Customer Churn

Abhay Kulkarni

7/20/2019

Contents

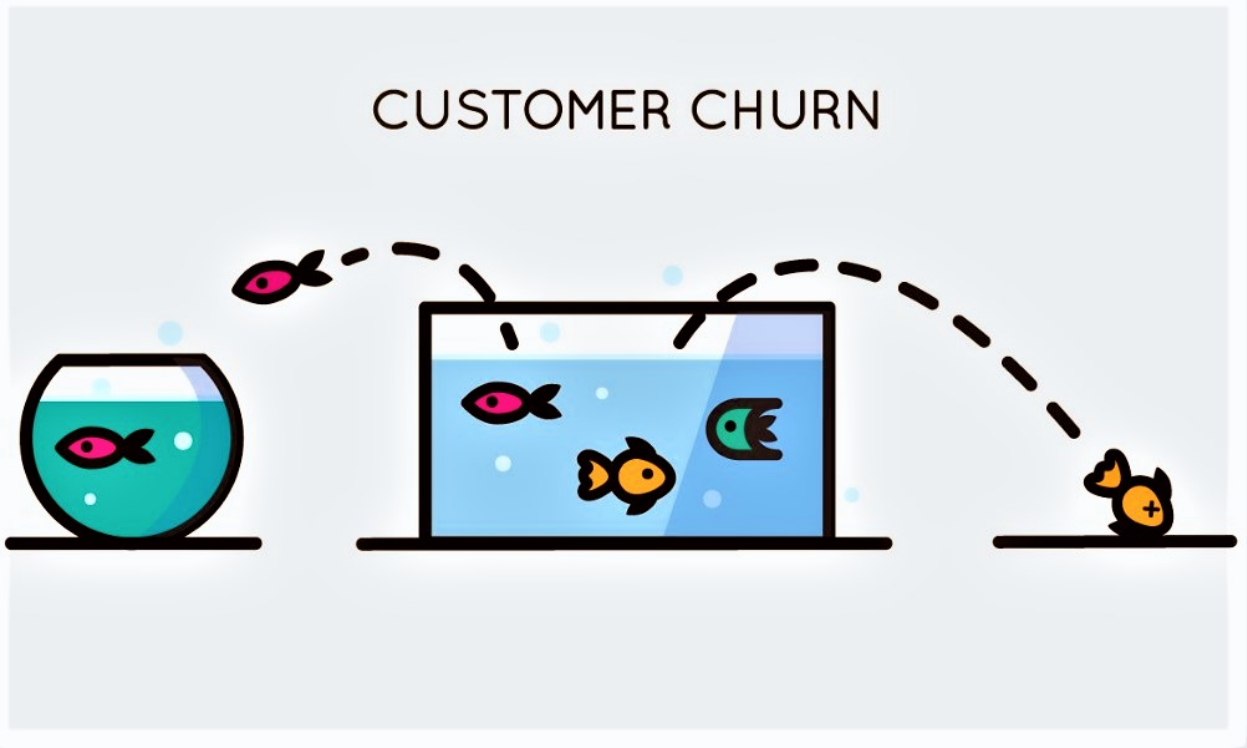


Figure 1: Customer Churn

Customer Churn in Telecom Industry

1 Introduction

1.1 What is Customer Churn

Customer attrition, also known as customer churn, customer turnover, or customer defection, is the loss of clients or customers.

1.2 Why Predict Customer Churn

Current organizations face a huge challenge: to be able to anticipate to customers' abandon in order to retain them on time, reducing this way costs and risks and gaining efficiency and competitivity.

2 Libraries/ Packages

```
library(DataExplorer)
```

```
## Warning: package 'DataExplorer' was built under R version 3.5.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.3
```

```
library(caTools)
```

```
## Warning: package 'caTools' was built under R version 3.5.3
```

```
library(xlsx)
```

```
## Warning: package 'xlsx' was built under R version 3.5.2
```

```
library(skimr)
```

```
## Warning: package 'skimr' was built under R version 3.5.3
```

```
##
## Attaching package: 'skimr'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.5.3
```

```
## Loading required package: lattice
```

```
## Warning: package 'lattice' was built under R version 3.5.2
```

```
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 3.5.3
```

```
##
## ************************************************************
```

```
## Note: As of version 1.0.0, cowplot does not change the
```

```
##   default ggplot2 theme anymore. To recover the previous
```

```
##   behavior, execute:
##   theme_set(theme_cowplot())
```

```
## ************************************************************
```

```
library(caTools)
library(ROSE)
```

```
## Warning: package 'ROSE' was built under R version 3.5.3
```

```
## Loaded ROSE 0.0-3
```

```
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 3.5.3
```

```
## Loading required package: gplots
```

```
## Warning: package 'gplots' was built under R version 3.5.2
##
## Attaching package: 'gplots'
## The following object is masked from 'package:stats':
##
##     lowess
library(MLmetrics)
## Warning: package 'MLmetrics' was built under R version 3.5.3
##
## Attaching package: 'MLmetrics'
## The following objects are masked from 'package:caret':
##
##     MAE, RMSE
## The following object is masked from 'package:base':
##
##     Recall
library(MASS)
## Warning: package 'MASS' was built under R version 3.5.3
library(class)
## Warning: package 'class' was built under R version 3.5.2
library(e1071)
## Warning: package 'e1071' was built under R version 3.5.3
library(car)
## Warning: package 'car' was built under R version 3.5.3
## Loading required package: carData
## Warning: package 'carData' was built under R version 3.5.2
```

3 Project 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 postpaid 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.

You are expected to do the following :

3.1 EDA

- How does the data look like, Univariate and bivariate analysis. Plots and charts which illustrate the relationships between variables
- Look out for outliers and missing value
- Check and treat for multicollinearity
- Summarize the insights you get from EDA

3.2 Build Models and compare them to get to the best one

- Logistic Regression
- KNN
- Naive Bayes
- Model Comparison using Model Performance metrics & Interpretation

3.3 Actionable Insights

- Interpretation & Recommendations from the best model

4 Step by Step Approach

4.1 EDA

4.1.1 Set Working Directory

```
setwd("Z:\\Projects\\Predictive\\Customer Churn")
getwd()
```

```
## [1] "Z:/Projects/Predictive/Customer Churn"
```

4.1.2 Import Data

```
mydata<-read.xlsx2("Cellphone.xlsx",  sheetIndex = 2, header = TRUE)
```

Imported the xlsx datasheet with index 2 as the data is in the 2nd sheet.

4.1.3 Check Dataset

```
skim(mydata)
```

```
## Skim summary statistics
##  n obs: 3333
##  n variables: 11
##
## -- Variable type:factor ------------------------------------------------------------------------
##         variable missing complete    n n_unique
##     AccountWeeks       0     3333 3333      212
##            Churn       0     3333 3333        2
##  ContractRenewal       0     3333 3333        2
##    CustServCalls       0     3333 3333       10
##         DataPlan       0     3333 3333        2
##        DataUsage       0     3333 3333      174
##         DayCalls       0     3333 3333      119
##          DayMins       0     3333 3333     1667
##    MonthlyCharge       0     3333 3333      656
##       OverageFee       0     3333 3333     1024
##         RoamMins       0     3333 3333      162
##                                    top_counts ordered
##   105: 43, 87: 42, 101: 40, 93: 40             FALSE
##             0: 2850, 1: 483, NA: 0             FALSE
##             1: 3010, 0: 323, NA: 0             FALSE
##    1: 1181, 2: 759, 0: 697, 3: 429             FALSE
##             0: 2411, 1: 922, NA: 0             FALSE
##                          0: 1813, 0.3: 41, 0    FALSE
##  102: 78, 105: 75, 107: 69, 95: 69             FALSE
##     154: 8, 159: 8, 174: 8, 162: 7             FALSE
##     50: 84, 46: 75, 45: 74, 49: 73             FALSE
##                                  8.5: 13, 10   FALSE
##                         10: 62, 11.: 59, 10    FALSE
```

Findings

*There are 3333 observations and 11 variables.

```r
str(mydata)
```

```
## 'data.frame':    3333 obs. of  11 variables:
##  $ Churn          : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ AccountWeeks   : Factor w/ 212 levels "1","10","100",..: 33 10 43 196 186 22 26 53 21 47 ...
##  $ ContractRenewal: Factor w/ 2 levels "0","1": 2 2 2 1 1 1 2 1 2 1 ...
##  $ DataPlan       : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 2 1 1 2 ...
##  $ DataUsage      : Factor w/ 174 levels "0","0.11","0.12",..: 104 141 1 1 1 1 79 1 10 116 ...
##  $ CustServCalls  : Factor w/ 10 levels "0","1","2","3",..: 2 2 1 3 4 1 4 1 2 1 ...
##  $ DayMins        : Factor w/ 1667 levels "0","100","100.1",..: 1306 477 1176 1442 521 1022 976 435 
##  $ DayCalls       : Factor w/ 119 levels "0","100","101",..: 12 25 16 91 15 118 108 99 117 104 ...
##  $ MonthlyCharge  : Factor w/ 656 levels "100","100.30000000000001",..: 593 530 224 272 125 272 580 8
##  $ OverageFee     : Factor w/ 1024 levels "0","1.56","10",..: 1012 1003 652 532 768 105 514 597 517 
##  $ RoamMins       : Factor w/ 162 levels "0","1.1","1.3",..: 4 41 26 129 5 126 138 134 150 16 ...
```

Findings

* AccountWeeks, DataUsage, DataUsage, CustServCalls, DayMins, DayCalls, MonthlyCharge, OverageFee, RoamMins are incorrectly imported as factors. Convert factors to numeric data types.
* Churn , ContractRenewal and DataPlan are factor variables.

4.1.4 Creating Backup of the original dataset

```
cleandata <- data.frame(mydata)
```

4.1.5 AccountWeeks, DataUsage, DataUsage, CustServCalls, DayMins, DayCalls, MonthlyCharge, OverageFee, RoamMins are incorrectly imported as factors. Convert factors to numeric data types.

```
cleandata$AccountWeeks<-as.numeric(as.character(cleandata$AccountWeeks))
```

```
cleandata$DataUsage<-as.numeric(as.character(cleandata$DataUsage))
```

```
cleandata$CustServCalls<-as.numeric(as.character(cleandata$CustServCalls))
```

```
cleandata$DayMins<-as.numeric(as.character(cleandata$DayMins))
```

```
cleandata$DayCalls<-as.numeric(as.character(cleandata$DayCalls))
```

```
cleandata$DayCalls<-as.numeric(as.character(cleandata$DayCalls))
```

```
cleandata$MonthlyCharge<-as.numeric(as.character(cleandata$MonthlyCharge))
```

```
cleandata$OverageFee<-as.numeric(as.character(cleandata$OverageFee))
```

```
cleandata$RoamMins<-as.numeric(as.character(cleandata$RoamMins))
```

4.1.6 Checking sturcture of data

```
str(cleandata)
```

```
## 'data.frame':    3333 obs. of  11 variables:
##  $ Churn          : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ AccountWeeks   : num  128 107 137 84 75 118 121 147 117 141 ...
##  $ ContractRenewal: Factor w/ 2 levels "0","1": 2 2 2 1 1 1 2 1 2 1 ...
##  $ DataPlan       : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 2 1 1 2 ...
##  $ 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 ...
```

4.1.7 Dataset details

```
introduce(cleandata)
```

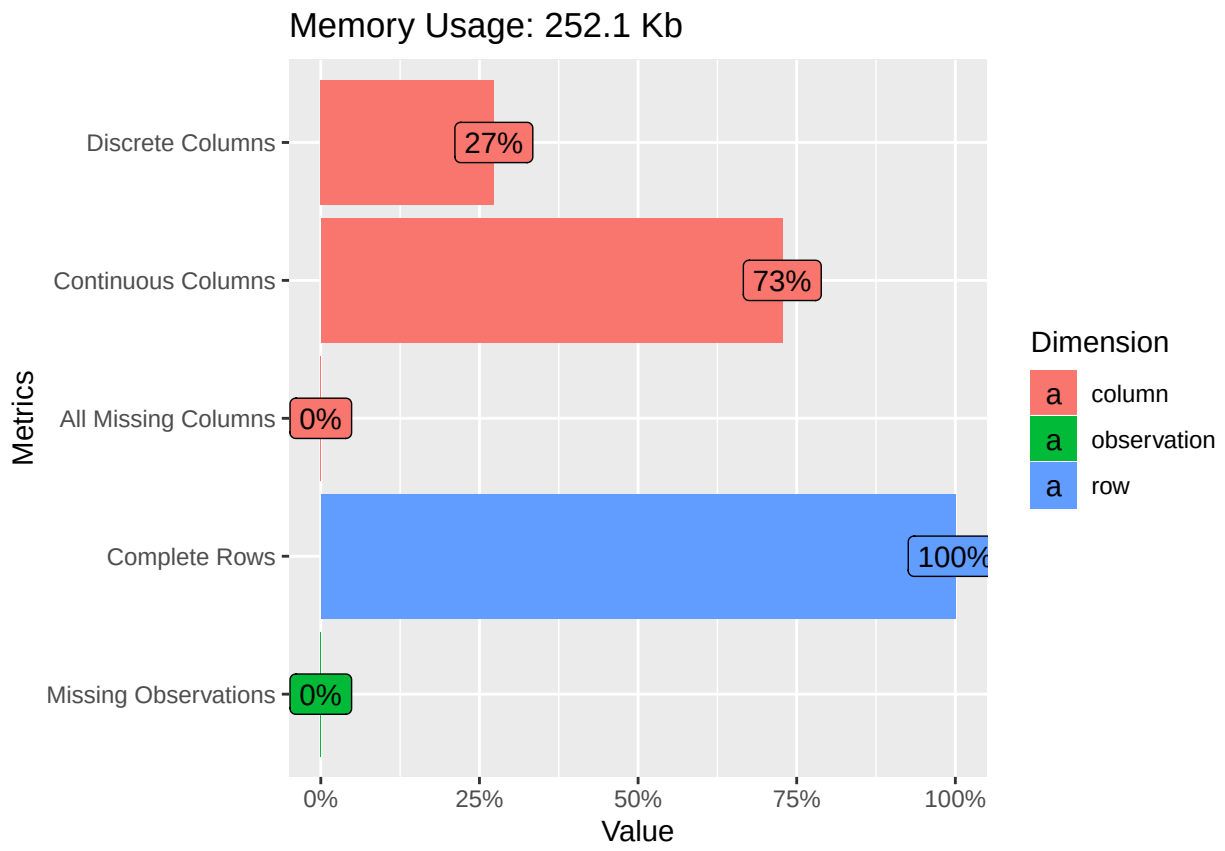
```
##   rows columns discrete_columns continuous_columns all_missing_columns
## 1 3333      11               3                 8                   0
##   total_missing_values complete_rows total_observations memory_usage
## 1                    0          3333              36663       258112
```

Findings

- There are **3333 Rows** and **11 Columns**
- 3 discrete colums and 8 continuous columns

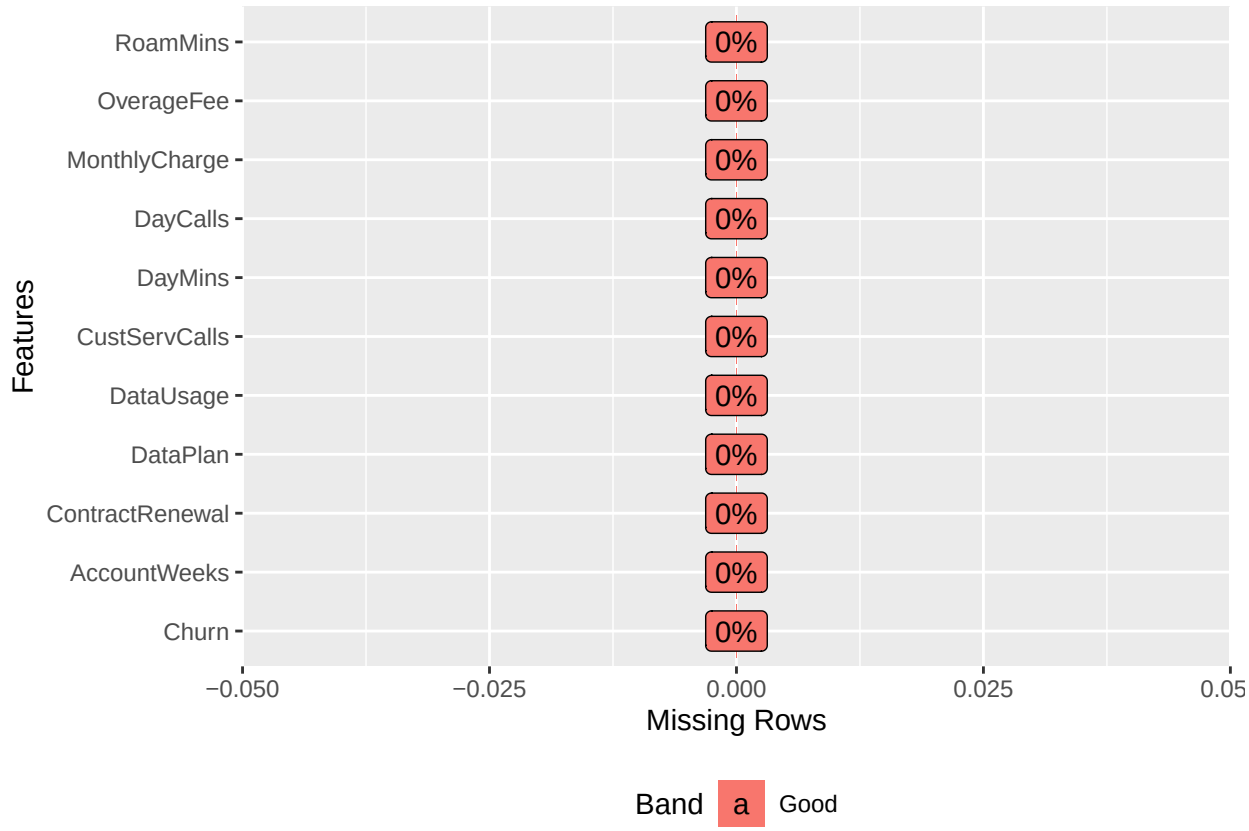
4.1.8 Visualize dataset

```
plot_intro(cleandata)
```



4.1.9 Checking for Missing Values

```
plot_missing(cleandata)
```



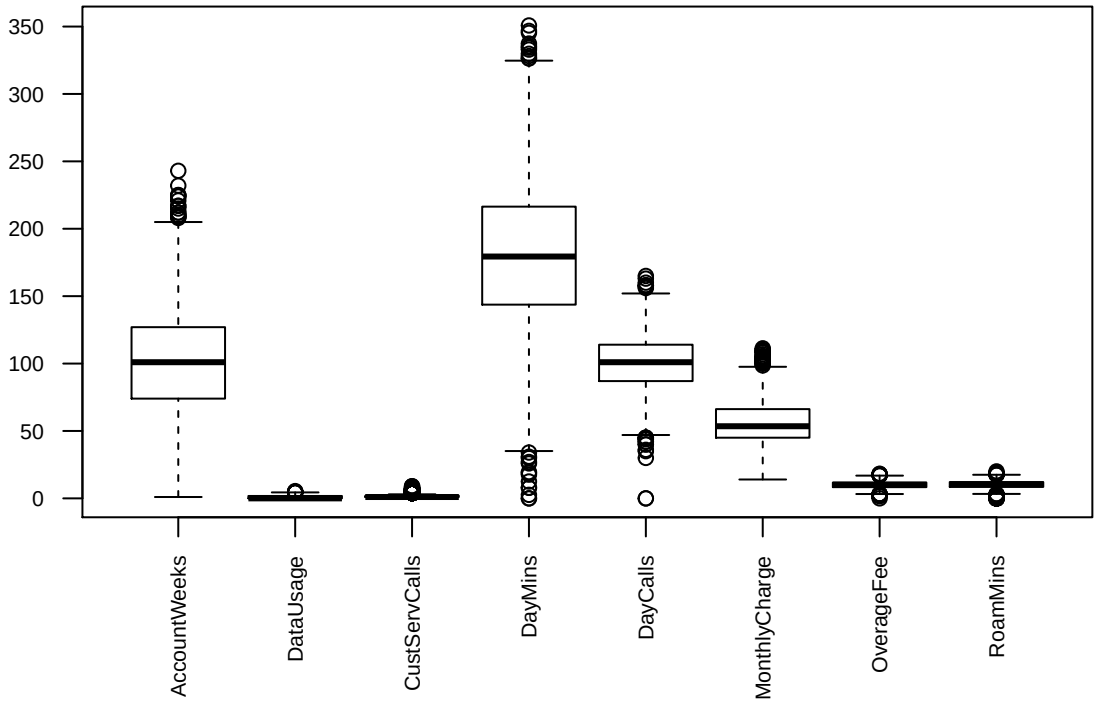
Findings

- There are no missing values in the dataset.

```
scaledata <- data.frame(cleandata)
```

4.1.10 Check if there are Outliers

```
boxplot(scaledata[,c(2,5,6,7,8,9,10,11)],las=2,cex.axis=0.7)
```



Findings

- There are outliers in the dataset.

4.1.11 Outlier Treatment. Winsorizing.

```
summary(scaledata$AccountWeeks)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     1.0    74.0   101.0   101.1   127.0   243.0
```

```
benchAccountWeeks <- 127.0 + 1.5* IQR(scaledata$AccountWeeks)
scaledata$AccountWeeks[scaledata$AccountWeeks>benchAccountWeeks] <- benchAccountWeeks
```

```
summary(scaledata$DataUsage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.0000  0.8165  1.7800  5.4000
```

```
benchDataUsage <- 1.78 + 1.5* IQR(scaledata$DataUsage)
scaledata$DataUsage[scaledata$DataUsage>benchDataUsage] <- benchDataUsage
```

```
summary(scaledata$DayMins)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   143.7   179.4   179.8   216.4   350.8
```

```
benchDayMins <- 216.4 + 1.5* IQR(scaledata$DayMins)
scaledata$DayMins[scaledata$DayMins>benchDayMins] <- benchDayMins
```

```
summary(scaledata$DayCalls)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0    87.0   101.0   100.4   114.0   165.0
```

```
benchDayCalls <- 114.0 + 1.5* IQR(scaledata$DayCalls)
scaledata$DayCalls[scaledata$DayCalls>benchDayCalls] <- benchDayCalls
```

```
summary(scaledata$MonthlyCharge)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.00   45.00   53.50   56.31   66.20  111.30
```

```
benchMonthlyCharge <- 66.20 + 1.5* IQR(scaledata$MonthlyCharge)
scaledata$MonthlyCharge[scaledata$MonthlyCharge>benchMonthlyCharge] <- benchMonthlyCharge
```

```
summary(scaledata$OverageFee)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    8.33   10.07   10.05   11.77   18.19
```

```
benchOverageFee <- 11.77 + 1.5* IQR(scaledata$OverageFee)
scaledata$OverageFee[scaledata$OverageFee>benchOverageFee] <- benchOverageFee
```

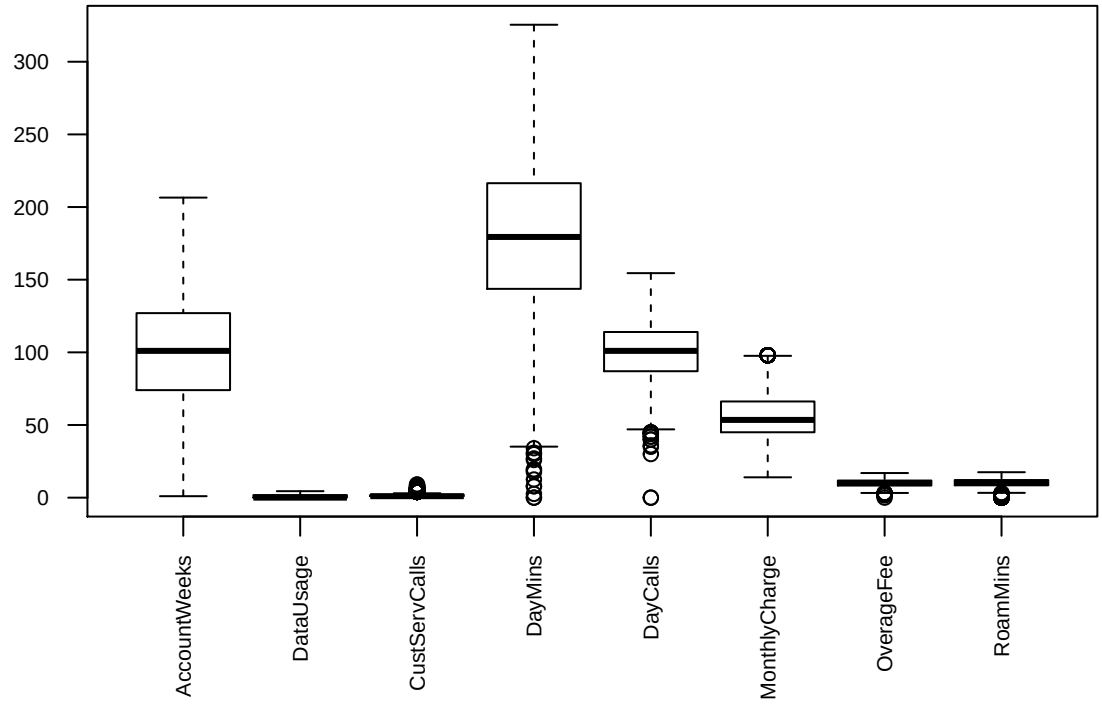
```
summary(scaledata$RoamMins)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    8.50   10.30   10.24   12.10   20.00
```

```
benchRoamMins <- 12.10 + 1.5* IQR(scaledata$RoamMins)
scaledata$RoamMins[scaledata$RoamMins>benchRoamMins] <- benchRoamMins
```

4.1.12 Outliers Treated.

```
boxplot(scaledata[,c(2,5,6,7,8,9,10,11)],las=2,cex.axis=0.7)
```



4.1.13 Checking Churn data split

```
table(scaledata$Churn)
```

```
##
##    0    1
## 2850  483
```

```
483/(2850+483)*100
```

```
## [1] 14.49145
```

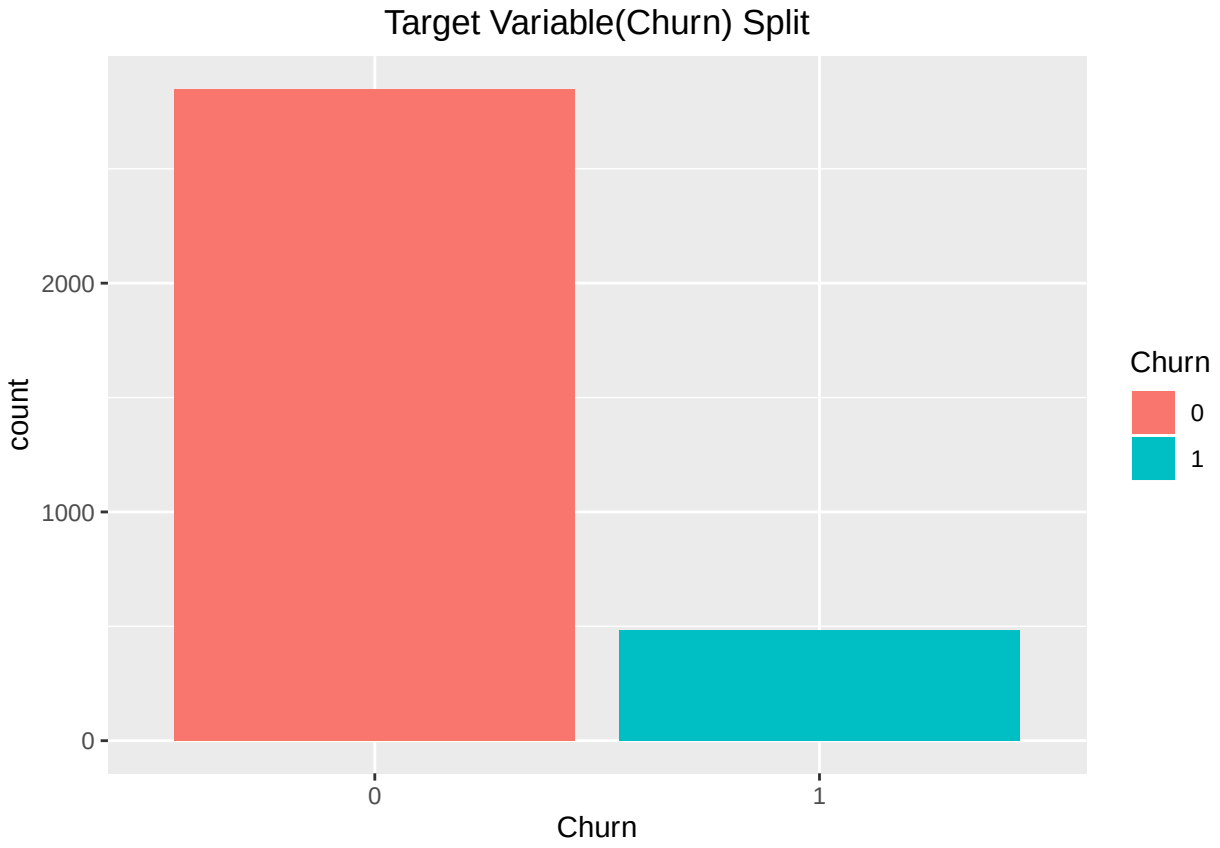
Findings

- Minority Class (1) is only 14.49 percent of the total Churn class. It is less than 15%.
- Data will be balanced.

4.1.14 Visualize Dependent Variable(Churn) proportion split

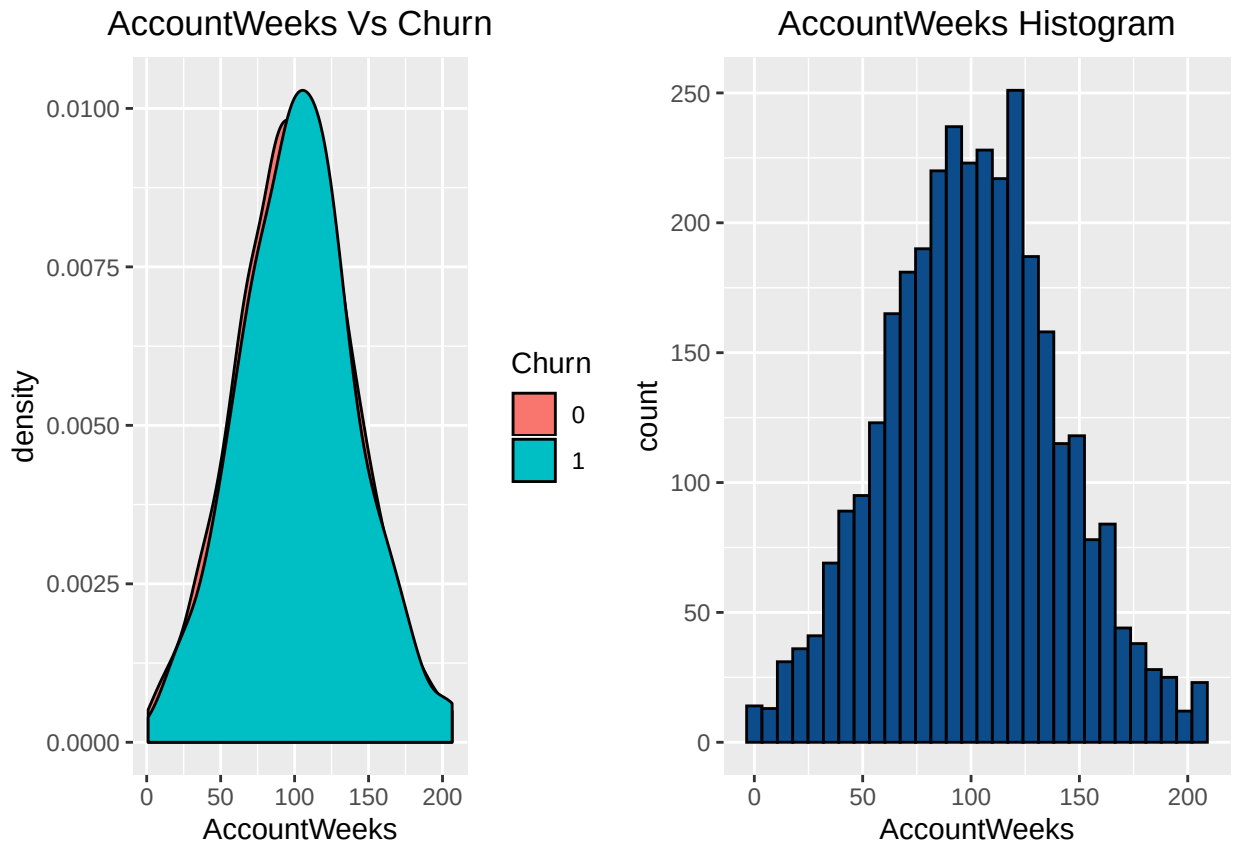
```
ggplot(scaledata) +
 aes(x = Churn, fill = Churn) +
 geom_bar() +
```

```
 scale_fill_hue() +
 labs(title = "Target Variable(Churn) Split") +theme(plot.title = element_text(hjust = 0.5))
```



4.1.15 Dependent Variables VS Independent Variables

```
d1 <- ggplot(scaledata) +
 aes(x = AccountWeeks, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "AccountWeeks Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h1 <- ggplot(scaledata) +
 aes(x = AccountWeeks) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "AccountWeeks Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d1,h1)
```

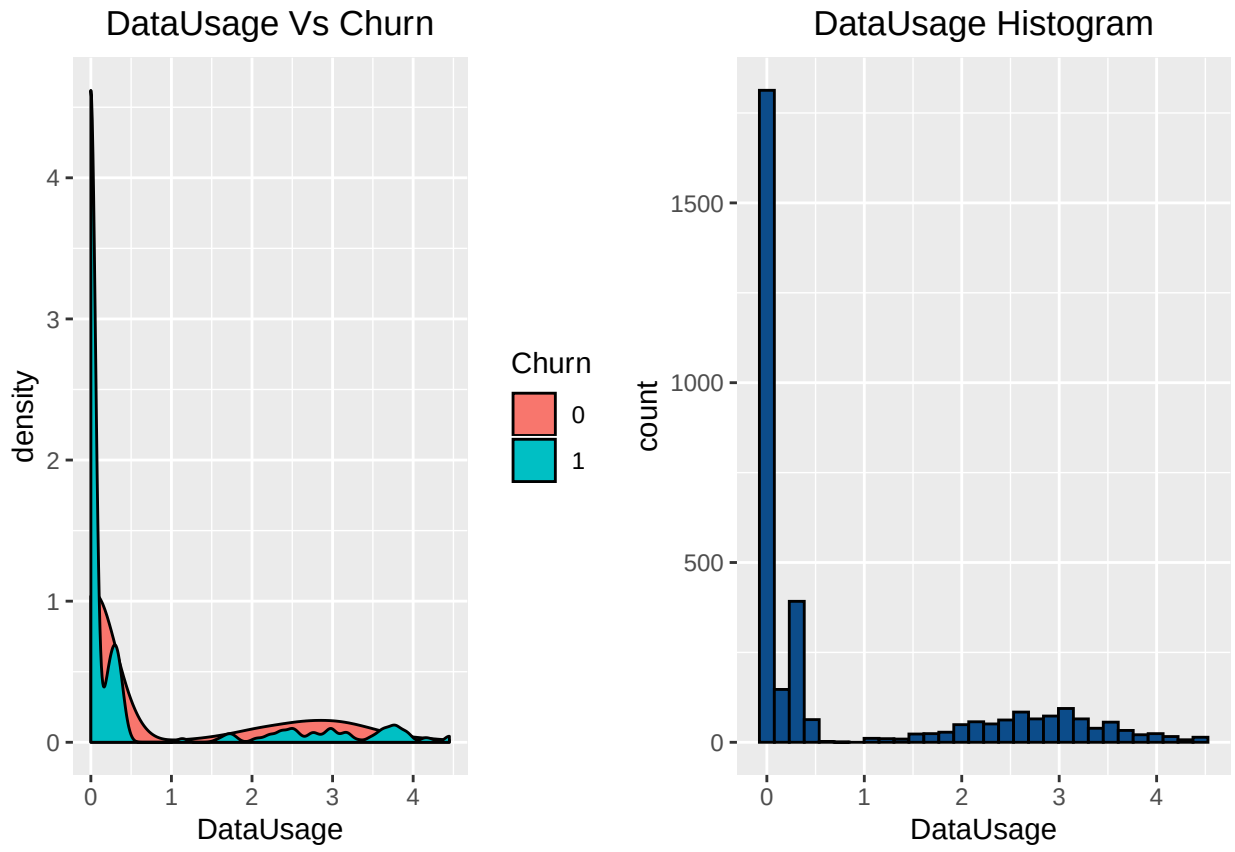


Findings

Density plot: AccountWeeks around 100 has more Churn Count.

Histogram: We can say that the frequency distribution for AccountWeeks are normally distributed.

```
d2 <- ggplot(scaledata) +
 aes(x = DataUsage, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "DataUsage Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h2 <- ggplot(scaledata) +
 aes(x = DataUsage) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "DataUsage Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d2,h2)
```

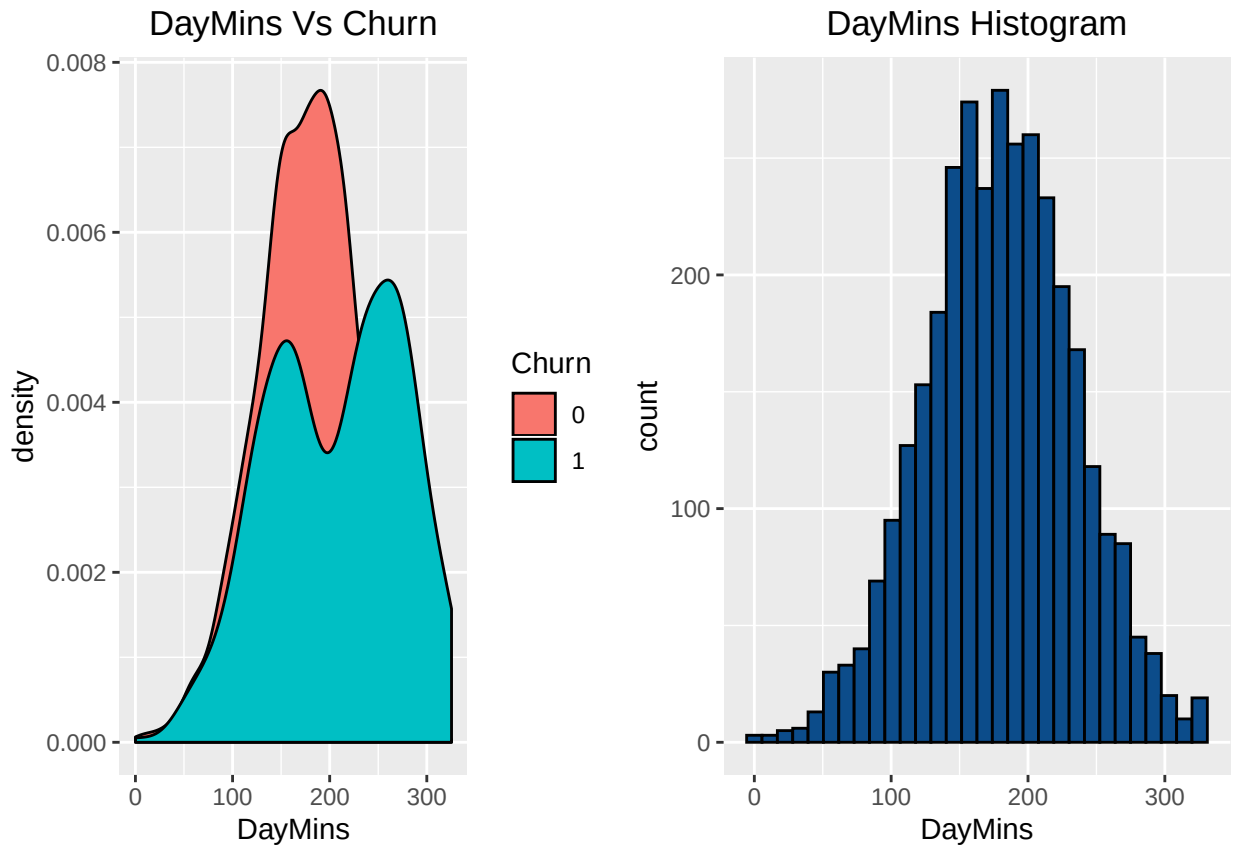


Findings

Density plot: There are more **0** DataUsage in the dataset.

Histogram: There are more **0** DataUsage in the dataset.

```
d3 <- ggplot(scaledata) +
 aes(x = DayMins, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "DayMins Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h3 <- ggplot(scaledata) +
 aes(x = DayMins) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "DayMins Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d3,h3)
```

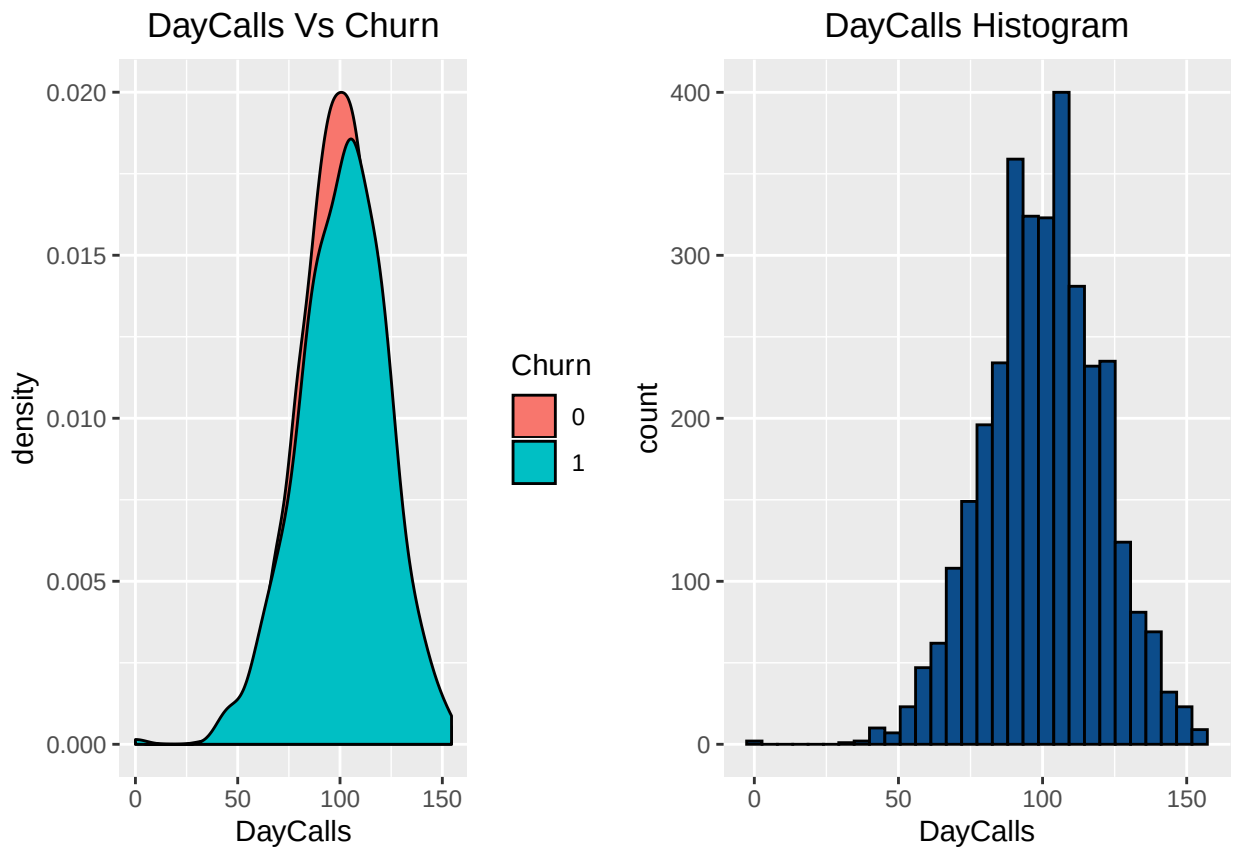


Findings

Density plot: Customers have Churned more for DayMins between **150 and 280** .

Histogram: DaysMins is normally Distributed

```
d4 <- ggplot(scaledata) +
 aes(x = DayCalls, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "DayCalls Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h4 <- ggplot(scaledata) +
 aes(x = DayCalls) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "DayCalls Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d4,h4)
```

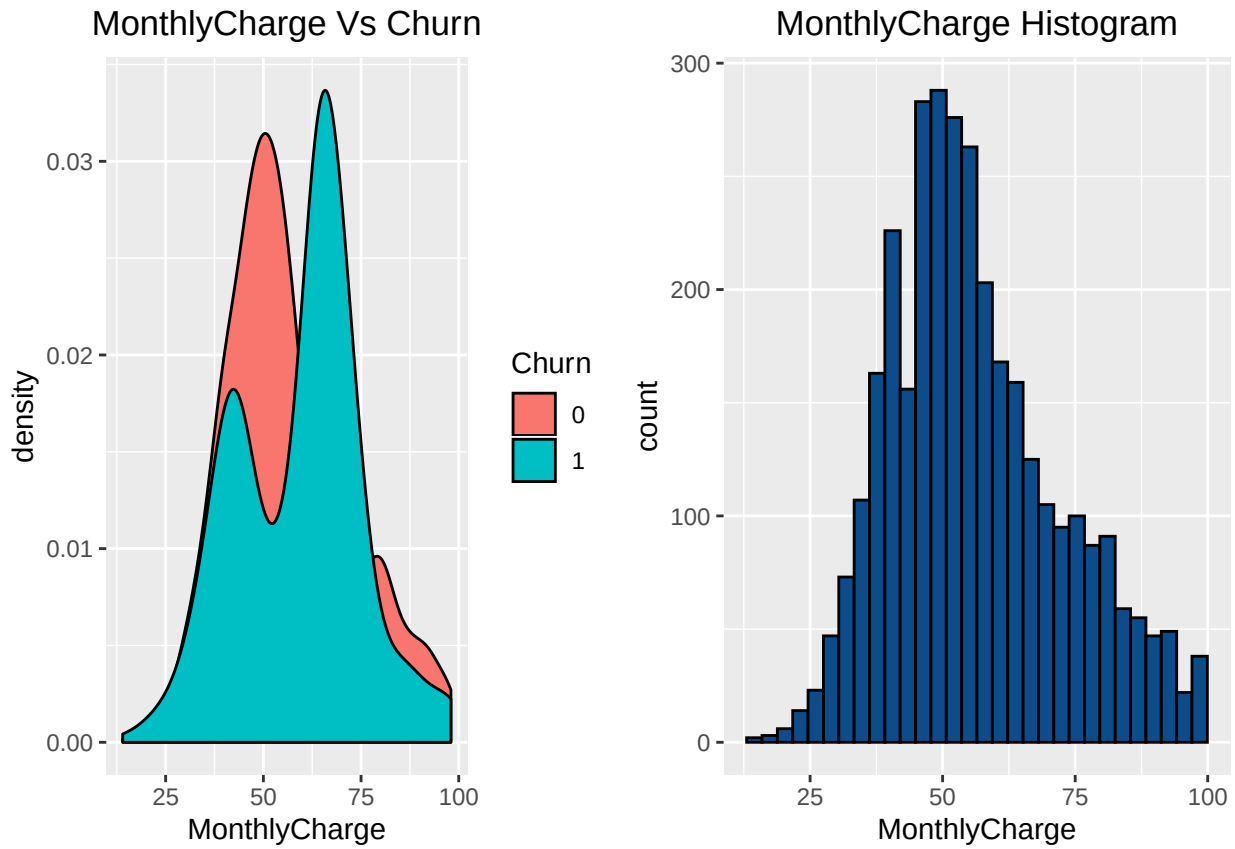


Findings

Density plot: DayCalls are almost evenly distributed for Churn and Not Churn .

Histogram: DayCalls is normally Distributed

```
d5 <- ggplot(scaledata) +
 aes(x = MonthlyCharge, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "MonthlyCharge Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h5 <- ggplot(scaledata) +
 aes(x = MonthlyCharge) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "MonthlyCharge Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d5,h5)
```

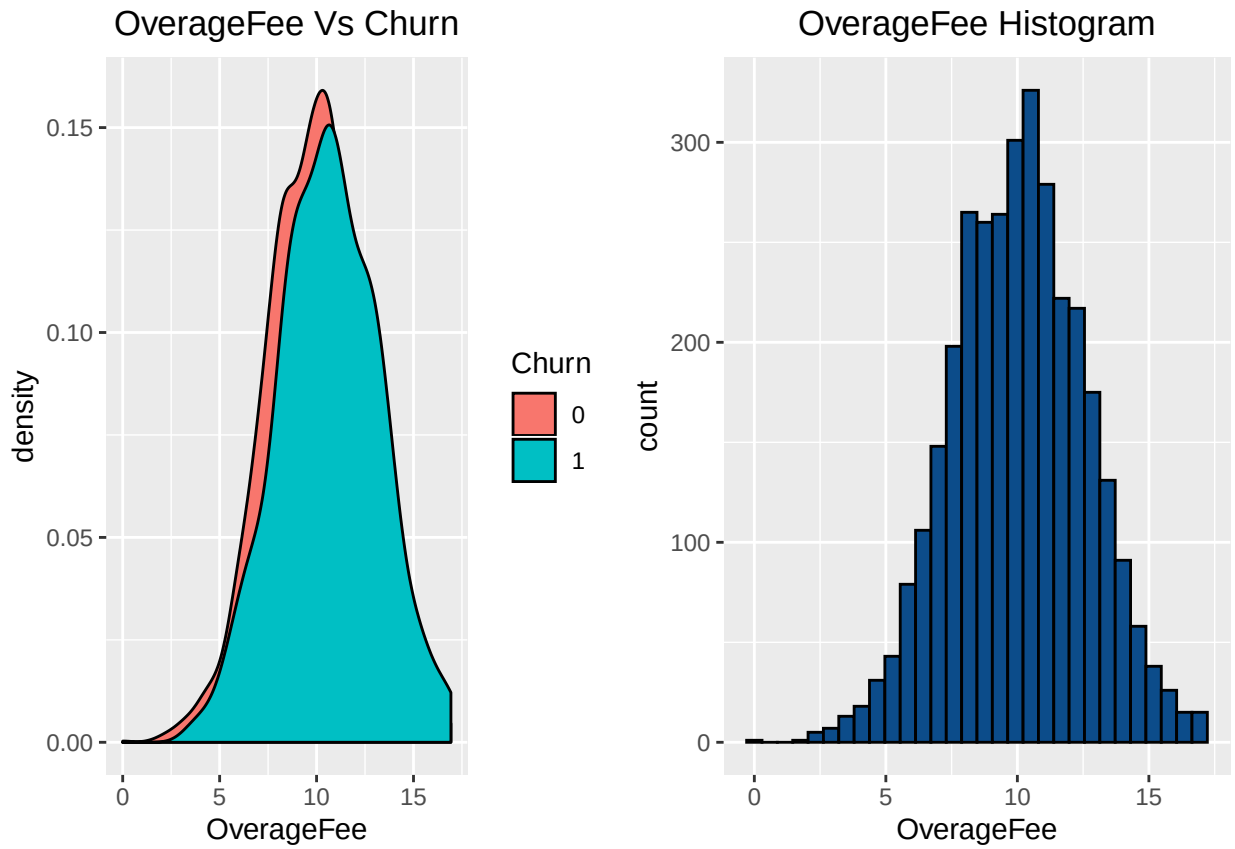


Findings

Density plot: Customer with average monthly bill between **50 and 75** is more likely to Churn.

Histogram: MonthlyCharge is normally Distributed

```
d6 <- ggplot(scaledata) +
 aes(x = OverageFee, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "OverageFee Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h6 <- ggplot(scaledata) +
 aes(x = OverageFee) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "OverageFee Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d6,h6)
```

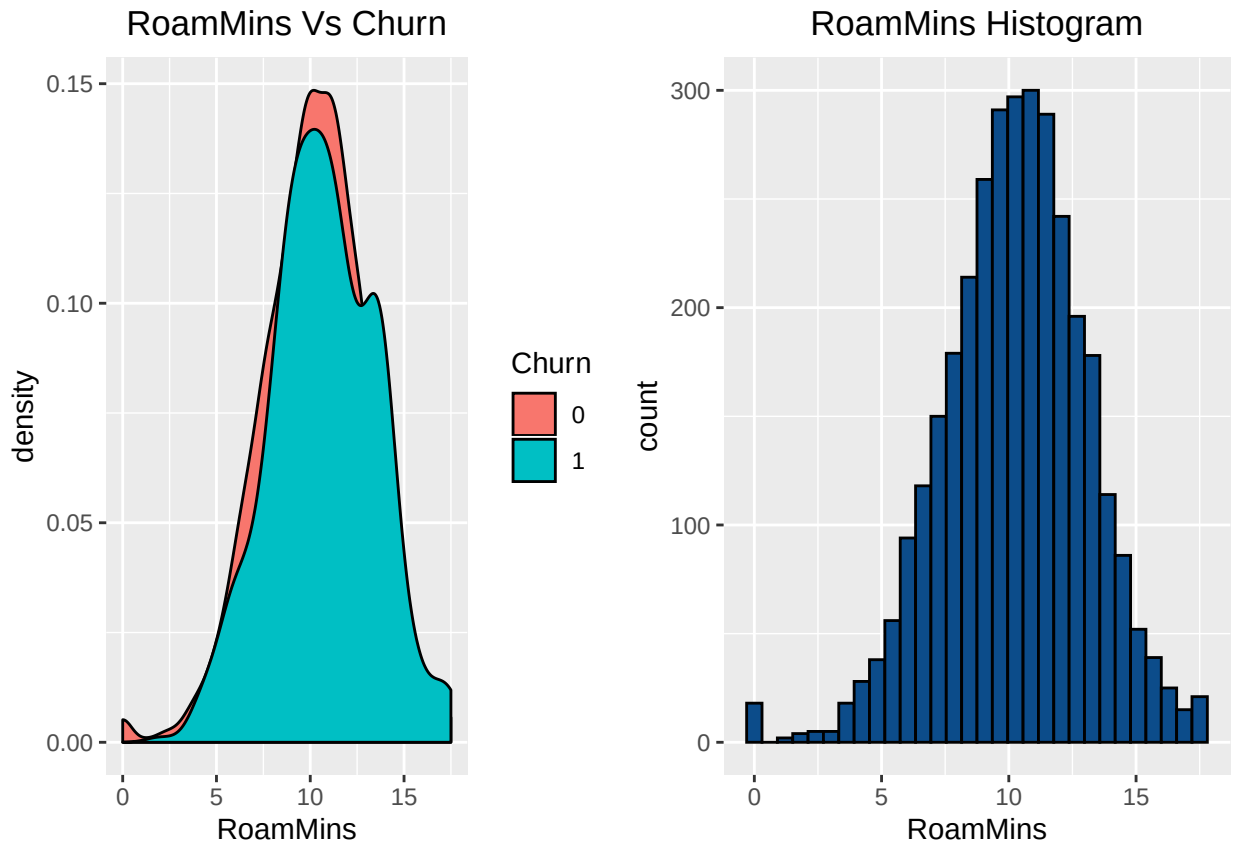


Findings

Density plot: Customer with Overcharge fee between **5 and 15** are more likely to cancel the service.

Histogram: Overcharge is normally Distributed

```
d7 <- ggplot(scaledata) +
 aes(x = RoamMins, fill = Churn) +
 geom_density(adjust = 1L) +
 scale_fill_hue() +
 labs(title = "RoamMins Vs Churn") +
 theme(plot.title = element_text(hjust = 0.5))


h7 <- ggplot(scaledata) +
 aes(x = RoamMins) +
 geom_histogram(bins = 30L, fill = "#0c4c8a",colour="black") +
 labs(title = "RoamMins Histogram") +
 theme(plot.title = element_text(hjust = 0.5))

plot_grid(d7,h7)
```

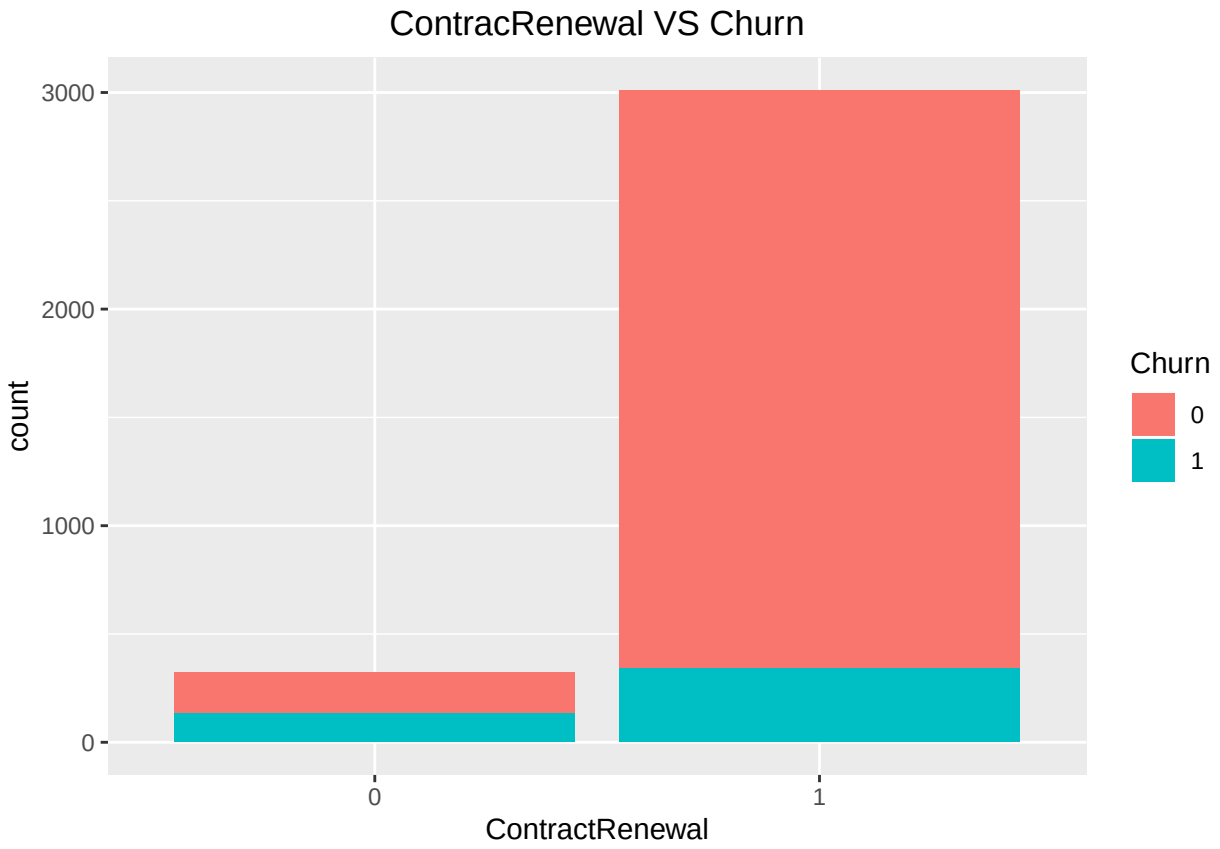


Findings

Density plot: Out of Customers who have cancelled the service for RoamMins are most arund 10 minutes

Histogram: RoamMins is normally Distributed

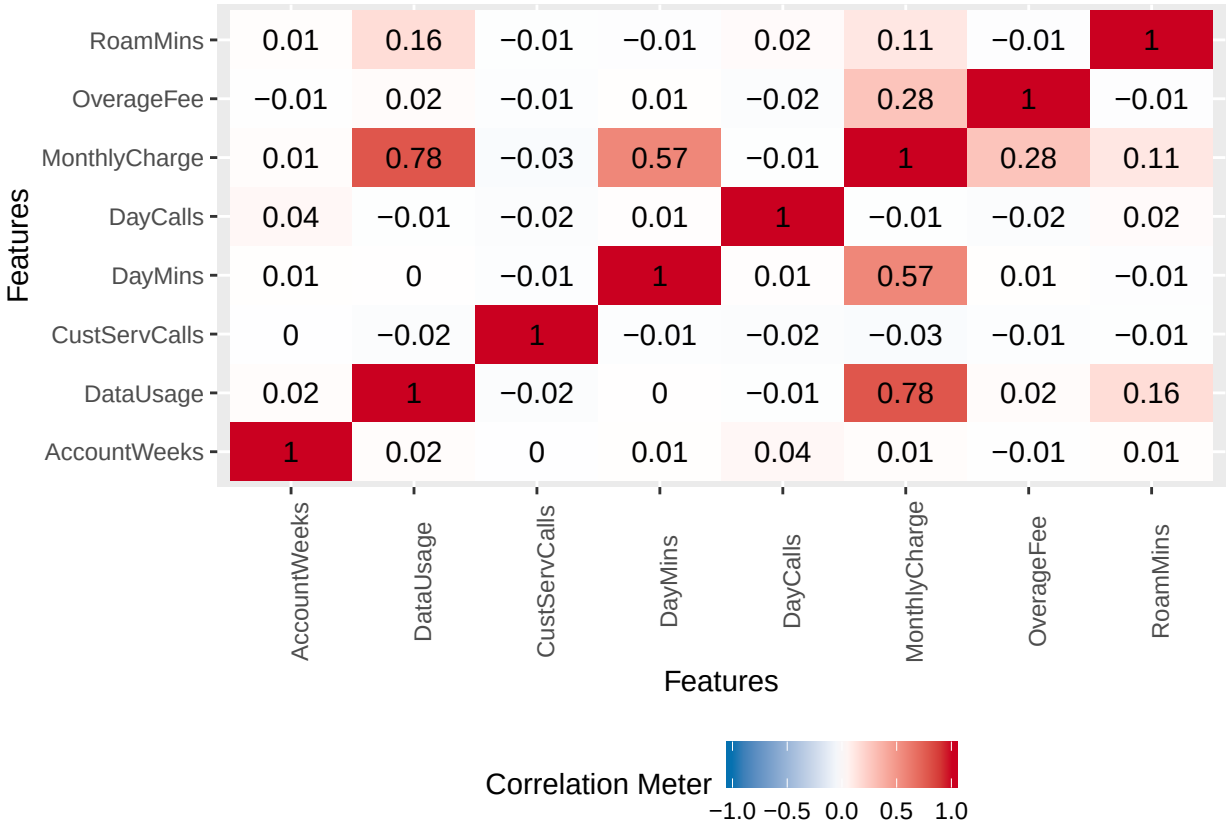
```
h8<- ggplot(scaledata) +
 aes(x = ContractRenewal, fill = Churn) +
 geom_bar() +
 scale_fill_hue() +
 labs(title = "ContracRenewal VS Churn") +
 theme(plot.title = element_text(hjust = 0.5))

h8
```



Findings

Bar Chart: Customers who have renewed the plan are more likely to stay with the company

```
plot_correlation(scaledata, type = c("continuous"))
```



Findings

- DataUsage and MonthlyCharge are Highly Correlated.
- MonthlyCharge and DayMins are Highly Correlated.

---

5 Build Models

5.1 Pre Process(Train and Test Split)

```
seed <- 101
set.seed(seed)
sample <- sample.split(scaledata,SplitRatio = 0.7)
churn.train <- subset(scaledata,sample == TRUE)
churn.test <- subset(scaledata,sample == FALSE)
nrow(churn.train)
```

```
## [1] 2121
```

```
nrow(churn.test)
```

```
## [1] 1212
```

Balance Dataset with ROSE package

```
BalancedData=ROSE(Churn~.,data=churn.train,seed=seed)$data
```

5.1.1 Before Balance VS After Balance

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

```
##
##    0    1
## 1830  291
```

```
table(BalancedData$Churn)
```

```
##
##    0    1
## 1062 1059
```

Findings

- Dataset is now BALANCED

5.2 Logistic Regression

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).

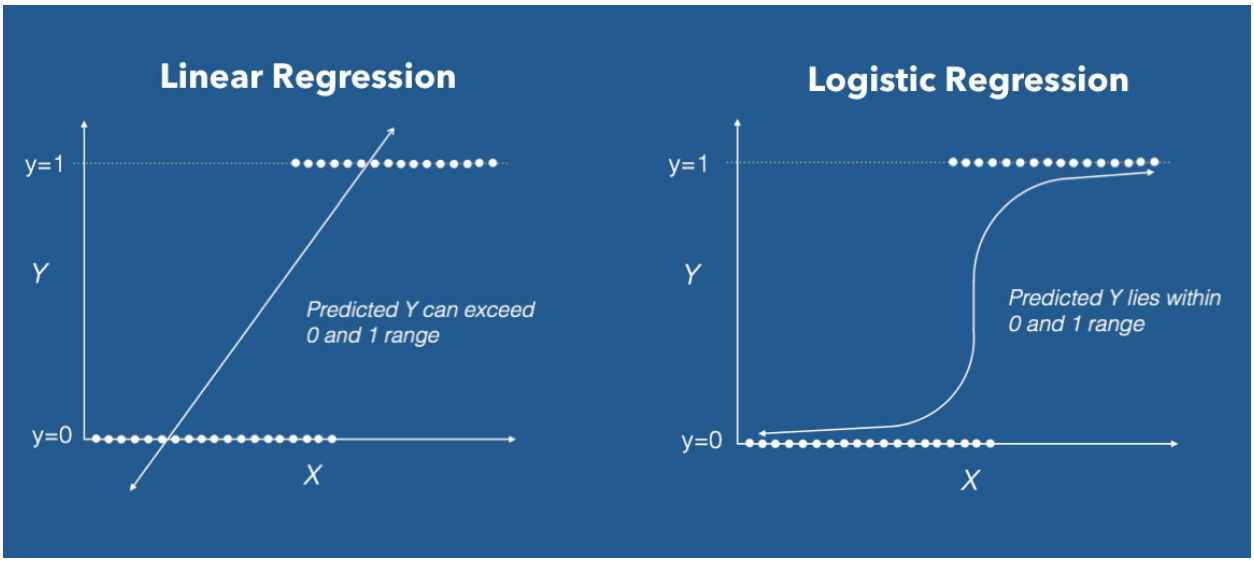


Figure 2: Logistic VS Linear

5.2.1 Full Model for stepwise variable selection and elimination

```
fullmod <- glm(Churn ~.,family=binomial,data = BalancedData)

summary(fullmod)
```

```
##
## Call:
## glm(formula = Churn ~ ., family = binomial, data = BalancedData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6895  -0.8910  -0.2079   0.9166   2.7396
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -2.2800535  0.4230727  -5.389 7.07e-08 ***
## AccountWeeks     -0.0016946  0.0011316  -1.498 0.134263
## ContractRenewal1 -2.4252907  0.1698024 -14.283  < 2e-16 ***
## DataPlan1        -1.2451647  0.2480346  -5.020 5.16e-07 ***
## DataUsage        -0.0460423  0.0692316  -0.665 0.506021
## CustServCalls     0.4355874  0.0318817  13.663  < 2e-16 ***
## DayMins           0.0060290  0.0009854   6.119 9.45e-10 ***
## DayCalls          0.0038169  0.0022627   1.687 0.091635 .
## MonthlyCharge     0.0171697  0.0046837   3.666 0.000247 ***
## OverageFee        0.0899965  0.0196366   4.583 4.58e-06 ***
## RoamMins          0.0567465  0.0167447   3.389 0.000702 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2940.3  on 2120  degrees of freedom
```

```
## Residual deviance: 2315.4  on 2110  degrees of freedom
## AIC: 2337.4
##
## Number of Fisher Scoring iterations: 4
```

5.2.2 Empty Model for stepwise variable selection and elimination

```
emptyModel<- glm(Churn ~ 1,family=binomial,data = BalancedData)

summary(emptyModel)
```

```
##
## Call:
## glm(formula = Churn ~ 1, family = binomial, data = BalancedData)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.176  -1.176  -1.176   1.179   1.179
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.002829   0.043427  -0.065    0.948
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2940.3  on 2120  degrees of freedom
## Residual deviance: 2940.3  on 2120  degrees of freedom
## AIC: 2942.3
##
## Number of Fisher Scoring iterations: 3
```

5.2.3 Baclward Selection of significant variables

```
backwards = step(fullmod)
```

```
## Start:  AIC=2337.4
## Churn ~ AccountWeeks + ContractRenewal + DataPlan + DataUsage +
##     CustServCalls + DayMins + DayCalls + MonthlyCharge + OverageFee +
##     RoamMins
##
##                   Df Deviance    AIC
## - DataUsage        1   2315.8 2335.8
## <none>                 2315.4 2337.4
## - AccountWeeks     1   2317.6 2337.6
## - DayCalls         1   2318.2 2338.2
## - RoamMins         1   2327.0 2347.0
## - MonthlyCharge    1   2329.0 2349.0
## - OverageFee       1   2336.7 2356.7
## - DataPlan         1   2341.2 2361.2
## - DayMins          1   2354.0 2374.0
## - CustServCalls    1   2534.1 2554.1
## - ContractRenewal  1   2587.1 2607.1
##
## Step:  AIC=2335.84
```

```
## Churn ~ AccountWeeks + ContractRenewal + DataPlan + CustServCalls +
##     DayMins + DayCalls + MonthlyCharge + OverageFee + RoamMins
##
##                   Df Deviance    AIC
## <none>                 2315.8 2335.8
## - AccountWeeks     1   2318.2 2336.2
## - DayCalls         1   2318.7 2336.7
## - RoamMins         1   2327.0 2345.0
## - MonthlyCharge    1   2329.0 2347.0
## - OverageFee       1   2337.5 2355.5
## - DayMins          1   2355.7 2373.7
## - DataPlan         1   2378.2 2396.2
## - CustServCalls    1   2535.0 2553.0
## - ContractRenewal  1   2587.3 2605.3
```

5.2.4 Forward Slection

```
forwards = step(emptyModel,scope=list(lower=formula(emptyModel),upper=formula(fullmod)), direction="forw
```

```
## Start:  AIC=2942.33
## Churn ~ 1
##
##                   Df Deviance    AIC
## + ContractRenewal  1   2736.0 2740.0
## + CustServCalls    1   2839.6 2843.6
## + DayMins          1   2854.5 2858.5
## + DataPlan         1   2880.7 2884.7
## + DataUsage        1   2902.3 2906.3
## + OverageFee       1   2907.8 2911.8
## + RoamMins         1   2920.0 2924.0
## + MonthlyCharge    1   2929.9 2933.9
## + DayCalls         1   2937.6 2941.6
## <none>                 2940.3 2942.3
## + AccountWeeks     1   2940.3 2944.3
##
## Step:  AIC=2739.97
## Churn ~ ContractRenewal
##
##                  Df Deviance    AIC
## + CustServCalls  1   2594.0 2600.0
## + DayMins        1   2630.6 2636.6
## + DataPlan       1   2675.4 2681.4
## + DataUsage      1   2692.8 2698.8
## + OverageFee     1   2697.7 2703.7
## + MonthlyCharge  1   2723.6 2729.6
## + RoamMins       1   2729.0 2735.0
## + DayCalls       1   2731.9 2737.9
## <none>               2736.0 2740.0
## + AccountWeeks   1   2735.3 2741.3
##
## Step:  AIC=2599.99
## Churn ~ ContractRenewal + CustServCalls
##
##                  Df Deviance    AIC
```

```
## + DayMins        1   2444.4 2452.4
## + DataPlan       1   2523.0 2531.0
## + OverageFee     1   2543.3 2551.3
## + DataUsage      1   2545.4 2553.4
## + MonthlyCharge  1   2570.2 2578.2
## + RoamMins       1   2585.5 2593.5
## + DayCalls       1   2589.4 2597.4
## <none>               2594.0 2600.0
## + AccountWeeks   1   2592.9 2600.9
##
## Step:  AIC=2452.39
## Churn ~ ContractRenewal + CustServCalls + DayMins
##
##                 Df Deviance    AIC
## + DataPlan       1   2386.2 2396.2
## + DataUsage      1   2406.6 2416.6
## + OverageFee     1   2407.8 2417.8
## + RoamMins       1   2430.8 2440.8
## + DayCalls       1   2441.4 2451.4
## <none>               2444.4 2452.4
## + AccountWeeks   1   2442.8 2452.8
## + MonthlyCharge  1   2443.4 2453.4
##
## Step:  AIC=2396.2
## Churn ~ ContractRenewal + CustServCalls + DayMins + DataPlan
##
##                 Df Deviance    AIC
## + OverageFee     1   2349.2 2361.2
## + MonthlyCharge  1   2350.8 2362.8
## + RoamMins       1   2371.8 2383.8
## <none>               2386.2 2396.2
## + DayCalls       1   2384.2 2396.2
## + AccountWeeks   1   2384.7 2396.7
## + DataUsage      1   2386.0 2398.0
##
## Step:  AIC=2361.25
## Churn ~ ContractRenewal + CustServCalls + DayMins + DataPlan +
##     OverageFee
##
##                 Df Deviance    AIC
## + MonthlyCharge  1   2331.7 2345.7
## + RoamMins       1   2333.4 2347.4
## + DayCalls       1   2347.0 2361.0
## <none>               2349.2 2361.2
## + AccountWeeks   1   2347.8 2361.8
## + DataUsage      1   2349.1 2363.1
##
## Step:  AIC=2345.65
## Churn ~ ContractRenewal + CustServCalls + DayMins + DataPlan +
##     OverageFee + MonthlyCharge
##
##                Df Deviance    AIC
## + RoamMins      1   2320.7 2336.7
## + DayCalls      1   2329.3 2345.3
```

```
## + AccountWeeks  1   2329.6 2345.6
## <none>              2331.7 2345.7
## + DataUsage     1   2331.6 2347.6
##
## Step:  AIC=2336.74
## Churn ~ ContractRenewal + CustServCalls + DayMins + DataPlan +
##     OverageFee + MonthlyCharge + RoamMins
##
##                Df Deviance    AIC
## + DayCalls      1   2318.2 2336.2
## + AccountWeeks  1   2318.7 2336.7
## <none>              2320.7 2336.7
## + DataUsage     1   2320.2 2338.2
##
## Step:  AIC=2336.16
## Churn ~ ContractRenewal + CustServCalls + DayMins + DataPlan +
##     OverageFee + MonthlyCharge + RoamMins + DayCalls
##
##                Df Deviance    AIC
## + AccountWeeks  1   2315.8 2335.8
## <none>              2318.2 2336.2
## + DataUsage     1   2317.6 2337.6
##
## Step:  AIC=2335.84
## Churn ~ ContractRenewal + CustServCalls + DayMins + DataPlan +
##     OverageFee + MonthlyCharge + RoamMins + DayCalls + AccountWeeks
##
##             Df Deviance    AIC
## <none>           2315.8 2335.8
## + DataUsage  1   2315.4 2337.4
```

Findings

- From both Forward Selection and Backward Elimination, we see that DataUsage is insignificant feature.
- Build model without DataUsage

5.2.5 Modelling Logistic Regression with Cross Validation

```
fitControl <- trainControl(
  method = "repeatedcv",
  number = 10,

  savePredictions = TRUE

)
```

5.2.6 Logistic Regression Model

```
lregmodel<- glm(Churn ~  ContractRenewal + CustServCalls + DayMins + DataPlan +
                  OverageFee + MonthlyCharge + RoamMins + DayCalls + AccountWeeks,data=BalancedData,fami
summary(lregmodel)
```

```
##
## Call:
```

```
## glm(formula = Churn ~ ContractRenewal + CustServCalls + DayMins +
##     DataPlan + OverageFee + MonthlyCharge + RoamMins + DayCalls +
##     AccountWeeks, family = "binomial", data = BalancedData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6848  -0.8882  -0.2024   0.9141   2.7517
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -2.2659985  0.4226733  -5.361 8.27e-08 ***
## ContractRenewal1 -2.4223623  0.1695870 -14.284  < 2e-16 ***
## CustServCalls     0.4360201  0.0318788  13.677  < 2e-16 ***
## DayMins           0.0060928  0.0009805   6.214 5.17e-10 ***
## DataPlan1        -1.3612594  0.1767530  -7.701 1.35e-14 ***
## OverageFee        0.0905252  0.0196241   4.613 3.97e-06 ***
## MonthlyCharge     0.0167706  0.0046417   3.613 0.000303 ***
## RoamMins          0.0549779  0.0165200   3.328 0.000875 ***
## DayCalls          0.0038420  0.0022621   1.698 0.089431 .
## AccountWeeks     -0.0017205  0.0011306  -1.522 0.128091
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2940.3  on 2120  degrees of freedom
## Residual deviance: 2315.8  on 2111  degrees of freedom
## AIC: 2335.8
##
## Number of Fisher Scoring iterations: 4
```

5.2.7 Checking for Multicollinearity

```
vif(lregmodel)
```

```
## ContractRenewal   CustServCalls         DayMins        DataPlan
##        1.105116        1.155222        1.777375        2.125319
##      OverageFee   MonthlyCharge        RoamMins        DayCalls
##        1.146748        2.959555        1.050082        1.008879
##    AccountWeeks
##        1.012100
```

Findings

- Values below 4 indicates that there is no Multicollinearity

5.2.8 Predicting for Train Data

```
logpred<- predict(lregmodel,BalancedData[-1],type = "response")
```

5.2.9 Probabilities

```
logpred
```

```
##         1         2         3         4         5         6
```

```
## 0.34584066 0.19929893 0.05871543 0.24279324 0.42689615 0.08637136
##          7          8          9         10         11         12
## 0.36184384 0.46336282 0.35992386 0.17423454 0.15590466 0.16936702
##         13         14         15         16         17         18
## 0.19613675 0.27503938 0.37828311 0.21942511 0.45267783 0.53286568
##         19         20         21         22         23         24
## 0.29498312 0.10203806 0.45848804 0.44216439 0.25036871 0.45866522
##         25         26         27         28         29         30
## 0.20152063 0.36478359 0.40224855 0.47804593 0.79313361 0.49090121
##         31         32         33         34         35         36
## 0.62100035 0.50584914 0.43813658 0.59947232 0.34346128 0.07400533
##         37         38         39         40         41         42
## 0.53125875 0.41332439 0.34348894 0.17318168 0.24580521 0.17620273
##         43         44         45         46         47         48
## 0.24198814 0.24424332 0.39056003 0.30875867 0.16719006 0.28579054
##         49         50         51         52         53         54
## 0.31124749 0.20923260 0.33965235 0.21683885 0.08490931 0.14589154
##         55         56         57         58         59         60
## 0.36440439 0.05009680 0.49542592 0.61039104 0.46311643 0.37831005
##         61         62         63         64         65         66
## 0.68890712 0.12844203 0.44795196 0.10221381 0.73580517 0.23183734
##         67         68         69         70         71         72
## 0.24827720 0.27487511 0.41421734 0.28658899 0.10968131 0.16900834
##         73         74         75         76         77         78
## 0.45467369 0.69312689 0.28030920 0.14508598 0.73434064 0.13838513
##         79         80         81         82         83         84
## 0.95373736 0.18942396 0.80110242 0.09165667 0.14669483 0.31974373
##         85         86         87         88         89         90
## 0.43142902 0.43068704 0.28602604 0.08876965 0.31474910 0.08338423
##         91         92         93         94         95         96
## 0.43466008 0.27082715 0.39563372 0.29699269 0.25099837 0.07679091
##         97         98         99        100        101        102
## 0.19452422 0.14048134 0.27058691 0.27042154 0.78590801 0.43965101
##        103        104        105        106        107        108
## 0.53761269 0.67635019 0.21628967 0.29400412 0.56659343 0.19677108
##        109        110        111        112        113        114
## 0.42501870 0.11144975 0.19336020 0.20223066 0.68140269 0.73593466
##        115        116        117        118        119        120
## 0.09239227 0.25132783 0.52597808 0.25795550 0.65821154 0.22459178
##        121        122        123        124        125        126
## 0.11465833 0.32199726 0.56818205 0.21646461 0.37960551 0.15445344
##        127        128        129        130        131        132
## 0.13349176 0.29185540 0.35614012 0.24209241 0.04826275 0.27135591
##        133        134        135        136        137        138
## 0.82087548 0.62110556 0.30355658 0.26128304 0.26756064 0.21972234
##        139        140        141        142        143        144
## 0.25011436 0.18164232 0.75014766 0.91856604 0.77774201 0.22153308
##        145        146        147        148        149        150
## 0.76654872 0.22417625 0.36808600 0.16144712 0.21087695 0.20759384
##        151        152        153        154        155        156
## 0.14002723 0.45804053 0.65869359 0.15221613 0.70083266 0.22300062
##        157        158        159        160        161        162
## 0.38289645 0.24794880 0.37257335 0.77978453 0.26708720 0.35264122
##        163        164        165        166        167        168
```

```
## 0.19501623 0.03147396 0.29860365 0.09145077 0.71434888 0.54102295
##        169        170        171        172        173        174
## 0.12749066 0.29008874 0.18126563 0.60314345 0.34385226 0.19427470
##        175        176        177        178        179        180
## 0.25763646 0.11367157 0.55589352 0.16876090 0.68932729 0.33185082
##        181        182        183        184        185        186
## 0.57590680 0.05178085 0.20199528 0.54890352 0.55218308 0.19555649
##        187        188        189        190        191        192
## 0.27912561 0.82178781 0.14010903 0.38008997 0.54439352 0.77130672
##        193        194        195        196        197        198
## 0.43902509 0.05026143 0.21827082 0.32870849 0.06167798 0.24562087
##        199        200        201        202        203        204
## 0.39094137 0.97278854 0.34364146 0.31084758 0.45518121 0.11245972
##        205        206        207        208        209        210
## 0.61829524 0.30216885 0.79333448 0.29540366 0.09418310 0.34917558
##        211        212        213        214        215        216
## 0.17384114 0.25908142 0.21002256 0.14755671 0.32296766 0.27979556
##        217        218        219        220        221        222
## 0.48500107 0.22393729 0.61590590 0.19395490 0.82193697 0.55199579
##        223        224        225        226        227        228
## 0.39544801 0.93404579 0.53784371 0.11703283 0.46825815 0.21675191
##        229        230        231        232        233        234
## 0.16175047 0.28408279 0.42703993 0.23387361 0.54205006 0.07143750
##        235        236        237        238        239        240
## 0.48296990 0.24594903 0.17426007 0.56420054 0.26539557 0.27249303
##        241        242        243        244        245        246
## 0.15433818 0.18666783 0.46292860 0.21858907 0.15337231 0.23834509
##        247        248        249        250        251        252
## 0.52709350 0.62147197 0.38833533 0.20344177 0.50154886 0.16178468
##        253        254        255        256        257        258
## 0.43856408 0.39210181 0.24220892 0.08890848 0.25598232 0.46056019
##        259        260        261        262        263        264
## 0.36653701 0.87720488 0.33609825 0.30049054 0.23878720 0.45920898
##        265        266        267        268        269        270
## 0.64201649 0.28525184 0.11513109 0.24413295 0.71487122 0.17592230
##        271        272        273        274        275        276
## 0.37087567 0.33040866 0.23266609 0.77513729 0.22047105 0.55083937
##        277        278        279        280        281        282
## 0.84298048 0.29612890 0.29198178 0.42743991 0.40329909 0.51070958
##        283        284        285        286        287        288
## 0.35713927 0.42030841 0.46202703 0.43249581 0.30465611 0.24511915
##        289        290        291        292        293        294
## 0.19336120 0.21728838 0.32000437 0.28003467 0.57141524 0.39766359
##        295        296        297        298        299        300
## 0.58015373 0.81787996 0.49247617 0.50070975 0.45433231 0.21963799
##        301        302        303        304        305        306
## 0.27846523 0.30445713 0.22901870 0.29744626 0.62965529 0.29664393
##        307        308        309        310        311        312
## 0.34356581 0.51919137 0.12733202 0.53734599 0.27148086 0.24856984
##        313        314        315        316        317        318
## 0.28855603 0.34709918 0.21869549 0.19241859 0.48100463 0.53807754
##        319        320        321        322        323        324
## 0.53844002 0.37662228 0.44958085 0.77096935 0.23787540 0.42561294
##        325        326        327        328        329        330
```

```
## 0.46513891 0.34623008 0.26661116 0.39898448 0.51031721 0.59334057
##        331        332        333        334        335        336
## 0.50874982 0.47368609 0.33034824 0.41630234 0.39028293 0.20044680
##        337        338        339        340        341        342
## 0.04903479 0.32234887 0.29838277 0.31961006 0.50193921 0.17193159
##        343        344        345        346        347        348
## 0.11906910 0.04763083 0.37542391 0.18415340 0.90183737 0.31939341
##        349        350        351        352        353        354
## 0.44652706 0.52734867 0.44980190 0.19084769 0.27980858 0.19845090
##        355        356        357        358        359        360
## 0.88014145 0.16839264 0.52118378 0.33771813 0.73949250 0.16776376
##        361        362        363        364        365        366
## 0.24929271 0.07221404 0.28242691 0.11353183 0.31143616 0.54478081
##        367        368        369        370        371        372
## 0.66601684 0.45855883 0.77781981 0.42495911 0.34754233 0.37469605
##        373        374        375        376        377        378
## 0.15242819 0.66531872 0.32837905 0.22743591 0.30025211 0.45766673
##        379        380        381        382        383        384
## 0.26329224 0.47456593 0.10074962 0.56100155 0.61357211 0.10770252
##        385        386        387        388        389        390
## 0.09388195 0.72925358 0.18902388 0.34505316 0.29503798 0.23109144
##        391        392        393        394        395        396
## 0.29239949 0.64002643 0.30291066 0.20669054 0.37541306 0.28871878
##        397        398        399        400        401        402
## 0.11326691 0.49916891 0.17031941 0.57070504 0.36861106 0.36126549
##        403        404        405        406        407        408
## 0.28032371 0.35177895 0.19431272 0.20779514 0.45190662 0.17613644
##        409        410        411        412        413        414
## 0.42632983 0.59927880 0.08391001 0.07416958 0.07562769 0.58095842
##        415        416        417        418        419        420
## 0.41290591 0.37094630 0.69377142 0.93213240 0.16649906 0.28668888
##        421        422        423        424        425        426
## 0.50173349 0.25178102 0.28660802 0.73275841 0.59182592 0.30299504
##        427        428        429        430        431        432
## 0.86122808 0.28750907 0.48334806 0.27815951 0.57033667 0.48679372
##        433        434        435        436        437        438
## 0.83983236 0.47324645 0.20752720 0.22474201 0.90428302 0.29878166
##        439        440        441        442        443        444
## 0.33841118 0.35464431 0.89555574 0.22657489 0.39450973 0.32220958
##        445        446        447        448        449        450
## 0.40249692 0.54019412 0.52172723 0.24900996 0.09817625 0.16513447
##        451        452        453        454        455        456
## 0.14722543 0.05872109 0.19345062 0.16525182 0.21552530 0.54925918
##        457        458        459        460        461        462
## 0.36109115 0.17811739 0.47287236 0.67600070 0.20634962 0.26638782
##        463        464        465        466        467        468
## 0.03246218 0.34166304 0.15430194 0.20663788 0.22483713 0.22699018
##        469        470        471        472        473        474
## 0.27809411 0.52291721 0.50444561 0.64961752 0.44403886 0.58802731
##        475        476        477        478        479        480
## 0.41759616 0.46005742 0.19835607 0.89848734 0.33276889 0.29058748
##        481        482        483        484        485        486
## 0.19081635 0.30009312 0.37006322 0.47898758 0.09201452 0.15538646
##        487        488        489        490        491        492
```

```
## 0.76463941 0.38392079 0.44406820 0.72275291 0.02026988 0.08239856
##        493        494        495        496        497        498
## 0.30985483 0.28307009 0.30123773 0.40047276 0.24765509 0.15334375
##        499        500        501        502        503        504
## 0.51029249 0.46050077 0.53050675 0.05775646 0.03219159 0.19570145
##        505        506        507        508        509        510
## 0.19999592 0.63785306 0.47281005 0.13878101 0.30998782 0.50164659
##        511        512        513        514        515        516
## 0.46210351 0.27169764 0.59823779 0.35303308 0.09850590 0.70692385
##        517        518        519        520        521        522
## 0.28359646 0.73033577 0.17008262 0.67269306 0.40299119 0.12711346
##        523        524        525        526        527        528
## 0.30457054 0.47962926 0.66192454 0.24326909 0.36343948 0.55795142
##        529        530        531        532        533        534
## 0.20377843 0.21726763 0.53407019 0.18736736 0.64884936 0.29308046
##        535        536        537        538        539        540
## 0.59391772 0.18075011 0.58092772 0.64688714 0.33433188 0.85954970
##        541        542        543        544        545        546
## 0.09363944 0.34722609 0.11205964 0.28929334 0.15648819 0.23798434
##        547        548        549        550        551        552
## 0.45404159 0.47137325 0.04675290 0.23293045 0.18477613 0.44716726
##        553        554        555        556        557        558
## 0.14829158 0.66690677 0.24201741 0.43687714 0.29849888 0.64631130
##        559        560        561        562        563        564
## 0.33297198 0.19864932 0.61665025 0.56616694 0.28923472 0.09732860
##        565        566        567        568        569        570
## 0.43538017 0.03310017 0.26102351 0.37915440 0.35212464 0.38289613
##        571        572        573        574        575        576
## 0.23023056 0.51617223 0.36249228 0.19257462 0.18336268 0.36961801
##        577        578        579        580        581        582
## 0.49765432 0.33341275 0.38356051 0.16415971 0.52696695 0.13663246
##        583        584        585        586        587        588
## 0.31073386 0.13119512 0.18314563 0.24940659 0.20763917 0.21299109
##        589        590        591        592        593        594
## 0.18488536 0.35166247 0.71576117 0.14683666 0.19636194 0.10018767
##        595        596        597        598        599        600
## 0.57609009 0.36487332 0.43255623 0.21467366 0.41629196 0.43243588
##        601        602        603        604        605        606
## 0.36779214 0.38062198 0.42857574 0.65194633 0.30297210 0.20670544
##        607        608        609        610        611        612
## 0.63641679 0.74226567 0.29172648 0.37267508 0.36238544 0.74472364
##        613        614        615        616        617        618
## 0.34975846 0.73474516 0.42481437 0.17212956 0.24564069 0.44090206
##        619        620        621        622        623        624
## 0.19511743 0.05652070 0.46496559 0.09716544 0.25268440 0.30521113
##        625        626        627        628        629        630
## 0.41302473 0.40782873 0.48318669 0.41832596 0.08867477 0.31636769
##        631        632        633        634        635        636
## 0.48357600 0.51040824 0.28583947 0.32308992 0.47970452 0.30484741
##        637        638        639        640        641        642
## 0.47283463 0.22372289 0.45411606 0.41799770 0.61816447 0.31890875
##        643        644        645        646        647        648
## 0.35600987 0.57271632 0.51283474 0.49187259 0.25541094 0.16686045
##        649        650        651        652        653        654
```

```
## 0.21819859 0.39347422 0.17932623 0.70611044 0.45834087 0.51195018
##        655        656        657        658        659        660
## 0.25921349 0.74841169 0.32725088 0.39231546 0.21626637 0.23133315
##        661        662        663        664        665        666
## 0.37737166 0.30501503 0.71554825 0.88740905 0.92536428 0.31608937
##        667        668        669        670        671        672
## 0.80626259 0.35551452 0.23271174 0.21341428 0.33922134 0.32595899
##        673        674        675        676        677        678
## 0.22222714 0.36281871 0.24817699 0.22142976 0.25675467 0.16200025
##        679        680        681        682        683        684
## 0.11758738 0.66779841 0.74064743 0.32241571 0.16574790 0.90041778
##        685        686        687        688        689        690
## 0.10374611 0.51641342 0.59901133 0.15542955 0.07096092 0.50946621
##        691        692        693        694        695        696
## 0.60843479 0.13444025 0.69084234 0.17104351 0.08507809 0.96396333
##        697        698        699        700        701        702
## 0.24459756 0.25254638 0.74763922 0.72137828 0.58155095 0.33028515
##        703        704        705        706        707        708
## 0.54924923 0.14687552 0.19056997 0.13526343 0.28755461 0.24425528
##        709        710        711        712        713        714
## 0.08661234 0.14786192 0.39471318 0.46768524 0.08682624 0.14870271
##        715        716        717        718        719        720
## 0.52866743 0.22304274 0.31270565 0.19436564 0.49833026 0.44719156
##        721        722        723        724        725        726
## 0.63021733 0.28129826 0.18481710 0.18772378 0.11209265 0.37576525
##        727        728        729        730        731        732
## 0.10105833 0.14831683 0.20325049 0.09580351 0.37529356 0.52156964
##        733        734        735        736        737        738
## 0.64565424 0.89286458 0.15347268 0.14081545 0.29538871 0.49247319
##        739        740        741        742        743        744
## 0.17939799 0.16787541 0.79468335 0.42959274 0.41627859 0.41511599
##        745        746        747        748        749        750
## 0.43734411 0.23542468 0.64733400 0.11611937 0.64496403 0.08511272
##        751        752        753        754        755        756
## 0.18802329 0.80036739 0.27949334 0.92767460 0.11548658 0.80725988
##        757        758        759        760        761        762
## 0.13111764 0.16747376 0.81862468 0.02994937 0.54086961 0.36966345
##        763        764        765        766        767        768
## 0.84628633 0.64655148 0.23355683 0.14324576 0.73741535 0.40590763
##        769        770        771        772        773        774
## 0.30057084 0.28350602 0.72295326 0.24458696 0.36548180 0.54630220
##        775        776        777        778        779        780
## 0.55794923 0.20868348 0.50790268 0.33301151 0.45828201 0.34237745
##        781        782        783        784        785        786
## 0.34564874 0.48099018 0.43486372 0.24716317 0.35079866 0.66076193
##        787        788        789        790        791        792
## 0.64619822 0.38211143 0.37863392 0.25785833 0.15557398 0.57323716
##        793        794        795        796        797        798
## 0.10851721 0.48198048 0.24011059 0.21842380 0.70916102 0.56802617
##        799        800        801        802        803        804
## 0.55414684 0.18693136 0.10837618 0.26948894 0.21196136 0.08559393
##        805        806        807        808        809        810
## 0.45710847 0.43646910 0.43504994 0.82500443 0.51676872 0.19309128
##        811        812        813        814        815        816
```

```
## 0.70123042 0.22829114 0.40666017 0.43107283 0.03324461 0.42477650
##        817        818        819        820        821        822
## 0.64610867 0.30428141 0.35278039 0.64806478 0.15064084 0.46927853
##        823        824        825        826        827        828
## 0.38654814 0.65887098 0.53001098 0.36672765 0.43519643 0.20397842
##        829        830        831        832        833        834
## 0.24673109 0.23082469 0.24906334 0.15866812 0.14995572 0.39864153
##        835        836        837        838        839        840
## 0.24772971 0.28173802 0.51489079 0.15838075 0.76594256 0.61087126
##        841        842        843        844        845        846
## 0.51771233 0.64586016 0.23216593 0.26694887 0.25130233 0.23792526
##        847        848        849        850        851        852
## 0.15377402 0.55831076 0.31173075 0.49651709 0.20362833 0.14617181
##        853        854        855        856        857        858
## 0.47628801 0.55343374 0.25268063 0.26623438 0.31734122 0.22511433
##        859        860        861        862        863        864
## 0.19123506 0.51282027 0.23344840 0.44556463 0.81996417 0.21676970
##        865        866        867        868        869        870
## 0.64323095 0.24040931 0.28628228 0.24287836 0.12951499 0.21779395
##        871        872        873        874        875        876
## 0.47730675 0.22808247 0.44339328 0.22319852 0.19410882 0.13910206
##        877        878        879        880        881        882
## 0.40232460 0.62306030 0.14966064 0.19786854 0.12105646 0.29764383
##        883        884        885        886        887        888
## 0.83424507 0.45973688 0.11605839 0.38464252 0.85109607 0.35297011
##        889        890        891        892        893        894
## 0.22239150 0.30762833 0.62324023 0.30333231 0.30109212 0.71886696
##        895        896        897        898        899        900
## 0.44414671 0.64268105 0.13379542 0.20093578 0.13724980 0.17567349
##        901        902        903        904        905        906
## 0.15517203 0.21744572 0.07878583 0.48496276 0.56440180 0.50169923
##        907        908        909        910        911        912
## 0.60161104 0.37229756 0.46218221 0.61757294 0.58798349 0.14600850
##        913        914        915        916        917        918
## 0.07166507 0.35719339 0.33794099 0.66712857 0.30912291 0.68347791
##        919        920        921        922        923        924
## 0.52371756 0.20693528 0.54270369 0.37020135 0.17242531 0.25596527
##        925        926        927        928        929        930
## 0.24625703 0.21185517 0.32676163 0.33967504 0.46206496 0.56660184
##        931        932        933        934        935        936
## 0.06947378 0.68466169 0.87645291 0.14247808 0.78995078 0.34734518
##        937        938        939        940        941        942
## 0.45900697 0.22765783 0.55629819 0.18965458 0.15274212 0.10350301
##        943        944        945        946        947        948
## 0.20324211 0.19312209 0.08940659 0.57287665 0.21721223 0.44401974
##        949        950        951        952        953        954
## 0.18711594 0.32214731 0.16462878 0.32790447 0.17518204 0.46365233
##        955        956        957        958        959        960
## 0.85197882 0.15992148 0.13139995 0.41659513 0.29491620 0.37002359
##        961        962        963        964        965        966
## 0.11548803 0.95142149 0.01481569 0.34325058 0.39328799 0.11171120
##        967        968        969        970        971        972
## 0.82994416 0.28974455 0.43078694 0.42865121 0.15454166 0.64518032
##        973        974        975        976        977        978
```

```
## 0.55694236 0.13382257 0.40668883 0.20316663 0.82831771 0.42083174
##        979        980        981        982        983        984
## 0.47576716 0.12556231 0.14866109 0.11125076 0.56653879 0.12535385
##        985        986        987        988        989        990
## 0.11352374 0.11724897 0.35181953 0.13436548 0.32498148 0.67795321
##        991        992        993        994        995        996
## 0.70953910 0.33924216 0.39134132 0.13652903 0.30821240 0.56384299
##        997        998        999       1000       1001       1002
## 0.43447362 0.55395433 0.17992096 0.94603000 0.25506712 0.43920888
##       1003       1004       1005       1006       1007       1008
## 0.52007317 0.43434769 0.44799938 0.51755514 0.43458523 0.19870775
##       1009       1010       1011       1012       1013       1014
## 0.59037209 0.39445498 0.16614249 0.18796127 0.64418936 0.07947131
##       1015       1016       1017       1018       1019       1020
## 0.21207680 0.54313061 0.49235524 0.09484359 0.22856256 0.95019288
##       1021       1022       1023       1024       1025       1026
## 0.17523237 0.51890287 0.36984288 0.17230941 0.34051191 0.28272115
##       1027       1028       1029       1030       1031       1032
## 0.19572939 0.09451496 0.09684715 0.42545217 0.14718406 0.52959191
##       1033       1034       1035       1036       1037       1038
## 0.17923403 0.23254593 0.23531173 0.82110792 0.07872139 0.34464525
##       1039       1040       1041       1042       1043       1044
## 0.49384185 0.61640245 0.10635898 0.44082935 0.05044867 0.83772805
##       1045       1046       1047       1048       1049       1050
## 0.85598867 0.65692415 0.79155826 0.81930649 0.13707331 0.87346514
##       1051       1052       1053       1054       1055       1056
## 0.32285172 0.24840973 0.17789422 0.44868590 0.85629681 0.55404493
##       1057       1058       1059       1060       1061       1062
## 0.26638500 0.11801231 0.05986661 0.44322460 0.09740258 0.08776564
##       1063       1064       1065       1066       1067       1068
## 0.24372879 0.79879020 0.90735761 0.78101838 0.67548273 0.19469561
##       1069       1070       1071       1072       1073       1074
## 0.63605039 0.52008232 0.19198692 0.44096944 0.59566613 0.18278445
##       1075       1076       1077       1078       1079       1080
## 0.85363838 0.31747618 0.87810250 0.77453296 0.82787623 0.89335049
##       1081       1082       1083       1084       1085       1086
## 0.68253276 0.48770570 0.91156668 0.92862392 0.80861149 0.93282363
##       1087       1088       1089       1090       1091       1092
## 0.74063207 0.58465648 0.14671688 0.75110168 0.51673207 0.51073354
##       1093       1094       1095       1096       1097       1098
## 0.93770233 0.21674230 0.64356455 0.56614689 0.97108034 0.86635407
##       1099       1100       1101       1102       1103       1104
## 0.94292510 0.59176061 0.63206749 0.50807293 0.78328588 0.89866981
##       1105       1106       1107       1108       1109       1110
## 0.40459465 0.88441469 0.95268297 0.58840360 0.12435205 0.95495338
##       1111       1112       1113       1114       1115       1116
## 0.46555046 0.58881523 0.37452312 0.15153556 0.88451949 0.53715038
##       1117       1118       1119       1120       1121       1122
## 0.30612616 0.71341989 0.62735620 0.83315124 0.61547116 0.48506901
##       1123       1124       1125       1126       1127       1128
## 0.64673007 0.29085738 0.29799973 0.50031374 0.76358978 0.95483468
##       1129       1130       1131       1132       1133       1134
## 0.08319340 0.58229499 0.53655067 0.36472949 0.12577039 0.45504620
##       1135       1136       1137       1138       1139       1140
```

```
## 0.97195274 0.75042123 0.14751586 0.30676941 0.94506701 0.47136196
##       1141       1142       1143       1144       1145       1146
## 0.74318848 0.23949375 0.95466258 0.73483359 0.90379685 0.81306477
##       1147       1148       1149       1150       1151       1152
## 0.74590490 0.93499789 0.43505138 0.59984600 0.63959152 0.58642762
##       1153       1154       1155       1156       1157       1158
## 0.65108950 0.37939685 0.91698002 0.90749731 0.86011258 0.25379104
##       1159       1160       1161       1162       1163       1164
## 0.91265740 0.68982809 0.66021266 0.71552734 0.62820312 0.82905940
##       1165       1166       1167       1168       1169       1170
## 0.91770576 0.87742287 0.90649914 0.95310319 0.47014682 0.18540461
##       1171       1172       1173       1174       1175       1176
## 0.76449041 0.18995383 0.78283825 0.33453956 0.59283584 0.66433189
##       1177       1178       1179       1180       1181       1182
## 0.38421203 0.40031823 0.20072351 0.89918022 0.66106939 0.56184633
##       1183       1184       1185       1186       1187       1188
## 0.79259889 0.14511403 0.57510226 0.63718605 0.79566735 0.54234771
##       1189       1190       1191       1192       1193       1194
## 0.29364310 0.84421497 0.83673895 0.69506896 0.80842096 0.25487434
##       1195       1196       1197       1198       1199       1200
## 0.23341102 0.89359519 0.68962544 0.57910930 0.77976157 0.52881039
##       1201       1202       1203       1204       1205       1206
## 0.66723667 0.55084724 0.93116529 0.96997013 0.58143330 0.88271263
##       1207       1208       1209       1210       1211       1212
## 0.65630082 0.70431054 0.41257057 0.70629683 0.50024186 0.57979616
##       1213       1214       1215       1216       1217       1218
## 0.62538692 0.58212796 0.37002718 0.17160565 0.70416216 0.41434901
##       1219       1220       1221       1222       1223       1224
## 0.24420089 0.70718595 0.50368782 0.71831515 0.29208989 0.55387296
##       1225       1226       1227       1228       1229       1230
## 0.60587061 0.71439248 0.50780061 0.77506108 0.80577614 0.73794657
##       1231       1232       1233       1234       1235       1236
## 0.13526719 0.64308006 0.51865927 0.32968406 0.96829110 0.59350277
##       1237       1238       1239       1240       1241       1242
## 0.69442665 0.69536562 0.46649746 0.45148255 0.64118507 0.37323189
##       1243       1244       1245       1246       1247       1248
## 0.91897976 0.41309186 0.49698362 0.98034992 0.92505155 0.72096407
##       1249       1250       1251       1252       1253       1254
## 0.37060251 0.70580203 0.74071943 0.72694866 0.65889187 0.56544540
##       1255       1256       1257       1258       1259       1260
## 0.97475735 0.26690673 0.08491678 0.96854773 0.35681838 0.88225140
##       1261       1262       1263       1264       1265       1266
## 0.80401035 0.48046661 0.83958126 0.96754299 0.98215853 0.33120125
##       1267       1268       1269       1270       1271       1272
## 0.93532352 0.71655975 0.38569160 0.90109830 0.60888096 0.86243277
##       1273       1274       1275       1276       1277       1278
## 0.17981125 0.71918593 0.87219978 0.93842223 0.86307409 0.87288033
##       1279       1280       1281       1282       1283       1284
## 0.64979440 0.37983811 0.63824412 0.47122518 0.38284314 0.69985511
##       1285       1286       1287       1288       1289       1290
## 0.37093253 0.51735771 0.80321926 0.63085120 0.67041328 0.30498062
##       1291       1292       1293       1294       1295       1296
## 0.94584161 0.95135043 0.52307853 0.91104599 0.11562801 0.83324367
##       1297       1298       1299       1300       1301       1302
```

```
## 0.95426728 0.79502536 0.35302761 0.54716510 0.76195846 0.66398728
##       1303       1304       1305       1306       1307       1308
## 0.69867268 0.79913608 0.93843561 0.43019609 0.87199932 0.78723675
##       1309       1310       1311       1312       1313       1314
## 0.69380334 0.84837942 0.85502255 0.73857626 0.18159730 0.62265522
##       1315       1316       1317       1318       1319       1320
## 0.29135044 0.92612827 0.90988100 0.48180938 0.76807328 0.50048274
##       1321       1322       1323       1324       1325       1326
## 0.19903403 0.24520456 0.66779331 0.41610156 0.49132434 0.78930632
##       1327       1328       1329       1330       1331       1332
## 0.88380083 0.71510646 0.62294672 0.71436685 0.66017061 0.72030056
##       1333       1334       1335       1336       1337       1338
## 0.84242614 0.56616066 0.12118702 0.81148975 0.54912355 0.80475713
##       1339       1340       1341       1342       1343       1344
## 0.64434498 0.39280941 0.86774903 0.75724899 0.68041035 0.74495874
##       1345       1346       1347       1348       1349       1350
## 0.93727507 0.44641987 0.98296295 0.74590399 0.71285144 0.46888857
##       1351       1352       1353       1354       1355       1356
## 0.30917189 0.66780708 0.72996658 0.96074752 0.89492399 0.67564944
##       1357       1358       1359       1360       1361       1362
## 0.84689881 0.66762752 0.62217779 0.31938606 0.20618554 0.22874538
##       1363       1364       1365       1366       1367       1368
## 0.54609591 0.58932498 0.74025399 0.70484739 0.72723723 0.93119491
##       1369       1370       1371       1372       1373       1374
## 0.45277516 0.72529052 0.19139688 0.80723378 0.70340589 0.78518951
##       1375       1376       1377       1378       1379       1380
## 0.86505212 0.86384249 0.37548672 0.81804507 0.15231979 0.84120731
##       1381       1382       1383       1384       1385       1386
## 0.27521364 0.63965013 0.92641178 0.40003911 0.75696398 0.89376078
##       1387       1388       1389       1390       1391       1392
## 0.58287361 0.89514236 0.98352479 0.38855111 0.45931147 0.07200530
##       1393       1394       1395       1396       1397       1398
## 0.57609194 0.95064715 0.61512040 0.54799176 0.58232065 0.89667274
##       1399       1400       1401       1402       1403       1404
## 0.73364785 0.68182909 0.14182653 0.52183615 0.91718960 0.43468897
##       1405       1406       1407       1408       1409       1410
## 0.95775753 0.77229422 0.65599517 0.97117969 0.93687298 0.77671408
##       1411       1412       1413       1414       1415       1416
## 0.83501142 0.79465000 0.97504354 0.31710556 0.17362954 0.62389253
##       1417       1418       1419       1420       1421       1422
## 0.56778791 0.69115271 0.88119231 0.75543314 0.22271307 0.15632449
##       1423       1424       1425       1426       1427       1428
## 0.58621011 0.94577983 0.49259359 0.27648848 0.43927273 0.77059143
##       1429       1430       1431       1432       1433       1434
## 0.49435061 0.31073930 0.53183167 0.78989101 0.71897335 0.62384341
##       1435       1436       1437       1438       1439       1440
## 0.83680106 0.05575673 0.36285551 0.24095523 0.75290195 0.85474865
##       1441       1442       1443       1444       1445       1446
## 0.64206863 0.71331712 0.46385444 0.45561028 0.15020224 0.77485224
##       1447       1448       1449       1450       1451       1452
## 0.39105353 0.71638384 0.76895603 0.65894342 0.22780451 0.51897576
##       1453       1454       1455       1456       1457       1458
## 0.52068806 0.46985252 0.86606992 0.64591211 0.85461119 0.53192345
##       1459       1460       1461       1462       1463       1464
```

```
## 0.49912787 0.95288133 0.19483351 0.47282409 0.36930556 0.72520014
##       1465       1466       1467       1468       1469       1470
## 0.32370692 0.46767935 0.38599762 0.41864472 0.60400854 0.90262023
##       1471       1472       1473       1474       1475       1476
## 0.62956693 0.16987253 0.82372777 0.53374969 0.55858701 0.97834177
##       1477       1478       1479       1480       1481       1482
## 0.84528761 0.51071100 0.95864280 0.85587117 0.53284949 0.64012280
##       1483       1484       1485       1486       1487       1488
## 0.86181466 0.90909095 0.96352880 0.71405262 0.78409247 0.53116658
##       1489       1490       1491       1492       1493       1494
## 0.34472363 0.81042694 0.67653248 0.87876608 0.47849871 0.47164732
##       1495       1496       1497       1498       1499       1500
## 0.37074609 0.88428650 0.56008987 0.45381378 0.44797671 0.46750473
##       1501       1502       1503       1504       1505       1506
## 0.79840581 0.52530256 0.96612178 0.63560011 0.52431765 0.48690417
##       1507       1508       1509       1510       1511       1512
## 0.38144830 0.60363754 0.89516794 0.69600756 0.90815345 0.57618675
##       1513       1514       1515       1516       1517       1518
## 0.65585022 0.86551570 0.40525927 0.70549746 0.63009728 0.89920234
##       1519       1520       1521       1522       1523       1524
## 0.50017607 0.53696175 0.45922806 0.40146602 0.78823796 0.74391168
##       1525       1526       1527       1528       1529       1530
## 0.70540414 0.88124758 0.65169214 0.94684552 0.85825820 0.54127261
##       1531       1532       1533       1534       1535       1536
## 0.59368259 0.30646940 0.45181892 0.77119987 0.63516217 0.37611743
##       1537       1538       1539       1540       1541       1542
## 0.74265600 0.88170380 0.55728467 0.42119401 0.24569365 0.74102018
##       1543       1544       1545       1546       1547       1548
## 0.55787187 0.31411250 0.93852108 0.69333568 0.97009120 0.60249602
##       1549       1550       1551       1552       1553       1554
## 0.67072835 0.96729970 0.57386011 0.71688717 0.49024591 0.49929989
##       1555       1556       1557       1558       1559       1560
## 0.54490874 0.61642608 0.64700272 0.96101978 0.34144232 0.67180790
##       1561       1562       1563       1564       1565       1566
## 0.75628111 0.74796697 0.25187668 0.65323862 0.65869126 0.35670623
##       1567       1568       1569       1570       1571       1572
## 0.62732793 0.41749793 0.96305880 0.76175352 0.19511882 0.83327964
##       1573       1574       1575       1576       1577       1578
## 0.71908408 0.81732686 0.63428671 0.86966801 0.91326473 0.52989831
##       1579       1580       1581       1582       1583       1584
## 0.84915022 0.92983725 0.38039968 0.42333959 0.55549579 0.87312220
##       1585       1586       1587       1588       1589       1590
## 0.79494367 0.36312098 0.48149148 0.89143405 0.24525395 0.52222168
##       1591       1592       1593       1594       1595       1596
## 0.31923026 0.73168362 0.44806304 0.46975862 0.75139586 0.67176286
##       1597       1598       1599       1600       1601       1602
## 0.45973570 0.78537667 0.99702231 0.65292379 0.43193361 0.83447784
##       1603       1604       1605       1606       1607       1608
## 0.70126536 0.55725462 0.59609737 0.76857328 0.27243802 0.78956604
##       1609       1610       1611       1612       1613       1614
## 0.39811949 0.43072404 0.58265605 0.92386269 0.83990436 0.92000510
##       1615       1616       1617       1618       1619       1620
## 0.50228284 0.56002258 0.95315074 0.73812852 0.23804117 0.36872849
##       1621       1622       1623       1624       1625       1626
```

```
## 0.84568590 0.90967828 0.26877077 0.85434688 0.95174171 0.55393616
##       1627       1628       1629       1630       1631       1632
## 0.74744680 0.79185113 0.77919529 0.12002065 0.91140386 0.94065482
##       1633       1634       1635       1636       1637       1638
## 0.59147742 0.85160369 0.25974475 0.56336658 0.83806545 0.65858277
##       1639       1640       1641       1642       1643       1644
## 0.78013615 0.92598058 0.46921505 0.60028526 0.23406159 0.96522693
##       1645       1646       1647       1648       1649       1650
## 0.50087034 0.87012298 0.95333780 0.33190392 0.89785332 0.41821397
##       1651       1652       1653       1654       1655       1656
## 0.32170738 0.48814494 0.83320235 0.89299438 0.88080642 0.41695220
##       1657       1658       1659       1660       1661       1662
## 0.34380927 0.71210971 0.69903514 0.88853119 0.11448264 0.81112082
##       1663       1664       1665       1666       1667       1668
## 0.92314353 0.86836096 0.13989286 0.32197170 0.44822694 0.75174280
##       1669       1670       1671       1672       1673       1674
## 0.74137169 0.81709128 0.83715617 0.86340458 0.43012772 0.41512524
##       1675       1676       1677       1678       1679       1680
## 0.46560906 0.46974971 0.96722427 0.73642964 0.32995288 0.45757360
##       1681       1682       1683       1684       1685       1686
## 0.36888447 0.64512696 0.50026412 0.45789435 0.69572629 0.82316277
##       1687       1688       1689       1690       1691       1692
## 0.96350853 0.92439693 0.96396354 0.95584645 0.45452283 0.45912747
##       1693       1694       1695       1696       1697       1698
## 0.38299252 0.92173062 0.47580056 0.48959417 0.71742576 0.80675410
##       1699       1700       1701       1702       1703       1704
## 0.78730238 0.38775341 0.26056734 0.83515482 0.92829263 0.80148639
##       1705       1706       1707       1708       1709       1710
## 0.63844507 0.82453884 0.45019640 0.76426720 0.42944984 0.74521603
##       1711       1712       1713       1714       1715       1716
## 0.90898348 0.82149583 0.63894735 0.62104130 0.84986457 0.61902631
##       1717       1718       1719       1720       1721       1722
## 0.41991411 0.94704688 0.87839532 0.49981607 0.96816742 0.64362167
##       1723       1724       1725       1726       1727       1728
## 0.77332250 0.71838914 0.72265882 0.98678087 0.79196987 0.65234506
##       1729       1730       1731       1732       1733       1734
## 0.49689952 0.96161890 0.55548794 0.63885431 0.41010106 0.64402014
##       1735       1736       1737       1738       1739       1740
## 0.54149412 0.15024621 0.29791585 0.70620054 0.95697127 0.84381266
##       1741       1742       1743       1744       1745       1746
## 0.25383323 0.95622999 0.91278259 0.36640686 0.92835347 0.49132010
##       1747       1748       1749       1750       1751       1752
## 0.18688687 0.23808434 0.62112168 0.32746559 0.51727099 0.20307795
##       1753       1754       1755       1756       1757       1758
## 0.75952096 0.02269038 0.65067704 0.88934408 0.30274743 0.79789817
##       1759       1760       1761       1762       1763       1764
## 0.17779883 0.49978137 0.42032306 0.70016505 0.92030987 0.40972840
##       1765       1766       1767       1768       1769       1770
## 0.72048307 0.91574418 0.90146115 0.38730674 0.92949411 0.81087360
##       1771       1772       1773       1774       1775       1776
## 0.34011497 0.70881937 0.40973246 0.86292765 0.90777122 0.15797114
##       1777       1778       1779       1780       1781       1782
## 0.49521420 0.35604199 0.96778872 0.60106928 0.59904456 0.60010276
##       1783       1784       1785       1786       1787       1788
```

```
## 0.64997923 0.65849773 0.91375795 0.91080665 0.27842510 0.65557266
##       1789       1790       1791       1792       1793       1794
## 0.98818589 0.45662678 0.33005043 0.70419632 0.86705000 0.66122259
##       1795       1796       1797       1798       1799       1800
## 0.76903000 0.89579269 0.21573015 0.56344912 0.75022968 0.58468860
##       1801       1802       1803       1804       1805       1806
## 0.81683955 0.63568914 0.61222886 0.63572082 0.69262353 0.36284751
##       1807       1808       1809       1810       1811       1812
## 0.88919333 0.48703279 0.74485644 0.46038921 0.82625663 0.23835382
##       1813       1814       1815       1816       1817       1818
## 0.65995267 0.73009807 0.23295473 0.68225797 0.79502116 0.88637929
##       1819       1820       1821       1822       1823       1824
## 0.37592848 0.90645652 0.85516410 0.82580341 0.76547007 0.64243580
##       1825       1826       1827       1828       1829       1830
## 0.81387047 0.74416539 0.32022166 0.66324973 0.49133186 0.39254702
##       1831       1832       1833       1834       1835       1836
## 0.81917644 0.84447159 0.26432238 0.33647017 0.20077637 0.92004376
##       1837       1838       1839       1840       1841       1842
## 0.30719240 0.92986253 0.55248020 0.73760586 0.51267320 0.48537756
##       1843       1844       1845       1846       1847       1848
## 0.26322753 0.55092222 0.73964723 0.44661647 0.39796337 0.47514531
##       1849       1850       1851       1852       1853       1854
## 0.57580388 0.43312879 0.61482642 0.61497297 0.68922498 0.31534168
##       1855       1856       1857       1858       1859       1860
## 0.54121482 0.62598013 0.87842697 0.63391190 0.78615461 0.43093844
##       1861       1862       1863       1864       1865       1866
## 0.80103002 0.87922879 0.95344683 0.40174039 0.80472382 0.76554963
##       1867       1868       1869       1870       1871       1872
## 0.57259880 0.92577807 0.88117509 0.55050163 0.61890595 0.71293413
##       1873       1874       1875       1876       1877       1878
## 0.58383993 0.71687555 0.58001956 0.97216018 0.31249998 0.31828995
##       1879       1880       1881       1882       1883       1884
## 0.67174542 0.85151487 0.50073433 0.15408353 0.84604691 0.57503356
##       1885       1886       1887       1888       1889       1890
## 0.24872791 0.55433032 0.58129965 0.73563906 0.12507083 0.14990262
##       1891       1892       1893       1894       1895       1896
## 0.64471236 0.95429118 0.91560112 0.93023751 0.55023207 0.96905885
##       1897       1898       1899       1900       1901       1902
## 0.16720913 0.76510433 0.53827771 0.66206225 0.67196398 0.95394465
##       1903       1904       1905       1906       1907       1908
## 0.31019183 0.84643488 0.59143510 0.90408222 0.68778430 0.30173429
##       1909       1910       1911       1912       1913       1914
## 0.79376506 0.89695766 0.88843801 0.63071432 0.69281982 0.46714573
##       1915       1916       1917       1918       1919       1920
## 0.88451946 0.69503540 0.41562079 0.48106657 0.94082871 0.73067786
##       1921       1922       1923       1924       1925       1926
## 0.49164672 0.49349435 0.95136180 0.38028195 0.63765683 0.66625482
##       1927       1928       1929       1930       1931       1932
## 0.26020089 0.51408286 0.33989021 0.48563324 0.51963881 0.79700060
##       1933       1934       1935       1936       1937       1938
## 0.73214186 0.73712219 0.33931787 0.61735371 0.91834356 0.12228080
##       1939       1940       1941       1942       1943       1944
## 0.63905528 0.33878116 0.70111076 0.77027679 0.78979717 0.41618678
##       1945       1946       1947       1948       1949       1950
```

```
## 0.70958417 0.67476352 0.26789067 0.13150066 0.93900807 0.67084171
##       1951       1952       1953       1954       1955       1956
## 0.13867575 0.94447746 0.70734067 0.72474156 0.79494707 0.53435233
##       1957       1958       1959       1960       1961       1962
## 0.81016558 0.86622307 0.61919754 0.83240303 0.64699876 0.65005600
##       1963       1964       1965       1966       1967       1968
## 0.69251430 0.63652141 0.73188653 0.95589402 0.71049798 0.85435745
##       1969       1970       1971       1972       1973       1974
## 0.49007349 0.37337773 0.57008338 0.75101483 0.54211171 0.52744529
##       1975       1976       1977       1978       1979       1980
## 0.48847258 0.63177177 0.92914921 0.75257020 0.15528047 0.92351130
##       1981       1982       1983       1984       1985       1986
## 0.50611309 0.92434862 0.53698427 0.66577616 0.67856590 0.95959105
##       1987       1988       1989       1990       1991       1992
## 0.87226501 0.82350504 0.44713441 0.56545646 0.35114337 0.37990932
##       1993       1994       1995       1996       1997       1998
## 0.86676482 0.82309855 0.70294987 0.76460778 0.76743242 0.41478120
##       1999       2000       2001       2002       2003       2004
## 0.82306551 0.86769920 0.87569438 0.95948271 0.77609196 0.96804465
##       2005       2006       2007       2008       2009       2010
## 0.45872076 0.29223317 0.78530329 0.71090012 0.40747672 0.59688588
##       2011       2012       2013       2014       2015       2016
## 0.98511271 0.65308326 0.70341210 0.48133928 0.25926080 0.95474281
##       2017       2018       2019       2020       2021       2022
## 0.67370066 0.70487006 0.57590747 0.79489137 0.44801441 0.71695130
##       2023       2024       2025       2026       2027       2028
## 0.61257832 0.78596642 0.40268541 0.78086040 0.80137865 0.65955207
##       2029       2030       2031       2032       2033       2034
## 0.85292417 0.69810640 0.64650226 0.61899777 0.61446798 0.92967833
##       2035       2036       2037       2038       2039       2040
## 0.37406193 0.38375491 0.57445351 0.85577267 0.93813827 0.66942534
##       2041       2042       2043       2044       2045       2046
## 0.75288707 0.78369968 0.16995037 0.81135600 0.75604272 0.62001925
##       2047       2048       2049       2050       2051       2052
## 0.96962551 0.34461341 0.44802556 0.85643521 0.56836365 0.27147031
##       2053       2054       2055       2056       2057       2058
## 0.58887776 0.98765678 0.37219134 0.65790322 0.55036261 0.65769635
##       2059       2060       2061       2062       2063       2064
## 0.86733201 0.53286135 0.60352762 0.36957402 0.91938430 0.45896666
##       2065       2066       2067       2068       2069       2070
## 0.95074899 0.80826542 0.40247190 0.29650393 0.47331786 0.66165167
##       2071       2072       2073       2074       2075       2076
## 0.90585834 0.80010879 0.70089487 0.96227804 0.63658203 0.83106654
##       2077       2078       2079       2080       2081       2082
## 0.56362285 0.53964653 0.54896697 0.47339930 0.95384206 0.54860025
##       2083       2084       2085       2086       2087       2088
## 0.15844043 0.23418441 0.70741735 0.94918450 0.92585945 0.60191378
##       2089       2090       2091       2092       2093       2094
## 0.69678318 0.76763533 0.97752328 0.92157889 0.10865254 0.96430500
##       2095       2096       2097       2098       2099       2100
## 0.70172001 0.32111389 0.39665399 0.97814822 0.91221833 0.97077856
##       2101       2102       2103       2104       2105       2106
## 0.13465015 0.31019297 0.74903665 0.72671279 0.57967003 0.73403772
##       2107       2108       2109       2110       2111       2112
```

```
## 0.58582887 0.71015643 0.36319724 0.50856624 0.39177683 0.65871816
##       2113       2114       2115       2116       2117       2118
## 0.78893024 0.54472068 0.76340527 0.89195661 0.53278499 0.66537012
##       2119       2120       2121
## 0.82538750 0.59534387 0.23004189
```

5.2.10 Coverting Probabilites to 0 and 1

```
ypredlog <- ifelse(logpred > 0.5,1,0)
ypredlog
```

```
##   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
##   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30
##   0   0   1   0   0   0   0   0   0   0   0   0   0   1   0
##  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45
##   1   1   0   1   0   0   1   0   0   0   0   0   0   0   0
##  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60
##   0   0   0   0   0   0   0   0   0   0   0   0   1   0   0
##  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75
##   1   0   0   0   1   0   0   0   0   0   0   0   0   1   0
##  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90
##   0   1   0   1   0   1   0   0   0   0   0   0   0   0   0
##  91  92  93  94  95  96  97  98  99 100 101 102 103 104 105
##   0   0   0   0   0   0   0   0   0   0   1   0   1   1   0
## 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120
##   0   1   0   0   0   0   0   1   1   0   0   1   0   1   0
## 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135
##   0   0   1   0   0   0   0   0   0   0   0   0   1   1   0
## 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150
##   0   0   0   0   0   1   1   1   0   1   0   0   0   0   0
## 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165
##   0   0   1   0   1   0   0   0   0   1   0   0   0   0   0
## 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180
##   0   1   1   0   0   0   1   0   0   0   0   1   0   1   0
## 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195
##   1   0   0   1   1   0   0   1   0   0   1   1   0   0   0
## 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210
##   0   0   0   0   1   0   0   0   0   1   0   1   0   0   0
## 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225
##   0   0   0   0   0   0   0   0   1   0   1   1   0   1   1
## 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240
##   0   0   0   0   0   0   0   1   0   0   0   0   1   0   0
## 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255
##   0   0   0   0   0   0   1   1   0   0   1   0   0   0   0
## 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270
##   0   0   0   0   1   0   0   0   0   1   0   0   0   1   0
## 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285
##   0   0   0   1   0   1   1   0   0   0   0   1   0   0   0
## 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300
##   0   0   0   0   0   0   0   1   0   1   1   0   1   0   0
## 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315
##   0   0   0   0   1   0   0   1   0   1   0   0   0   0   0
## 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330
```

```
##   0   0   1   1   0   0   1   0   0   0   0   0   0   1   1
##  331 332 333 334 335 336 337 338 339 340 341 342 343 344 345
##   1   0   0   0   0   0   0   0   0   0   1   0   0   0   0
##  346 347 348 349 350 351 352 353 354 355 356 357 358 359 360
##   0   1   0   0   1   0   0   0   0   1   0   1   0   1   0
##  361 362 363 364 365 366 367 368 369 370 371 372 373 374 375
##   0   0   0   0   0   1   1   0   1   0   0   0   0   1   0
##  376 377 378 379 380 381 382 383 384 385 386 387 388 389 390
##   0   0   0   0   0   0   1   1   0   0   1   0   0   0   0
##  391 392 393 394 395 396 397 398 399 400 401 402 403 404 405
##   0   1   0   0   0   0   0   0   0   1   0   0   0   0   0
##  406 407 408 409 410 411 412 413 414 415 416 417 418 419 420
##   0   0   0   0   1   0   0   0   1   0   0   1   1   0   0
##  421 422 423 424 425 426 427 428 429 430 431 432 433 434 435
##   1   0   0   1   1   0   1   0   0   0   1   0   1   0   0
##  436 437 438 439 440 441 442 443 444 445 446 447 448 449 450
##   0   1   0   0   0   1   0   0   0   0   1   1   0   0   0
##  451 452 453 454 455 456 457 458 459 460 461 462 463 464 465
##   0   0   0   0   0   1   0   0   0   1   0   0   0   0   0
##  466 467 468 469 470 471 472 473 474 475 476 477 478 479 480
##   0   0   0   0   1   1   1   0   1   0   0   0   1   0   0
##  481 482 483 484 485 486 487 488 489 490 491 492 493 494 495
##   0   0   0   0   0   0   1   0   0   1   0   0   0   0   0
##  496 497 498 499 500 501 502 503 504 505 506 507 508 509 510
##   0   0   0   1   0   1   0   0   0   0   1   0   0   0   1
##  511 512 513 514 515 516 517 518 519 520 521 522 523 524 525
##   0   0   1   0   0   1   0   1   0   1   0   0   0   0   1
##  526 527 528 529 530 531 532 533 534 535 536 537 538 539 540
##   0   0   1   0   0   1   0   1   0   1   0   1   1   0   1
##  541 542 543 544 545 546 547 548 549 550 551 552 553 554 555
##   0   0   0   0   0   0   0   0   0   0   0   0   0   1   0
##  556 557 558 559 560 561 562 563 564 565 566 567 568 569 570
##   0   0   1   0   0   1   1   0   0   0   0   0   0   0   0
##  571 572 573 574 575 576 577 578 579 580 581 582 583 584 585
##   0   1   0   0   0   0   0   0   0   0   1   0   0   0   0
##  586 587 588 589 590 591 592 593 594 595 596 597 598 599 600
##   0   0   0   0   0   1   0   0   0   1   0   0   0   0   0
##  601 602 603 604 605 606 607 608 609 610 611 612 613 614 615
##   0   0   0   1   0   0   1   1   0   0   0   1   0   1   0
##  616 617 618 619 620 621 622 623 624 625 626 627 628 629 630
##   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
##  631 632 633 634 635 636 637 638 639 640 641 642 643 644 645
##   0   1   0   0   0   0   0   0   0   0   1   0   0   1   1
##  646 647 648 649 650 651 652 653 654 655 656 657 658 659 660
##   0   0   0   0   0   0   1   0   1   0   1   0   0   0   0
##  661 662 663 664 665 666 667 668 669 670 671 672 673 674 675
##   0   0   1   1   1   0   1   0   0   0   0   0   0   0   0
##  676 677 678 679 680 681 682 683 684 685 686 687 688 689 690
##   0   0   0   0   1   1   0   0   1   0   1   1   0   0   1
##  691 692 693 694 695 696 697 698 699 700 701 702 703 704 705
##   1   0   1   0   0   1   0   0   1   1   1   0   1   0   0
##  706 707 708 709 710 711 712 713 714 715 716 717 718 719 720
##   0   0   0   0   0   0   0   0   0   1   0   0   0   0   0
##  721 722 723 724 725 726 727 728 729 730 731 732 733 734 735
```

```
##    1    0    0    0    0    0    0    0    0    0    0    1    1    1    0
##  736  737  738  739  740  741  742  743  744  745  746  747  748  749  750
##    0    0    0    0    0    1    0    0    0    0    0    1    0    1    0
##  751  752  753  754  755  756  757  758  759  760  761  762  763  764  765
##    0    1    0    1    0    1    0    0    1    0    1    0    1    1    0
##  766  767  768  769  770  771  772  773  774  775  776  777  778  779  780
##    0    1    0    0    0    1    0    0    1    1    0    1    0    0    0
##  781  782  783  784  785  786  787  788  789  790  791  792  793  794  795
##    0    0    0    0    0    1    1    0    0    0    0    1    0    0    0
##  796  797  798  799  800  801  802  803  804  805  806  807  808  809  810
##    0    1    1    1    0    0    0    0    0    0    0    0    1    1    0
##  811  812  813  814  815  816  817  818  819  820  821  822  823  824  825
##    1    0    0    0    0    0    1    0    0    1    0    0    0    1    1
##  826  827  828  829  830  831  832  833  834  835  836  837  838  839  840
##    0    0    0    0    0    0    0    0    0    0    0    1    0    1    1
##  841  842  843  844  845  846  847  848  849  850  851  852  853  854  855
##    1    1    0    0    0    0    0    1    0    0    0    0    0    1    0
##  856  857  858  859  860  861  862  863  864  865  866  867  868  869  870
##    0    0    0    0    1    0    0    1    0    1    0    0    0    0    0
##  871  872  873  874  875  876  877  878  879  880  881  882  883  884  885
##    0    0    0    0    0    0    0    1    0    0    0    0    1    0    0
##  886  887  888  889  890  891  892  893  894  895  896  897  898  899  900
##    0    1    0    0    0    1    0    0    1    0    1    0    0    0    0
##  901  902  903  904  905  906  907  908  909  910  911  912  913  914  915
##    0    0    0    0    1    1    1    0    0    1    1    0    0    0    0
##  916  917  918  919  920  921  922  923  924  925  926  927  928  929  930
##    1    0    1    1    0    1    0    0    0    0    0    0    0    0    1
##  931  932  933  934  935  936  937  938  939  940  941  942  943  944  945
##    0    1    1    0    1    0    0    0    1    0    0    0    0    0    0
##  946  947  948  949  950  951  952  953  954  955  956  957  958  959  960
##    1    0    0    0    0    0    0    0    0    1    0    0    0    0    0
##  961  962  963  964  965  966  967  968  969  970  971  972  973  974  975
##    0    1    0    0    0    0    1    0    0    0    0    1    1    0    0
##  976  977  978  979  980  981  982  983  984  985  986  987  988  989  990
##    0    1    0    0    0    0    0    1    0    0    0    0    0    0    1
##  991  992  993  994  995  996  997  998  999 1000 1001 1002 1003 1004 1005
##    1    0    0    0    0    1    0    1    0    1    0    0    1    0    0
## 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020
##    1    0    0    1    0    0    0    1    0    0    1    0    0    0    1
## 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035
##    0    1    0    0    0    0    0    0    0    0    0    1    0    0    0
## 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 1048 1049 1050
##    1    0    0    0    1    0    0    0    1    1    1    1    1    0    1
## 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065
##    0    0    0    0    1    1    0    0    0    0    0    0    0    1    1
## 1066 1067 1068 1069 1070 1071 1072 1073 1074 1075 1076 1077 1078 1079 1080
##    1    1    0    1    1    0    0    1    0    1    0    1    1    1    1
## 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095
##    1    0    1    1    1    1    1    1    0    1    1    1    1    0    1
## 1096 1097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110
##    1    1    1    1    1    1    1    1    1    0    1    1    1    0    1
## 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 1121 1122 1123 1124 1125
##    0    1    0    0    1    1    0    1    1    1    1    0    1    0    0
## 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 1138 1139 1140
```

```
##    1    1    1    0    1    1    0    0    0    1    1    0    0    1    0
## 1141 1142 1143 1144 1145 1146 1147 1148 1149 1150 1151 1152 1153 1154 1155
##    1    0    1    1    1    1    1    1    0    1    1    1    1    0    1
## 1156 1157 1158 1159 1160 1161 1162 1163 1164 1165 1166 1167 1168 1169 1170
##    1    1    0    1    1    1    1    1    1    1    1    1    1    0    0
## 1171 1172 1173 1174 1175 1176 1177 1178 1179 1180 1181 1182 1183 1184 1185
##    1    0    1    0    1    1    0    0    0    1    1    1    1    0    1
## 1186 1187 1188 1189 1190 1191 1192 1193 1194 1195 1196 1197 1198 1199 1200
##    1    1    1    0    1    1    1    1    0    0    1    1    1    1    1
## 1201 1202 1203 1204 1205 1206 1207 1208 1209 1210 1211 1212 1213 1214 1215
##    1    1    1    1    1    1    1    1    0    1    1    1    1    1    0
## 1216 1217 1218 1219 1220 1221 1222 1223 1224 1225 1226 1227 1228 1229 1230
##    0    1    0    0    1    1    1    0    1    1    1    1    1    1    1
## 1231 1232 1233 1234 1235 1236 1237 1238 1239 1240 1241 1242 1243 1244 1245
##    0    1    1    0    1    1    1    1    0    0    1    0    1    0    0
## 1246 1247 1248 1249 1250 1251 1252 1253 1254 1255 1256 1257 1258 1259 1260
##    1    1    1    0    1    1    1    1    1    1    0    0    1    0    1
## 1261 1262 1263 1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 1274 1275
##    1    0    1    1    1    0    1    1    0    1    1    1    0    1    1
## 1276 1277 1278 1279 1280 1281 1282 1283 1284 1285 1286 1287 1288 1289 1290
##    1    1    1    1    0    1    0    0    1    0    1    1    1    1    0
## 1291 1292 1293 1294 1295 1296 1297 1298 1299 1300 1301 1302 1303 1304 1305
##    1    1    1    1    0    1    1    1    0    1    1    1    1    1    1
## 1306 1307 1308 1309 1310 1311 1312 1313 1314 1315 1316 1317 1318 1319 1320
##    0    1    1    1    1    1    1    0    1    0    1    1    0    1    1
## 1321 1322 1323 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 1334 1335
##    0    0    1    0    0    1    1    1    1    1    1    1    1    1    0
## 1336 1337 1338 1339 1340 1341 1342 1343 1344 1345 1346 1347 1348 1349 1350
##    1    1    1    1    0    1    1    1    1    1    0    1    1    1    0
## 1351 1352 1353 1354 1355 1356 1357 1358 1359 1360 1361 1362 1363 1364 1365
##    0    1    1    1    1    1    1    1    1    0    0    0    1    1    1
## 1366 1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380
##    1    1    1    0    1    0    1    1    1    1    1    0    1    0    1
## 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395
##    0    1    1    0    1    1    1    1    1    0    0    0    1    1    1
## 1396 1397 1398 1399 1400 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410
##    1    1    1    1    1    0    1    1    0    1    1    1    1    1    1
## 1411 1412 1413 1414 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425
##    1    1    1    0    0    1    1    1    1    1    0    0    1    1    0
## 1426 1427 1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440
##    0    0    1    0    0    1    1    1    1    1    0    0    0    1    1
## 1441 1442 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1455
##    1    1    0    0    0    1    0    1    1    1    0    1    1    0    1
## 1456 1457 1458 1459 1460 1461 1462 1463 1464 1465 1466 1467 1468 1469 1470
##    1    1    1    0    1    0    0    0    1    0    0    0    0    1    1
## 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 1481 1482 1483 1484 1485
##    1    0    1    1    1    1    1    1    1    1    1    1    1    1    1
## 1486 1487 1488 1489 1490 1491 1492 1493 1494 1495 1496 1497 1498 1499 1500
##    1    1    1    0    1    1    1    0    0    0    1    1    0    0    0
## 1501 1502 1503 1504 1505 1506 1507 1508 1509 1510 1511 1512 1513 1514 1515
##    1    1    1    1    1    0    0    1    1    1    1    1    1    1    0
## 1516 1517 1518 1519 1520 1521 1522 1523 1524 1525 1526 1527 1528 1529 1530
##    1    1    1    1    1    0    0    1    1    1    1    1    1    1    1
## 1531 1532 1533 1534 1535 1536 1537 1538 1539 1540 1541 1542 1543 1544 1545
```

```
##    1    0    0    1    1    0    1    1    1    0    0    1    1    0    1
## 1546 1547 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560
##    1    1    1    1    1    1    1    0    0    1    1    1    1    0    1
## 1561 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575
##    1    1    0    1    1    0    1    0    1    1    0    1    1    1    1
## 1576 1577 1578 1579 1580 1581 1582 1583 1584 1585 1586 1587 1588 1589 1590
##    1    1    1    1    1    0    0    1    1    1    0    0    1    0    1
## 1591 1592 1593 1594 1595 1596 1597 1598 1599 1600 1601 1602 1603 1604 1605
##    0    1    0    0    1    1    0    1    1    1    0    1    1    1    1
## 1606 1607 1608 1609 1610 1611 1612 1613 1614 1615 1616 1617 1618 1619 1620
##    1    0    1    0    0    1    1    1    1    1    1    1    1    0    0
## 1621 1622 1623 1624 1625 1626 1627 1628 1629 1630 1631 1632 1633 1634 1635
##    1    1    0    1    1    1    1    1    1    0    1    1    1    1    0
## 1636 1637 1638 1639 1640 1641 1642 1643 1644 1645 1646 1647 1648 1649 1650
##    1    1    1    1    1    0    1    0    1    1    1    1    0    1    0
## 1651 1652 1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665
##    0    0    1    1    1    0    0    1    1    1    0    1    1    1    0
## 1666 1667 1668 1669 1670 1671 1672 1673 1674 1675 1676 1677 1678 1679 1680
##    0    0    1    1    1    1    1    0    0    0    0    1    1    0    0
## 1681 1682 1683 1684 1685 1686 1687 1688 1689 1690 1691 1692 1693 1694 1695
##    0    1    1    0    1    1    1    1    1    1    0    0    0    1    0
## 1696 1697 1698 1699 1700 1701 1702 1703 1704 1705 1706 1707 1708 1709 1710
##    0    1    1    1    0    0    1    1    1    1    1    0    1    0    1
## 1711 1712 1713 1714 1715 1716 1717 1718 1719 1720 1721 1722 1723 1724 1725
##    1    1    1    1    1    1    0    1    1    0    1    1    1    1    1
## 1726 1727 1728 1729 1730 1731 1732 1733 1734 1735 1736 1737 1738 1739 1740
##    1    1    1    0    1    1    1    0    1    1    0    0    1    1    1
## 1741 1742 1743 1744 1745 1746 1747 1748 1749 1750 1751 1752 1753 1754 1755
##    0    1    1    0    1    0    0    0    1    0    1    0    1    0    1
## 1756 1757 1758 1759 1760 1761 1762 1763 1764 1765 1766 1767 1768 1769 1770
##    1    0    1    0    0    0    1    1    0    1    1    1    0    1    1
## 1771 1772 1773 1774 1775 1776 1777 1778 1779 1780 1781 1782 1783 1784 1785
##    0    1    0    1    1    0    0    0    1    1    1    1    1    1    1
## 1786 1787 1788 1789 1790 1791 1792 1793 1794 1795 1796 1797 1798 1799 1800
##    1    0    1    1    0    0    1    1    1    1    1    0    1    1    1
## 1801 1802 1803 1804 1805 1806 1807 1808 1809 1810 1811 1812 1813 1814 1815
##    1    1    1    1    1    0    1    0    1    0    1    0    1    1    0
## 1816 1817 1818 1819 1820 1821 1822 1823 1824 1825 1826 1827 1828 1829 1830
##    1    1    1    0    1    1    1    1    1    1    1    0    1    0    0
## 1831 1832 1833 1834 1835 1836 1837 1838 1839 1840 1841 1842 1843 1844 1845
##    1    1    0    0    0    1    0    1    1    1    1    0    0    1    1
## 1846 1847 1848 1849 1850 1851 1852 1853 1854 1855 1856 1857 1858 1859 1860
##    0    0    0    1    0    1    1    1    0    1    1    1    1    1    0
## 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871 1872 1873 1874 1875
##    1    1    1    0    1    1    1    1    1    1    1    1    1    1    1
## 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887 1888 1889 1890
##    1    0    0    1    1    1    0    1    1    0    1    1    1    0    0
## 1891 1892 1893 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905
##    1    1    1    1    1    1    0    1    1    1    1    1    0    1    1
## 1906 1907 1908 1909 1910 1911 1912 1913 1914 1915 1916 1917 1918 1919 1920
##    1    1    0    1    1    1    1    1    0    1    1    0    0    1    1
## 1921 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 1935
##    0    0    1    0    1    1    0    1    0    0    1    1    1    1    0
## 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949 1950
```

```
##    1    1    0    1    0    1    1    1    0    1    1    0    0    1    1
## 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965
##    0    1    1    1    1    1    1    1    1    1    1    1    1    1    1
## 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980
##    1    1    1    0    0    1    1    1    1    0    1    1    1    0    1
## 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995
##    1    1    1    1    1    1    1    1    0    1    0    0    1    1    1
## 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
##    1    1    0    1    1    1    1    1    1    0    0    1    1    0    1
## 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024 2025
##    1    1    1    0    0    1    1    1    1    1    0    1    1    1    0
## 2026 2027 2028 2029 2030 2031 2032 2033 2034 2035 2036 2037 2038 2039 2040
##    1    1    1    1    1    1    1    1    1    0    0    1    1    1    1
## 2041 2042 2043 2044 2045 2046 2047 2048 2049 2050 2051 2052 2053 2054 2055
##    1    1    0    1    1    1    1    0    0    1    1    0    1    1    0
## 2056 2057 2058 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 2069 2070
##    1    1    1    1    1    1    0    1    0    1    1    0    0    0    1
## 2071 2072 2073 2074 2075 2076 2077 2078 2079 2080 2081 2082 2083 2084 2085
##    1    1    1    1    1    1    1    1    1    0    1    1    0    0    1
## 2086 2087 2088 2089 2090 2091 2092 2093 2094 2095 2096 2097 2098 2099 2100
##    1    1    1    1    1    1    1    0    1    1    0    0    1    1    1
## 2101 2102 2103 2104 2105 2106 2107 2108 2109 2110 2111 2112 2113 2114 2115
##    0    0    1    1    1    1    1    1    0    1    0    1    1    1    1
## 2116 2117 2118 2119 2120 2121
##    1    1    1    1    1    0
```

5.2.11 ConfusionMatrix and Training Model Evaluation

```
ConfusionMatrix(ypredlog, BalancedData$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 802 260
##      1 308 751
```

5.2.12 Accuracy of Train, Precision , Recall, Sensitivity and F score

```
accuracy.meas(ypredlog,BalancedData$Churn)
```

```
##
## Call:
## accuracy.meas(response = ypredlog, predicted = BalancedData$Churn)
##
## Examples are labelled as positive when predicted is greater than 0.5
##
## precision: 0.477
## recall: 1.000
## F: 0.323
```

```
Accuracy(ypredlog,BalancedData$Churn)
```

```
## [1] 0.7322018
```

```
Sensitivity(BalancedData$Churn, ypredlog)
```

```
## [1] 0.7551789
```

```
Specificity(BalancedData$Churn,ypredlog)
```

```
## [1] 0.7091596
```

```
AUC(y_pred = ypredlog,y_true = BalancedData$Churn)
```

```
## [1] 0.7321692
```

5.2.13 Model Evaluation Table

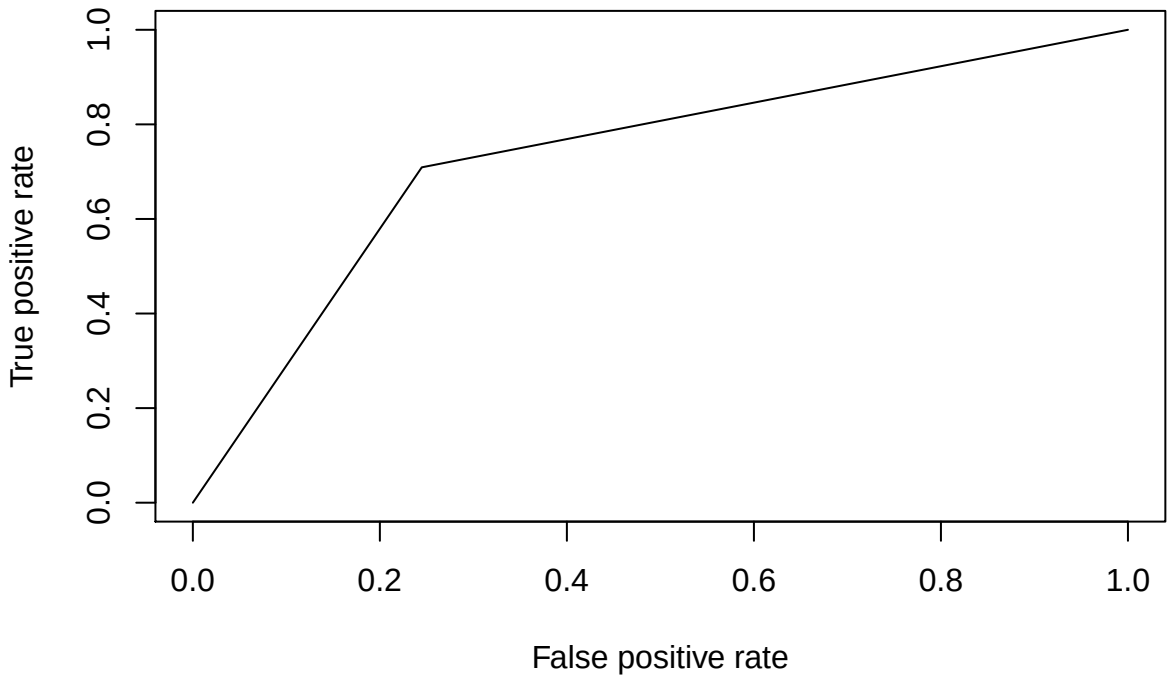
Accuracy	0.7322018
Sensitivity	0.7551789
Specificity	0.7091596
F Score	0.323
AUC	0.7321692

5.2.14 ROC Curve Plot

```
rocrpred2=prediction(ypredlog,BalancedData$Churn)
as.numeric(performance(rocrpred2,"auc")@y.values)
```

```
## [1] 0.7321692
```

```
pref2=performance(rocrpred2,"tpr","fpr")
plot(pref2)
```



5.2.15 Predicting Test Data

```
logpredtest <- predict(lregmodel,churn.test[-1],type = "response")
```

5.2.16 Coverting Probabilites to 0 and 1

```
logpredtestprob <- ifelse(logpredtest > 0.5,1,0)
logpredtestprob
```

```
##    3    4    9   11   14   15   20   22   25   26   31   33   36   37   42
##    0    1    1    1    1    1    0    0    0    1    0    0    1    0    1
##   44   47   48   53   55   58   59   64   66   69   70   75   77   80   81
##    0    1    0    1    1    0    1    0    0    0    1    0    1    0    1
##   86   88   91   92   97   99  102  103  108  110  113  114  119  121  124
##    0    0    0    0    0    0    0    0    0    1    1    0    0    1    0
##  125  130  132  135  136  141  143  146  147  152  154  157  158  163  165
##    0    0    0    0    1    0    0    1    0    0    0    1    0    0    0
##  168  169  174  176  179  180  185  187  190  191  196  198  201  202  207
##    0    0    0    1    1    1    1    0    0    0    0    1    0    1    0
##  209  212  213  218  220  223  224  229  231  234  235  240  242  245  246
##    1    1    0    0    0    0    0    1    1    0    1    0    1    0    0
##  251  253  256  257  262  264  267  268  273  275  278  279  284  286  289
##    0    1    1    0    0    0    0    0    0    0    0    0    0    0    0
##  290  295  297  300  301  306  308  311  312  317  319  322  323  328  330
##    1    0    0    0    0    1    1    1    0    0    0    0    0    0    0
##  333  334  339  341  344  345  350  352  355  356  361  363  366  367  372
##    1    0    1    0    0    0    0    1    1    0    1    0    1    1    1
##  374  377  378  383  385  388  389  394  396  399  400  405  407  410  411
##    1    0    0    1    0    0    0    1    0    0    0    1    0    0    0
##  416  418  421  422  427  429  432  433  438  440  443  444  449  451  454
##    1    0    0    1    0    1    0    0    0    0    0    1    0    0    0
##  455  460  462  465  466  471  473  476  477  482  484  487  488  493  495
##    1    0    1    0    1    0    0    0    1    0    0    1    0    1    0
##  498  499  504  506  509  510  515  517  520  521  526  528  531  532  537
##    1    1    1    0    0    1    1    0    1    0    0    0    0    0    0
##  539  542  543  548  550  553  554  559  561  564  565  570  572  575  576
##    0    0    1    1    1    1    1    0    0    0    0    1    0    1    0
##  581  583  586  587  592  594  597  598  603  605  608  609  614  616  619
##    1    0    0    1    0    0    0    0    0    0    0    0    1    0    0
##  620  625  627  630  631  636  638  641  642  647  649  652  653  658  660
##    1    0    1    1    0    0    0    0    0    0    0    0    0    0    0
##  663  664  669  671  674  675  680  682  685  686  691  693  696  697  702
##    0    0    0    0    1    0    1    0    0    0    1    1    0    0    0
##  704  707  708  713  715  718  719  724  726  729  730  735  737  740  741
##    0    0    0    0    0    0    0    1    0    1    0    1    1    0    0
##  746  748  751  752  757  759  762  763  768  770  773  774  779  781  784
##    0    0    0    0    1    0    1    1    0    0    0    1    1    1    1
##  785  790  792  795  796  801  803  806  807  812  814  817  818  823  825
##    0    0    1    1    0    0    0    0    0    0    0    0    0    0    0
##  828  829  834  836  839  840  845  847  850  851  856  858  861  862  867
##    0    0    0    0    0    0    0    0    1    0    1    0    0    0    0
##  869  872  873  878  880  883  884  889  891  894  895  900  902  905  906
##    0    1    0    1    1    0    0    0    0    1    1    0    1    1    1
##  911  913  916  917  922  924  927  928  933  935  938  939  944  946  949
```

```
##    0    0    0    0    0    0    0    0    1    0    0    0    0    0    0
##  950  955  957  960  961  966  968  971  972  977  979  982  983  988  990
##    1    0    0    0    1    0    0    0    1    0    1    0    0    1    1
##  993  994  999 1001 1004 1005 1010 1012 1015 1016 1021 1023 1026 1027 1032
##    0    0    0    1    0    0    1    1    0    0    0    0    0    0    0
## 1034 1037 1038 1043 1045 1048 1049 1054 1056 1059 1060 1065 1067 1070 1071
##    1    0    1    0    1    0    0    0    0    0    0    1    0    0    0
## 1076 1078 1081 1082 1087 1089 1092 1093 1098 1100 1103 1104 1109 1111 1114
##    0    1    0    0    0    0    0    0    0    1    1    0    0    0    1
## 1115 1120 1122 1125 1126 1131 1133 1136 1137 1142 1144 1147 1148 1153 1155
##    0    1    1    0    0    0    0    0    1    0    0    0    0    0    0
## 1158 1159 1164 1166 1169 1170 1175 1177 1180 1181 1186 1188 1191 1192 1197
##    0    1    0    1    0    1    0    1    0    0    1    0    0    1    0
## 1199 1202 1203 1208 1210 1213 1214 1219 1221 1224 1225 1230 1232 1235 1236
##    0    0    0    0    1    0    0    0    0    0    0    0    0    0    1
## 1241 1243 1246 1247 1252 1254 1257 1258 1263 1265 1268 1269 1274 1276 1279
##    0    1    0    1    0    1    0    0    1    0    0    0    1    0    0
## 1280 1285 1287 1290 1291 1296 1298 1301 1302 1307 1309 1312 1313 1318 1320
##    0    1    0    0    0    0    0    1    0    1    0    0    0    0    1
## 1323 1324 1329 1331 1334 1335 1340 1342 1345 1346 1351 1353 1356 1357 1362
##    1    0    0    0    0    1    1    0    0    0    1    0    1    0    0
## 1364 1367 1368 1373 1375 1378 1379 1384 1386 1389 1390 1395 1397 1400 1401
##    0    0    1    0    0    0    1    0    0    1    0    1    1    1    0
## 1406 1408 1411 1412 1417 1419 1422 1423 1428 1430 1433 1434 1439 1441 1444
##    0    1    0    0    1    1    0    0    1    0    0    0    1    0    1
## 1445 1450 1452 1455 1456 1461 1463 1466 1467 1472 1474 1477 1478 1483 1485
##    1    1    0    0    0    1    1    0    0    0    1    0    0    1    1
## 1488 1489 1494 1496 1499 1500 1505 1507 1510 1511 1516 1518 1521 1522 1527
##    0    0    0    0    0    0    0    1    0    0    0    0    0    1    0
## 1529 1532 1533 1538 1540 1543 1544 1549 1551 1554 1555 1560 1562 1565 1566
##    0    1    0    1    1    1    0    0    0    0    0    0    0    0    0
## 1571 1573 1576 1577 1582 1584 1587 1588 1593 1595 1598 1599 1604 1606 1609
##    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
## 1610 1615 1617 1620 1621 1626 1628 1631 1632 1637 1639 1642 1643 1648 1650
##    0    1    0    0    0    0    0    1    0    0    1    1    0    0    0
## 1653 1654 1659 1661 1664 1665 1670 1672 1675 1676 1681 1683 1686 1687 1692
##    0    1    0    1    0    0    0    1    1    0    0    0    0    0    1
## 1694 1697 1698 1703 1705 1708 1709 1714 1716 1719 1720 1725 1727 1730 1731
##    0    0    0    1    1    1    1    1    0    1    1    0    0    0    0
## 1736 1738 1741 1742 1747 1749 1752 1753 1758 1760 1763 1764 1769 1771 1774
##    0    0    0    1    0    0    0    1    1    0    0    0    1    0    1
## 1775 1780 1782 1785 1786 1791 1793 1796 1797 1802 1804 1807 1808 1813 1815
##    0    0    0    1    0    0    0    0    0    0    0    0    1    1    0
## 1818 1819 1824 1826 1829 1830 1835 1837 1840 1841 1846 1848 1851 1852 1857
##    0    1    0    1    0    0    0    0    0    0    1    0    0    1    0
## 1859 1862 1863 1868 1870 1873 1874 1879 1881 1884 1885 1890 1892 1895 1896
##    0    0    1    1    1    1    0    1    1    0    1    0    0    0    0
## 1901 1903 1906 1907 1912 1914 1917 1918 1923 1925 1928 1929 1934 1936 1939
##    0    0    0    0    1    0    0    0    1    0    0    0    1    1    0
## 1940 1945 1947 1950 1951 1956 1958 1961 1962 1967 1969 1972 1973 1978 1980
##    0    0    0    0    0    0    0    0    1    0    0    0    0    1    0
## 1983 1984 1989 1991 1994 1995 2000 2002 2005 2006 2011 2013 2016 2017 2022
##    0    0    0    0    0    0    0    1    1    0    0    0    0    1    0
## 2024 2027 2028 2033 2035 2038 2039 2044 2046 2049 2050 2055 2057 2060 2061
```

```
##    0    0    1    0    0    0    1    0    1    0    0    0    1    1    0
## 2066 2068 2071 2072 2077 2079 2082 2083 2088 2090 2093 2094 2099 2101 2104
##    0    1    0    1    1    0    0    0    1    0    0    0    0    0    1
## 2105 2110 2112 2115 2116 2121 2123 2126 2127 2132 2134 2137 2138 2143 2145
##    0    0    0    1    1    0    0    0    0    0    0    1    0    0    0
## 2148 2149 2154 2156 2159 2160 2165 2167 2170 2171 2176 2178 2181 2182 2187
##    1    0    0    1    1    0    1    1    0    0    0    0    0    0    1
## 2189 2192 2193 2198 2200 2203 2204 2209 2211 2214 2215 2220 2222 2225 2226
##    0    0    0    0    1    1    1    0    0    1    1    0    0    0    0
## 2231 2233 2236 2237 2242 2244 2247 2248 2253 2255 2258 2259 2264 2266 2269
##    1    0    0    1    0    0    0    0    0    0    0    1    0    0    0
## 2270 2275 2277 2280 2281 2286 2288 2291 2292 2297 2299 2302 2303 2308 2310
##    0    1    0    0    0    0    1    1    0    0    0    0    0    1    1
## 2313 2314 2319 2321 2324 2325 2330 2332 2335 2336 2341 2343 2346 2347 2352
##    0    1    0    0    0    1    0    0    0    1    0    0    0    0    0
## 2354 2357 2358 2363 2365 2368 2369 2374 2376 2379 2380 2385 2387 2390 2391
##    1    1    0    1    0    1    0    0    0    1    0    1    1    0    0
## 2396 2398 2401 2402 2407 2409 2412 2413 2418 2420 2423 2424 2429 2431 2434
##    1    1    1    1    0    0    0    1    0    1    0    0    1    0    1
## 2435 2440 2442 2445 2446 2451 2453 2456 2457 2462 2464 2467 2468 2473 2475
##    0    0    0    1    0    0    0    0    1    0    0    0    1    0    0
## 2478 2479 2484 2486 2489 2490 2495 2497 2500 2501 2506 2508 2511 2512 2517
##    0    0    0    0    0    1    0    0    1    0    1    0    1    0    0
## 2519 2522 2523 2528 2530 2533 2534 2539 2541 2544 2545 2550 2552 2555 2556
##    0    0    0    0    0    0    1    0    0    0    0    1    0    0    0
## 2561 2563 2566 2567 2572 2574 2577 2578 2583 2585 2588 2589 2594 2596 2599
##    0    0    0    0    0    1    0    0    0    1    1    0    0    1    0
## 2600 2605 2607 2610 2611 2616 2618 2621 2622 2627 2629 2632 2633 2638 2640
##    0    1    1    0    0    0    0    0    0    0    1    0    0    1    0
## 2643 2644 2649 2651 2654 2655 2660 2662 2665 2666 2671 2673 2676 2677 2682
##    1    0    0    0    0    0    0    0    1    1    0    1    0    0    0
## 2684 2687 2688 2693 2695 2698 2699 2704 2706 2709 2710 2715 2717 2720 2721
##    0    0    1    1    0    0    0    0    1    1    0    0    0    0    0
## 2726 2728 2731 2732 2737 2739 2742 2743 2748 2750 2753 2754 2759 2761 2764
##    0    1    0    0    0    0    1    0    0    1    0    0    0    1    0
## 2765 2770 2772 2775 2776 2781 2783 2786 2787 2792 2794 2797 2798 2803 2805
##    1    0    0    1    0    1    1    1    1    0    1    0    0    0    0
## 2808 2809 2814 2816 2819 2820 2825 2827 2830 2831 2836 2838 2841 2842 2847
##    0    0    0    0    1    0    1    0    0    0    0    0    0    0    0
## 2849 2852 2853 2858 2860 2863 2864 2869 2871 2874 2875 2880 2882 2885 2886
##    0    0    0    0    0    0    1    1    0    0    1    0    0    0    0
## 2891 2893 2896 2897 2902 2904 2907 2908 2913 2915 2918 2919 2924 2926 2929
##    0    0    1    1    0    1    0    0    0    1    0    1    0    0    0
## 2930 2935 2937 2940 2941 2946 2948 2951 2952 2957 2959 2962 2963 2968 2970
##    1    0    0    0    0    0    1    0    0    0    1    0    0    0    0
## 2973 2974 2979 2981 2984 2985 2990 2992 2995 2996 3001 3003 3006 3007 3012
##    0    0    0    1    0    0    1    1    1    0    1    1    1    0    0
## 3014 3017 3018 3023 3025 3028 3029 3034 3036 3039 3040 3045 3047 3050 3051
##    0    0    1    0    0    1    0    0    1    0    0    0    0    0    1
## 3056 3058 3061 3062 3067 3069 3072 3073 3078 3080 3083 3084 3089 3091 3094
##    0    0    1    1    1    0    0    0    0    1    1    0    0    0    1
## 3095 3100 3102 3105 3106 3111 3113 3116 3117 3122 3124 3127 3128 3133 3135
##    0    0    0    0    0    0    1    1    0    0    0    0    1    1    0
## 3138 3139 3144 3146 3149 3150 3155 3157 3160 3161 3166 3168 3171 3172 3177
```

```
##    0    0    0    1    1    0    0    0    0    0    0    0    0    0    0
## 3179 3182 3183 3188 3190 3193 3194 3199 3201 3204 3205 3210 3212 3215 3216
##    0    1    1    0    1    0    0    0    1    0    1    1    0    1    0
## 3221 3223 3226 3227 3232 3234 3237 3238 3243 3245 3248 3249 3254 3256 3259
##    0    1    1    1    0    0    0    1    0    0    0    0    0    1    1
## 3260 3265 3267 3270 3271 3276 3278 3281 3282 3287 3289 3292 3293 3298 3300
##    0    0    0    1    0    0    0    1    0    0    0    1    0    0    0
## 3303 3304 3309 3311 3314 3315 3320 3322 3325 3326 3331 3333
##    1    0    0    0    0    0    1    0    0    0    1    0
```

5.2.17 ConfusionMatrix and Test Data Model Evaluation

```
ConfusionMatrix(logpredtestprob, churn.test$Churn)
```

```
##        y_pred
## y_true   0   1
##      0 801 219
##      1  51 141
```

5.2.18 Accuracy of Train, Precision , Recall, Sensitivity and F score

```
accuracy.meas(logpredtestprob,churn.test$Churn)
```

```
##
## Call:
## accuracy.meas(response = logpredtestprob, predicted = churn.test$Churn)
##
## Examples are labelled as positive when predicted is greater than 0.5
##
## precision: 0.297
## recall: 1.000
## F: 0.229
```

```
Accuracy(logpredtestprob,churn.test$Churn)
```

```
## [1] 0.7772277
```

```
Sensitivity(churn.test$Churn, logpredtestprob)
```

```
## [1] 0.7852941
```

```
Specificity(churn.test$Churn,logpredtestprob)
```

```
## [1] 0.734375
```

```
AUC(y_pred = logpredtestprob,y_true = churn.test$Churn)
```

```
## [1] 0.7598346
```

5.2.19 Model Evaluation Table

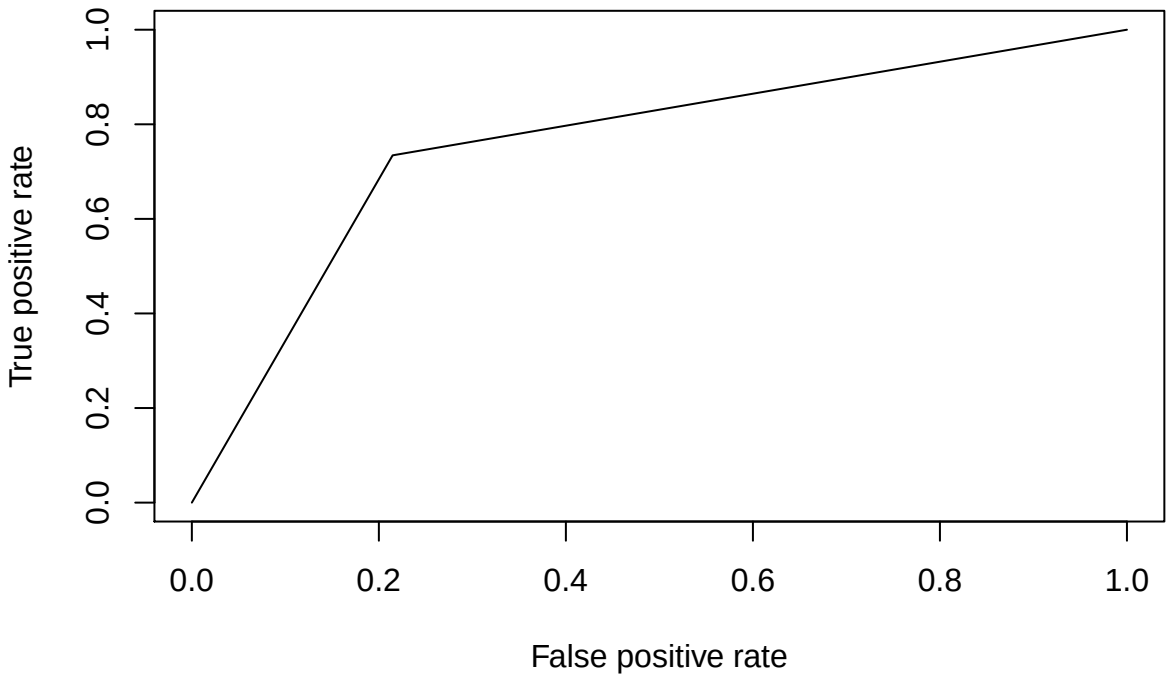
Accuracy	0.7772277
Sensitivity	0.7852941
Specificity	0.734375
F Score	0.229
AUC	0.7598346

5.2.20 ROC Curve Plot

```
rocrpred=prediction(logpredtestprob,churn.test$Churn)
as.numeric(performance(rocrpred,"auc")@y.values)
```

```
## [1] 0.7598346
```

```
pref=performance(rocrpred,"tpr","fpr")
plot(pref)
```



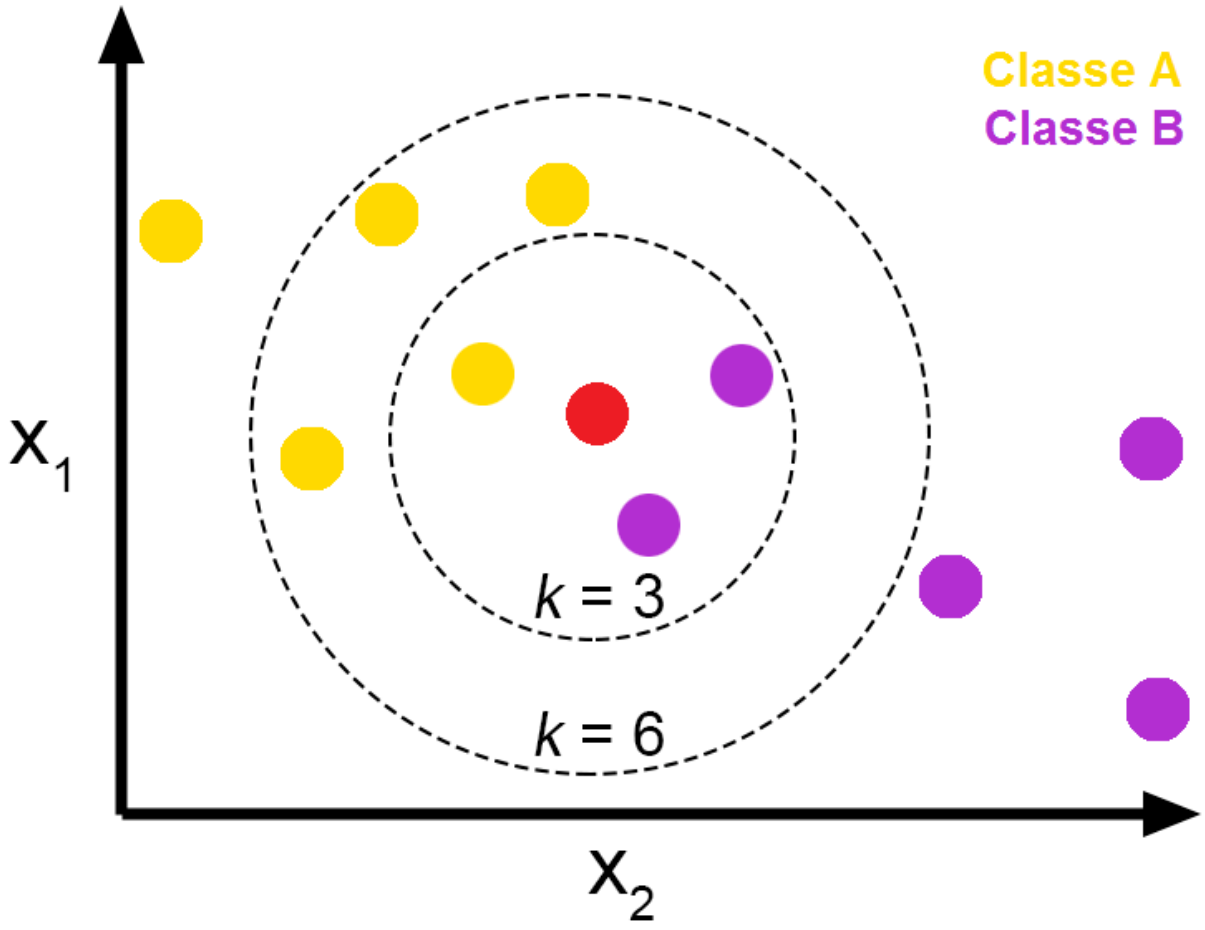


Figure 3: KNN

5.3 KNN

5.3.1 What is KNN

KNN (K — Nearest Neighbors) is one of many (supervised learning) algorithms used in data mining and machine learning, it's a classifier algorithm where the learning is based "how similar" is a data (a vector) from other .

5.3.2 Fitting KNN to the Training set and Predicting the Test set results

```
ypredKNN <- knn(train = BalancedData[,-1],
                test = churn.test[,-1],
                cl= BalancedData[,1],
                k=5
                )
```

```
ypredKNN
```

```
##    [1] 1 1 0 0 0 0 0 1 1 0 0 0 1 1 0 0 1 0 0 1 0 0 1 1 0 0 0 1 1 0 1 0 1 0
##   [35] 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 1 0 1 0 0 1 1 1 0 1 0 0 1 0 1 0 0 0 0
##   [69] 1 1 0 1 1 1 1 1 0 0 1 0 1 0 1 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0
##  [103] 0 1 0 0 0 0 0 0 1 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 1 0
```

```
##  [137] 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0 1 1 0 1 1 0 1 0 1 0 0 1 0
##  [171] 1 1 1 1 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 0 0
##  [205] 0 1 1 0 0 1 1 0 1 0 0 1 1 0 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 0 0 1 0
##  [239] 0 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 1 0 0 0
##  [273] 0 0 0 0 0 0 1 0 0 0 0 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0
##  [307] 1 0 1 1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 1 0 1 1 1 0 0 0 1 1 1 0 0 0 0 1
##  [341] 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0
##  [375] 0 0 0 0 1 0 0 1 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 0 1 1 0 0 0 1 0 0 0 1
##  [409] 1 0 1 0 0 0 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 1 0 0 0 0 0 0 0
##  [443] 0 0 0 0 0 0 1 0 1 1 1 1 0 0 1 0 0 0 0 0 0 0 1 1 0 0 1 0 0 1 1 1 0 0 0 0
##  [477] 1 1 0 0 0 0 0 0 0 1 1 0 0 1 1 1 0 1 0 0 0 1 0 0 0 0 0 0 1 1 0 1 1 0 0 0
##  [511] 1 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
##  [545] 0 0 0 1 0 1 0 0 1 0 0 0 1 1 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0
##  [579] 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 1 0 1 1 1 1 0 0 1
##  [613] 0 0 1 0 0 0 0 1 0 1 0 1 0 0 1 0 1 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [647] 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 1 0 0 0 1 0 0 0 1 0 1 0 1
##  [681] 0 1 1 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 1 0 1 0 0 0
##  [715] 0 1 0 0 1 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 1 0 1 1 0 1 1 0 1 1 0 0 0 0 0 1 0
##  [749] 1 1 0 0 1 0 0 0 0 0 1 0 0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1 0
##  [783] 0 1 1 0 0 0 1 0 1 0 0 0 1 1 1 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 1
##  [817] 0 0 1 0 1 1 1 1 1 0 1 1 0 0 0 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 0 1 1 0 0 0
##  [851] 0 1 1 0 0 1 1 0 0 1 1 1 0 1 0 0 0 0 0 0 1 1 1 1 0 0 1 1 0 1 0 1 0 0
##  [885] 1 1 1 1 0 0 1 0 1 0 0 0 0 0 0 1 1 1 1 1 1 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0
##  [919] 1 1 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 1 1 0 1 0 1 0 0 0
##  [953] 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 1 1 0
##  [987] 0 1 0 0 1 0 0 1 1 0 1 1 1 0 0 1 0 1 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 1 1
## [1021] 0 0 0 0 0 1 1 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 1 0 0 1
## [1055] 0 1 0 0 0 1 0 1 1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 1 0
## [1089] 1 1 0 1 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 1 0 1
## [1123] 1 1 0 1 0 1 0 1 1 0 0 0 0 0 1 1 1 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
## [1157] 1 0 1 0 0 1 1 1 0 0 1 1 0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 1 1 0 0 0 0 0
## [1191] 0 1 0 0 0 0 0 0 1 1 1 0 0 0 1 0 1 1 0 0 0 0
## Levels: 0 1
```

5.3.3 Confusion Matrix

```
ConfusionMatrix(ypredKNN, churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 712 308
##      1 100  92
```

```
Accuracy(ypredKNN,churn.test$Churn)
```

```
## [1] 0.6633663
```

5.3.4 Trying Different K value

```
ypredKNN2 <- knn(train = BalancedData[,-1],
                 test = churn.test[,-1],
                 cl= BalancedData[,1],
                 k=3
                 )
```

```
ConfusionMatrix(ypredKNN2, churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 677 343
##      1 100  92
```

```
Accuracy(ypredKNN2,churn.test$Churn)
```

```
## [1] 0.6344884
```

5.3.5 Try k 9

```
ypredKNN3 <- knn(train = BalancedData[,-1],
                 test = churn.test[,-1],
                 cl= BalancedData[,1],
                 k=9
                 )
```

```
ConfusionMatrix(ypredKNN3, churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 754 266
##      1 103  89
```

```
Accuracy(ypredKNN3,churn.test$Churn)
```

```
## [1] 0.6955446
```

5.3.6 Try K 15

```
ypredKNN4 <- knn(train = BalancedData[,-1],
                 test = churn.test[,-1],
                 cl= BalancedData[,1],
                 k=15
                 )
```

```
ConfusionMatrix(ypredKNN4, churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 789 231
##      1 110  82
```

```
Accuracy(ypredKNN4,churn.test$Churn)
```

```
## [1] 0.7186469
```

5.3.7 Try K 25

```
ypredKNN5 <- knn(train = BalancedData[,-1],
                 test = churn.test[,-1],
                 cl= BalancedData[,1],
                 k=25
                 )
```

```
ConfusionMatrix(ypredKNN5, churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 815 205
##      1 110  82
```

```
Accuracy(ypredKNN5,churn.test$Churn)
```

```
## [1] 0.740099
```

5.3.8 Try K 35

```
ypredKNN6 <- knn(train = BalancedData[,-1],
                 test = churn.test[,-1],
                 cl= BalancedData[,1],
                 k=35
                 )
```

```
ConfusionMatrix(ypredKNN6, churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 827 193
##      1 109  83
```

```
Accuracy(ypredKNN6,churn.test$Churn)
```

```
## [1] 0.7508251
```

```
Sensitivity(churn.test$Churn, ypredKNN6)
```

```
## [1] 0.8107843
```

```
Specificity(churn.test$Churn,ypredKNN6)
```

```
## [1] 0.4322917
```

```
accuracy.meas(ypredKNN6,churn.test$Churn)
```

```
##
## Call:
## accuracy.meas(response = ypredKNN6, predicted = churn.test$Churn)
##
## Examples are labelled as positive when predicted is greater than 0.5
##
## precision: 0.228
## recall: 1.000
## F: 0.185
```

```
KNNAUC<- AUC(ypredKNN6,churn.test$Churn)

KNNAUC
```

```
## [1] 0.621538
```

5.3.9 Model Evaluation Table

Accuracy	0.7508251
Sensitivity	0.8107843
Specificity	0.4322917
F Score	0.185
AUC	0.621538

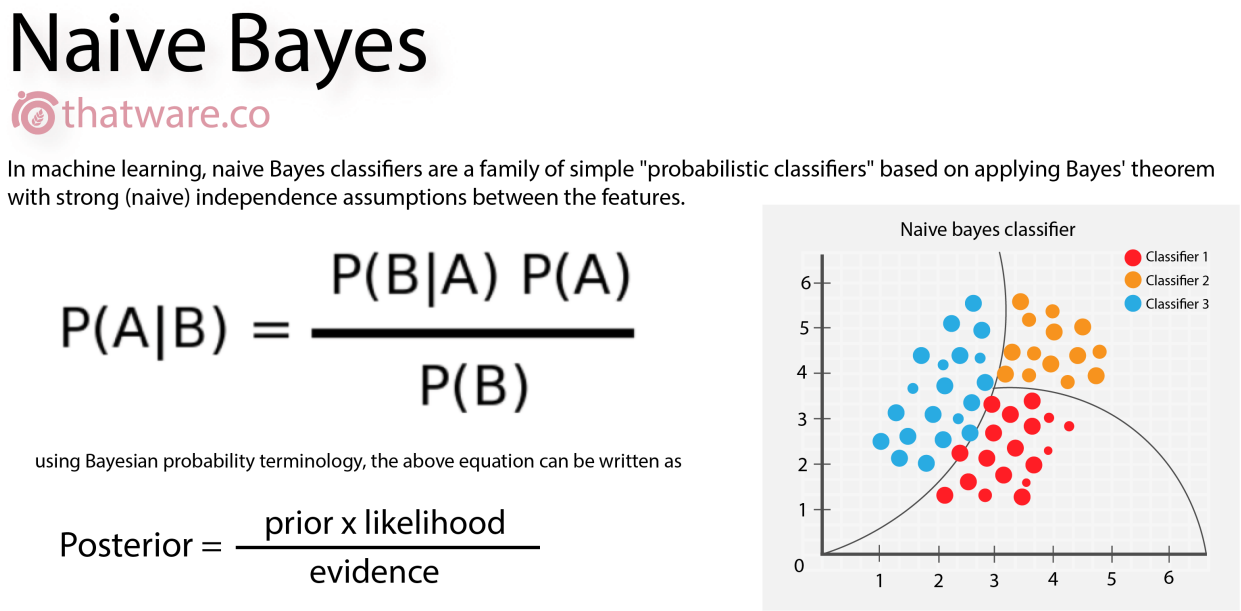


Figure 4: Naive Bayes

5.4 Naive Bayes

5.4.1 What is Naive Bayes

In machine learning, naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

5.4.2 Naive Bayes. Is it applicable here?

YES it is applicable. Naibe Bayes is a **probabilistic classifiers**. Here we'll use it to predict probability if Customer Cancelled(Churn) or Not Cancelled the service.

5.4.3 Fitting Naive Bayes to training set

```
classnaive <- naiveBayes(x=BalancedData[-1],
                         y=BalancedData$Churn)
```

5.4.4 Prediciting Naive Bayes Test Dataset

```
ypredNB<- predict(classnaive,newdata = churn.test[-1])
ypredNB
```

```
##    [1] 0 1 0 1 0 1 0 1 0 1 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0
##   [35] 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 0 0 0 1 1 1 0
##   [69] 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [103] 0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 0 1 0 1 1 0 0
##  [137] 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1
##  [171] 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0
##  [205] 0 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0 0 0 0 0 0 0 0
##  [239] 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0
##  [273] 0 0 0 0 0 0 1 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [307] 0 0 0 0 1 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 1 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0
```

```
##  [341] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0
##  [375] 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1
##  [409] 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0
##  [443] 0 0 0 0 0 1 0 1 0 1 0 1 0 1 0 0 1 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0
##  [477] 0 0 0 1 1 0 0 0 0 1 1 0 1 1 1 0 1 0 0 0 0 1 0 0 0 0 0 0 1 0 1 1 1 0
##  [511] 0 1 0 0 1 1 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0 0
##  [545] 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
##  [579] 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 1 1 0 0 0 0 1 0 1 0 0 0 0 1 0 0 0
##  [613] 0 0 1 0 0 0 1 1 1 0 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 0 1 0 0 0
##  [647] 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 1 0 1
##  [681] 1 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
##  [715] 0 0 0 0 1 0 0 0 0 0 0 0 0 1 1 0 0 0 1 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0
##  [749] 1 0 0 1 0 0 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0
##  [783] 0 1 1 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 1 0 0 1 0 0
##  [817] 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 1
##  [851] 0 0 1 0 0 1 1 0 1 0 1 0 0 0 0 0 1 1 0 0 1 1 1 1 0 0 0 1 0 1 0 0 1 0
##  [885] 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0
##  [919] 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 0
##  [953] 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0
##  [987] 0 0 0 0 0 0 1 0 1 1 0 0 0 1 0 0 1 0 1 0 0 0 0 1 0 0 0 1 1 0 0 1 0 0 0
## [1021] 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 1 0 0 0 1 0
## [1055] 0 1 0 0 0 0 0 0 1 1 0 0 1 0 1 0 0 0 0 1 0 0 0 0 1 1 0 0 0 0 0 0 1 0 0 1 1
## [1089] 1 0 1 1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 0 1 0 0
## [1123] 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
## [1157] 1 0 0 1 0 0 0 1 0 0 1 0 1 0 0 1 0 1 0 0 0 1 0 0 1 0 0 1 1 0 0 0 1 0
## [1191] 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 1 0
## Levels: 0 1
```

5.4.5 ConfusionMatrix

```
ConfusionMatrix(ypredNB,churn.test$Churn)
```

```
##       y_pred
## y_true   0   1
##      0 886 134
##      1  44 148
```

```
Accuracy(ypredNB,churn.test$Churn)
```

```
## [1] 0.8531353
```

```
Sensitivity(churn.test$Churn, ypredNB)
```

```
## [1] 0.8686275
```

```
Specificity(churn.test$Churn,ypredNB)
```

```
## [1] 0.7708333
```

```
accuracy.meas(ypredNB,churn.test$Churn)
```

```
##
## Call:
## accuracy.meas(response = ypredNB, predicted = churn.test$Churn)
##
## Examples are labelled as positive when predicted is greater than 0.5
##
```

```
## precision: 0.233
## recall: 1.000
## F: 0.189
```

```
NBAUC<- AUC(ypredNB,churn.test$Churn)

NBAUC
```

```
## [1] 0.8197304
```

5.4.6 Model Evaluation Table

Accuracy	0.8531353
Sensitivity	0.8686275
Specificity	0.7708333
F Score	0.189
AUC	0.8197304

5.5 Model Comparison using Model Performance metrics

	Logistic Regression	KNN	Naive Bayes
Accuracy	0.7772277	0.7508251	0.8531353
Sensitivity	0.7852941	0.8107843	0.8686275
Specificity	0.734375	0.4322917	0.7708333
F Score	0.229	0.185	0.189
AUC	0.7598346	0.621538	0.8197304

5.5.1 Interpretation

- From the above model Comparison we can see that **Naive Bayes** is performing the best.

5.5.2 Conclusion:

- We can see that all the models are performing similarly. However, Naive Bayes is performing the best. Naive Bayes is able to predict Customers who are going to cancel the service.
- ContractRenewal + CustServCalls + DayMins are the main factors which makes customer continue or cancel the service.

5.5.3 Recommendation:

- CustServCalls greatly contribute influence customer's decision to continue or cancel the service. It look like Customer Service has to be improved. There are many customers calling customer service 10 times. Company must invest in training resources and improve their customer service quality and reduce the number of customer service calls.