

## TELECOM CHURN ANALYSIS

Churn is a one of the biggest problems in the telecom industry. Research has shown that the average monthly churn rate among the top 4 wireless carriers in the US is 1.9% - 2%. So, for the analysis in this report I will be working with the telecom churn analysis from IBM dataset.

Initially we will need to pre-process our dataset for better testing results and the analysis, for pre-processing target will be to remove any null values if present in the dataset

### PREPROCESSING AND ANALYSING DATA

```
churn <- read.csv("TelcoChurnData.csv")
```

```
glimpse(churn)
```

```
> glimpse(churn)
Rows: 7,043
Columns: 21
$ customerID      <chr> "7590-VHVEG", "5575-GNVDE", "3668-QPYBK", "7795-CFOCW", "9~
$ gender          <chr> "Female", "Male", "Male", "Male", "Female", "Female", "Mal~
$ SeniorCitizen   <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0~
$ Partner         <chr> "Yes", "No", "No", "No", "No", "No", "No", "No", "Yes", "N~
$ Dependents      <chr> "No", "No", "No", "No", "No", "No", "No", "Yes", "No", "No", "Ye~
$ tenure          <int> 1, 34, 2, 45, 2, 8, 22, 10, 28, 62, 13, 16, 58, 49, 25, 69~
$ PhoneService    <chr> "No", "Yes", "Yes", "No", "Yes", "Yes", "Yes", "No", "Yes"~
$ MultipleLines   <chr> "No phone service", "No", "No", "No phone service", "No", ~
$ InternetService <chr> "DSL", "DSL", "DSL", "DSL", "Fiber optic", "Fiber optic", ~
$ OnlineSecurity  <chr> "No", "Yes", "Yes", "Yes", "No", "No", "No", "Yes", "No", ~
$ OnlineBackup    <chr> "Yes", "No", "Yes", "No", "No", "No", "Yes", "No", "No", ~
$ DeviceProtection <chr> "No", "Yes", "No", "Yes", "No", "Yes", "No", "No", "Yes", ~
$ TechSupport     <chr> "No", "No", "No", "Yes", "No", "No", "No", "No", "Yes", ~
$ StreamingTV     <chr> "No", "No", "No", "No", "No", "Yes", "Yes", "No", "Yes", ~
$ StreamingMovies <chr> "No", "No", "No", "No", "No", "Yes", "No", "No", "Yes", ~
$ Contract        <chr> "Month-to-month", "One year", "Month-to-month", "One year"~
$ PaperlessBilling <chr> "Yes", "No", "Yes", "No", "Yes", "Yes", "Yes", "No", "Yes"~
$ PaymentMethod   <chr> "Electronic check", "Mailed check", "Mailed check", "Bank ~
$ MonthlyCharges  <dbl> 29.85, 56.95, 53.85, 42.30, 70.70, 99.65, 89.10, 29.75, 10~
$ TotalCharges    <dbl> 29.85, 1889.50, 108.15, 1840.75, 151.65, 820.50, 1949.40, ~
$ Churn           <chr> "No", "No", "Yes", "No", "Yes", "Yes", "No", "No", "Yes", ~
```

- So, we obtained a summary of our dataset now our next target is to find out how many null values are in each of the columns

```
sapply(churn, function(x) sum(is.na(x)))
```

```
sapply(churn, function(x) sum(is.na(x)))
```

customerID	gender	SeniorCitizen	Partner	Dependents
0	0	0	0	0
tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity
0	0	0	0	0
OnlineBackup	DeviceProtection	TechSupport	StreamingTV	StreamingMovies
0	0	0	0	0
Contract	PaperlessBilling	PaymentMethod	MonthlyCharges	TotalCharges
0	0	0	0	11
Churn				
0				

- So, we saw some of our data rows are having NULL values so next our aim would be to handle them. Firstly, we will see how much of the proportion of data is having this null values. We ran the following command in R.

```
sum(is.na(churn$TotalCharges))/nrow(churn)
```

```
> sum(is.na(churn$TotalCharges))/nrow(churn)
[1] 0.001561834
```

- So, as we can see it is small part of our dataset so we will remove them for them dataset for the further analysis on it

```
churn_clean <- churn[complete.cases(churn), ]
```

- After having glimpse on the dataset, for making the analysis easier we will transform some of the value of the attributes in our dataset for further analysis. We will also drop the customer id column as we will not be using it in future

```
churn_clean$SeniorCitizen <- as.factor(mapvalues(churn_clean$SeniorCitizen, from=c("0", "1"), to=c("No", "Yes"))) <-
```

```
churn_clean$MultipleLines <- as.factor(mapvalues(churn_clean$MultipleLines, from=c("No phone service"), to=c("No")))
```

```
for(i in 10:15){
```

```
  churn_clean[,i] <- as.factor(mapvalues(churn_clean[,i], from= c("No internet service"), to= c("No")))
```

```
}
```

```
churn_clean$customerID <- NULL
```

```
glimpse(churn_clean)
```

```
> glimpse(churn_clean)
```

```
Rows: 7,032
```

```
Columns: 20
```

```
$ gender      <chr> "Female", "Male", "Male", "Male", "Female", "Female", "~
$ SeniorCitizen <fct> No, No, No, No, No, No, No, No, No, No, No, No, No, No, No, ~
$ Partner      <chr> "Yes", "No", "No", "No", "No", "No", "No", "No", "No", "Yes", ~
$ Dependents   <chr> "No", "No", "No", "No", "No", "No", "No", "Yes", "No", "No", ~
$ tenure       <int> 1, 34, 2, 45, 2, 8, 22, 10, 28, 62, 13, 16, 58, 49, 25, ~
$ PhoneService <chr> "No", "Yes", "Yes", "No", "Yes", "Yes", "Yes", "No", "Y~
$ MultipleLines <fct> No, No, No, No, No, Yes, Yes, No, Yes, No, No, No, Yes, ~
$ InternetService <chr> "DSL", "DSL", "DSL", "DSL", "Fiber optic", "Fiber optic~
$ OnlineSecurity <fct> No, Yes, Yes, Yes, No, No, No, Yes, No, Yes, Yes, No, N~
$ OnlineBackup <fct> Yes, No, Yes, No, No, No, Yes, No, No, Yes, No, No, No, ~
$ DeviceProtection <fct> No, Yes, No, Yes, No, Yes, No, No, Yes, No, No, No, Yes~
$ TechSupport  <fct> No, No, No, Yes, No, No, No, No, Yes, No, No, No, No, N~
$ StreamingTV  <fct> No, No, No, No, No, Yes, Yes, No, Yes, No, No, No, Yes, ~
$ StreamingMovies <fct> No, No, No, No, No, Yes, No, No, Yes, No, No, No, Yes, ~
$ Contract     <chr> "Month-to-month", "One year", "Month-to-month", "One ye~
$ PaperlessBilling <chr> "Yes", "No", "Yes", "No", "Yes", "Yes", "Yes", "No", "Y~
$ PaymentMethod <chr> "Electronic check", "Mailed check", "Mailed check", "Ba~
$ MonthlyCharges <dbl> 29.85, 56.95, 53.85, 42.30, 70.70, 99.65, 89.10, 29.75, ~
$ TotalCharges <dbl> 29.85, 1889.50, 108.15, 1840.75, 151.65, 820.50, 1949.4~
$ Churn        <chr> "No", "No", "Yes", "No", "Yes", "Yes", "No", "No", "Yes~
```

## 1. ANALYSING GENDER, SENIOR CITIZEN, PARTNER AND DEPENDENTS WITH CHURN

- So, finally we have cleaned our dataset now we will apply some of the statistical analysis techniques on the dataset to observe some of the patterns and draw some conclusions from it
- Firstly, we will look into some of bar charts for the gender, Senior Citizen, Partner and Dependents and see how our churn varies with these factors

#Gender plot

```
p1 <- ggplot(churn_clean, aes(x = gender)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%')),  
            stat = 'count',  
            position = position_dodge(.1),  
            size = 3)
```

#Senior citizen plot

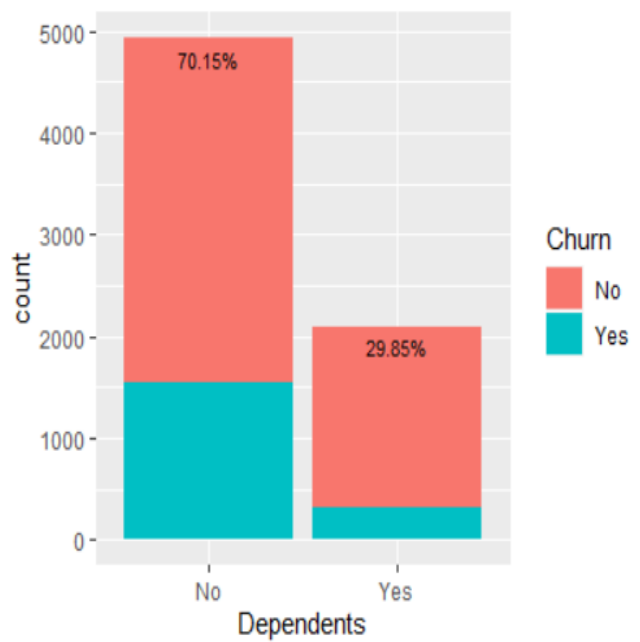
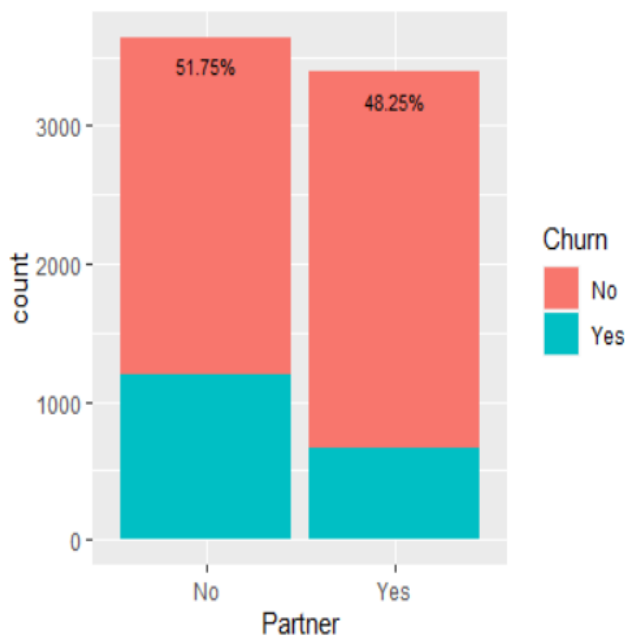
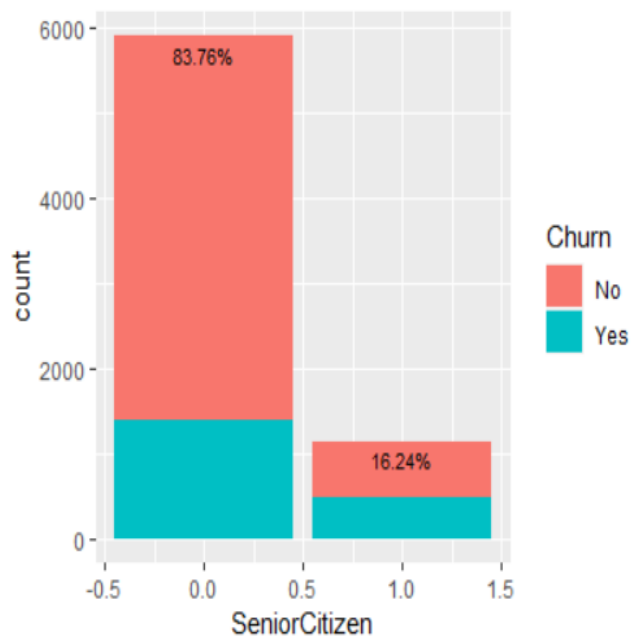
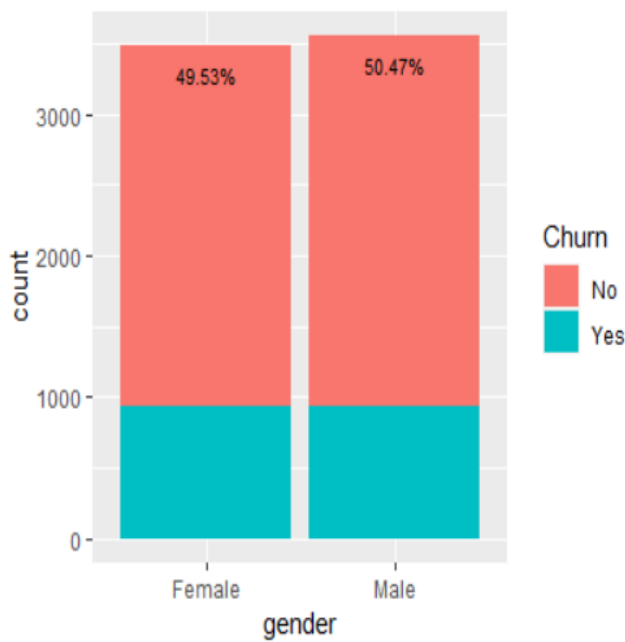
```
p2 <- ggplot(churn_clean, aes(x = SeniorCitizen)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%')),  
            stat = 'count',  
            position = position_dodge(.1),  
            size = 3)
```

#Partner plot

```
p3 <- ggplot(churn_clean, aes(x = Partner)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%')),  
            stat = 'count', position = position_dodge(.1), size = 3)
```

```
#Dependents plot
```

```
p4 <- ggplot(churn_clean, aes(x = Dependents)) + geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200, label = paste0(round(prop.table(..count..),4) * 100, '%')),  
  stat = 'count',  
  position = position_dodge(.1), size = 3)  
#Plot demographic data within a grid  
grid.arrange(p1, p2, p3, p4, ncol=2)
```



Now we will draw some conclusions from these bar plots of the following attributes in our dataset.

## QUESTION 1

### 1. How many females are there who are Senior citizen?

```
> df2 <- filter(churn_clean, SeniorCitizen=="Yes" & gender=="Female")
> print(nrow(df2))
[1] 568
```

### 2. How many females are dependent and Senior Citizen?

```
> df2 <- filter(churn_clean, SeniorCitizen=="Yes" & gender=="Female" & Dependents=="Yes")
> print(nrow(df2))
[1] 42
```

### 3. How many men are dependent and Senior Citizen?

```
> df2 <- filter(churn_clean, SeniorCitizen=="Yes" & gender=="Male" & Dependents=="Yes")
> print(nrow(df2))
[1] 49
```

### 4. How many are not Senior Citizen and Are dependents?

```
> df2 <- filter(churn_clean, SeniorCitizen=="No" & Dependents=="Yes")
> print(nrow(df2))
[1] 2008
```

### 5. How many of females have Partners?

```
> df2 <- filter(churn_clean, gender=="Female" & Partner=="Yes")
> print(nrow(df2))
[1] 1683
```

## QUESTION 2

### 1. We now want to verify that whether the churn is dependent on the gender or not?

We will apply chi square test between the churn and gender attributes and if they are dependent then the p values at the X2 observed will be less than 0.05 which is our significance level.

**Null hypothesis: Churn is independent of gender**

**Alternate Hypothesis: Churn is dependent on gender**

```
> print(chisq.test(table(churn_clean$gender, churn_clean$Churn)))
```

Pearson's Chi-squared test with Yates' continuity correction

```
data: table(churn_clean$gender, churn_clean$Churn)
X-squared = 0.47545, df = 1, p-value = 0.4905
```

We will accept our null hypothesis and hence Churn is independent of the gender.

## **2. We now want to verify that whether the Senior Citizen is a variable on which churn depends?**

**Null hypothesis: Churn is independent of Senior Citizen**

**Alternate Hypothesis: Churn is dependent on Senior Citizen**

```
> print(chisq.test(table(churn_clean$SeniorCitizen,churn_clean$Churn)))
```

Pearson's Chi-squared test with Yates' continuity correction

```
data: table(churn_clean$SeniorCitizen, churn_clean$Churn)
X-squared = 158.44, df = 1, p-value < 2.2e-16
```

As p is less than 0.05 so we will ACCEPT the ALTERNATE HYPOTHESIS and hence we can say that churn is dependent on the Senior Citizen attribute

## **3. Is chances of staying with the telecom company of people who are dependents greater than the chances of people who are not dependents.**

**Null Hypothesis: Chances of staying of not dependents = chances of staying of dependents**

**Alternate Hypothesis: Chances of staying of dependents > chances of staying of not dependents**

```
d1 <- filter(churn_clean,Dependents=="Yes")
```

```
d2 <- filter(churn_clean,Dependents=="No")
```

```
d3 <- filter(d1,Churn=="No")
```

```
d4 <- filter(d2,Churn=="No")
```

```
prop.test(c(nrow(d3),nrow(d4)),c(nrow(d1),nrow(d2)),alternative=c("greater"))
```

2-sample test for equality of proportions with continuity correction

```
data: c(nrow(d3), nrow(d4)) out of c(nrow(d1), nrow(d2))
X-squared = 186.32, df = 1, p-value < 2.2e-16
alternative hypothesis: greater
95 percent confidence interval:
 0.1401989 1.0000000
sample estimates:
 prop 1      prop 2 
0.8446879 0.6872086
```

ALTERNATE HYPOTHESIS is ACCEPTED. Hence, we can say that chances of staying of who are dependents is greater than chances of people who are not dependents staying with the telecom company as per the dataset analysis.

## **Summary of the above section**

Most of the customers who are not senior citizens have a comparatively less chances of churn as compared to the ones who are senior citizen. People with a partner have a higher chance of not being churned by the telecom company service as per the bar chart.

## 2. ANALYSING COLUMNS OF SERVICES PROVIDED BY THE TELECOM COMPANY

In this section will work on columns which include phone service, internet service, multiple Lines, Online security, Online backup, Device Protection, Tech support, Streaming Movies, Streaming TV.

Firstly, we will plot some charts in R to visualize these columns:

#Phone service plot

```
p5 <- ggplot(churn_clean, aes(x = PhoneService)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%'),  
                stat = 'count',  
                position = position_dodge(.1),  
                size = 3)
```

#Multiple phone lines plot

```
p6 <- ggplot(churn_clean, aes(x = MultipleLines)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%'),  
                stat = 'count',  
                position = position_dodge(.1),  
                size = 3)
```

#Internet service plot

```
p7 <- ggplot(churn_clean, aes(x = InternetService)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%'),  
                stat = 'count',
```

```
position = position_dodge(.1),  
size = 3)
```

#Online security service plot

```
p8 <- ggplot(churn_clean, aes(x = OnlineSecurity)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
    label = paste0(round(prop.table(..count..),4) * 100, '%'),  
    stat = 'count',  
    position = position_dodge(.1),  
    size = 3)
```

#Online backup service plot

```
p9 <- ggplot(churn_clean, aes(x = OnlineBackup)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
    label = paste0(round(prop.table(..count..),4) * 100, '%'),  
    stat = 'count',  
    position = position_dodge(.1),  
    size = 3)
```

#Device Protection service plot

```
p10 <- ggplot(churn_clean, aes(x = DeviceProtection)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
    label = paste0(round(prop.table(..count..),4) * 100, '%'),  
    stat = 'count',  
    position = position_dodge(.1),  
    size = 3)
```



```
#Tech Support service plot
```

```
p11 <- ggplot(churn_clean, aes(x = TechSupport)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
    label = paste0(round(prop.table(..count..),4) * 100, '%')),  
    stat = 'count',  
    position = position_dodge(.1),  
    size = 3)
```

```
#Streaming TV service plot
```

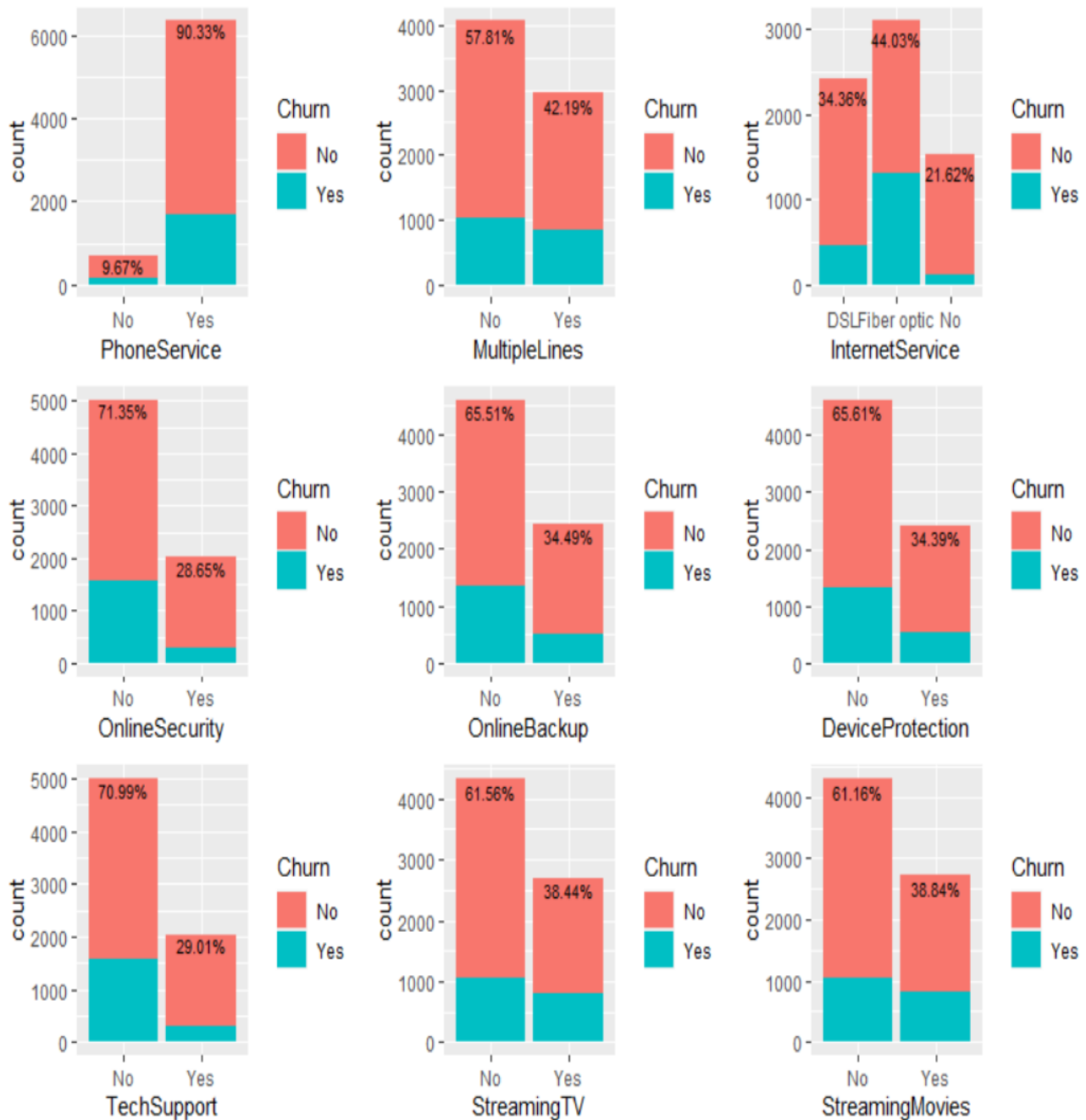
```
p12 <- ggplot(churn_clean, aes(x = StreamingTV)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
    label = paste0(round(prop.table(..count..),4) * 100, '%')),  
    stat = 'count',  
    position = position_dodge(.1),  
    size = 3)
```

```
#Streaming Movies service plot
```

```
p13 <- ggplot(churn_clean, aes(x = StreamingMovies)) +geom_bar(aes(fill = Churn))  
+ geom_text(aes(y = ..count.. -200, label = paste0(round(prop.table(..count..),4) * 100, '%')),  
  stat = 'count',  
    position = position_dodge(.1),  
    size = 3)
```

```
#Plot service data within a grid
```

```
grid.arrange(p5, p6, p7,p8, p9, p10,p11, p12, p13,ncol=3)
```



### Question3:

**1. How many of the customers opted for Phone Service as well as Internet Service?**

```
> df<-filter(churn_clean,InternetService=="DSL" | InternetService=="Fiber optic" & PhoneService=="Yes")
> print(nrow(df))
[1] 5512
```

**2. How many Customers opted for only Internet Service?**

```
> df<-filter(churn_clean,InternetService=="DSL" | InternetService=="Fiber optic" & PhoneService=="No")
> print(nrow(df))
[1] 2416
```

**3. How many customers opted for DSL as Internet Service?**

```
> print(nrow(filter(churn_clean,InternetService=="DSL")))
```

```
[1] 2416
```

#### 4. How many Customers opted for Fabric Optic as their Internet Service?

```
> print(nrow(filter(churn_clean,InternetService=="Fiber optic")))
```

```
[1] 3096
```

#### 5. Is churn rate of customers who opted for DSL service less than churn rate of Fiber optic ones in our sample dataset?

```
cr1 <- (nrow(filter(churn_clean,InternetService=="Fiber optic" &
Churn=="Yes")))/nrow(filter(churn_clean,InternetService=="Fiber optic"))
```

```
cr2 <- (nrow(filter(churn_clean,InternetService=="DSL" &
Churn=="Yes")))/nrow(filter(churn_clean,InternetService=="DSL"))
```

```
print(cr2 < cr1)
```

```
> print(cr2 < cr1)
```

```
[1] TRUE
```

#### Question 4:

##### 1. We will verify that whether the churn rate of people with multiple lines is less than churn rate of people with no multiple lines for the population?

As we know here success can be made similar to the churn rate being Yes. So, we will be applying test for comparing two proportions where we know total customers in that category and no of customers in that proportions who will be churned.

**NULL HYPOTHESIS: Is churning rate of people with multiple Lines = Churning rate of people with no multiple lines**

**ALTERNATE HYPOTHESIS: Is churning rate of people with multiple Lines < Churning rate of people with no multiple lines**

```
t1 <- table(churn_clean$Churn,churn_clean$MultipleLines)
```

```
tab_values <- as.numeric(t1) # Extract values
```

```
N1 <- tab_values[2]+tab_values[1]
```

```
N2 <- tab_values[3]+tab_values[4]
```

```
prop.test(c(tab_values[4],tab_values[1]),c(N2,N1),alternative="less")
```

2-sample test for equality of proportions with continuity correction

```
data: c(tab_values[4], tab_values[1]) out of c(N2, N1)
X-squared = 1485.2, df = 1, p-value < 2.2e-16
alternative hypothesis: less
95 percent confidence interval:
-1.0000000 -0.4449003
sample estimates:
prop 1 prop 2
0.2864847 0.7493235
```

**We can now safely conclude that the churn rate of the customers with multiple is less than the customers with no multiple lines as the values of  $p < 0.05$  we will ACCEPT the ALTERNATE HYPOTHESIS.**

### **3. ANALYSING OTHER PAYMENT MODES, PAPER BILLING AND CONTRACT**

#Contract status plot

```
p14 <- ggplot(churn_clean, aes(x = Contract)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%'),  
                stat = 'count',  
                position = position_dodge(.1),  
                size = 3)
```

#Paperless billing plot

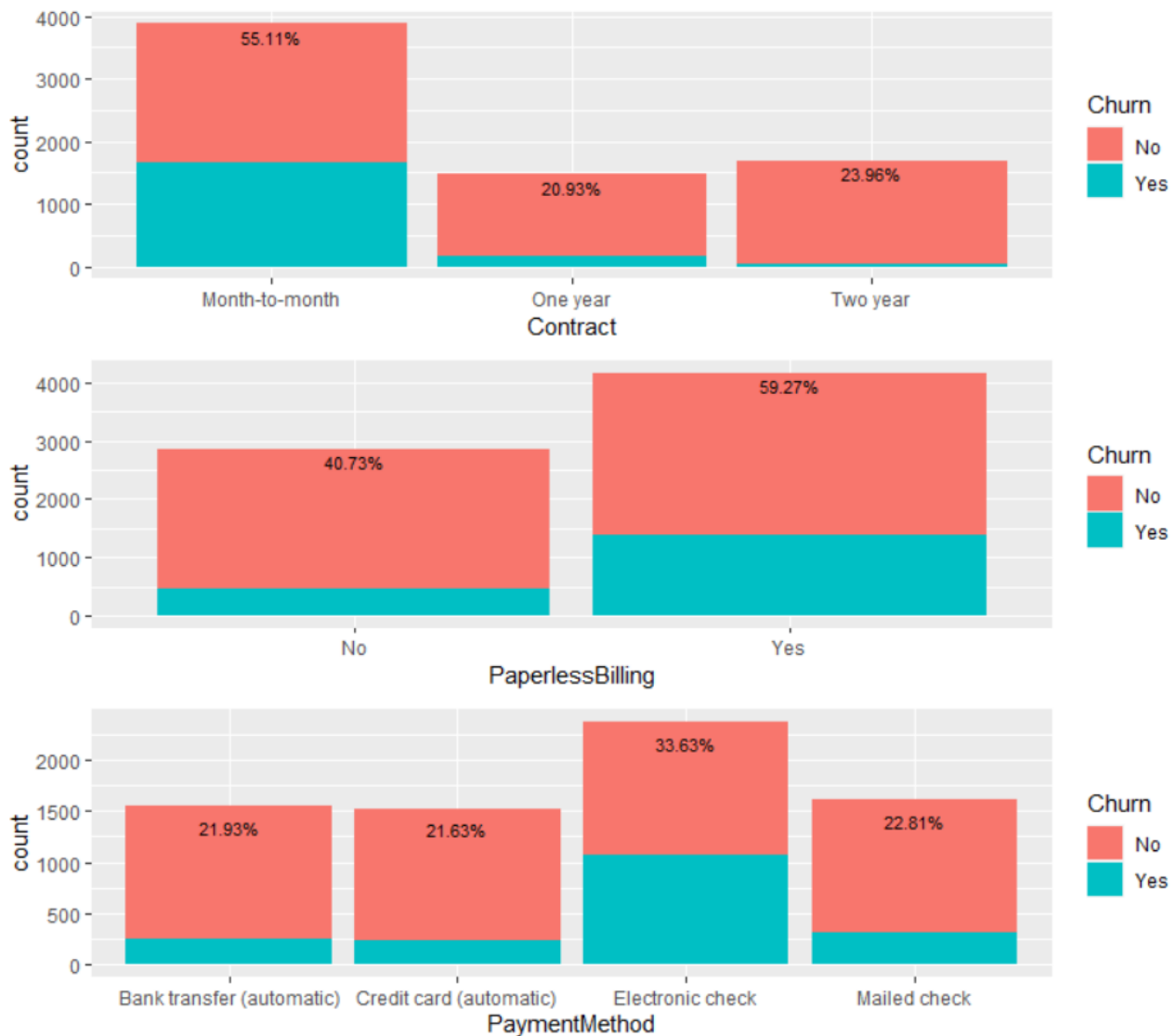
```
p15 <- ggplot(churn_clean, aes(x = PaperlessBilling)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%'),  
                stat = 'count',  
                position = position_dodge(.1),  
                size = 3)
```

#Payment method plot

```
p16 <- ggplot(churn_clean, aes(x = PaymentMethod)) +  
  geom_bar(aes(fill = Churn)) +  
  geom_text(aes(y = ..count.. -200,  
                label = paste0(round(prop.table(..count..),4) * 100, '%'),  
                stat = 'count',  
                position = position_dodge(.1),  
                size = 3)
```

#Plot contract data within a grid

```
grid.arrange(p14,p15,p16,ncol=1)
```



## QUESTION 5

### 1. Is paperless billing dependent on Payment Method?

We do this test to check if according to our data paperless billing dependent on payment method which must be true in real scenario. So, we again perform a chi square test to test the two attributes are dependent or not

**NULL HYPOTHESIS: Paperless Billing is independent of Payment Method**

**ALTERNATE HYPOTHESIS: Paperless Billing is dependent on Payment Method**

```
print(chisq.test(table(churn_clean$PaymentMethod,churn_clean$PaperlessBilling)))
```

Pearson's Chi-squared test

```
data: table(churn_clean$PaymentMethod, churn_clean$PaperlessBilling)
X-squared = 431.3, df = 3, p-value < 2.2e-16
```

As the p value  $< 0.05$  so the ALTERNATE HYPOTHESIS is ACCEPTED. Hence, we say that paper billing is dependent on the payment method.

## 2. Which Payment Method have maximum Churn Rate in the dataset?

```
c1 <- nrow(filter(churn_clean,PaymentMethod == 'Credit card (automatic)' &
Churn=="Yes"))/nrow(filter(churn_clean,PaymentMethod == 'Credit card (automatic)'))

c2 <- nrow(filter(churn_clean,PaymentMethod == 'Bank transfer (automatic)' &
Churn=="Yes"))/nrow(filter(churn_clean,PaymentMethod == 'Bank transfer (automatic)'))

c3 <- nrow(filter(churn_clean,PaymentMethod == 'Electronic check' &
Churn=="Yes"))/nrow(filter(churn_clean,PaymentMethod == 'Electronic check'))

c4 <- nrow(filter(churn_clean,PaymentMethod == 'Mailed check' &
Churn=="Yes"))/nrow(filter(churn_clean,PaymentMethod == 'Mailed check'))

maxrate <- which.max(c(c1,c2,c3,c4))

print(maxrate)
```

```
> maxrate <- which.max(c(c1,c2,c3,c4))
> print(maxrate)
[1] 3
```

**So, from the above output we conclude that customers who opt for electronic check have higher chances of being churned than other payment methods.**

## ANALYSING CONTINUOUS ATTRIBUTES TENURE, TOTAL CHARGES AND MONTHLY CHARGES

#Tenure histogram

```
p17 <- ggplot(data = churn_clean, aes(tenure, color = Churn))+
  geom_freqpoly(binwidth = 5, size = 1)
```

#Monthly charges histogram

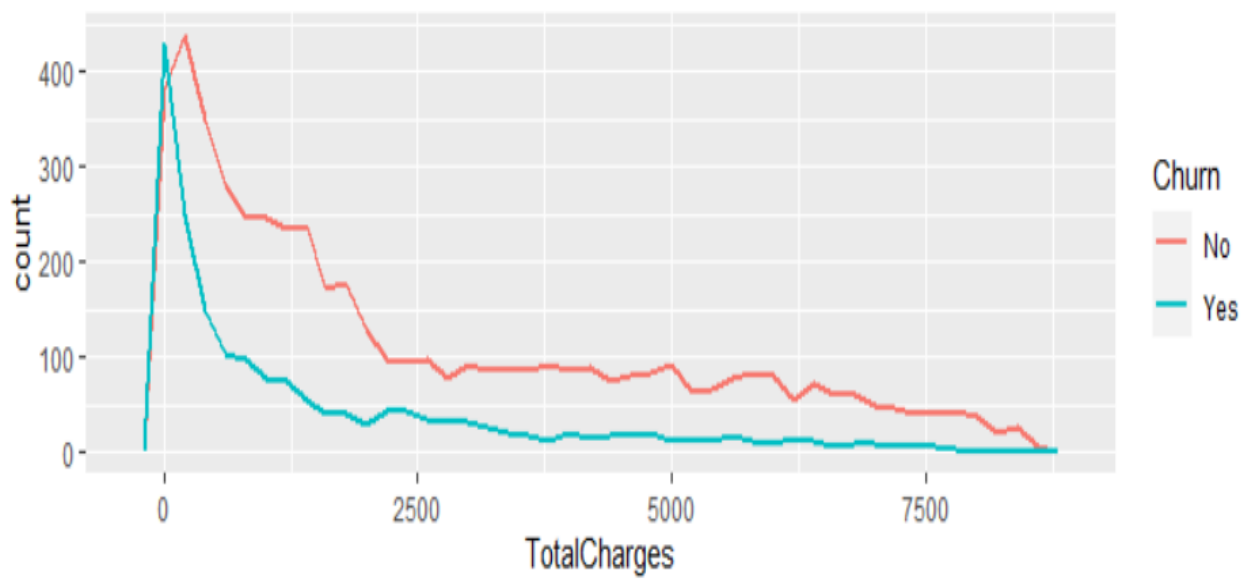
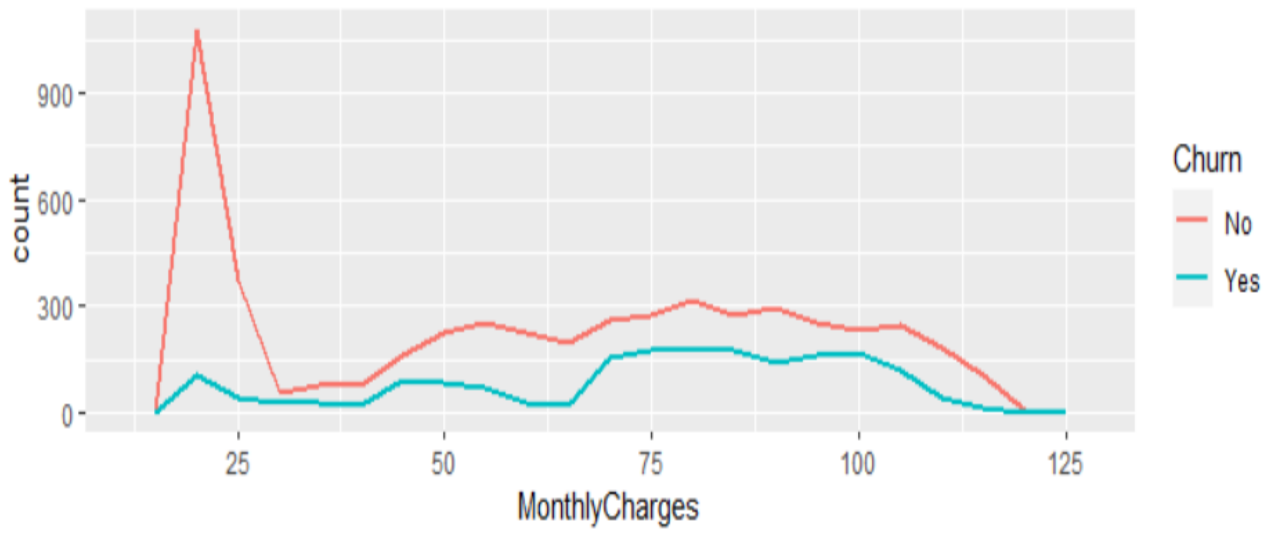
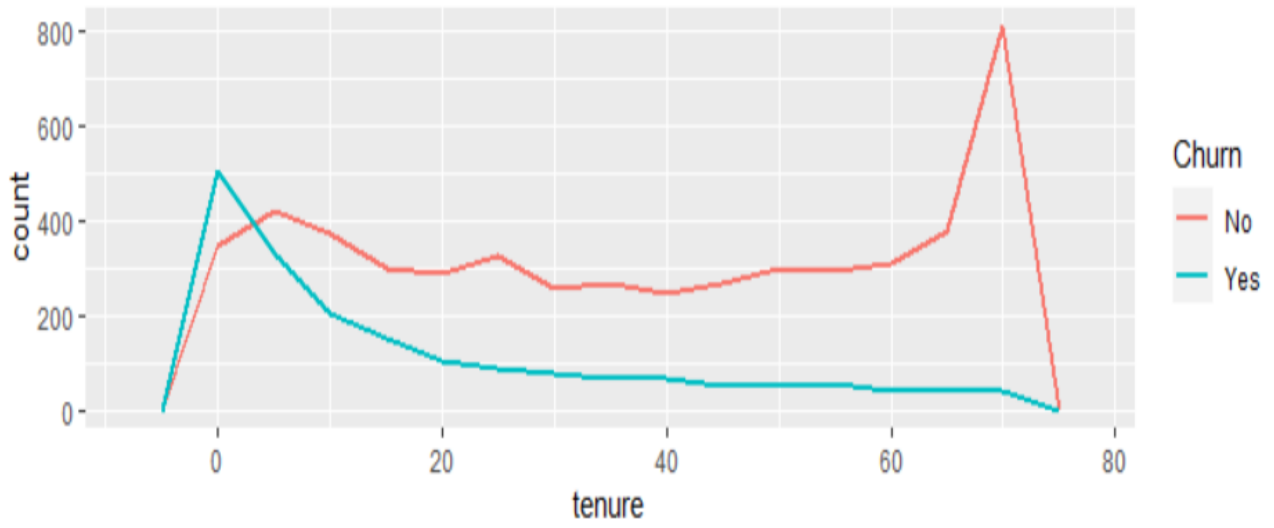
```
p18 <- ggplot(data = churn_clean, aes(MonthlyCharges, color = Churn))+
  geom_freqpoly(binwidth = 5, size = 1)
```

#Total charges histogram

```
p19 <- ggplot(data = churn_clean, aes(TotalCharges, color = Churn))+
  geom_freqpoly(binwidth = 200, size = 1)
```

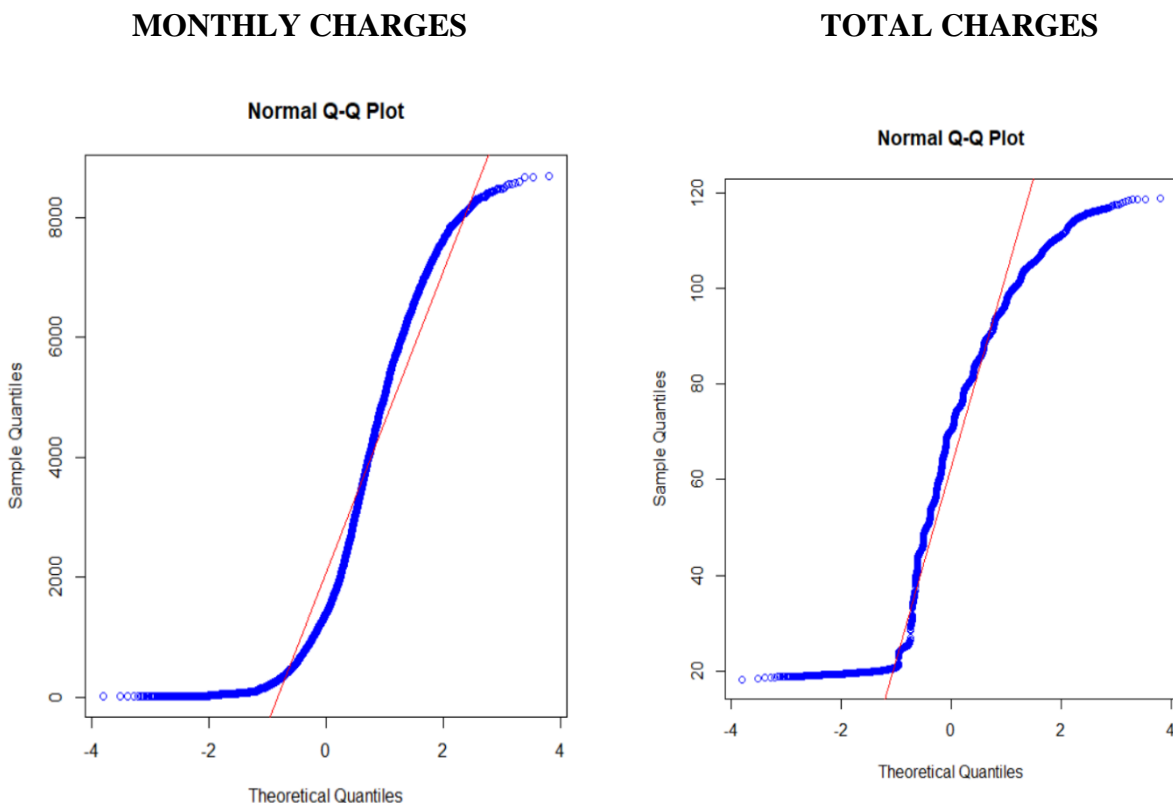
#Plot quantitative data within a grid

```
grid.arrange(p17,p18,p19,ncol=1)
```



The tenure variable is stacked at the tails, so a large proportion of customers have either been had the shortest (1 month) or longest (72 month) tenure. It appears as if the Monthly Charges variable is roughly normally distributed around \$80 per month with a large stack near the lowest rates. The Total Charges variable is positively skewed with a large stack near the lower amounts.

We also check how total charges and Monthly Charges deviate from the Normal Distribution by using QQ plot



From the above two plots we can conclude that the 2 attributes monthly charges and total charges have negative kurtosis effect which is usually seen in the platykurtic distribution which is basically having thinner tails compared to normal distribution this may due to some of extreme negative or positive outcomes.

## Question 6

### 1. We will verify is tenure dependent on the Contract type opted by the customer

As we know that there are more than 2 categories in the contract and the tenure is continuous variable so we will opt for the anova test.

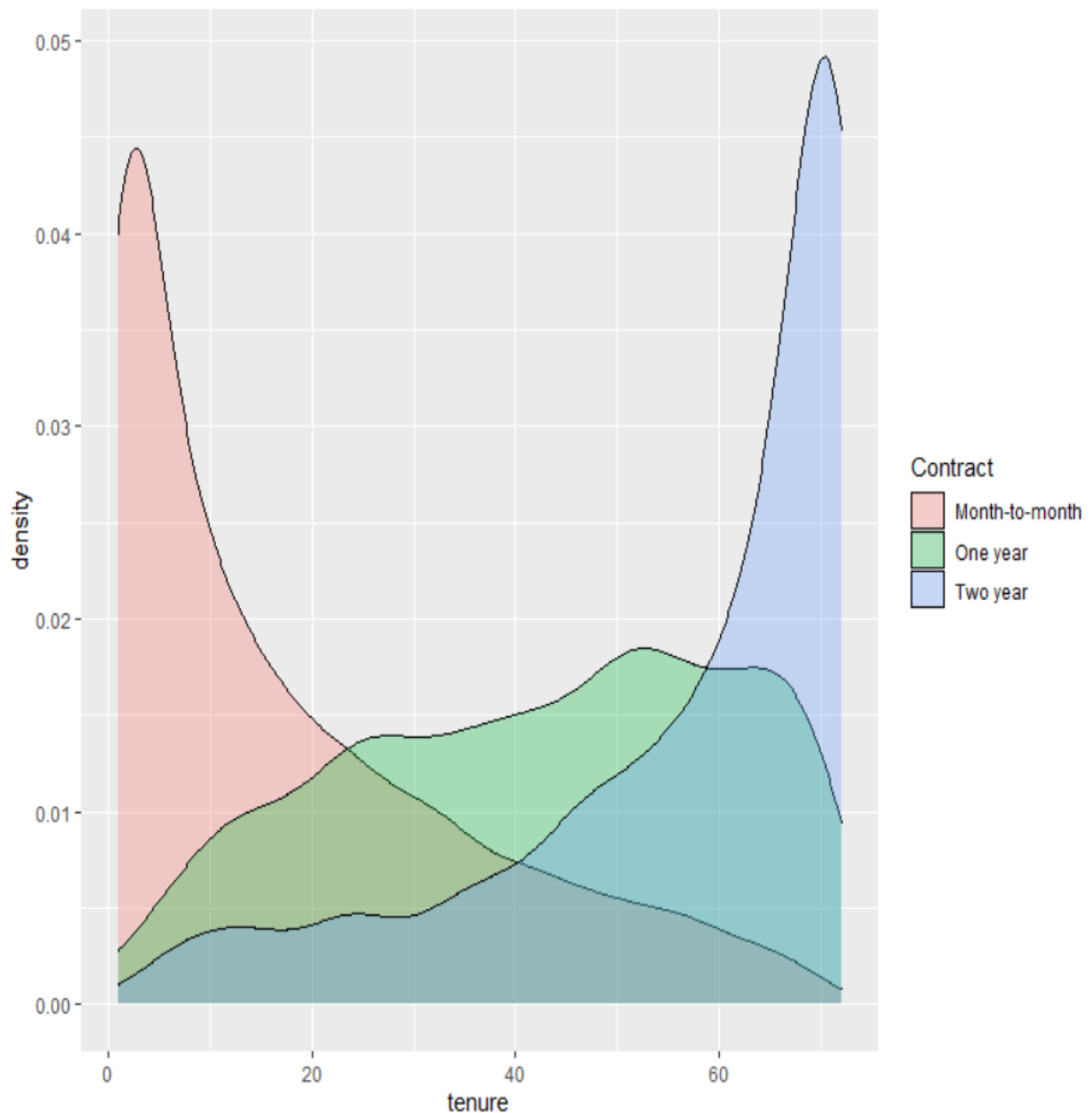
But before applying that we need two main conditions to be satisfied before hand by our sample is that all of them must be normally distributed and according to central limit theorem as  $n \rightarrow \infty$  we can assume normality in distribution.

We can also see through the plot below how the tenure is distributed in different contract types. This can be done by R commands as follows:



```
ptc<-ggplot(churn_clean, aes(x = tenure, fill = Contract)) +geom_density(alpha = .3)
```

ptc



After seeing the above plot, we can somewhat assume that for month-to-month contract our data is positively skewed normal distribution, and negatively skewed for the one year and the two-year contract types.

So, basically to compare population variances of different groups  $> 2$  together we have a test in R named as Bartlett test in which the hypothesis is as follows,

**NULL HYPOTHESIS - The population variances of all the groups are equal**

**ALTERNATE HYPOTHESIS – The population variances of all the groups are not equal.**

```
> bartlett.test(tenure~Contract, data=churn_clean)
```

```
Bartlett test of homogeneity of variances
```

```
data: tenure by Contract
```

```
Bartlett's K-squared = 12.187, df = 2, p-value = 0.002258
```

After running the test on our dataset, we got the following output:

```
> bartlett.test(TotalCharges~PaymentMethod, data=churn_clean)
```

```
Bartlett test of homogeneity of variances
```

```
data: TotalCharges by PaymentMethod
```

```
Bartlett's K-squared = 465.3, df = 3, p-value < 2.2e-16
```

As, the p value is less than 0.05 we can say that we need to reject our null hypothesis and so the population variance are not equal in all these groups.

So, now we cannot apply simple one-way anova test. After a study, we can apply welch one way anova test in our case which is available in R by command `oneway.test()`

So, next step is to apply welchs one-way anova test on our two attributes:

**NULL HYPOTHESIS: Means of all the groups are same**

**ALTERNATE HYPTHESIS: Means of all groups are not same**

```
> oneway.test(tenure~Churn,data=churn_clean, var.equal = FALSE)
```

```
One-way analysis of means (not assuming equal variances)
```

```
data: tenure and Churn
```

```
F = 1223, num df = 1.0, denom df = 4045.5, p-value < 2.2e-16
```

As we can see that p value < 0.05, we reject null hypothesis so we can say means of all groups are not same.

We can also draw an interesting inference from f value that tenure is dependent on the monthly contract because the larger the F value, the more likely it is that the variation caused by the independent variable is real and not due to chance.

## **2. Is total Charges dependent on the Payment Method?**

Again, we need to first check population variances are equal or not to decide whether to apply one way anova or welchs one way anova test with the total charges and the payment methods

**NULL HYPOTHESIS: Population Variances of all groups are equal**

**ALTERNATE HYPOTHESIS: Population Variances of all groups are not equal**

So, we conclude that population variances are not equal and so we need to apply welchs one way anova test and we did it in R by running the command and output is as follows:

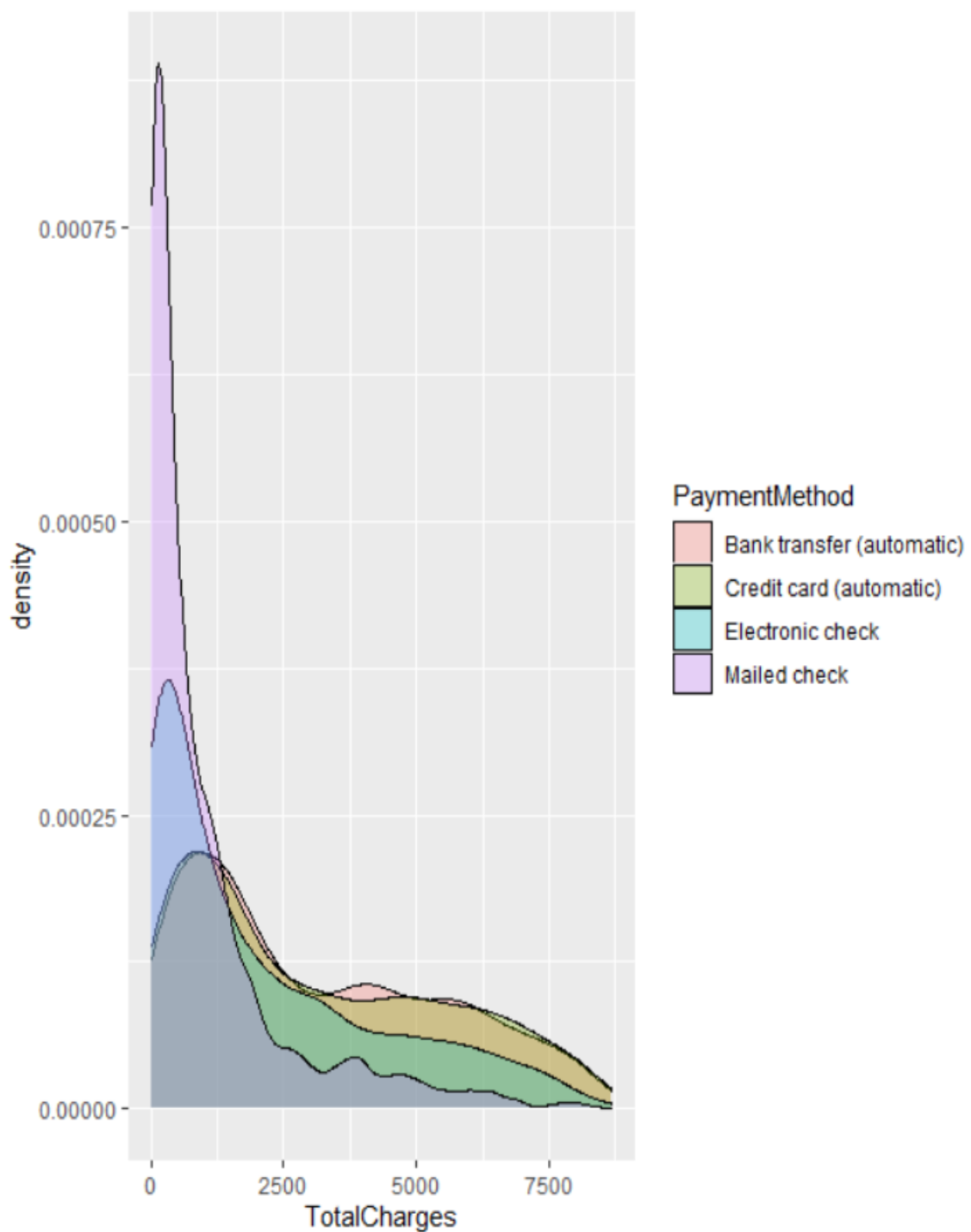
```
> oneway.test(TotalCharges~PaymentMethod,data=churn_clean, var.equal = FALSE)
```

One-way analysis of means (not assuming equal variances)

data: TotalCharges and PaymentMethod

F = 437.07, num df = 3.0, denom df = 3637.2, p-value < 2.2e-16

From the above test we can draw the conclusion that total Charges is dependent on the payment methods. Similarly, we can also visualize how total Charges are distributed according to different payment methods which we can see it with help of the distribution plot in R



### 3. Is churn dependent on the Monthly Charges?

As, we know that churn is having 2 categories and monthly charges is continuous variable so we will use t test.

But before applying we need to test whether the population variances of the two samples is equal or not to decide the type of t test.

```
> var.test(MonthlyCharges ~ Churn, churn_clean, alternative = "two.sided")
```

F test to compare two variances

```
data: MonthlyCharges by Churn
F = 1.5892, num df = 5162, denom df = 1868, p-value < 2.2e-16
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 1.473565 1.711558
sample estimates:
ratio of variances
 1.589167
```

From this we can conclude that the population variances of the two samples is not equal and so we will go for Welch's t test.

**NULL HYPOTHESIS: Churn is independent of the Monthly Charges**

**ALTERNATE HYPOTHESIS: Churn is dependent on the Monthly Charges**

```
result <- t.test(MonthlyCharges~Churn,data=churn_clean)
```

```
print(result)
```

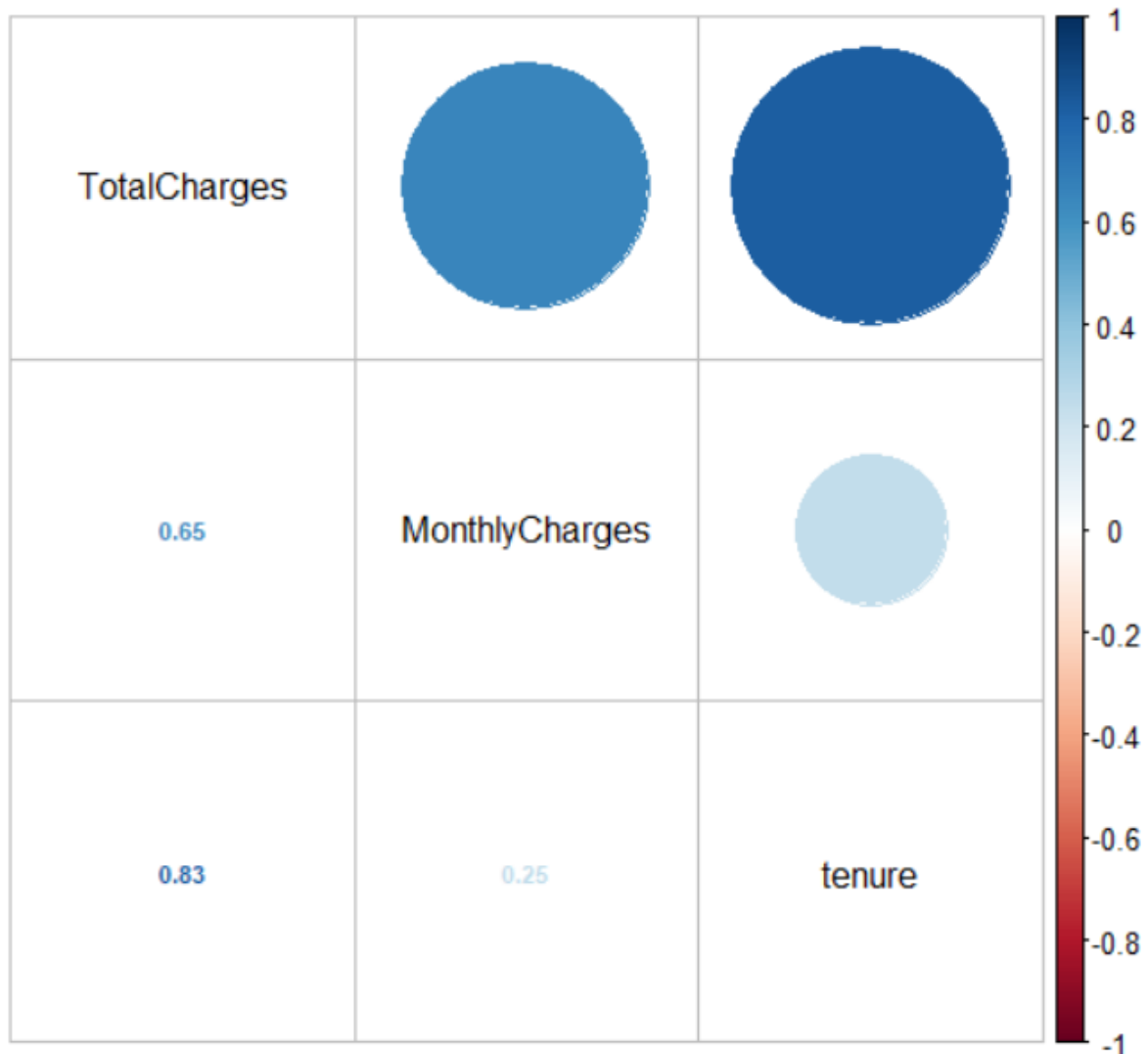
Welch Two Sample t-test

```
data: MonthlyCharges by Churn
t = -18.341, df = 4139.7, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -14.53786 -11.72998
sample estimates:
mean in group No mean in group Yes
 61.30741          74.44133
```

## QUESTION 7: COMPARE THE CORRELATION AMONG THE CONTINUOUS VARIABLES THAT ARE TENURE, TOTAL CHARGES AND MONTHLY CHARGES

So, to accomplish above task we will make correlation plot in R

```
churn_clean %>% dplyr::select (TotalCharges, MonthlyCharges, tenure) %>% cor()  
%>% corrplot.mixed(upper = "circle", tl.col = "black", number.cex = 0.7)
```



The plot shows high correlations between Total charges & tenure and between Total Charges and Monthly Charges. Pay attention to these variables while training models later. Multicollinearity does not reduce the predictive power or reliability of the model as a whole, at least within the sample data set. But it affects calculations regarding individual predictors. Assume a number of linearly correlated covariates/features present in the data set and Random Forest as the method. Obviously, random selection per node may pick only (or mostly) collinear features which may/will result in a poor split, and this can happen repeatedly, thus negatively affecting the performance.

## MODELS TRAINING AND PREDICTIONS

In order to assess the performance of our various modelling techniques, we can split the data into training and test subsets. We will model the training data and use these model parameters to make predictions with the test data. Let's call these data subsets dtrain and dtest.

We will randomly sample from the entire sample to create these subsets. The 'set.seed()' function argument can be changed in order to reset the random number generator used for sampling. The training subset will be roughly 70% of the original sample, with the remaining being the test subset.

### Splitting dataset into training and testing

```
set.seed(56)

split_train_test <- createDataPartition(churn_clean$Churn,p=0.7,list=FALSE)

dtrain<- churn_clean[split_train_test,]

dtest<- churn_clean[-split_train_test,]

# Remove Total Charges from the training dataset

dtrain <- dtrain[,-19]

dtest <- dtest[,-19]
```

## QUESTION 8

### APPLY DECISION TREE AND FIND ITS ACCURACY ON THE DATASET?

```
tr_fit <- rpart(Churn ~., data = dtrain, method="class")

rpart.plot(tr_fit)

tr_prob1 <- predict(tr_fit, dtest)

tr_pred1 <- ifelse(tr_prob1[,2] > 0.5,"Yes","No")

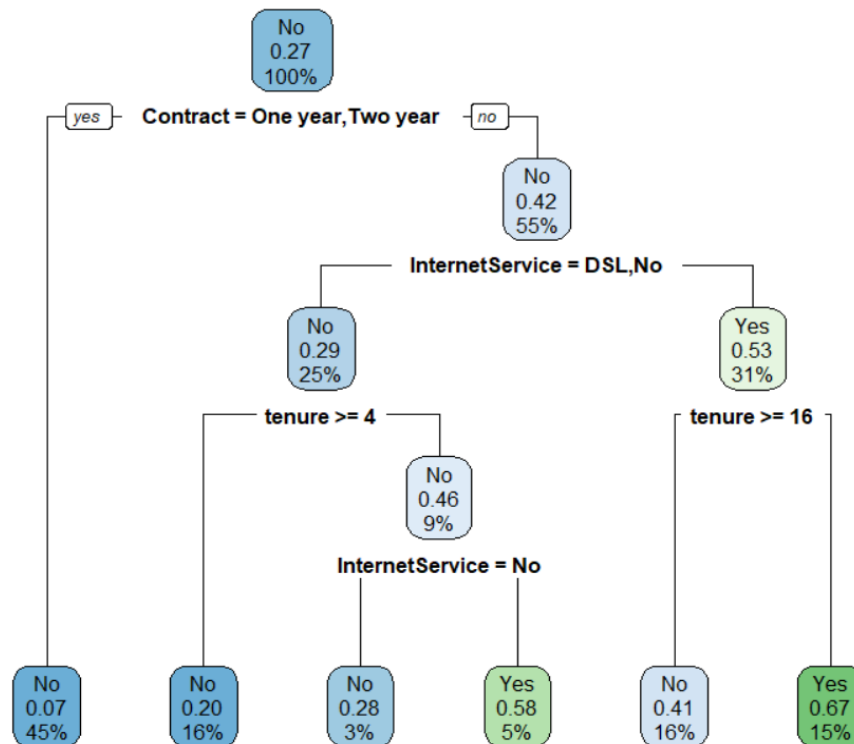
table(Predicted = tr_pred1, Actual = dtest$Churn)

tr_prob2 <- predict(tr_fit, dtrain)

tr_pred2 <- ifelse(tr_prob2[,2] > 0.5,"Yes","No")

tr_tab1 <- table(Predicted = tr_pred2, Actual = dtrain$Churn)

tr_tab2 <- table(Predicted = tr_pred1, Actual = dtest$Churn)
```



The contract variable is the most important. Customers with month-to-month contracts are more likely to churn. Customers with DSL internet service are less likely to churn. Customers who have stayed longer than 15 months are less likely to churn. Now let's assess the prediction accuracy of the decision tree model by investigating how well it predicts churn in the test subset. We will begin with the confusion matrix, which is a useful display of classification accuracy. It displays the following information.

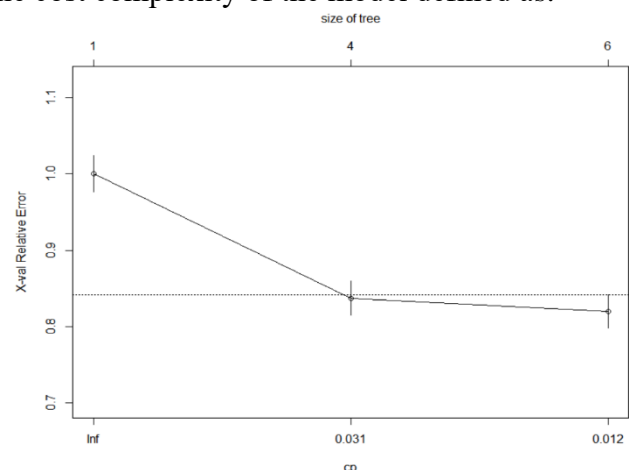
Sometimes, due to overfitting we may need to prune our tree and for this we have executed the command for our datasets:

```
printcp(tr_fit), plotcp(tr_fit)
```

So we for this we have the complexity parameter (cp) in rpart is the minimum improvement in the model needed at each node. It's based on the cost complexity of the model defined as:

$$\sum_{\text{Terminal Nodes}} \text{Misclass}_i + \lambda * (\text{Splits})$$

- For the given tree, add up the misclassification at every terminal node.
- Then multiply the number of splits time a penalty term (lambda) and add it to the total misclassification.
- The lambda is determined through cross-validation and not reported in R.
- The cp we see using `printcp()` is the scaled version of lambda over the misclassification rate of the overall data.



Now we will make a confusion matrix on the testing data and find out the accuracy of the model

```
# Test
```

```
confusionMatrix(
```

```
  as.factor(tr_pred1),
```

```
  as.factor(dtest$Churn),
```

```
  positive = "Yes"
```

```
)
```

```
tr_acc <- sum(diag(tr_tab2))/sum(tr_tab2)
```

```
tr_acc
```

### Confusion Matrix and Statistics

	Reference	
Prediction	No	Yes
No	1406	284
Yes	142	276

Accuracy : 0.7979

95% CI : (0.7801, 0.8149)

No Information Rate : 0.7343

P-Value [Acc > NIR] : 6.398e-12

Kappa : 0.4364

Mcnemar's Test P-Value : 8.405e-12

Sensitivity : 0.4929

Specificity : 0.9083

Pos Pred Value : 0.6603

Neg Pred Value : 0.8320

Prevalence : 0.2657

Detection Rate : 0.1309

Detection Prevalence : 0.1983

Balanced Accuracy : 0.7006

'Positive' Class : Yes

```
>
```

```
> tr_acc <- sum(diag(tr_tab2))/sum(tr_tab2)
```

```
> tr_acc
```

```
[1] 0.7979127
```

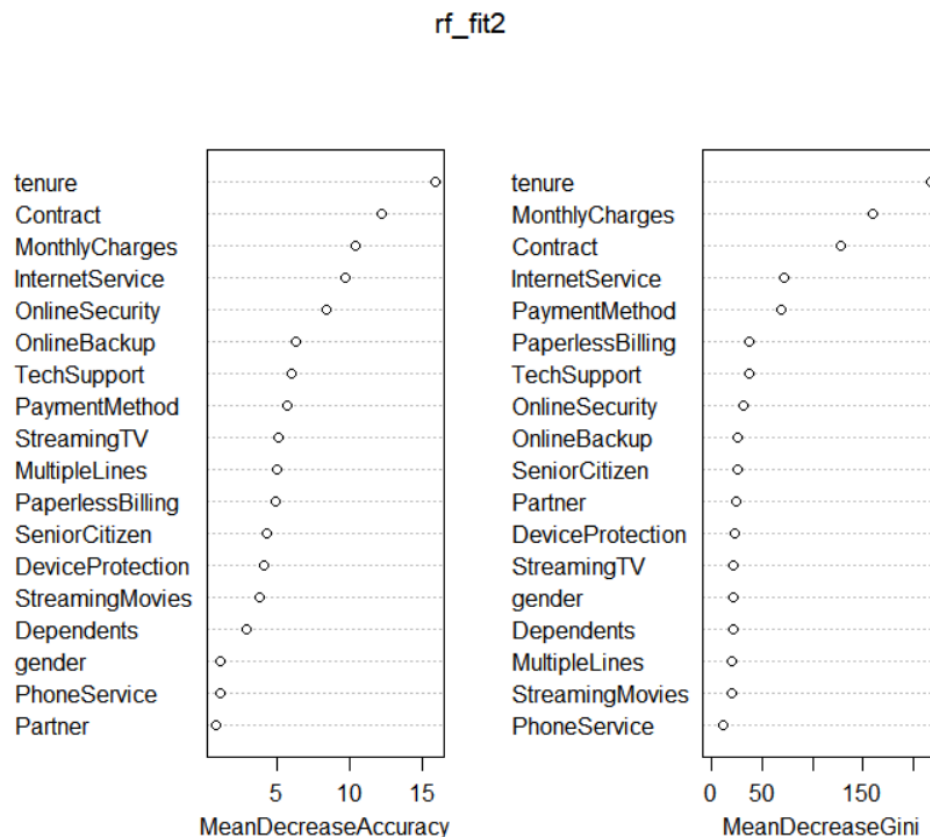


## QUESTION 9: TRAIN A RANDOM FOREST ON YOUR DATASET AND FIND ITS ACCURACY ON THE TEST DATASET

```
rf_fit2 <- randomForest(as.factor(Churn) ~., data = dtrain, ntree = 75, mtry = 2, importance = TRUE, proximity = TRUE)
```

**Display variable importance from random tree**

```
varImpPlot(rf_fit2)
```



Similar to the decision tree, this random forest model has identified contract status and tenure length as important predictors for churn.

Now, as we have trained our random forest on our training dataset, we will plot confusion matrix and find accuracy of it on test dataset

```
rf_pred1 <- predict(rf_fit2, dtest)
```

```
rf_tab2 <- table(Predicted = rf_pred1, Actual = dtest$Churn)
```

```
# Test
```

```
confusionMatrix(
```

```

as.factor(rf_pred1),

as.factor(dtest$Churn),

positive = "Yes"

)

rf_acc <- sum(diag(rf_tab2))/sum(rf_tab2)

rf_acc

pred1=predict(rf_fit2,dtest,type = "prob")

library(ROCR)

perf = prediction(pred1[,2], dtest$Churn)

auc = performance(perf, "auc")

pred3 = performance(perf, "tpr","fpr")

plot(pred3,main="ROC Curve for Random Forest",col=2,lwd=2)

abline(a=0,b=1,lwd=2,lty=2,col="gray")

```

#### Confusion Matrix and Statistics

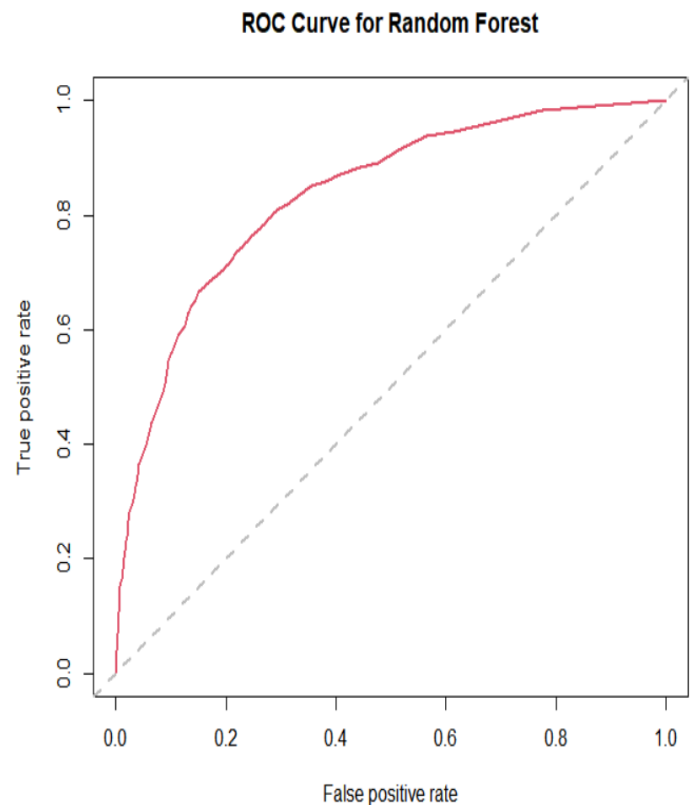
	Reference	
Prediction	No	Yes
No	1424	293
Yes	124	267

Accuracy : 0.8022  
 95% CI : (0.7845, 0.819)  
 No Information Rate : 0.7343  
 P-Value [Acc > NIR] : 2.183e-13  
  
 Kappa : 0.439  
  
 Mcnemar's Test P-Value : < 2.2e-16  
  
 Sensitivity : 0.4768  
 Specificity : 0.9199  
 Pos Pred Value : 0.6829  
 Neg Pred Value : 0.8294  
 Prevalence : 0.2657  
 Detection Rate : 0.1267  
 Detection Prevalence : 0.1855  
 Balanced Accuracy : 0.6983  
  
 'Positive' Class : Yes

```

> rf_acc <- sum(diag(rf_tab2))/sum(rf_tab2)
> rf_acc
[1] 0.8021822

```



## QUESTION 10: TRAIN A LOGISTIC REGRESSION MODEL ON YOUR DATASET AND FIND ITS ACCURACY ON THE TEST DATASET

```
lr_fit <- glm(as.factor(Churn) ~., data = dtrain,family=binomial(link='logit'))
```

```
summary(lr_fit)
```

```
Call:
glm(formula = as.factor(Churn) ~ ., family = binomial(link = "logit"),
    data = dtrain)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9796  -0.6746  -0.2913   0.7076   3.1075

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.441724   0.961202   0.460 0.645837
genderMale    -0.007747   0.077510  -0.100 0.920383
SeniorCitizen  0.235996   0.100570   2.347 0.018946 *
PartnerYes    -0.071058   0.092374  -0.769 0.441748
DependentsYes -0.211223   0.106934  -1.975 0.048238 *
tenure        -0.032358   0.002813 -11.505 < 2e-16 ***
PhoneServiceYes -0.165520   0.774096  -0.214 0.830685
MultipleLinesYes 0.402807   0.210212   1.916 0.055341 .
InternetServiceFiber optic 1.265800   0.951204   1.331 0.183277
InternetServiceNo -1.445403   0.964685  -1.498 0.134051
OnlineSecurityYes -0.219603   0.211715  -1.037 0.299614
OnlineBackupYes -0.085143   0.208896  -0.408 0.683578
DeviceProtectionYes 0.057192   0.209502   0.273 0.784860
TechSupportYes -0.296010   0.215162  -1.376 0.168898
StreamingTVYes  0.427724   0.388682   1.100 0.271138
StreamingMoviesYes 0.440124   0.389786   1.129 0.258837
ContractOne year -0.616267   0.127595  -4.830 1.37e-06 ***
ContractTwo year -1.331978   0.210434  -6.330 2.46e-10 ***
PaperlessBillingYes 0.306490   0.088674   3.456 0.000548 ***
PaymentMethodCredit card (automatic) -0.202668   0.135368  -1.497 0.134350
PaymentMethodElectronic check  0.256878   0.112405   2.285 0.022296 *
PaymentMethodMailed check -0.148949   0.136194  -1.094 0.274106
MonthlyCharges -0.017919   0.037834  -0.474 0.635769
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 5702.8  on 4923  degrees of freedom
Residual deviance: 4104.9  on 4901  degrees of freedom
AIC: 4150.9

Number of Fisher Scoring iterations: 6
```

By examining the significance values, we see similar predictor variables of importance. Tenure length and contract status have the lowest p-values and can be identified as the best predictors of customer churn.

Now, we will again make confusion matrix and check accuracy of our logistic regression model on the dataset

```
lr_prob1 <- predict(lr_fit, dtest, type="response")
```

```
lr_pred1 <- ifelse(lr_prob1 > 0.5,"Yes","No")
```

```
table(Predicted = lr_pred1, Actual = dtest$Churn)
```

```
lr_tab2 <- table(Predicted = lr_pred1, Actual = dtest$Churn)
```

```
# Test
confusionMatrix(
  as.factor(lr_pred1),
  as.factor(dtest$Churn),
  positive = "Yes"
)
lr_acc <- sum(diag(lr_tab2))/sum(lr_tab2)
lr_acc
```

ROC Curve for the logistic regression is one of the other evaluations of the model we will do in our testing.

```
library(pROC)
res.roc <- roc(dtest$Churn, lr_prob1)
plot.roc(res.roc, print.auc = TRUE)
```

#### Confusion Matrix and Statistics

```

      Reference
Prediction No  Yes
No      1404  249
Yes      144  311

      Accuracy : 0.8136
      95% CI   : (0.7963, 0.83)
No Information Rate : 0.7343
P-Value [Acc > NIR] : < 2.2e-16

      Kappa : 0.4918

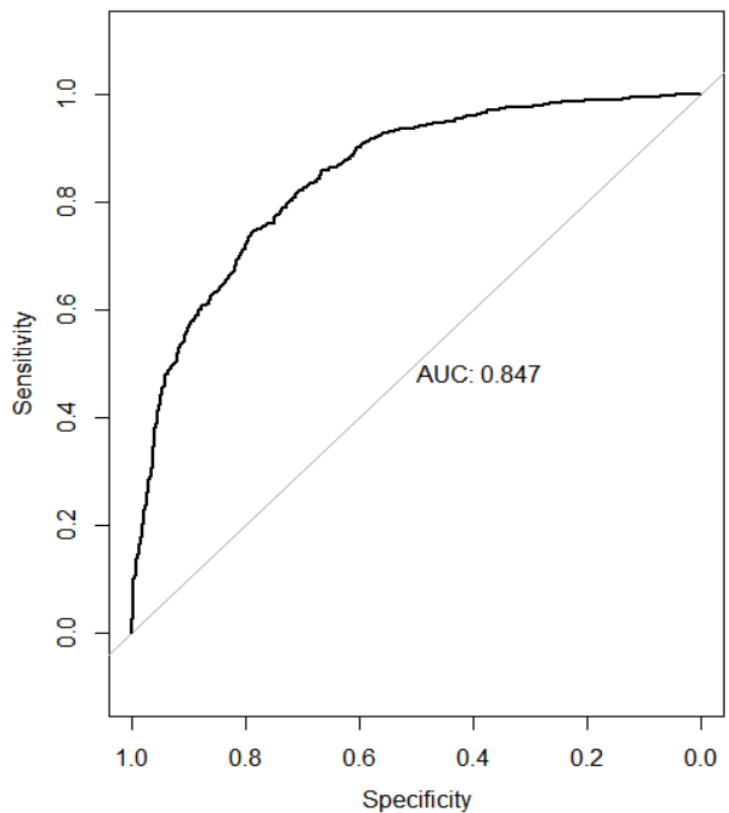
McNemar's Test P-Value : 1.553e-07

      Sensitivity : 0.5554
      Specificity : 0.9070
      Pos Pred Value : 0.6835
      Neg Pred Value : 0.8494
      Prevalence : 0.2657
      Detection Rate : 0.1475
      Detection Prevalence : 0.2158
      Balanced Accuracy : 0.7312

      'Positive' Class : Yes

>
> lr_acc <- sum(diag(lr_tab2))/sum(lr_tab2)
> lr_acc
[1] 0.8135674
```

#### MODEL ACCURACY



#### LOGISTIC REGRESSION ROC CURVE

Lastly, we will compare some models together for classification on our dataset:

```
control <- trainControl(method="repeatedcv", number=10, repeats=3)
set.seed(7)

fit.cart <- train(Churn~., data=dtrain, method="rpart", trControl=control)
set.seed(7)

fit.lor <- train(Churn~., data=dtrain, method="glm", trControl=control)
#RF
set.seed(7)

fit.rf <- train(Churn~., data=dtrain, method="rf", trControl=control)
set.seed(7)

fit.knn <- train(Churn~., data=dtrain, method="knn", trControl=control)
results <- resamples(list(CART=fit.cart,RF=fit.rf, LOR=fit.lor, KNN=fit.knn))
scales <- list(x=list(relation="free"), y=list(relation="free"))
densityplot(results, scales=scales, pch = "|")
summary(results)
```

