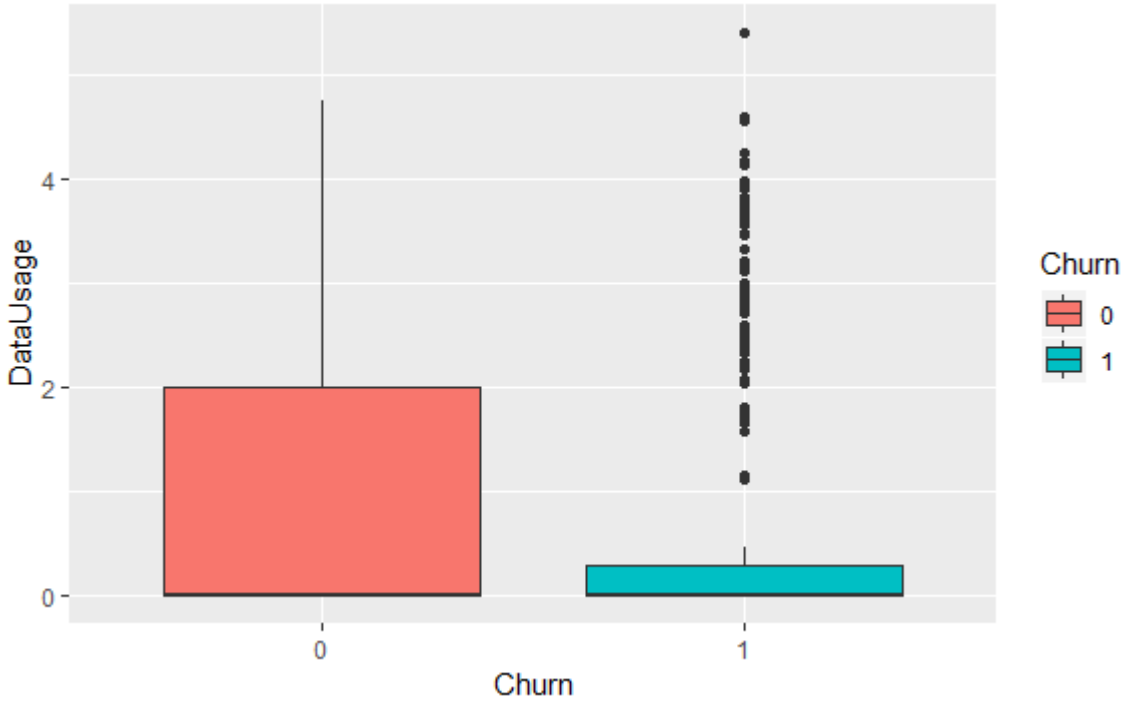
By the above plot, we can infer that account weeks does not have an impact on Churn ratio.



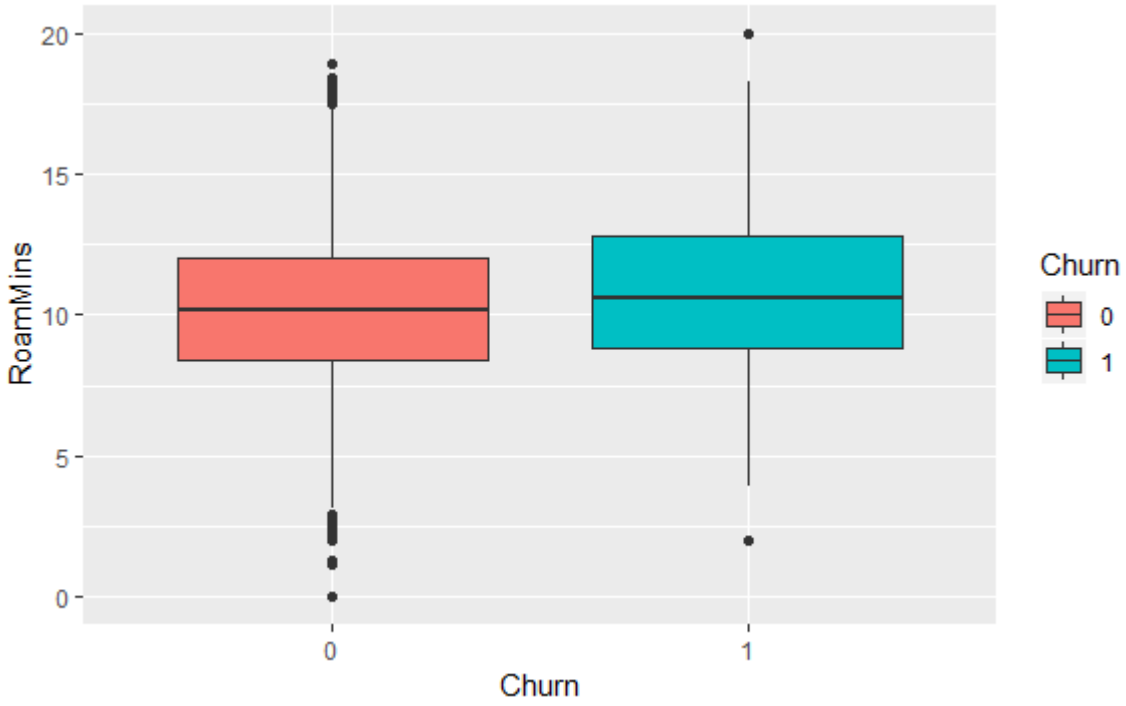
By this plot, we can infer that the one who has a high average day minutes per month are more likely to churn.



From the above plot, we can see that average day calls does not impact the Churn ratio.



For the one who uses more data, are unlikely to Churn. We can infer that the data services are strong.



From the above plot, we can infer that there is no much impact of roaming minutes in Churn ratio.

All the above inferences from the above plots are made disregarding the outliers.

4. Feature Engineering

It can be seen from the structure of the data frame that all the data points are numeric. For easy interpretation of the results from the model evaluation, we convert Contract renewal and data Plan as factors. Since both can be categorized as yes or no based on the values that they contain (0 or 1).

5. Split data

Split the dataset into 70%-30% of training and test data with a seed of 222.

set.seed(222)

pd<-sample(2,nrow(mydata),replace=TRUE, prob=c(0.7,0.3))

train<-mydata[pd==1,]

val<-mydata[pd==2,]

```
dim(train)
[1] 2316   11

dim(val)
[1] 1017   11
```

It can be seen that the training data has 2316 rows of 11 columns and test data has 1017 rows of 11 columns.

The proportion of observations:

```
prop.table(table(train$Churn))

        0         1
0.8536269 0.1463731

prop.table(table(val$Churn))

        0         1
0.8584071 0.1415929
```

It can be seen that the proportions of who churn out is 14.67% in training data and 14.15% in test data and the proportion who doesn't opt to churn out is 85% in both training data and test data. Hence it can be seen that the training and test data are equally distributed.

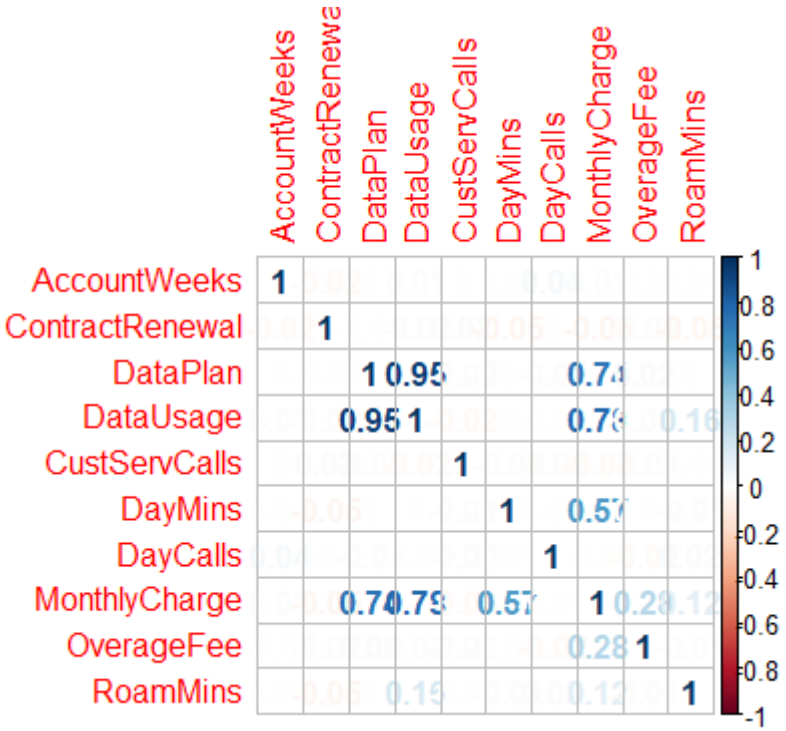
6. Relationship between dependent and independent variables:

The correlation between the variables is shown here,

```
                AccountWeeks ContractRenewal DataPlan DataUsage
AccountWeeks            1.00           -0.02     0.00      0.01
ContractRenewal        -0.02            1.00    -0.01     -0.02
DataPlan                0.00           -0.01     1.00      0.95
DataUsage               0.01           -0.02     0.95      1.00
```

```
CustServCalls           0.00           0.02     -0.02      -0.02
DayMins                 0.01          -0.05      0.00       0.00
DayCalls                0.04           0.00     -0.01      -0.01
MonthlyCharge           0.01          -0.05      0.74       0.78
OverageFee             -0.01          -0.02      0.02       0.02
RoamMins                0.01          -0.05      0.00       0.16
                CustServCalls DayMins DayCalls MonthlyCharge OverageFee
AccountWeeks             0.00    0.01     0.04          0.01      -0.01
ContractRenewal          0.02   -0.05     0.00         -0.05      -0.02
DataPlan                -0.02    0.00    -0.01          0.74       0.02
DataUsage               -0.02    0.00    -0.01          0.78       0.02
CustServCalls            1.00   -0.01    -0.02         -0.03      -0.01
DayMins                 -0.01    1.00     0.01          0.57       0.01
DayCalls                -0.02    0.01     1.00         -0.01      -0.02
MonthlyCharge           -0.03    0.57    -0.01          1.00       0.28
OverageFee              -0.01    0.01    -0.02          0.28       1.00
RoamMins                -0.01   -0.01     0.02          0.12      -0.01
                RoamMins
AccountWeeks        0.01
ContractRenewal    -0.05
DataPlan            0.00
DataUsage           0.16
CustServCalls      -0.01
DayMins            -0.01
DayCalls            0.02
MonthlyCharge       0.12
OverageFee         -0.01
RoamMins            1.00
```

Correlation Plot:



It can be seen from the plot that

- Data Usage and Data Plan are correlated.
- Monthly Charge and Daymins are correlated.
- Monthly Charge and Data Usage are correlated.
- Monthly Charge and Data Plan are correlated

7. Logistic Regression Model:

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

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0719  -0.4930  -0.3368  -0.1948   3.0722

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      -6.125741   0.661666  -9.258  < 2e-16 ***
AccountWeeks     -0.000879   0.001689  -0.521  0.60269
ContractRenewal1 -2.031544   0.175609 -11.569  < 2e-16 ***
DataPlan1        -1.802663   0.648675  -2.779  0.00545 **
DataUsage         1.550327   2.371856   0.654  0.51335
CustServCalls     0.572342   0.047928  11.942  < 2e-16 ***
DayMins           0.034614   0.040005   0.865  0.38691
DayCalls          0.007427   0.003337   2.226  0.02604 *
MonthlyCharge    -0.122446   0.235040  -0.521  0.60240
OverageFee        0.342634   0.400620   0.855  0.39241
RoamMins          0.052604   0.027157   1.937  0.05274 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1928.6  on 2315  degrees of freedom
Residual deviance: 1487.9  on 2305  degrees of freedom
AIC: 1509.9

Number of Fisher Scoring iterations: 6
```

Inference:

Initially constructing a model with all the variables.It can be seen that Contract Renewal (Yes), Data Plan(Yes), Customer Service Calls, DayCalls and Roaming Mins are significant.

Effect of multicollinearity:

For a regression model, the independent varibles or the predictors should be indepe ndent of each other. In the other way round, they should not be correlated to each other.
We use vif to predict multicollinearity,

AccountWeeks	ContractRenewal	DataPlan	DataUsage
1.006358	1.072247	15.155780	1795.712746
CustServCalls	DayMins	DayCalls	MonthlyCharge
1.113026	1007.073793	1.010103	3045.722845
OverageFee	RoamMins		
208.204039	1.204431		

It can be seen that the highlighted ones are the highly correlated ones.

In order to avoid multicollinearity, we go with Stepwise Regression,

Initially removing Monthly charge as Data Usage and Monthly Charge are highly correlated (evident from vif and correlation plot)

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

Deviance Residuals:
```

```
    Min       1Q   Median       3Q      Max
-2.0726  -0.4919  -0.3364  -0.1947   3.0526

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -6.1890847  0.6507472  -9.511  < 2e-16 ***
AccountWeeks     -0.0008741  0.0016887  -0.518  0.60474
ContractRenewal1 -2.0312775  0.1755790 -11.569  < 2e-16 ***
DataPlan1        -1.7864676  0.6479873  -2.757  0.00583 **
DataUsage         0.3200565  0.2198948   1.455  0.14553
CustServCalls     0.5717000  0.0478819  11.940  < 2e-16 ***
DayMins           0.0137865  0.0012889  10.696  < 2e-16 ***
DayCalls          0.0074440  0.0033347   2.232  0.02560 *
OverageFee        0.1344602  0.0280039   4.801 1.57e-06 ***
RoamMins          0.0530690  0.0271412   1.955  0.05055 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1928.6  on 2315  degrees of freedom
Residual deviance: 1488.2  on 2306  degrees of freedom
AIC: 1508.2

Number of Fisher Scoring iterations: 6
```

The ones which are highlighted with blue are significant with pvalues less than alpha.

We now consider vif of the model to detect multicollinearity,

```
 AccountWeeks ContractRenewal         DataPlan        DataUsage
     1.006380        1.072099        15.114983        15.417496
CustServCalls         DayMins         DayCalls      OverageFee
     1.111930        1.045097         1.010142         1.017563
    RoamMins
     1.202927
```

It can be seen that there is still multicollinearity in the model. So we can remove any one say Data Plan or Data usage from the model, which are highly correlated.

Removing Data Usage from the model,

```
Call:
glm(formula = Churn ~ . - DataUsage - MonthlyCharge, family = "binomial",
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0878  -0.4972  -0.3325  -0.1970   3.0658

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      -6.3011317  0.6470078  -9.739  < 2e-16 ***
AccountWeeks     -0.0008385  0.0016892  -0.496  0.61964
ContractRenewal1 -2.0354403  0.1755573 -11.594  < 2e-16 ***
DataPlan1        -0.8850554  0.1681274  -5.264 1.41e-07 ***
CustServCalls     0.5659433  0.0476472  11.878  < 2e-16 ***
DayMins           0.0137563  0.0012867  10.691  < 2e-16 ***
DayCalls          0.0073911  0.0033330   2.218  0.02659 *
OverageFee        0.1334319  0.0279271   4.778 1.77e-06 ***
RoamMins          0.0690309  0.0249530   2.766  0.00567 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 1928.6  on 2315  degrees of freedom
Residual deviance: 1490.3  on 2307  degrees of freedom
AIC: 1508.3

Number of Fisher Scoring iterations: 5
```

```
vif(logit2)
   AccountWeeks ContractRenewal        DataPlan   CustServCalls
       1.005970        1.071980        1.023502        1.101192
        DayMins        DayCalls      OverageFee        RoamMins
       1.044051        1.010051        1.017034        1.009876
```

The vif of the model shows that there is no multicollinearity.

As the Account weeks is insignificant, we can remove that and construct a new model.

```
Call:
glm(formula = Churn ~ . - DataUsage - MonthlyCharge - AccountWeeks,
    family = "binomial", data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0867  -0.4969  -0.3339  -0.1981   3.0653

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      -6.379277   0.628016 -10.158  < 2e-16 ***
ContractRenewal1 -2.032171   0.175403 -11.586  < 2e-16 ***
DataPlan1        -0.888324   0.168029  -5.287 1.25e-07 ***
CustServCalls     0.566076   0.047647  11.881  < 2e-16 ***
DayMins           0.013749   0.001286  10.690  < 2e-16 ***
DayCalls          0.007314   0.003329   2.197  0.02803 *
OverageFee        0.133616   0.027918   4.786 1.70e-06 ***
RoamMins          0.068950   0.024948   2.764  0.00571 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1928.6  on 2315  degrees of freedom
Residual deviance: 1490.906  on 2308  degrees of freedom
AIC: 1506.6

Number of Fisher Scoring iterations: 5
```

All the variables are significant and the predictors in the model are highly independent of eac h other.

Also we can see the AIC (Akaike information criterion), estimator of the relative quality of st atistical models for a given set of data. It keeps on decreasing indicating the quality of the mo del before and after removing the correlated variables

Overall Significance:

Testing the overall significance of the model,

```
Likelihood ratio test

Model 1: Churn ~ (AccountWeeks + ContractRenewal + DataPlan + DataUsage +
    CustServCalls + DayMins + DayCalls + MonthlyCharge + OverageFee +
    RoamMins) - DataUsage - MonthlyCharge - AccountWeeks
Model 2: Churn ~ 1
```

```
  #Df  LogLik Df  Chisq Pr(>Chisq)
1   8 -745.29
2   1 -964.30 -7 438.02  < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

We get the p-value (0.0000000000000002) which is very much lower than the alpha value (0.05). The null hypothesis that no variable is a predictor of Churn can be rejected. We have alteast one independent variable which is a predictor of Churn.

McFadden Rsquare

```
pR2(logit3)
         llh      llhNull           G2     McFadden         r2ML
-745.2918263 -964.3033196  438.0229866    0.2271189    0.1723204
        r2CU
   0.3049161
```

Based on the value of the McFadden R^2, we can conclude that almost 23% of the u ncertanity of the intercept only model has been explained by the full model. This value also i ndicates a goodness of fit.*

Individual coefficient significance

Odds ratio:

With the summary of the model, from the coefficients, we can calculate the odds ratio as th e exponents of the coefficients

```
(Intercept)         ContractRenewal1        DataPlan1         CustServCalls
     0.001696349       0.131050656       0.411344705       1.761342223
         DayMins           DayCalls        OverageFee           RoamMins
     1.013844194       1.007341188       1.142954076       1.071382465
```

Probability

With the odds ratio, we can calculate the probability as (oddsratio/(1+oddsratio)).

```
(Intercept)         ContractRenewal1        DataPlan1         CustServCalls
     0.001693476       0.115866301       0.291455874       0.637857274
         DayMins           DayCalls        OverageFee           RoamMins
     0.503437256       0.501828585       0.533354442       0.517230634
```

For finding the significance of individual variables, we create a new data frame of coefficient obtained from the model, p-values from the model for each predictor, odds ratio and probabili ty. The data frame is filtered with the p-values<=0.05 (taking only the significant ones). Then it is sorted based on the odds ratio largest to smallest. The resultant data frame is

```
                    coeff         pval        odds        prob
CustServCalls   0.566076145 1.493490e-32 1.761342223 0.637857274
OverageFee      0.133616206 1.701652e-06 1.142954076 0.533354442
RoamMins        0.068949838 5.714221e-03 1.071382465 0.517230634
DayMins         0.013749239 1.136969e-26 1.013844194 0.503437256
```

```
DayCalls          0.007314373 2.803084e-02 1.007341188 0.501828585
DataPlan1        -0.888323717 1.245164e-07 0.411344705 0.291455874
ContractRenewal1 -2.032171343 4.866684e-31 0.131050656 0.115866301
(Intercept)      -6.379277189 3.058330e-24 0.001696349 0.001693476
```

From this, we can take the cutoff value for the probabilities as 0.5. We have five main predict ors like CustServiceCalls, OverageFee, Roaming minutes, Day minutes and DayCalls.

Inferences:

It can be concluded that,

- For increase in **customer service calls** by one unit, odds of getting a customer churne d out increases by 1.76 times. In the other words, the probability of getting a customer churned out is 64% more when the customer service calls increases by one unit.
- For increase in **overage fee** by one unit, odds of getting a customer churned out increa ses by 1.14 times. In the other words, the probability of getting a customer churned ou t is 53% more when the overage fee becomes greater by one unit.
- For increase in **Roaming minutes** by one unit, odds of getting a customer churned out increases by 1.07 times. In the other words, the probability of getting a customer churn ed out is 52% more when the roaming minutes increases by one unit.
- For increase in **average day minutes per month** by one unit, odds of getting a custom er churned out increases by 1.01 times. In the other words, the probability of getting a customer churned out is 50% more when the average day minutes increases by one un it.
- For increase in **average day time calls per day** by one unit, odds of getting a custome r churned out increases by 1 times. In the other words, the probability of getting a cust omer churned out is 50% more when the average day time calls increases by one unit.

8. Model Performance Measures

Confusion Matrix:

The values are predicted at a cutoff value of probability being greater than 0.5. The confusion matrix for training dataset as follows

	0	**1**
0	**1921**	**56**
1	**264**	**75**

The confusion matrix for test dataset as follows

	0	**1**
0	**847**	**26**
1	**120**	**24**

From the above confusion matrices, it can be found that the false negative rate is hig her. In order to minimize false classification of true negatives, we try decreasing the cutoff va lue initially,

By reducing the cutoff to 0.3, we get the classification matrix for training dataset as follows,

	0	1
0	**1961**	**16**
1	**312**	**27**

And for the test dataset as follows,

	0	1
0	**866**	**7**
1	**136**	**8**

By increasing the cutoff to 0.7, we get the classification matrix for training dataset as follows,

	0	1
0	**1808**	**169**
1	**185**	**154**

And for the test dataset as follows,

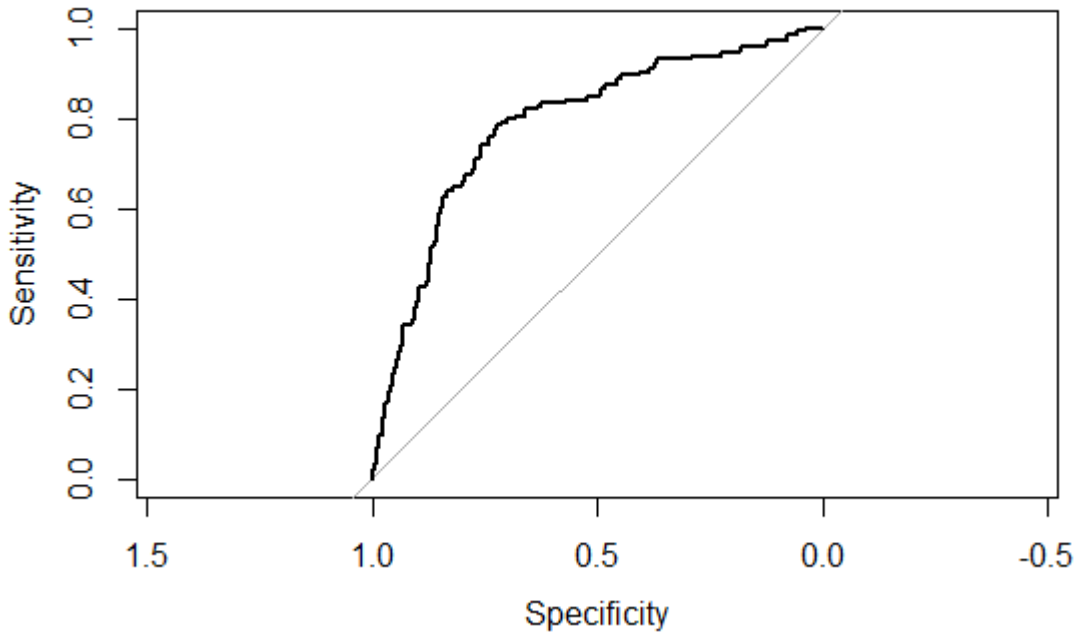
	0	1
0	**789**	**84**
1	**89**	**55**

We can see from the above, that the false negatives reduced to half.

ROC Plot

```
Call:
roc.default(response = val$Churn, predictor = pred.logit)

Data: pred.logit in 873 controls (val$Churn 0) < 144 cases (val$Churn 1
).
Area under the curve: 0.7867
```

We get area under the curve as **78.67**, which states the model is good.



From the above graph, it can be seen that the model has a performance measure of 79%.

The test dataset performance metrics:

Accuracy	0.8298918
Loss	0.09621993
Opportunity Loss	0.6180556
Total Loss	0.1223117

As the dataset is imbalanced, we do not consider the sensitivity and specificity.

9. Appendix A – Source Code

```
---
title: "Logistic Regression"
output:
  html_document:
    df_print: paged
---
```

```{r}
library(SDMTools)
```

```
library(pROC)
library(Hmisc)
library(ggplot2)
library(DataExplorer)
library(PerformanceAnalytics)
library(car)
library(nFactors)
library(psych)
library(dplyr)
```

```{r}
setwd("C:/Users/HP/Desktop/Mini Project5/")
getwd()
```

```{r}
mydata = read.csv("Cellphone.csv", header = T)
attach(mydata)
```

```{r}
#Exploratory Data Analysis
View(mydata)
names(mydata)
dim(mydata)
summary(mydata)
str(mydata)
colSums(is.na(mydata))
plot_missing(mydata)
```

```{r}
corr_mydata1 = cor(mydata)
```

```
library(corrplot)
round(corrplot(corr_mydata1, method = "number"), 2)
```

```{r}
#Feature Engineering

mydata$Churn = factor(mydata$Churn)
mydata$ContractRenewal = factor(mydata$ContractRenewal)
mydata$DataPlan = factor(mydata$DataPlan)
View(mydata)
str(mydata)
```

```{r}

#Proportion
table(Churn)
prop.table(table(Churn))
#which shows it is an imbalanced dataset
```
```{r}
#Data visualization

#chart.Correlation(mydata, histogram = TRUE, pch = 19)

```
```{r}
#Split data
set.seed(222)
pd<-sample(2,nrow(mydata),replace=TRUE, prob=c(0.7,0.3))
train<-mydata[pd==1,]
val<-mydata[pd==2,]
head(train)
```

```
dim(train)
dim(val)

#Proportion
prop.table(table(train$Churn))
prop.table(table(val$Churn))
```

```{r}
logit = glm(Churn ~ ., data = train, family = "binomial")
```

```{r}
summary(logit)
```

```{r}
#test multicollinearity
vif(logit)#this shows the presence of multicollinearity.
```

```{r}
#It seems that Data usage and monthly charge are highly correlated.
#hence we remove monthly charge,
#attach(mydata)
logit1= glm(Churn~.- MonthlyCharge, data = train, family = "binomial")
summary(logit1)
vif(logit1)
```

```{r}
#removing data usage from model
logit2= glm(Churn~. - DataUsage - MonthlyCharge, data = train, family = "binomial")
summary(logit2)
vif(logit2)
```

```

```{r}
#removing Account weeks as it is shown to be insignificant
attach(mydata)
logit3= glm(Churn~.-DataUsage - MonthlyCharge - AccountWeeks , data = train, family = "binomial")
summary(logit3)
vif(logit3)
```

```{r}
#Overall Significance
#install.packages("lmtest")
library(lmtest)
lrtest(logit3)
#The overall significance is more. p value is very low.Hence the null hypothesis
#is rejected. Alteast one variable is a predictor of Churn.
```
```{r}
#McFaden Rsquare computation
#install.packages("pscl")
library(pscl)
pR2(logit3)
#Only 20% of the variations in Churn is explained by the model. So the model is not good.
```

```{r}
#Individual coefficient significance
summary(logit3)
odds = exp(logit3$coefficients)
prob = odds/(1+odds)
```
```

```
newdf = data.frame(coeff = logit3$coefficients, pval = summary(logit3)$coefficients[ ,4], odds, prob)
newdf
#filter by probabilities(take only significant ones)
newdf[newdf$pval <= 0.05, ]
#sort by odds
sorted <- newdf[order(-odds),]
sorted
```

```{r}
predict(logit3, type = "response", data = train)

```

```{r}
#prediction at a cutoff value
pred_train = floor(predict(logit3, type = "response", data = train)+0.5)
confusionmat = table(Actual = train$Churn, Predicted = pred_train)
confusionmat

#predict test data
pred_test = floor(predict(logit3, type = "response", newdata = val[-1])+0.5)
confusionmat = table(Actual = val$Churn, Predicted = pred_test)
confusionmat

```

```{r}
#prediction at a cutoff value(0.3)
pred_train = floor(predict(logit3, type = "response", data = train)+0.3)
confusionmat = table(Actual = train$Churn, Predicted = pred_train)
confusionmat
```

```
#predict test data
pred_test = floor(predict(logit3, type = "response", newdata = val[-1])+0.3)
confusionmat = table(Actual = val$Churn, Predicted = pred_test)
confusionmat
```

```{r}
#prediction at a cutoff value(0.7)
pred_train = floor(predict(logit3, type = "response", data = train)+0.7)
confusionmat = table(Actual = train$Churn, Predicted = pred_train)
confusionmat

#predict test data
pred_test = floor(predict(logit3, type = "response", newdata = val[-1])+0.7)
confusionmat = table(Actual = val$Churn, Predicted = pred_test)
confusionmat
```
```{r}
pred.logit <- predict.glm(logit3, newdata=val[-1], type="response")
roc.logit<-roc(val$Churn,pred.logit )
roc.logit
plot(roc.logit)
```
```{r}
confusionmat
accuracy.logit<-sum(diag(confusionmat))/sum(confusionmat)
accuracy.logit
loss.logit<-confusionmat[1,2]/(confusionmat[1,2]+confusionmat[1,1])
loss.logit
opp.loss.logit<-confusionmat[2,1]/(confusionmat[2,1]+confusionmat[2,2])
opp.loss.logit
tot.loss.logit<-0.95*loss.logit+0.05*opp.loss.logit
tot.loss.logit
```

10. References

1. http://cowles.yale.edu/sites/default/files/files/pub/d04/d0474.pdf