Assignment 1

Arundhati Das

06/09/2020

Case Study 1A – Telco Churn

1. Introduction – The client is a telecom company. They have an issue with customer churn. Currently, they want to improve their customer retention efforts. For this, they have engaged on a churn prediction exercise.

2. Challenge – The customer has stated that the objective of the churn prediction exercise is to predict churn. That is, the model should be able to identify (based on past data), who is going to churn in the next time period.

3. Your role – You are the lead data scientist for the project. In this phase, you have to confirm that you understand the business requirements (business understanding) as well as the data (data understanding) for the project.

4. Details of the data – The customer has shared the data with your team. The data is in the form of a text file . The description of the various fields are given and the customer has stated that the target variable is "churn".

    (a) On the basis of the problem statement and the data provided, carry out the following steps for the client. The response for this exercise (Data Preprocessing and Exploratory Data Analysis) will be a report, with separate sections for the deliverables below.

(i) Business Understanding – Goals & Success Criterion.

(ii) Data Understanding – Data Exploration & Quality Report.

(b) Perform initial analysis (Variable Selection & Dimension Reduction) to identify possible variables that could be impacting churn. Provide a report of the same.

Introduction:

Customer churn occurs when customers or subscribers stop doing business with a company or service.It is also referred as loss of clients or customers. One industry in which churn rates are particularly useful is the telecommunications industry, because most customers have multiple options from which to choose within a geographic location.In many geographical areas, several companies are competing for customers, making it easy for people to transfer from one provider to another.

Churn rates are often used to indicate the strength of a company's customer service division and its overall growth prospects. Lower churn rates suggest a company is, or will be, in a better or stronger competitive state.

Data Preprocessing: A Telco Churn Data is being provided which contains 1000 rows (customers) and 42 columns (features). The “churn” column is our target.Some of the features which were in 0-1 form,I replaced them with No & Yes respectively.In Gender column,I replaced 0 & 1 with Male & Female respectively.

```
#reading data
df <- read.delim("C:/Users/AK DAS/Desktop/telco.txt")
head(df)

##   region tenure age marital address income ed employ retire gender reside
## 1      2     13  44     Yes       9     64  4      5     No   Male      2
## 2      3     11  33     Yes       7    136  5      5     No   Male      6
## 3      3     68  52     Yes      24    116  1     29     No Female      2
## 4      2     33  33      No      12     33  2      0     No Female      1
## 5      2     23  30     Yes       9     30  1      2     No   Male      4
## 6      2     41  39      No      17     78  2     16     No Female      1
##   tollfree equip callcard wireless longmon tollmon equipmon cardmon
wiremon
## 1       No    No      Yes       No    3.70    0.00        0    7.50
0.0
## 2      Yes    No      Yes      Yes    4.40   20.75        0   15.25
35.7
## 3      Yes    No      Yes       No   18.15   18.00        0   30.25
0.0
## 4       No    No       No       No    9.45    0.00        0    0.00
0.0
## 5       No    No       No       No    6.30    0.00        0    0.00
0.0
## 6      Yes    No      Yes       No   11.80   19.25        0   13.50
0.0
##   longten tollten equipten cardten wireten multline voice pager internet
callid
## 1   37.45    0.00        0     110    0.00       No    No    No       No
No
## 2   42.00  211.45        0     125  380.35       No   Yes   Yes       No
Yes
## 3 1300.60 1247.20        0    2150    0.00       No    No    No       No
Yes
## 4  288.80    0.00        0       0    0.00       No    No    No       No
No
## 5  157.05    0.00        0       0    0.00       No    No    No       No
Yes
## 6  487.40  798.40        0     570    0.00       No    No    No       No
Yes
##   callwait forward confer ebill  loglong  logtoll logequi  logcard
logwire
## 1       No     Yes     No    No 1.308333       NA      NA 2.014903
NA
## 2      Yes     Yes    Yes    No 1.481605 3.032546      NA 2.724580
3.575151
```

```
## 3      Yes      No    Yes    No 2.898671 2.890372      NA 3.409496
NA
## 4       No      No     No    No 2.246015       NA      NA       NA
NA
## 5       No     Yes    Yes    No 1.840550       NA      NA       NA
NA
## 6      Yes      No     No    No 2.468100 2.957511      NA 2.602690
NA
##      lninc custcat churn
## 1 4.158883       1   Yes
## 2 4.912655       4   Yes
## 3 4.753590       3    No
## 4 3.496508       1   Yes
## 5 3.401197       3    No
## 6 4.356709       3    No
```

```
#View structure and summary of the data
dim(df)
```

```
## [1] 1000   42
```

```
str(df)
```

```
## 'data.frame':    1000 obs. of  42 variables:
##  $ region  : int  2 3 3 2 2 2 3 2 3 1 ...
##  $ tenure  : int  13 11 68 33 23 41 45 38 45 68 ...
##  $ age     : int  44 33 52 33 30 39 22 35 59 41 ...
##  $ marital : Factor w/ 2 levels "No","Yes": 2 2 2 1 2 1 2 1 2 2 ...
##  $ address : int  9 7 24 12 9 17 2 5 7 21 ...
##  $ income  : int  64 136 116 33 30 78 19 76 166 72 ...
##  $ ed      : int  4 5 1 2 1 2 2 2 4 1 ...
##  $ employ  : int  5 5 29 0 2 16 4 10 31 22 ...
##  $ retire  : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ gender  : Factor w/ 2 levels "Female","Male": 2 2 1 1 2 1 1 2 2 2 ...
##  $ reside  : int  2 6 2 1 4 1 5 3 5 3 ...
##  $ tollfree: Factor w/ 2 levels "No","Yes": 1 2 2 1 1 2 1 2 2 1 ...
##  $ equip   : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 2 1 1 ...
##  $ callcard: Factor w/ 2 levels "No","Yes": 2 2 2 1 1 2 2 2 2 2 ...
##  $ wireless: Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 2 1 1 ...
##  $ longmon : num  3.7 4.4 18.15 9.45 6.3 ...
##  $ tollmon : num  0 20.8 18 0 0 ...
##  $ equipmon: num  0 0 0 0 0 0 0 50.1 0 0 ...
##  $ cardmon : num  7.5 15.2 30.2 0 0 ...
##  $ wiremon : num  0 35.7 0 0 0 0 0 64.9 0 0 ...
##  $ longten : num  37.5 42 1300.6 288.8 157.1 ...
##  $ tollten : num  0 211 1247 0 0 ...
##  $ equipten: num  0 0 0 0 0 ...
##  $ cardten : num  110 125 2150 0 0 ...
##  $ wireten : num  0 380 0 0 0 ...
##  $ multline: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 2 2 2 ...
##  $ voice   : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 2 1 1 ...
```

```
##  $ pager   : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 2 1 1 ...
##  $ internet: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 2 1 1 ...
##  $ callid  : Factor w/ 2 levels "No","Yes": 1 2 2 1 2 2 1 2 2 1 ...
##  $ callwait: Factor w/ 2 levels "No","Yes": 1 2 2 1 1 2 2 2 2 1 ...
##  $ forward : Factor w/ 2 levels "No","Yes": 2 2 1 1 2 1 1 2 2 1 ...
##  $ confer  : Factor w/ 2 levels "No","Yes": 1 2 2 1 2 1 1 2 2 1 ...
##  $ ebill   : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 2 1 1 ...
##  $ loglong : num  1.31 1.48 2.9 2.25 1.84 ...
##  $ logtoll : num  NA 3.03 2.89 NA NA ...
##  $ logequi : num  NA NA NA NA NA ...
##  $ logcard : num  2.01 2.72 3.41 NA NA ...
##  $ logwire : num  NA 3.58 NA NA NA ...
##  $ lninc   : num  4.16 4.91 4.75 3.5 3.4 ...
##  $ custcat : int  1 4 3 1 3 3 2 4 3 2 ...
##  $ churn   : Factor w/ 2 levels "No","Yes": 2 2 1 2 1 1 2 1 1 1 ...
```

```
summary(df)
```

```
##      region          tenure           age        marital      address     
##  Min.   :1.000   Min.   : 1.00   Min.   :18.00   No :505   Min.   : 0.00  
##  1st Qu.:1.000   1st Qu.:17.00   1st Qu.:32.00   Yes:495   1st Qu.: 3.00  
##  Median :2.000   Median :34.00   Median :40.00             Median : 9.00  
##  Mean   :2.022   Mean   :35.53   Mean   :41.68             Mean   :11.55  
##  3rd Qu.:3.000   3rd Qu.:54.00   3rd Qu.:51.00             3rd Qu.:18.00  
##  Max.   :3.000   Max.   :72.00   Max.   :77.00             Max.   :55.00  
##                                                                           
##      income             ed           employ      retire       gender     
##  Min.   :   9.00   Min.   :1.000   Min.   : 0.00   No :953   Female:517  
##  1st Qu.:  29.00   1st Qu.:2.000   1st Qu.: 3.00   Yes: 47   Male  :483  
##  Median :  47.00   Median :3.000   Median : 8.00                         
##  Mean   :  77.53   Mean   :2.671   Mean   :10.99                         
##  3rd Qu.:  83.00   3rd Qu.:4.000   3rd Qu.:17.00                         
##  Max.   :1668.00   Max.   :5.000   Max.   :47.00                         
##                                                                          
##      reside      tollfree  equip     callcard  wireless     longmon      
##  Min.   :1.000   No :526   No :614   No :322   No :704   Min.   : 0.900  
##  1st Qu.:1.000   Yes:474   Yes:386   Yes:678   Yes:296   1st Qu.: 5.200  
##  Median :2.000                                         Median : 8.525  
##  Mean   :2.331                                         Mean   :11.723  
##  3rd Qu.:3.000                                         3rd Qu.:14.412  
##  Max.   :8.000                                         Max.   :99.950  
##                                                                        
##     tollmon          equipmon        cardmon          wiremon      
##  Min.   :  0.00   Min.   : 0.00   Min.   :  0.00   Min.   :  0.00  
##  1st Qu.:  0.00   1st Qu.: 0.00   1st Qu.:  0.00   1st Qu.:  0.00  
##  Median :  0.00   Median : 0.00   Median : 12.00   Median :  0.00  
##  Mean   : 13.27   Mean   :14.22   Mean   : 13.78   Mean   : 11.58  
##  3rd Qu.: 24.25   3rd Qu.:31.48   3rd Qu.: 20.50   3rd Qu.: 24.71  
##  Max.   :173.00   Max.   :77.70   Max.   :109.25   Max.   :111.95  
##                                                                    
```

```
##     longten           tollten          equipten         cardten
##  Min.   :   0.90   Min.   :   0.0   Min.   :   0.0   Min.   :   0.0
##  1st Qu.:  90.14   1st Qu.:   0.0   1st Qu.:   0.0   1st Qu.:   0.0
##  Median : 285.48   Median :   0.0   Median :   0.0   Median : 332.5
##  Mean   : 574.05   Mean   : 551.3   Mean   : 465.6   Mean   : 605.8
##  3rd Qu.: 755.02   3rd Qu.: 846.9   3rd Qu.: 579.5   3rd Qu.: 910.0
##  Max.   :7257.60   Max.   :5916.0   Max.   :5028.6   Max.   :7515.0
##
##     wireten       multline  voice     pager     internet  callid
callwait
##  Min.   :   0.0   No :525   No :696   No :739   No :632   No :519   No
:515
##  1st Qu.:   0.0   Yes:475   Yes:304   Yes:261   Yes:368   Yes:481
Yes:485
##  Median :   0.0
##  Mean   : 442.7
##  3rd Qu.: 316.5
##  Max.   :7856.9
##
##  forward   confer    ebill        loglong           logtoll
##  No :507   No :498   No :629   Min.   :-0.1054   Min.   :1.749
##  Yes:493   Yes:502   Yes:371   1st Qu.: 1.6487   1st Qu.:2.970
##                                Median : 2.1430   Median :3.209
##                                Mean   : 2.1821   Mean   :3.240
##                                3rd Qu.: 2.6681   3rd Qu.:3.489
##                                Max.   : 4.6047   Max.   :5.153
##                                                  NA's   :525
##     logequi         logcard         logwire          lninc
##  Min.   :2.734   Min.   :1.012   Min.   :2.701   Min.   :2.197
##  1st Qu.:3.368   1st Qu.:2.464   1st Qu.:3.333   1st Qu.:3.367
##  Median :3.572   Median :2.848   Median :3.595   Median :3.850
##  Mean   :3.568   Mean   :2.854   Mean   :3.598   Mean   :3.957
##  3rd Qu.:3.757   3rd Qu.:3.209   3rd Qu.:3.862   3rd Qu.:4.419
##  Max.   :4.353   Max.   :4.694   Max.   :4.718   Max.   :7.419
##  NA's   :614     NA's   :322     NA's   :704
##     custcat      churn
##  Min.   :1.000   No :726
##  1st Qu.:1.000   Yes:274
##  Median :3.000
##  Mean   :2.487
##  3rd Qu.:3.000
##  Max.   :4.000
##
```

```
#To view the column names
names(df)
```

```
##  [1] "region"   "tenure"   "age"      "marital"  "address"  "income"
##  [7] "ed"       "employ"   "retire"   "gender"   "reside"   "tollfree"
## [13] "equip"    "callcard" "wireless" "longmon"  "tollmon"  "equipmon"
```

```
## [19] "cardmon"  "wiremon"  "longten"  "tollten"  "equipten" "cardten"
## [25] "wireten"  "multline" "voice"    "pager"    "internet" "callid"
## [31] "callwait" "forward"  "confer"   "ebill"    "loglong"  "logtoll"
## [37] "logequi"  "logcard"  "logwire"  "lninc"    "custcat"  "churn"
```

We use sapply to check the number if missing values in each columns.

```
#checking for the missing values
sapply(df, function(x) sum(is.na(x)))
```

```
##   region   tenure      age  marital  address   income       ed   employ
##        0        0        0        0        0        0        0        0
##   retire   gender   reside tollfree    equip callcard wireless  longmon
##        0        0        0        0        0        0        0        0
##  tollmon equipmon  cardmon  wiremon  longten  tollten equipten  cardten
##        0        0        0        0        0        0        0        0
##  wireten multline    voice    pager internet   callid callwait  forward
##        0        0        0        0        0        0        0        0
##   confer    ebill  loglong  logtoll  logequi  logcard  logwire    lninc
##        0        0        0      525      614      322      704        0
##  custcat    churn
##        0        0
```

I found that there are some missing values in logtoll,logequi,logcard & logcard columns. I removed all rows with missing values.

```
#removing all rows with missing values
df <- df[complete.cases(df), ]
#no missing values
sapply(df, function(x) sum(is.na(x)))
```

```
##   region   tenure      age  marital  address   income       ed   employ
##        0        0        0        0        0        0        0        0
##   retire   gender   reside tollfree    equip callcard wireless  longmon
##        0        0        0        0        0        0        0        0
##  tollmon equipmon  cardmon  wiremon  longten  tollten equipten  cardten
##        0        0        0        0        0        0        0        0
##  wireten multline    voice    pager internet   callid callwait  forward
##        0        0        0        0        0        0        0        0
##   confer    ebill  loglong  logtoll  logequi  logcard  logwire    lninc
##        0        0        0        0        0        0        0        0
##  custcat    churn
##        0        0
```

```
#min & max tenure
min(df$tenure); max(df$tenure)
```

```
## [1] 2
```

```
## [1] 72
```

Since the minimum tenure is 2 months and maximum tenure is 72 months, I grouped them into five tenure groups: “0–12 Month”, “12–24 Month”, “24–48 Months”, “48–60 Month”, “> 60 Month”

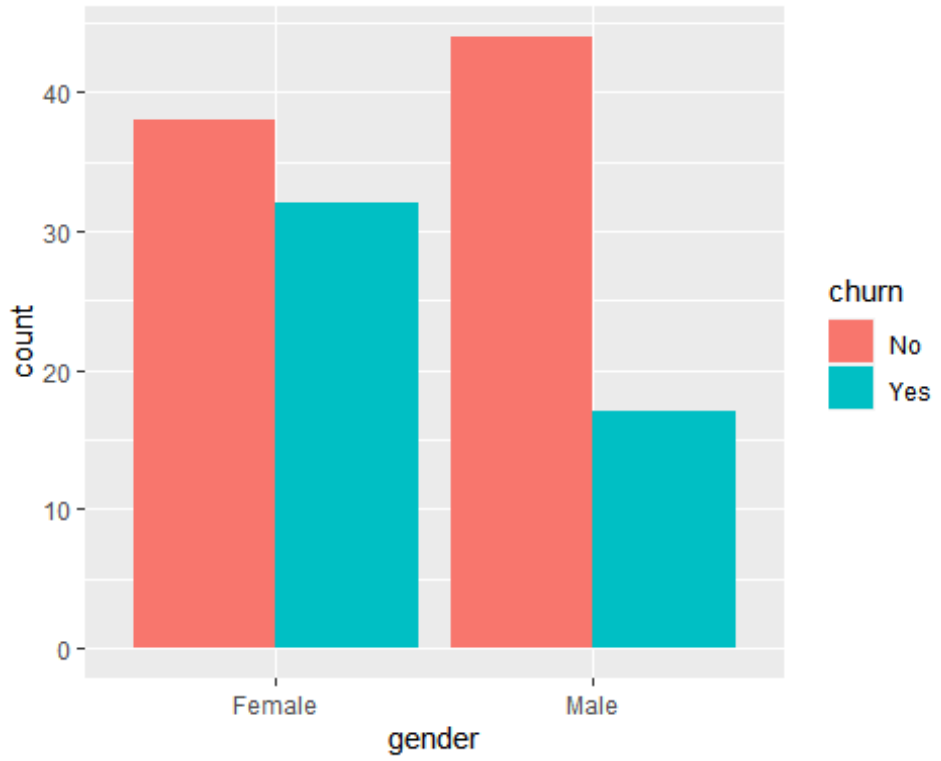
```
#grouping them into five tenure groups: “0-12 Month”, “12-24 Month”, “24-48 Months”, “48-60 Month”, “> 60 Month”
group_tenure <- function(tenure){
  if (tenure >= 0 & tenure <= 12){
    return('0-12 Month')
  }else if(tenure > 12 & tenure <= 24){
    return('12-24 Month')
  }else if (tenure > 24 & tenure <= 48){
    return('24-48 Month')
  }else if (tenure > 48 & tenure <=60){
    return('48-60 Month')
  }else if (tenure > 60){
    return('> 60 Month')
  }
}
df$tenure_group <- sapply(df$tenure,group_tenure)
df$tenure_group <- as.factor(df$tenure_group)
df$tenure_group
```

```
##   [1] 24-48 Month 24-48 Month 48-60 Month 48-60 Month 0-12 Month  24-48
Month
##   [7] 24-48 Month 48-60 Month 0-12 Month  24-48 Month 12-24 Month 0-12
Month
##  [13] 24-48 Month 24-48 Month > 60 Month  24-48 Month 24-48 Month 12-24
Month
##  [19] > 60 Month  24-48 Month 24-48 Month > 60 Month  12-24 Month 12-24
Month
##  [25] 0-12 Month  > 60 Month  24-48 Month 12-24 Month > 60 Month  48-60
Month
##  [31] 48-60 Month 24-48 Month 48-60 Month 24-48 Month 12-24 Month 24-48
Month
##  [37] 24-48 Month > 60 Month  > 60 Month  48-60 Month 24-48 Month 0-12
Month
##  [43] > 60 Month  48-60 Month 24-48 Month > 60 Month  48-60 Month 24-48
Month
##  [49] 24-48 Month 48-60 Month 0-12 Month  0-12 Month  24-48 Month 0-12
Month
##  [55] 48-60 Month 24-48 Month 24-48 Month 24-48 Month 12-24 Month > 60
Month
##  [61] > 60 Month  24-48 Month 48-60 Month 24-48 Month 24-48 Month > 60
Month
##  [67] 0-12 Month  0-12 Month  24-48 Month 24-48 Month 0-12 Month  24-48
Month
##  [73] 12-24 Month 12-24 Month 12-24 Month 24-48 Month 0-12 Month  0-12
Month
##  [79] 0-12 Month  12-24 Month 0-12 Month  48-60 Month 48-60 Month 24-48
```

```
Month
##  [85] > 60 Month  0-12 Month  12-24 Month 24-48 Month 48-60 Month 24-48
Month
##  [91] > 60 Month  12-24 Month 12-24 Month 24-48 Month 0-12 Month  24-48
Month
##  [97] 12-24 Month 24-48 Month > 60 Month  12-24 Month 48-60 Month 0-12
Month
## [103] 12-24 Month 24-48 Month 48-60 Month 48-60 Month 24-48 Month 12-24
Month
## [109] 48-60 Month 48-60 Month > 60 Month  0-12 Month  12-24 Month 24-48
Month
## [115] 12-24 Month 12-24 Month 0-12 Month  24-48 Month 0-12 Month  0-12
Month
## [121] 48-60 Month 48-60 Month 24-48 Month 24-48 Month > 60 Month  0-12
Month
## [127] 0-12 Month  48-60 Month 0-12 Month  0-12 Month  > 60 Month
## Levels: > 60 Month 0-12 Month 12-24 Month 24-48 Month 48-60 Month
```

Exploratory data analysis:(Variable Selection)

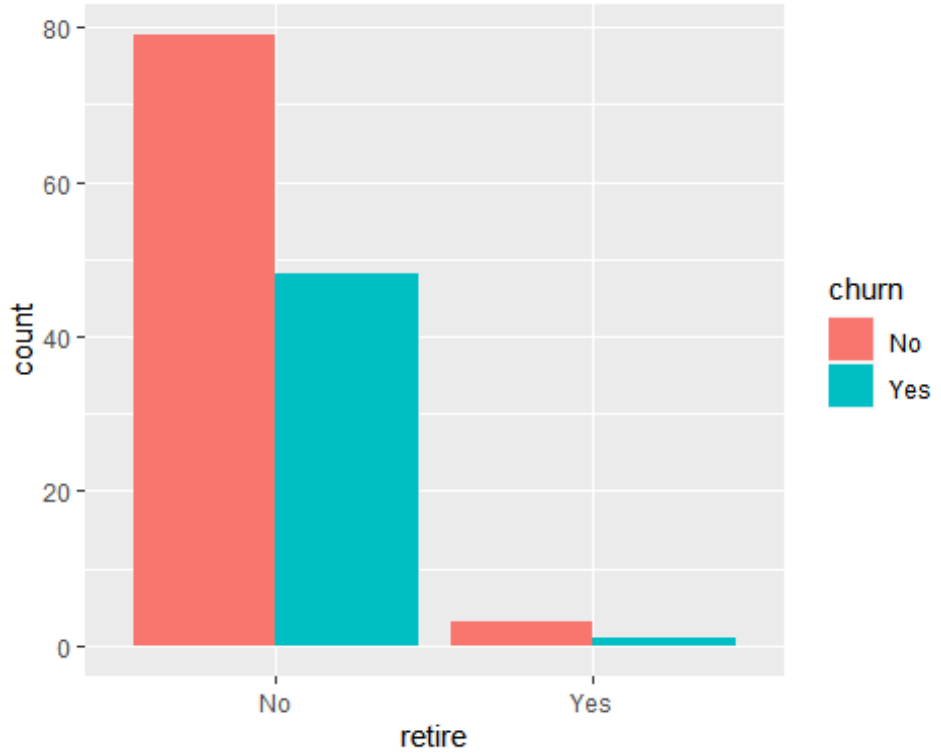
```
#Gender Overview
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.6.3

ggplot(df) +
  geom_bar(aes(x = gender, fill = churn), position = "dodge")
```



```
library(magrittr)

## Warning: package 'magrittr' was built under R version 3.6.3

library(dplyr)

## Warning: package 'dplyr' was built under R version 3.6.3

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

df %>%
  group_by(gender) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 2 x 3
##   gender     n  freq
##   <fct>  <int> <dbl>
## 1 Female    70 0.534
## 2 Male      61 0.466

df %>%
  group_by(gender, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'gender' (override with `.groups`
argument)

## # A tibble: 4 x 4
## # Groups:   gender [2]
##   gender churn     n  freq
##   <fct>  <fct> <int> <dbl>
## 1 Female No       38 0.543
## 2 Female Yes      32 0.457
## 3 Male   No       44 0.721
## 4 Male   Yes      17 0.279
```

Roughly there are 53.4% female customers 46.5% male customers. On the other hand, of the 53% females , 45% churn & Of the 46% males, only 27% churn.

```r
#Senior Citizen overview
ggplot(df) +
  geom_bar(aes(x = retire, fill = churn), position = "dodge")
```




```r
df %>%
  group_by(retire) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##   retire     n   freq
##   <fct>  <int>  <dbl>
## 1 No       127 0.969
## 2 Yes        4 0.0305
```

```r
df %>%
  group_by(retire, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

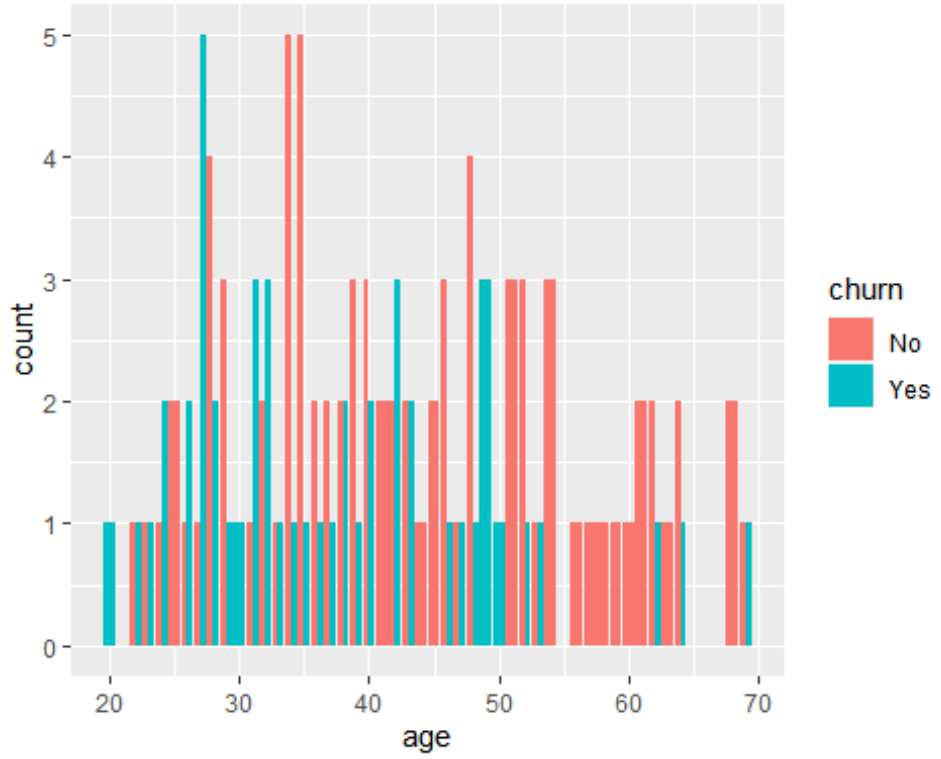
```
## `summarise()` regrouping output by 'retire' (override with `.groups`
argument)
```

```
## # A tibble: 4 x 4
## # Groups:   retire [2]
##   retire churn     n  freq
```

```
##   <fct>  <fct> <int> <dbl>
## 1 No     No       79 0.622
## 2 No     Yes      48 0.378
## 3 Yes    No        3 0.75
## 4 Yes    Yes       1 0.25
```

There are 3% customers who are retired & out of those 25% of the retired customers churn.

```
#Age overview
ggplot(df) +
  geom_bar(aes(x = age, fill = churn), position = "dodge")
```




```
df %>%
  group_by(age) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 45 x 3
##      age     n    freq
##    <int> <int>   <dbl>
##  1    20     1 0.00763
##  2    22     2 0.0153
##  3    23     2 0.0153
##  4    24     3 0.0229
##  5    25     2 0.0153
```

```
##  6    26     3 0.0229
##  7    27     6 0.0458
##  8    28     6 0.0458
##  9    29     4 0.0305
## 10    30     1 0.00763
## # ... with 35 more rows
```

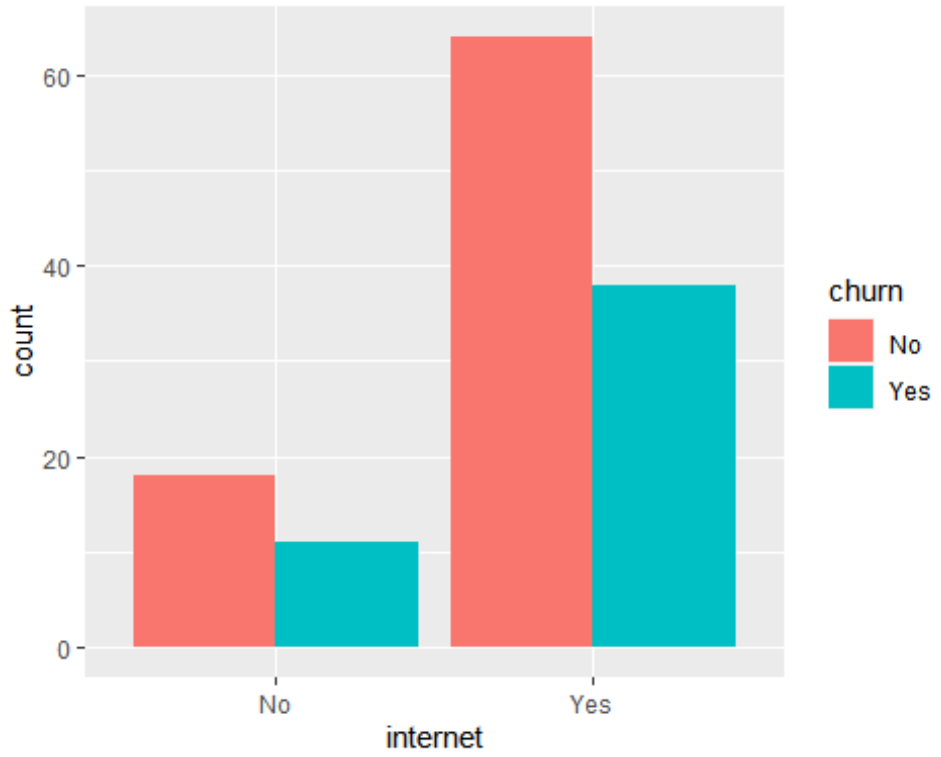
```
df %>%
  group_by(age, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'age' (override with `.groups`
argument)
```

```
## # A tibble: 72 x 4
## # Groups:   age [45]
##      age churn     n  freq
##    <int> <fct> <int> <dbl>
##  1    20 Yes       1 1
##  2    22 No        1 0.5
##  3    22 Yes       1 0.5
##  4    23 No        1 0.5
##  5    23 Yes       1 0.5
##  6    24 No        1 0.333
##  7    24 Yes       2 0.667
##  8    25 No        2 1
##  9    26 No        1 0.333
## 10    26 Yes       2 0.667
## # ... with 62 more rows
```

Age group of customers:20-69 100% Customers churn having age group 20,30,49,50

```
#Internet overview
ggplot(df) +
  geom_bar(aes(x = internet, fill = churn), position = "dodge")
```

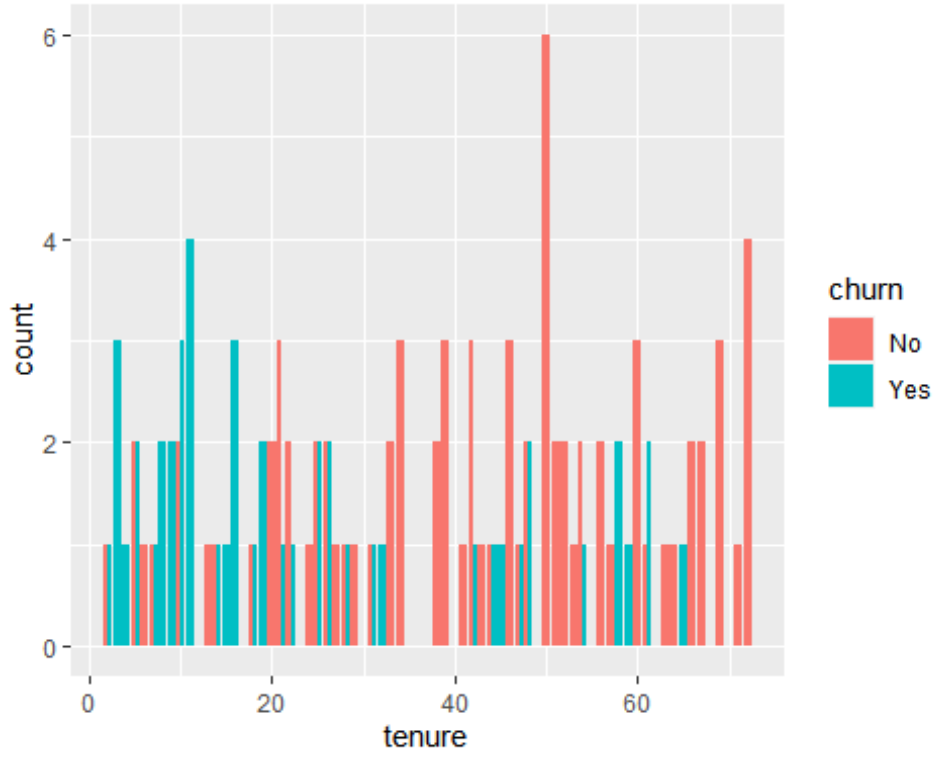



```
df %>%
  group_by(internet) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 2 x 3
##   internet     n  freq
##   <fct>    <int> <dbl>
## 1 No          29 0.221
## 2 Yes        102 0.779

df %>%
  group_by(internet, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'internet' (override with `.groups`
argument)

## # A tibble: 4 x 4
## # Groups:   internet [2]
##   internet churn     n  freq
##   <fct>    <fct> <int> <dbl>
## 1 No       No       18 0.621
## 2 No       Yes      11 0.379
```

```
## 3 Yes       No        64 0.627
## 4 Yes       Yes       38 0.373
```

Roughly 77% of the customers have internet & out of them around 37.2% of the customers churn.Roughly 22% of the customers do not have internet & out of them around 37.9% of the customers churn.

```
#Tenure overview
ggplot(df) +
  geom_bar(aes(x = tenure, fill = churn), position = "dodge")
```




```
df %>%
  group_by(tenure) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 58 x 3
##    tenure     n    freq
##     <int> <int>   <dbl>
##  1      2     2 0.0153
##  2      3     3 0.0229
##  3      4     1 0.00763
##  4      5     4 0.0305
##  5      6     1 0.00763
##  6      7     2 0.0153
```

```
##  7      8     2 0.0153
##  8      9     2 0.0153
##  9     10     5 0.0382
## 10     11     4 0.0305
## # ... with 48 more rows
```

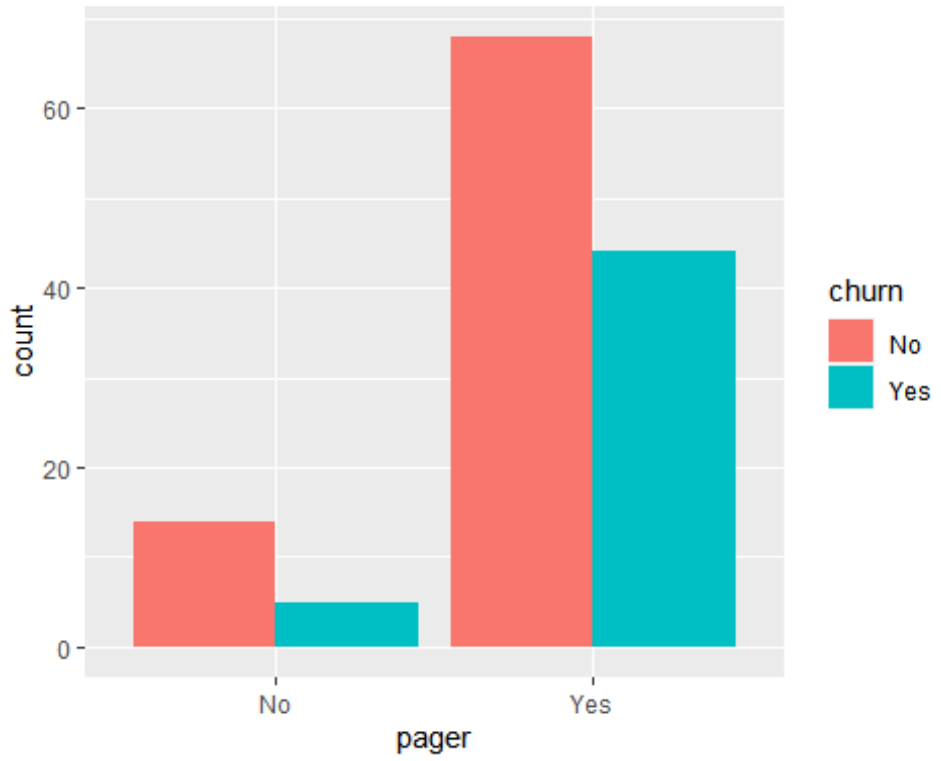
```
df %>%
  group_by(tenure, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'tenure' (override with `.groups`
argument)
```

```
## # A tibble: 76 x 4
## # Groups:   tenure [58]
##    tenure churn     n  freq
##     <int> <fct> <int> <dbl>
##  1      2 No        1   0.5
##  2      2 Yes       1   0.5
##  3      3 Yes       3   1
##  4      4 Yes       1   1
##  5      5 No        2   0.5
##  6      5 Yes       2   0.5
##  7      6 No        1   1
##  8      7 No        1   0.5
##  9      7 Yes       1   0.5
## 10      8 Yes       2   1
## # ... with 66 more rows
```

Tenure Period:2-72 months 100% Customers churn having tenure 3,4,6,8,11,15,16,19,24,32,45,58,59 & 65 months.

```
#Paging service overview
ggplot(df) +
  geom_bar(aes(x =pager, fill = churn), position = "dodge")
```

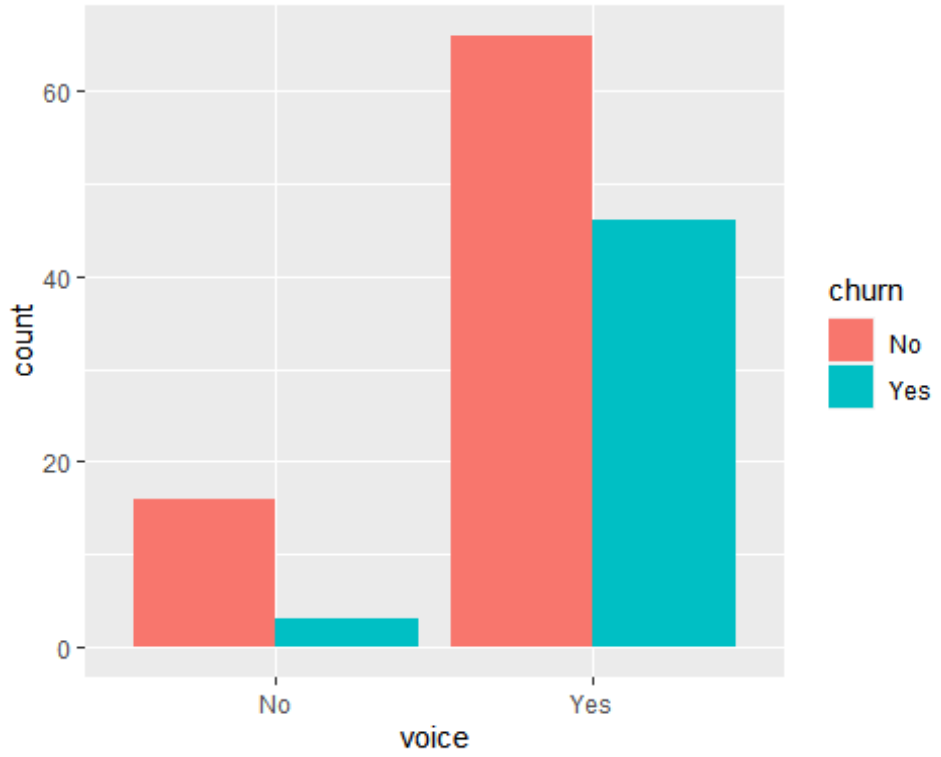



```
df %>%
  group_by(pager) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 2 x 3
##   pager     n  freq
##   <fct> <int> <dbl>
## 1 No       19 0.145
## 2 Yes     112 0.855

df %>%
  group_by(pager, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'pager' (override with `.groups`
argument)

## # A tibble: 4 x 4
## # Groups:   pager [2]
##   pager churn     n  freq
##   <fct> <fct> <int> <dbl>
## 1 No    No       14 0.737
## 2 No    Yes       5 0.263
```

```
## 3 Yes   No        68 0.607
## 4 Yes   Yes       44 0.393
```

Around 85% of the customers use Pager & out of them around 39% of the customers churn.

```
#Voice mail overview
ggplot(df) +
  geom_bar(aes(x =voice, fill = churn), position = "dodge")
```




```
df %>%
  group_by(voice) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

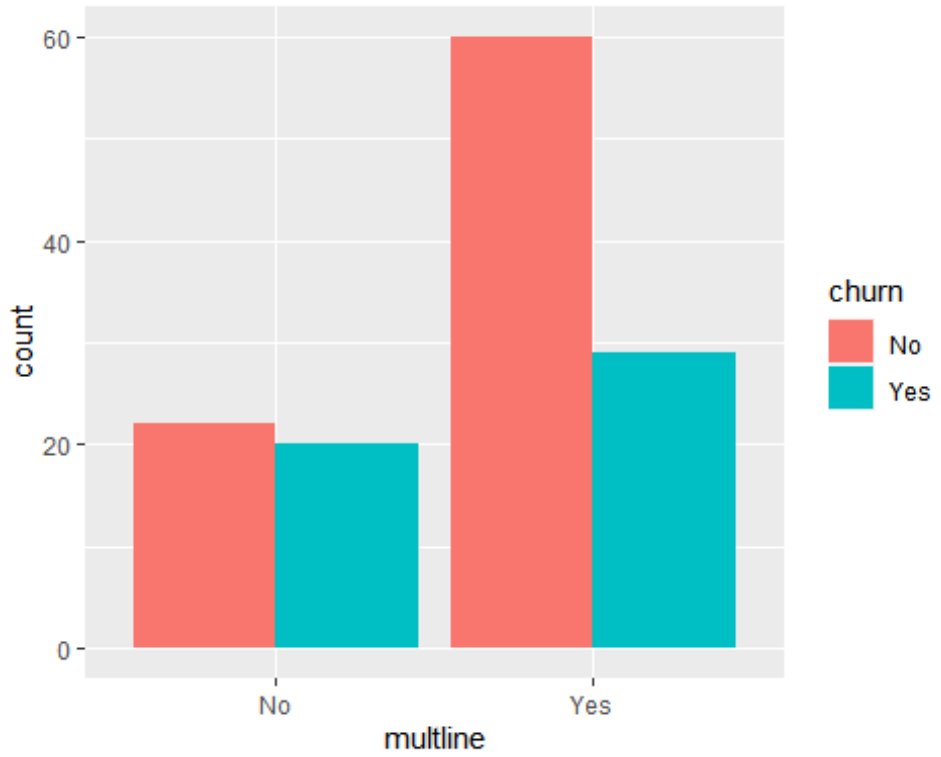
```
## # A tibble: 2 x 3
##   voice     n  freq
##   <fct> <int> <dbl>
## 1 No       19 0.145
## 2 Yes     112 0.855
```

```
df %>%
  group_by(voice, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'voice' (override with `.groups`
argument)

## # A tibble: 4 x 4
## # Groups:   voice [2]
##   voice churn     n  freq
##   <fct> <fct> <int> <dbl>
## 1 No    No       16 0.842
## 2 No    Yes       3 0.158
## 3 Yes   No       66 0.589
## 4 Yes   Yes      46 0.411
```

Around 85% of the customers used Voice Mail & out of those around 41% of the customers churn.

```
#Multiple lines overview
ggplot(df) +
  geom_bar(aes(x =multline, fill = churn), position = "dodge")
```




```
df %>%
  group_by(multline) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 2 x 3
##   multline     n  freq
```

```
##   <fct>    <int> <dbl>
## 1 No          42 0.321
## 2 Yes         89 0.679
```

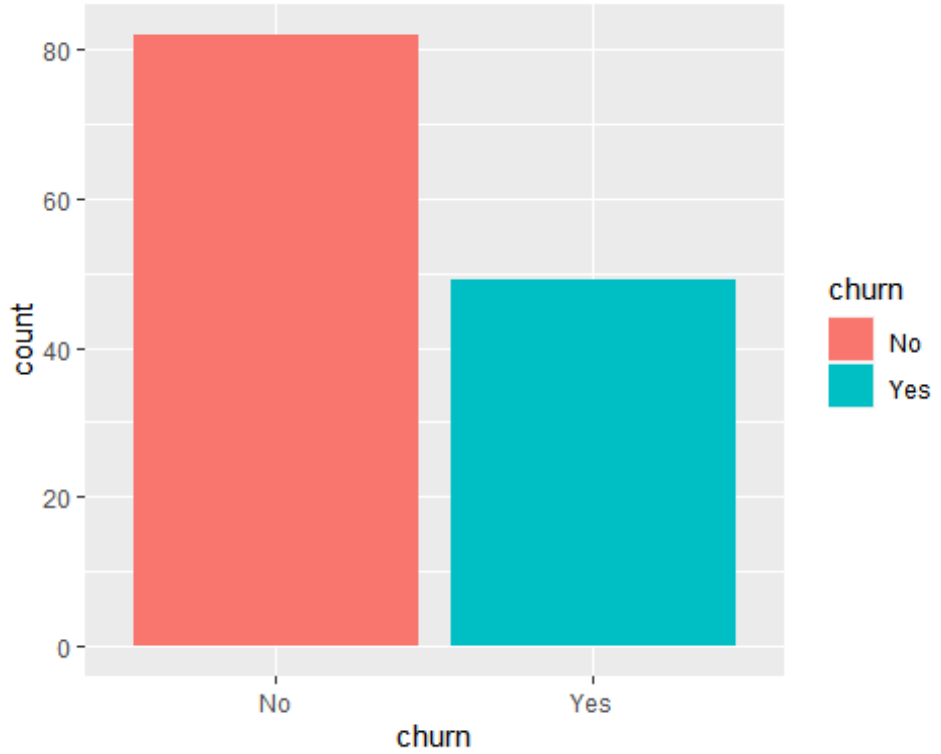
```
df %>%
  group_by(multline, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'multline' (override with `.groups`
argument)
```

```
## # A tibble: 4 x 4
## # Groups:   multline [2]
##   multline churn     n  freq
##   <fct>    <fct> <int> <dbl>
## 1 No       No       22 0.524
## 2 No       Yes      20 0.476
## 3 Yes      No       60 0.674
## 4 Yes      Yes      29 0.326
```

Around 68% of the customers used Multiple Lines & out of those around 32.5% of the customers churn.

```
#Customer Churn overview
ggplot(df) +
  geom_bar(aes(x =churn, fill = churn), position = "dodge")
```



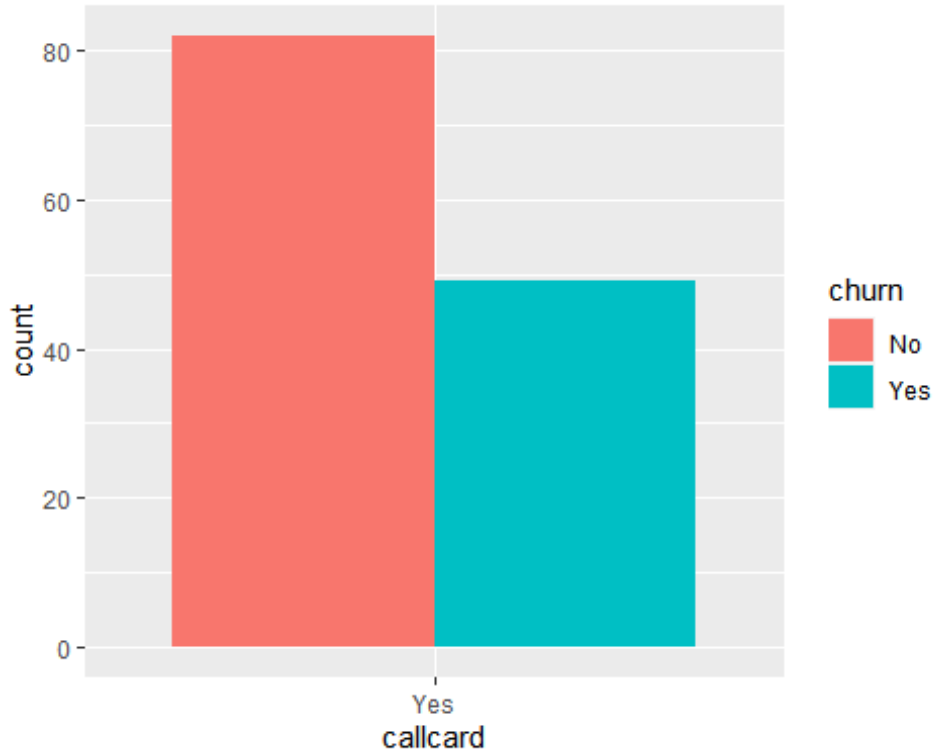
```
df %>%
  group_by(churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##   churn     n  freq
##   <fct> <int> <dbl>
## 1 No       82 0.626
## 2 Yes      49 0.374
```

Around 37% of the customers churn.

```
#Calling card service overview
ggplot(df) +
  geom_bar(aes(x =callcard, fill = churn), position = "dodge")
```




```
df %>%
  group_by(callcard) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 3
##   callcard     n  freq
```

```
##   <fct>    <int> <dbl>
## 1 Yes        131     1
```

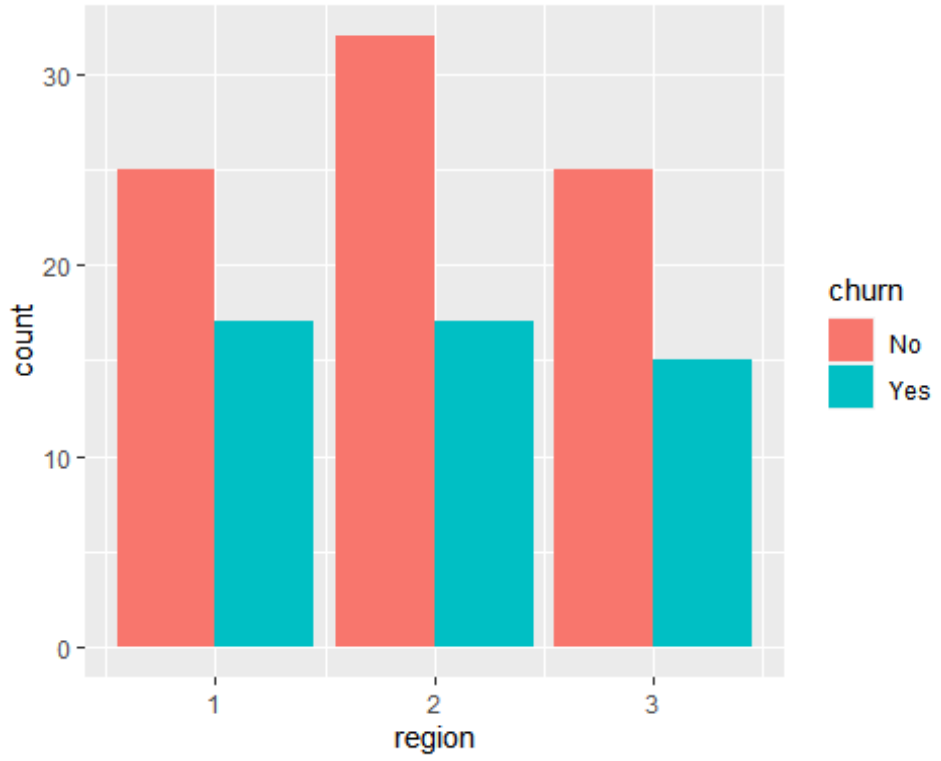
```
df %>%
  group_by(callcard, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'callcard' (override with `.groups`
argument)
```

```
## # A tibble: 2 x 4
## # Groups:   callcard [1]
##   callcard churn     n  freq
##   <fct>    <fct> <int> <dbl>
## 1 Yes      No       82 0.626
## 2 Yes      Yes      49 0.374
```

All customers are using the calling card service .Around 37.4% of the customers churn.

```
#Region overview
ggplot(df) +
  geom_bar(aes(x =region, fill = churn), position = "dodge")
```

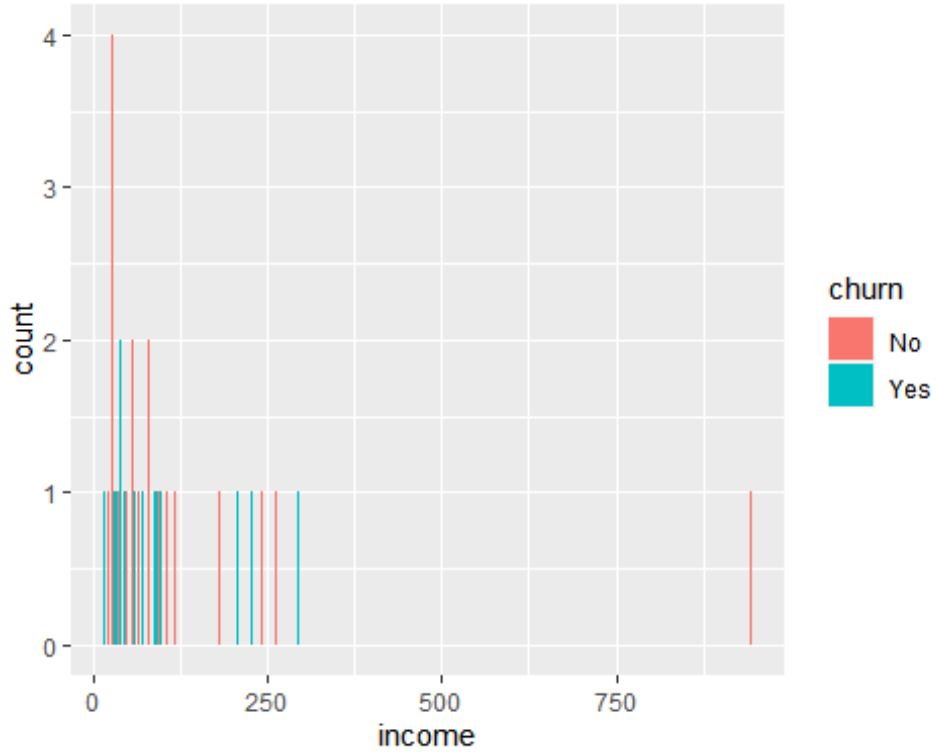



```
df %>%
  group_by(region) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 3 x 3
##   region     n  freq
##    <int> <int> <dbl>
## 1      1    42 0.321
## 2      2    49 0.374
## 3      3    40 0.305

df %>%
  group_by(region, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'region' (override with `.groups`
argument)

## # A tibble: 6 x 4
## # Groups:   region [3]
##   region churn     n  freq
##    <int> <fct> <int> <dbl>
## 1      1 No       25 0.595
## 2      1 Yes      17 0.405
## 3      2 No       32 0.653
## 4      2 Yes      17 0.347
## 5      3 No       25 0.625
## 6      3 Yes      15 0.375
```

Around 32%,37.4% & 30.5% of the customers belong to region 1,2 & 3 respectively. Churn % in region 1: 40.4% Churn % in region 2: 34.6% Churn % in region 3: 37.5%

```
#Income overview
ggplot(df) +
  geom_bar(aes(x =income, fill = churn), position = "dodge")
```




```
df %>%
  group_by(income) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

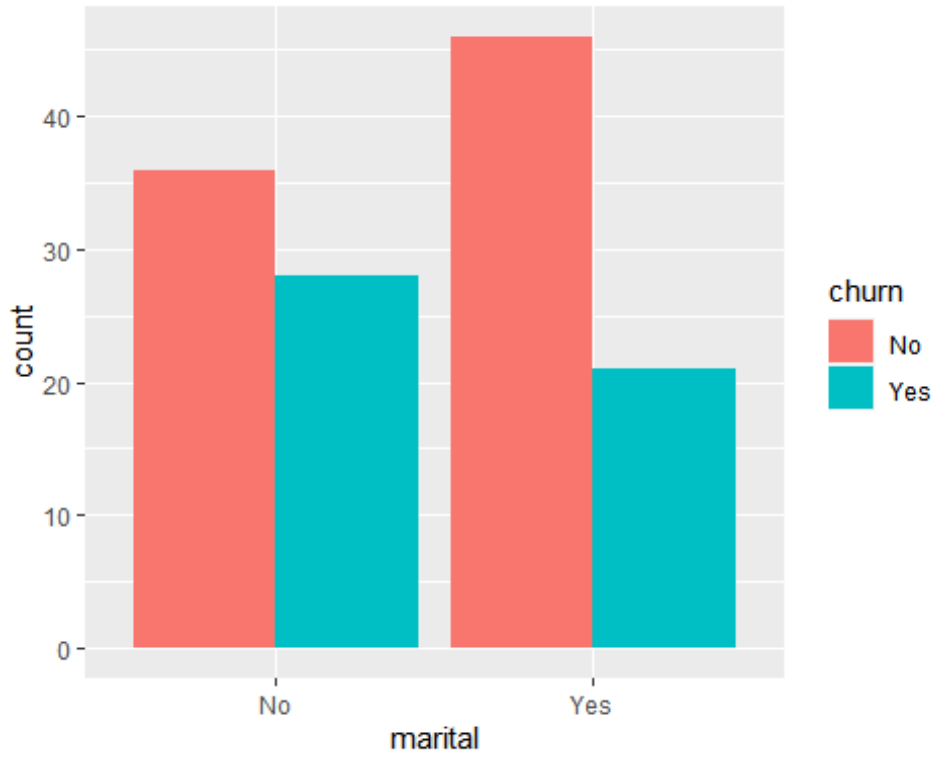
```
## # A tibble: 87 x 3
##    income     n    freq
##     <int> <int>   <dbl>
##  1     15     1 0.00763
##  2     16     1 0.00763
##  3     18     2 0.0153
##  4     20     3 0.0229
##  5     22     1 0.00763
##  6     23     3 0.0229
##  7     24     1 0.00763
##  8     25     1 0.00763
##  9     26     2 0.0153
## 10     27     4 0.0305
## # ... with 77 more rows
```

```
df %>%
  group_by(income, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'income' (override with `.groups`
argument)

## # A tibble: 101 x 4
## # Groups:   income [87]
##    income churn     n  freq
##     <int> <fct> <int> <dbl>
##  1     15 Yes       1 1
##  2     16 Yes       1 1
##  3     18 No        1 0.5
##  4     18 Yes       1 0.5
##  5     20 No        2 0.667
##  6     20 Yes       1 0.333
##  7     22 No        1 1
##  8     23 No        2 0.667
##  9     23 Yes       1 0.333
## 10     24 No        1 1
## # ... with 91 more rows
```

Customers having more income are most likely to churn.

```
#Marital overview
ggplot(df) +
  geom_bar(aes(x =marital, fill = churn), position = "dodge")
```

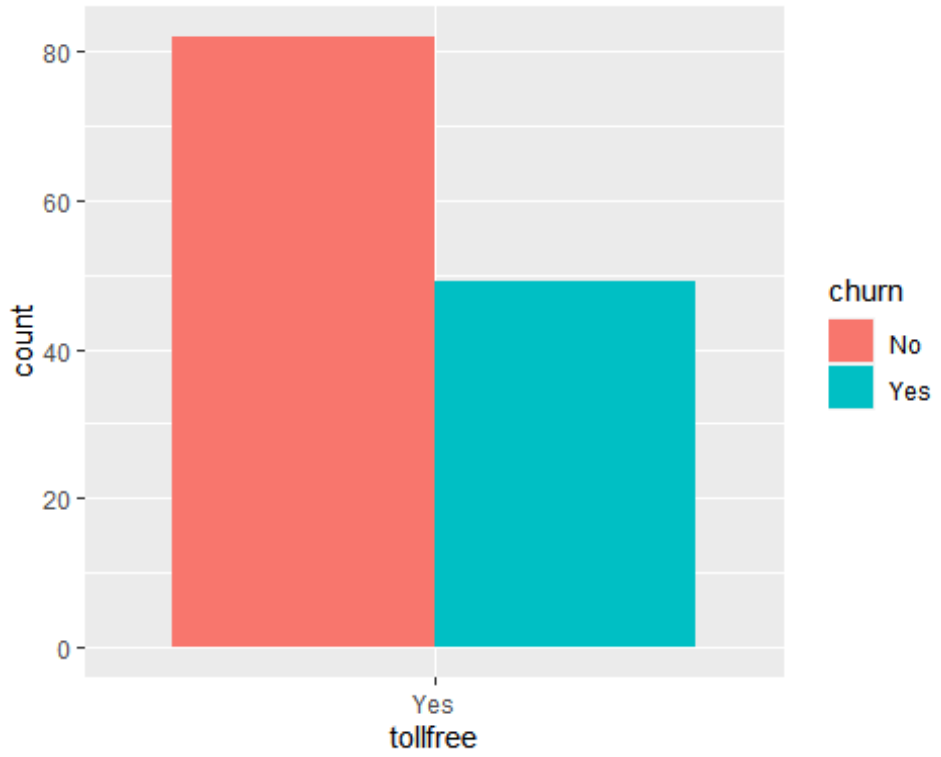



```
df %>%
  group_by(marital) %>%
```

```
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 2 x 3
##   marital     n  freq
##   <fct>   <int> <dbl>
## 1 No         64 0.489
## 2 Yes        67 0.511

df %>%
  group_by(marital, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'marital' (override with `.groups`
argument)

## # A tibble: 4 x 4
## # Groups:   marital [2]
##   marital churn     n  freq
##   <fct>   <fct> <int> <dbl>
## 1 No      No       36 0.562
## 2 No      Yes      28 0.438
## 3 Yes     No       46 0.687
## 4 Yes     Yes      21 0.313
```

Around 51% of the customers are married & out of them around 31% of the customers churn.

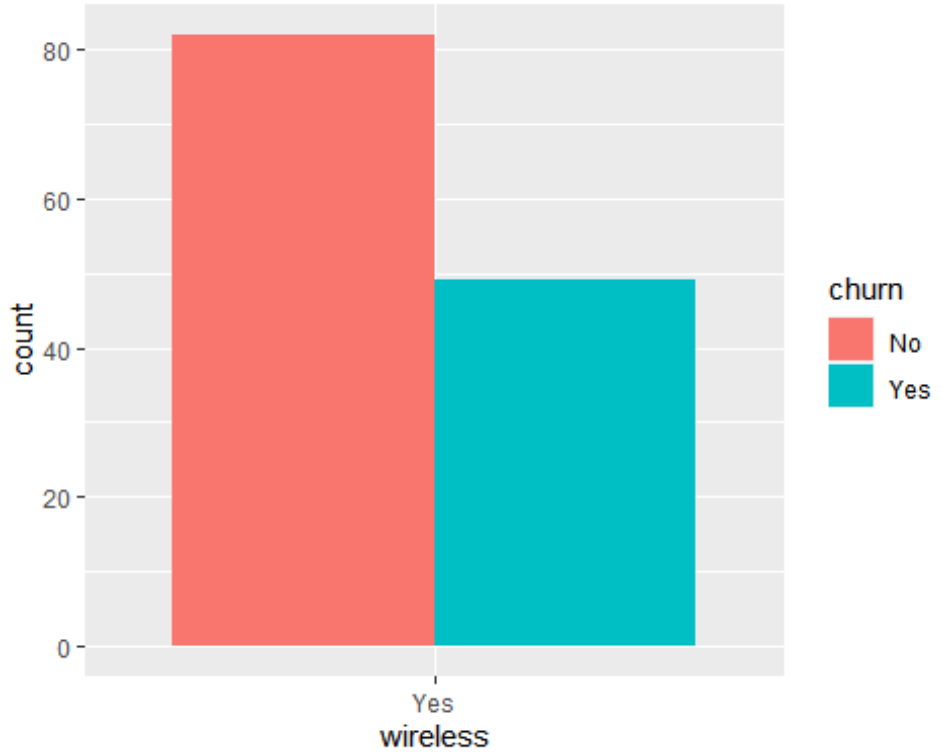
```
#Toll free overview
ggplot(df) +
  geom_bar(aes(x =tollfree, fill = churn), position = "dodge")
```




```
df %>%
  group_by(tollfree) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 1 x 3
##   tollfree     n  freq
##   <fct>    <int> <dbl>
## 1 Yes        131     1

df %>%
  group_by(tollfree, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'tollfree' (override with `.groups`
argument)

## # A tibble: 2 x 4
## # Groups:   tollfree [1]
##   tollfree churn     n  freq
##   <fct>    <fct> <int> <dbl>
## 1 Yes      No       82 0.626
## 2 Yes      Yes      49 0.374
```

All are using the tollfree service & out of them around 37% of the customers churn.

```r
#Wireless service overview
ggplot(df) +
  geom_bar(aes(x =wireless, fill = churn), position = "dodge")
```




```r
df %>%
  group_by(wireless) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 3
##   wireless     n  freq
##   <fct>    <int> <dbl>
## 1 Yes        131     1
```

```r
df %>%
  group_by(wireless, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

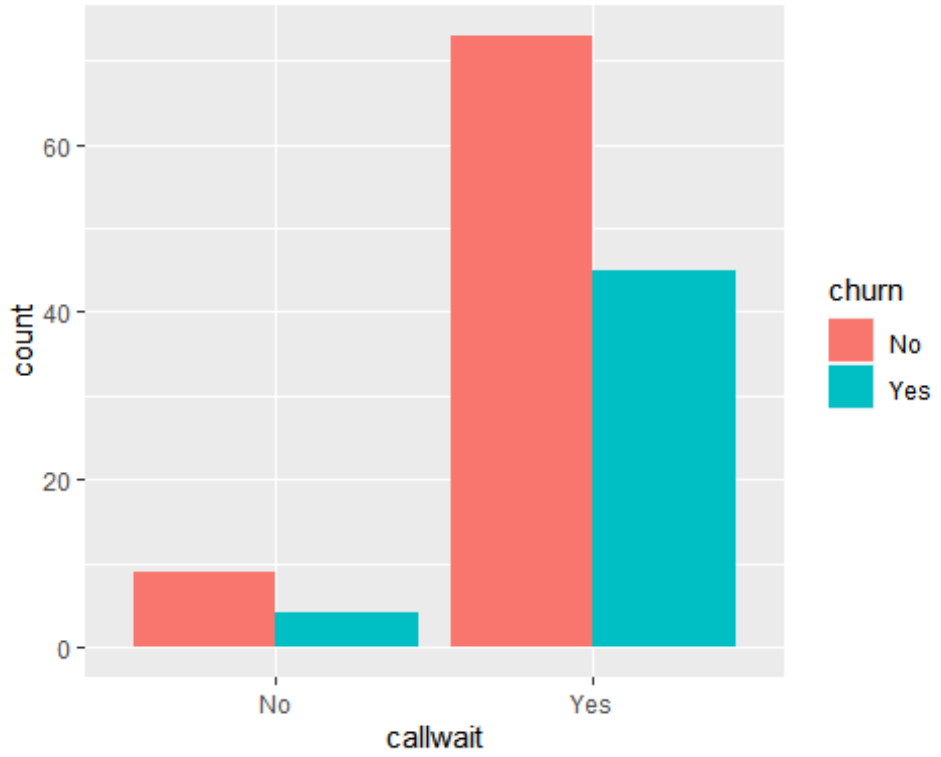
```
## `summarise()` regrouping output by 'wireless' (override with `.groups`
argument)
```

```
## # A tibble: 2 x 4
## # Groups:   wireless [1]
##   wireless churn     n  freq
##   <fct>    <fct> <int> <dbl>
```

```
## 1 Yes       No        82 0.626
## 2 Yes       Yes       49 0.374
```

All are using the wireless services & out of them around 37% of the customers churn.

```
#Call-waiting service overview
ggplot(df) +
  geom_bar(aes(x =callwait, fill = churn), position = "dodge")
```




```
df %>%
  group_by(callwait) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

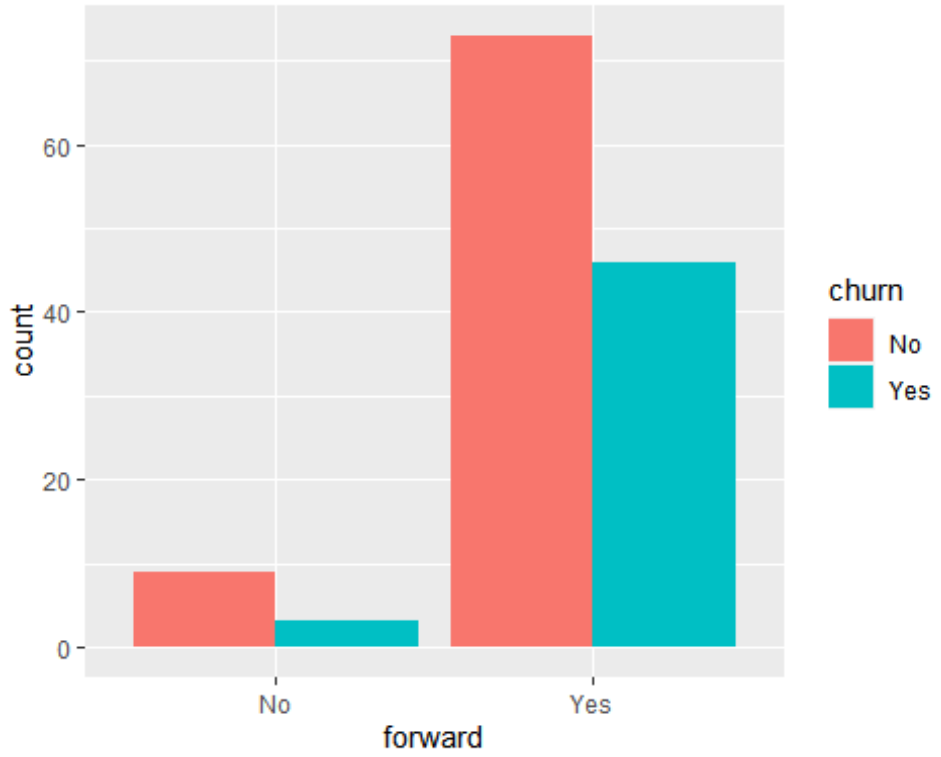
```
## # A tibble: 2 x 3
##   callwait     n   freq
##   <fct>    <int>  <dbl>
## 1 No          13 0.0992
## 2 Yes        118 0.901
```

```
df %>%
  group_by(callwait, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'callwait' (override with `.groups`
argument)

## # A tibble: 4 x 4
## # Groups:   callwait [2]
##   callwait churn     n  freq
##   <fct>    <fct> <int> <dbl>
## 1 No       No        9 0.692
## 2 No       Yes       4 0.308
## 3 Yes      No       73 0.619
## 4 Yes      Yes      45 0.381
```

90% of the customers' calls were waiting & out of the around 38% of the customers churn.

```
#Call-forwarding service overview
ggplot(df) +
  geom_bar(aes(x =forward, fill = churn), position = "dodge")
```




```
df %>%
  group_by(forward) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 2 x 3
##   forward     n   freq
##   <fct>   <int>  <dbl>
```

```
## 1 No          12 0.0916
## 2 Yes        119 0.908
```

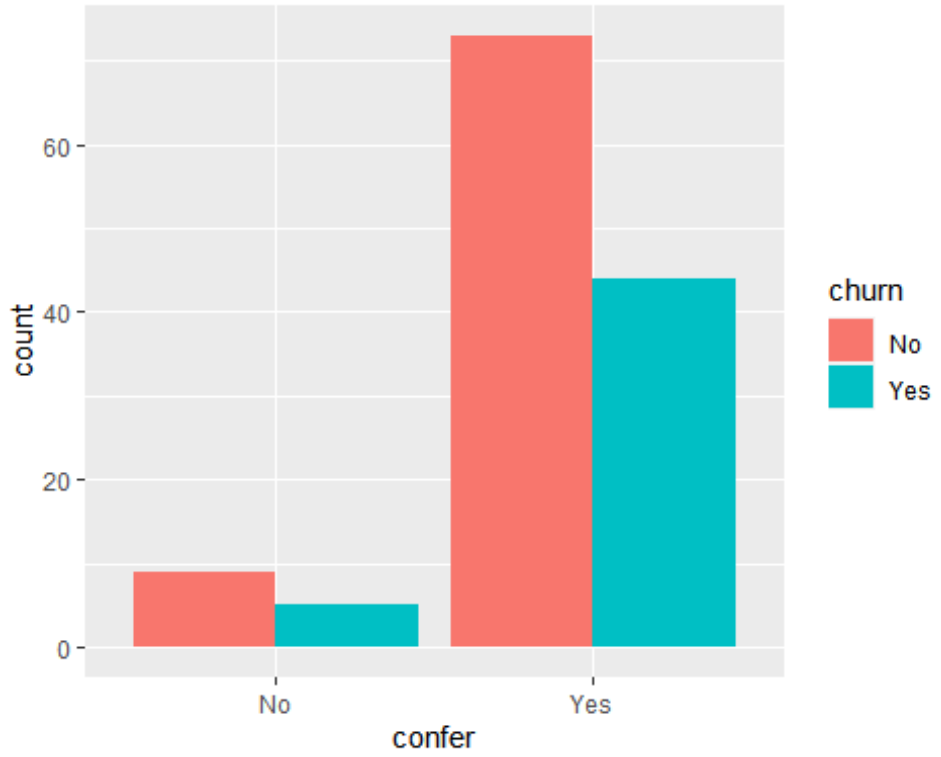
```
df %>%
  group_by(forward, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'forward' (override with `.groups`
argument)
```

```
## # A tibble: 4 x 4
## # Groups:   forward [2]
##   forward churn     n  freq
##   <fct>   <fct> <int> <dbl>
## 1 No      No        9 0.75
## 2 No      Yes       3 0.25
## 3 Yes     No       73 0.613
## 4 Yes     Yes      46 0.387
```

Around 91% of the customers calls were forwarded out of them around 39% of the customers churn.

```
#3 way calling service overview
ggplot(df) +
  geom_bar(aes(x =confer, fill = churn), position = "dodge")
```



```
df %>%
  group_by(confer) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 2 x 3
##   confer     n  freq
##   <fct>  <int> <dbl>
## 1 No        14 0.107
## 2 Yes      117 0.893
```

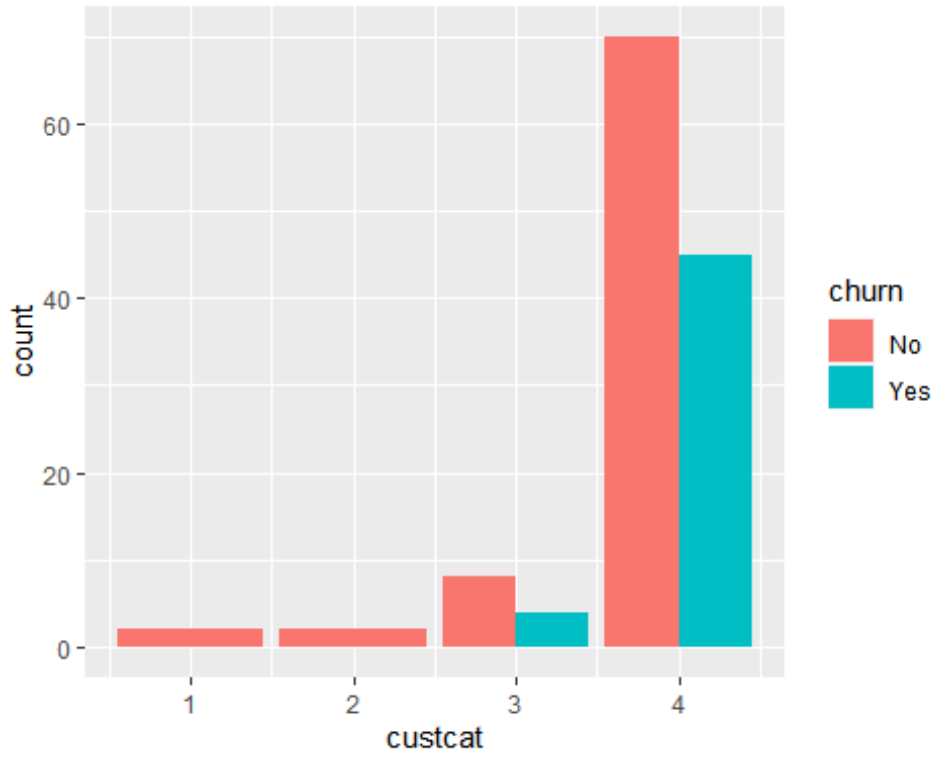
```
df %>%
  group_by(confer, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))
```

```
## `summarise()` regrouping output by 'confer' (override with `.groups`
argument)
```

```
## # A tibble: 4 x 4
## # Groups:   confer [2]
##   confer churn     n  freq
##   <fct>  <fct> <int> <dbl>
## 1 No     No        9 0.643
## 2 No     Yes       5 0.357
## 3 Yes    No       73 0.624
## 4 Yes    Yes      44 0.376
```

Almost 90% of the customers did conference calls out of which around 38% of the customers churn.

```
#customer category overview
ggplot(df) +
  geom_bar(aes(x =custcat, fill = churn), position = "dodge")
```



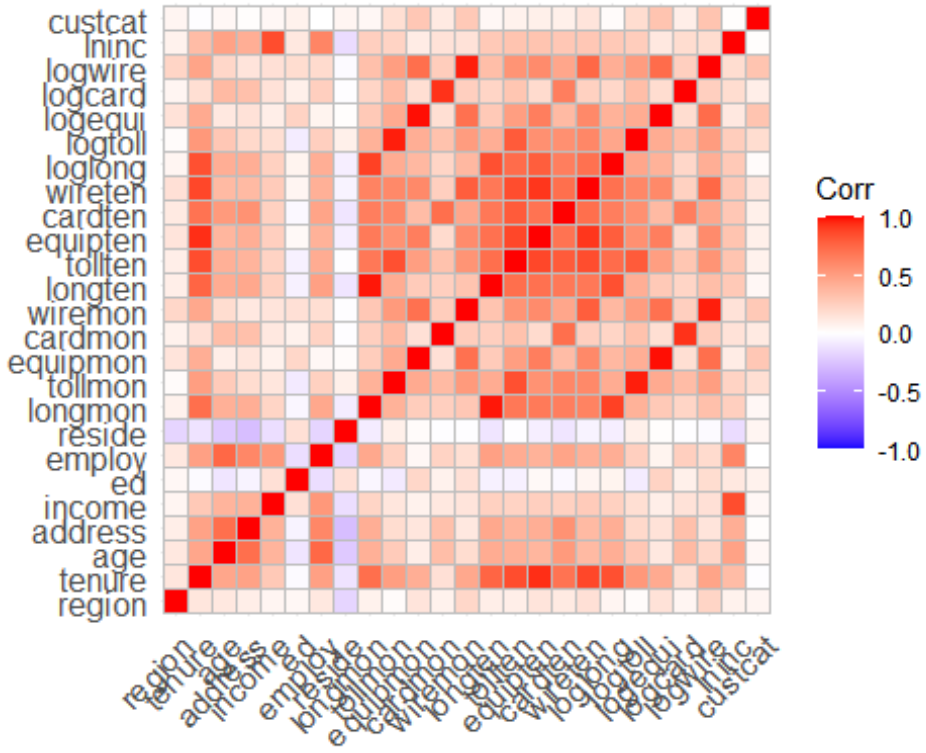

```
df %>%
  group_by(custcat) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 4 x 3
##   custcat     n   freq
##     <int> <int>  <dbl>
## 1       1     2 0.0153
## 2       2     2 0.0153
## 3       3    12 0.0916
## 4       4   115 0.878

df %>%
  group_by(custcat, churn) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n))

## `summarise()` regrouping output by 'custcat' (override with `.groups`
argument)

## # A tibble: 6 x 4
## # Groups:   custcat [4]
##   custcat churn     n  freq
##     <int> <fct> <int> <dbl>
## 1       1 No        2 1
```

```
## 2       2 No        2 1
## 3       3 No        8 0.667
## 4       3 Yes       4 0.333
## 5       4 No       70 0.609
## 6       4 Yes      45 0.391
```

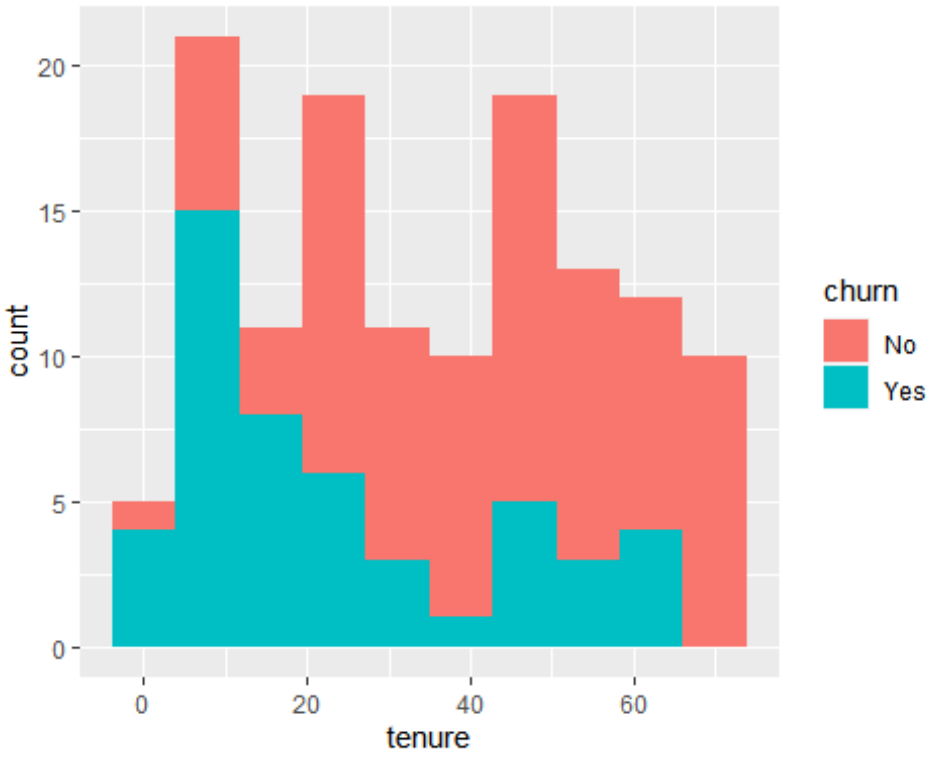
Only customers belonging to category 3 & 4 churn.

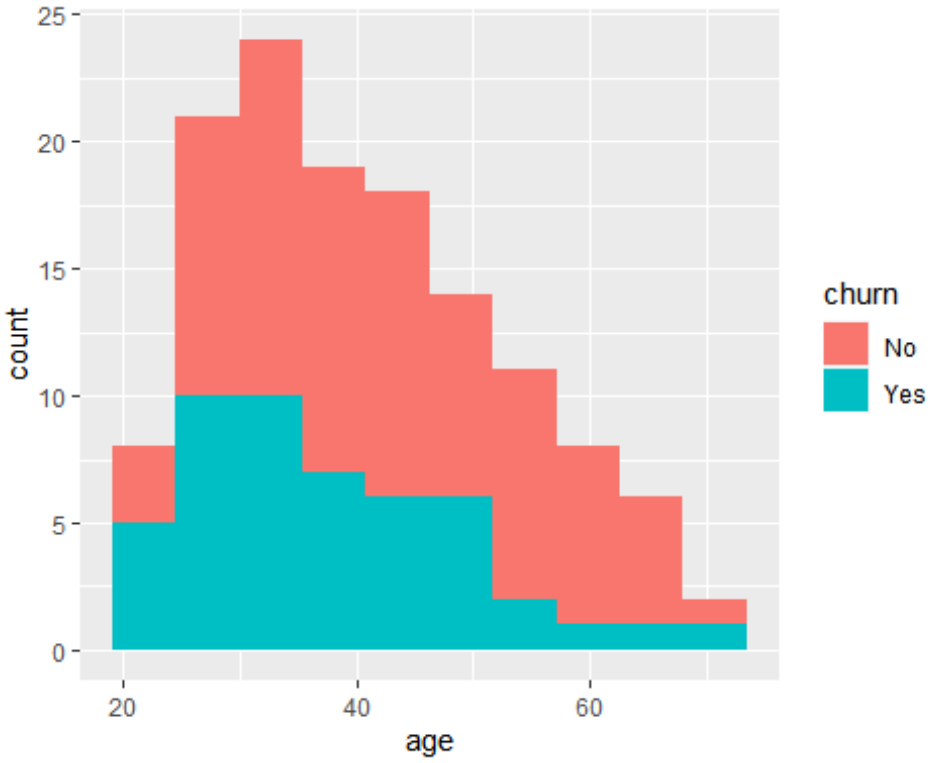
Correlation between numeric variables:

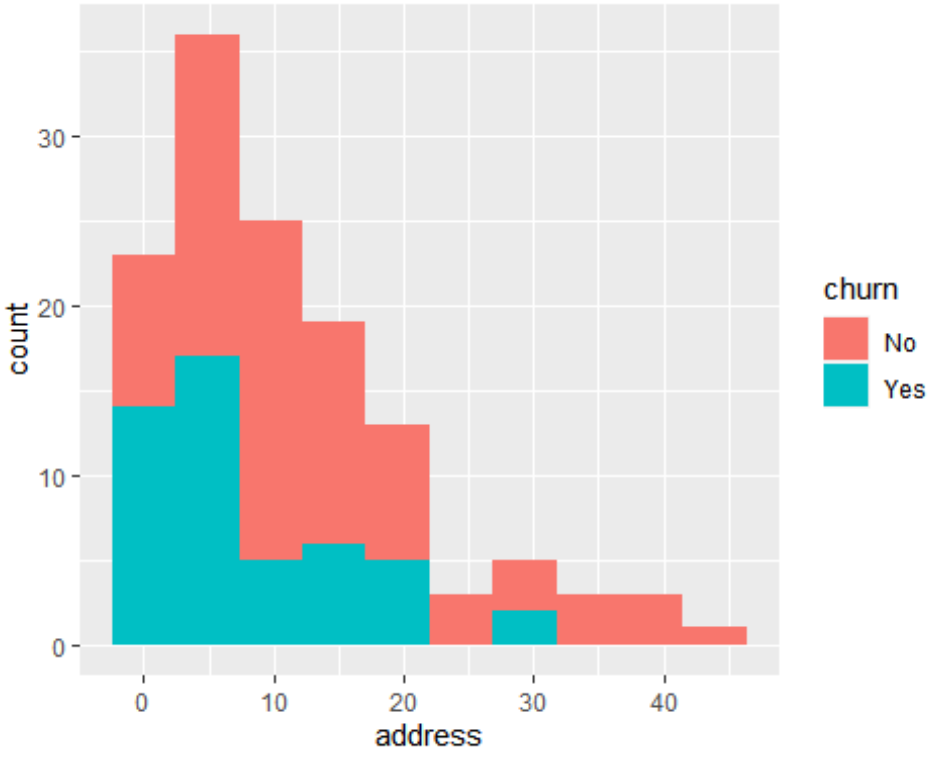
```
# Correlation plot
df1 <- sapply(df, is.numeric)
corr.matrix <- cor(df[,df1])
library(ggplot2)
library(ggcorrplot)

## Warning: package 'ggcorrplot' was built under R version 3.6.3

ggcorrplot(corr.matrix)
```

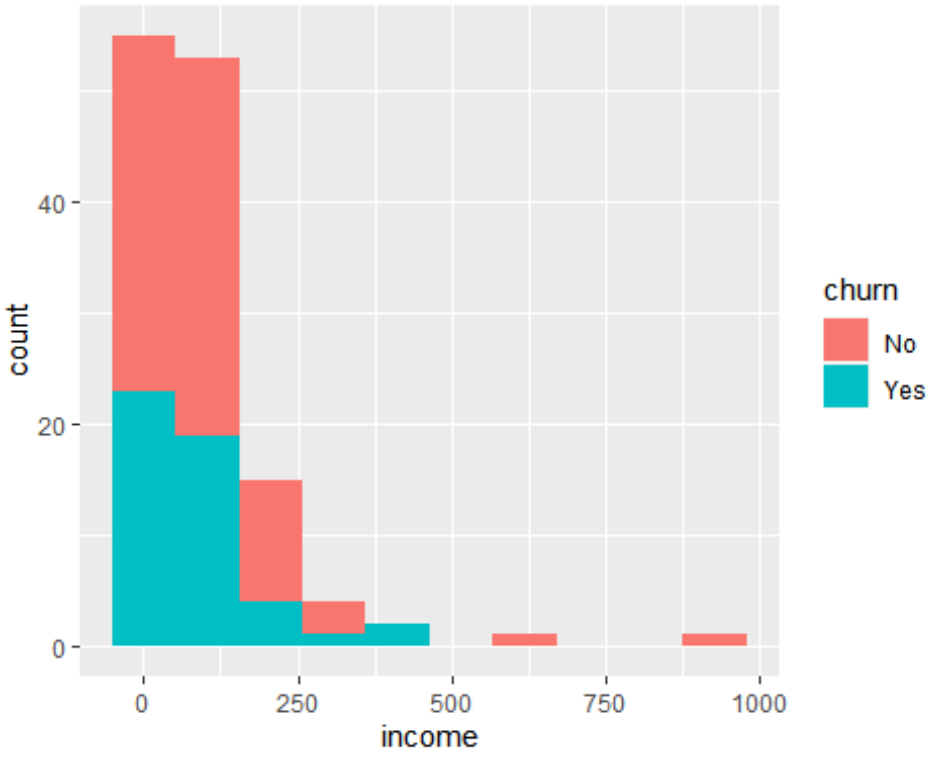


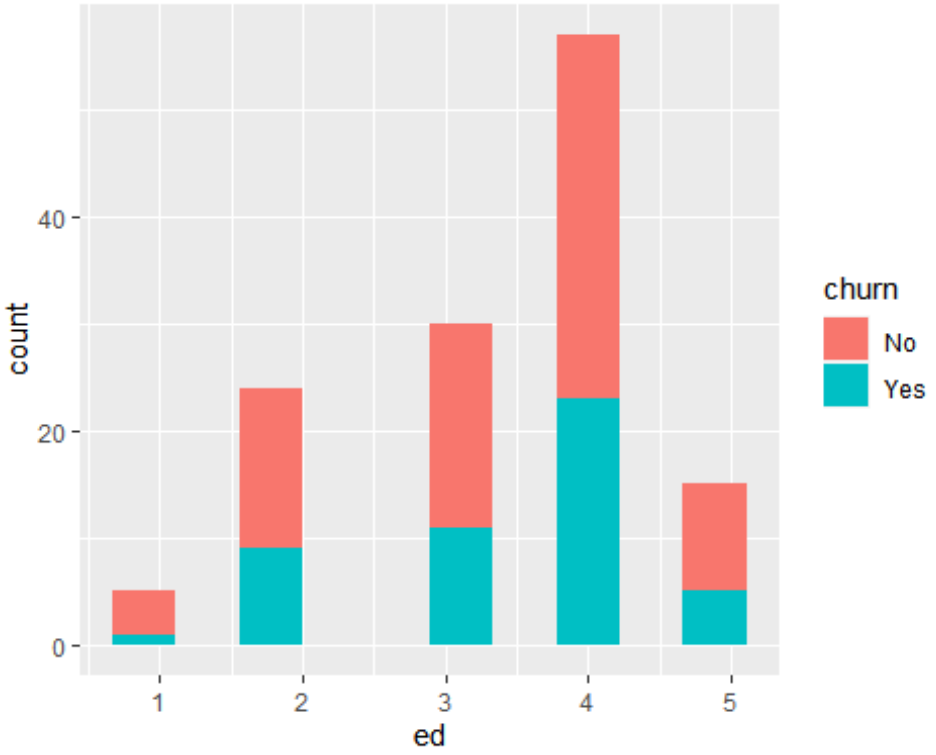

```
#Histogram of Numeric variables grouped by Churn
for (i in c(2:length(colnames(df)))){
  if (is.numeric(df[,i]) == T ){
    print(ggplot(df, aes(x = df[,i], fill = churn)) +
            geom_histogram(bins = 10) + labs(x = colnames(df)[i]))
  }
}
```

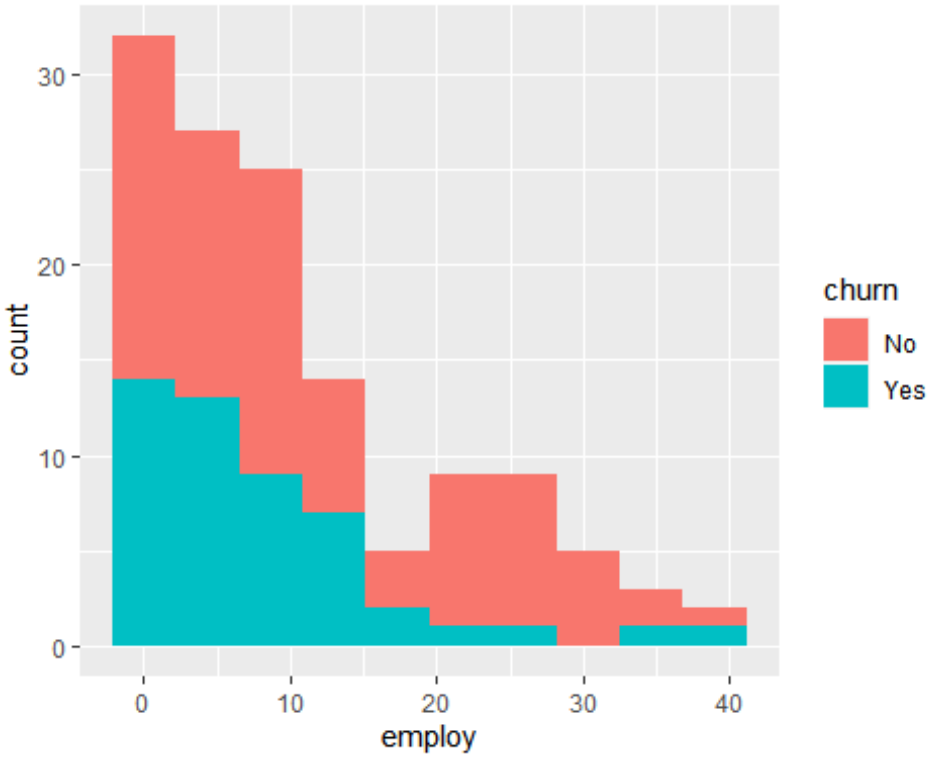

count
20
15
10
5
0
0
20
40
60
tenure
churn
No
Yes

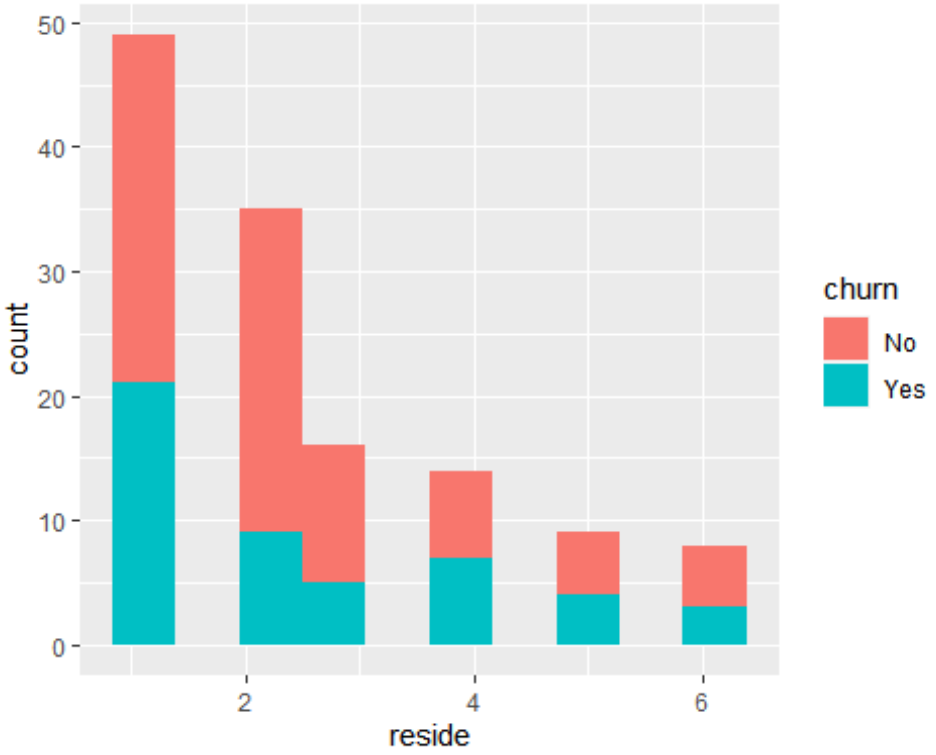

count
25
20
15
10
5
0
20
40
60
age
churn
No
Yes

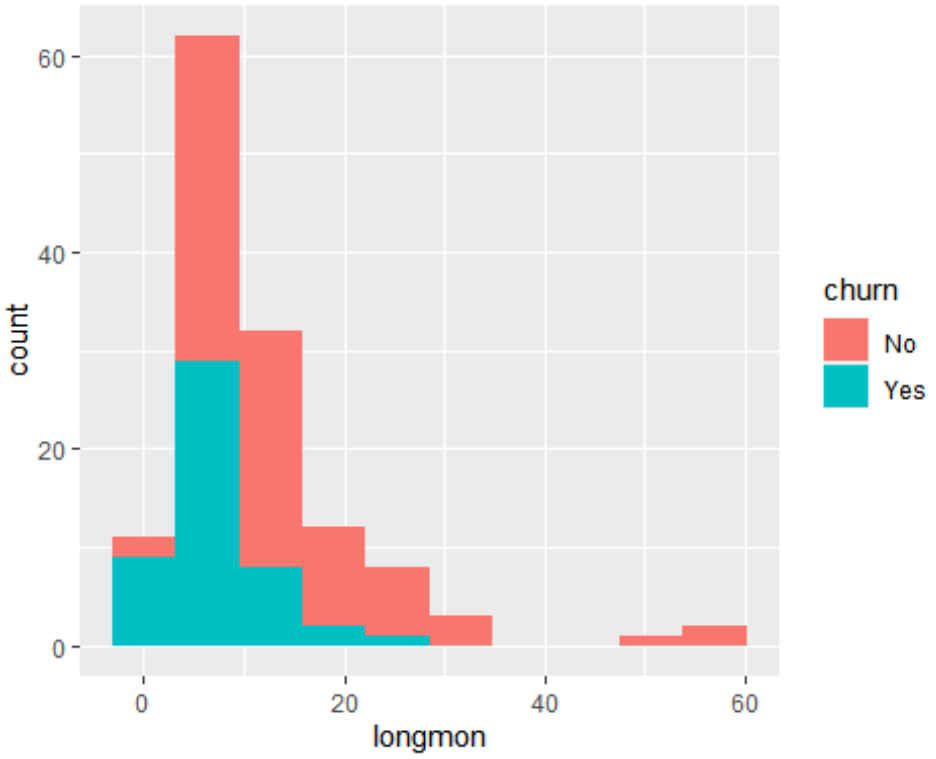

30
20
10
0
count
0
10
20
30
40
address
churn
No
Yes

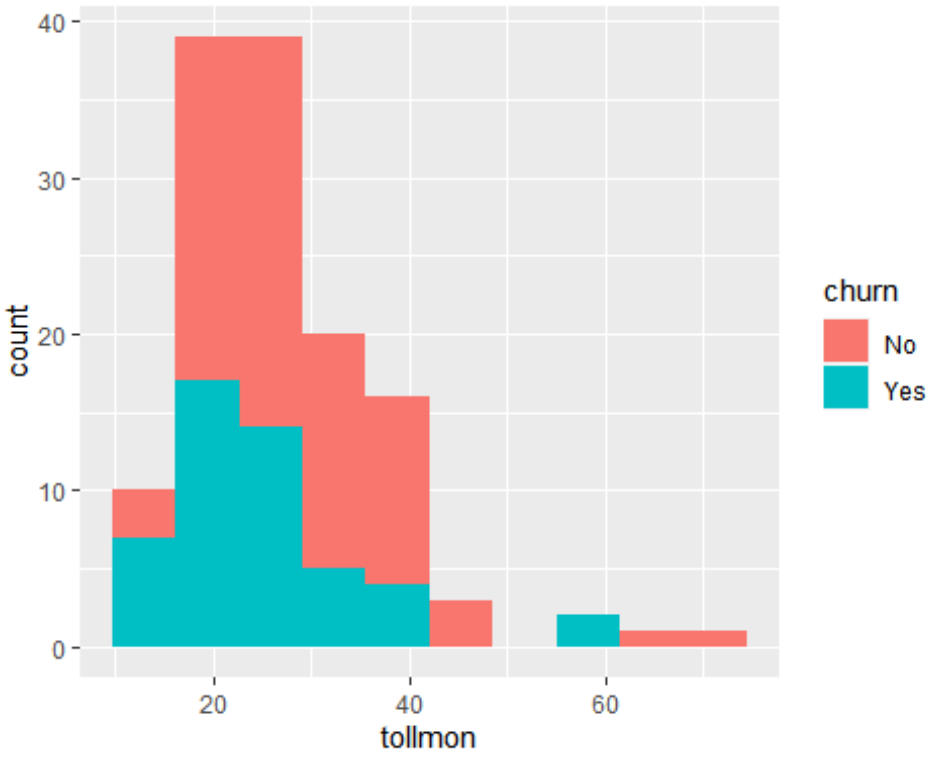

40
20
0
count
0
250
500
750
1000
income
churn
No
Yes

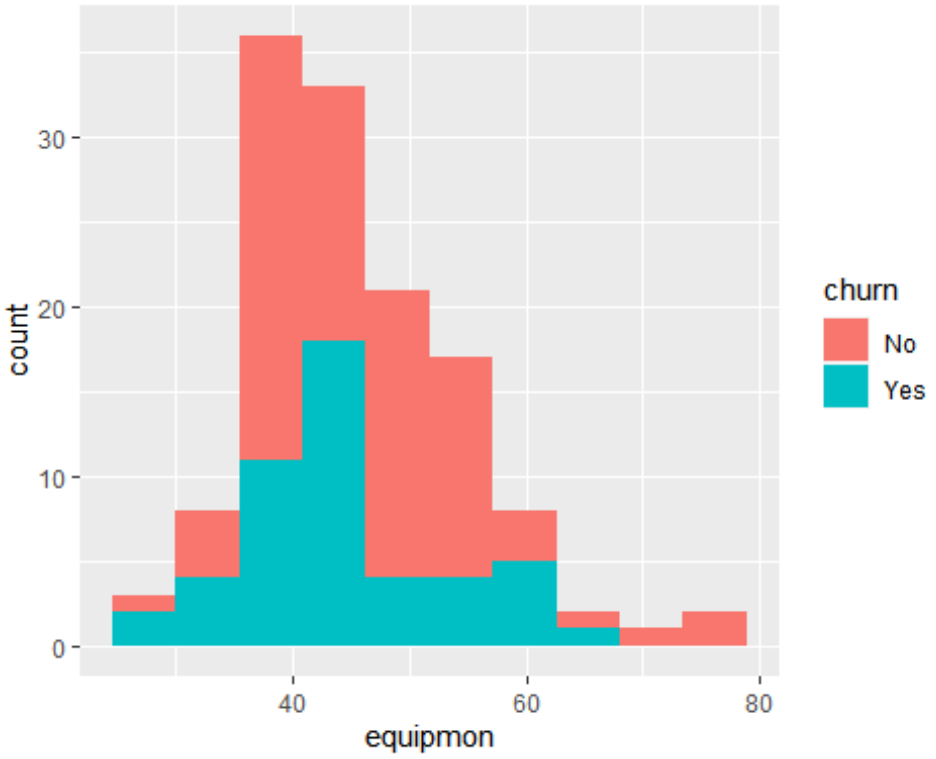

count
40
20
0
1
2
3
4
5
ed
churn
No
Yes

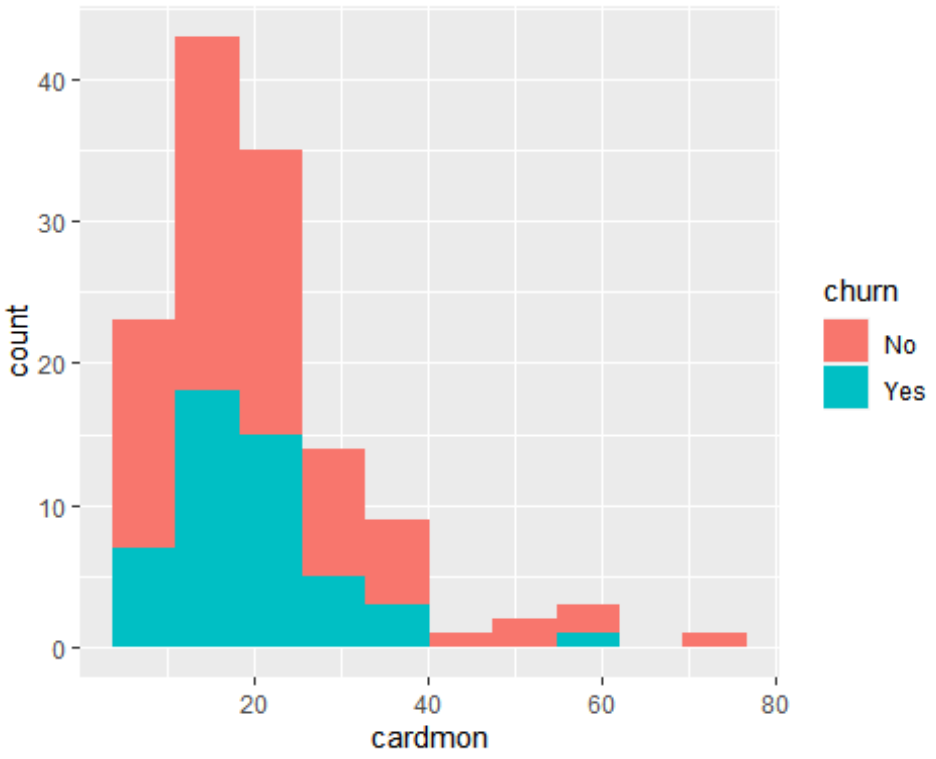

count
30
20
10
0
0
10
20
30
40
employ
churn
No
Yes

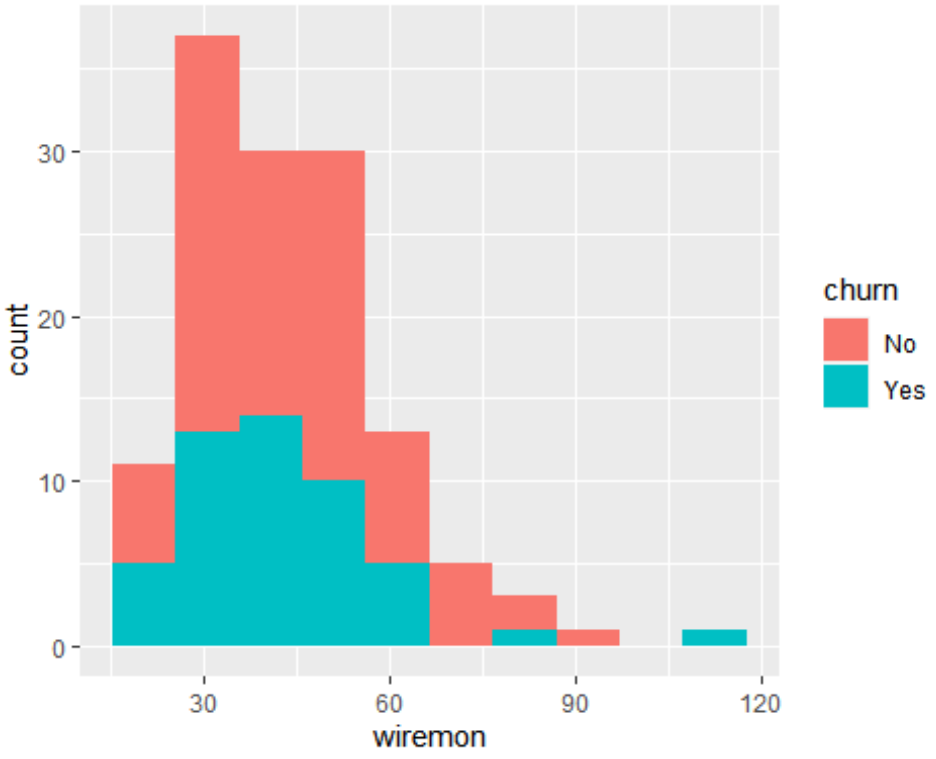

50
40
30
20
10
0
count
2
4
6
reside
churn
No
Yes

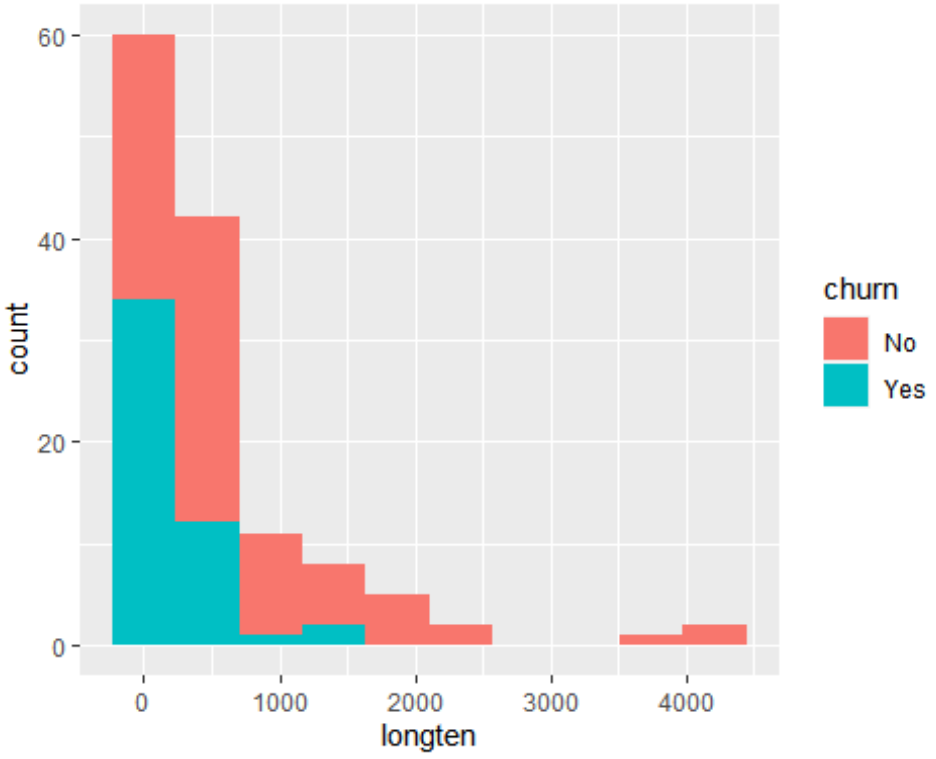

60
40
20
0
count
0
20
40
60
longmon
churn
No
Yes

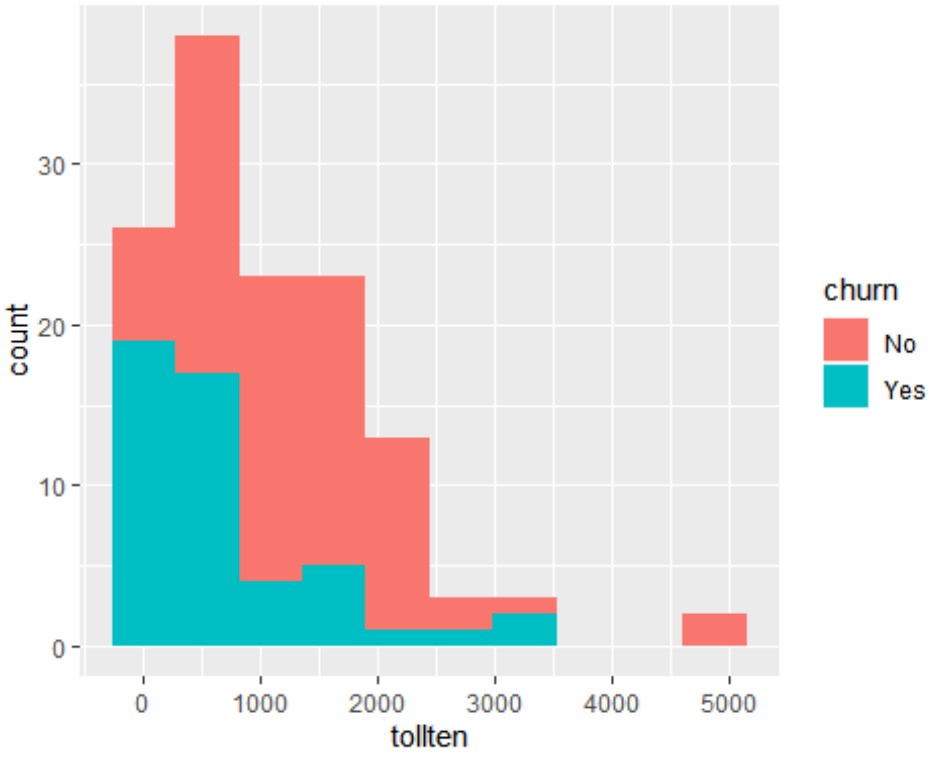

40
30
20
10
0
count
20
40
60
tollmon
churn
No
Yes

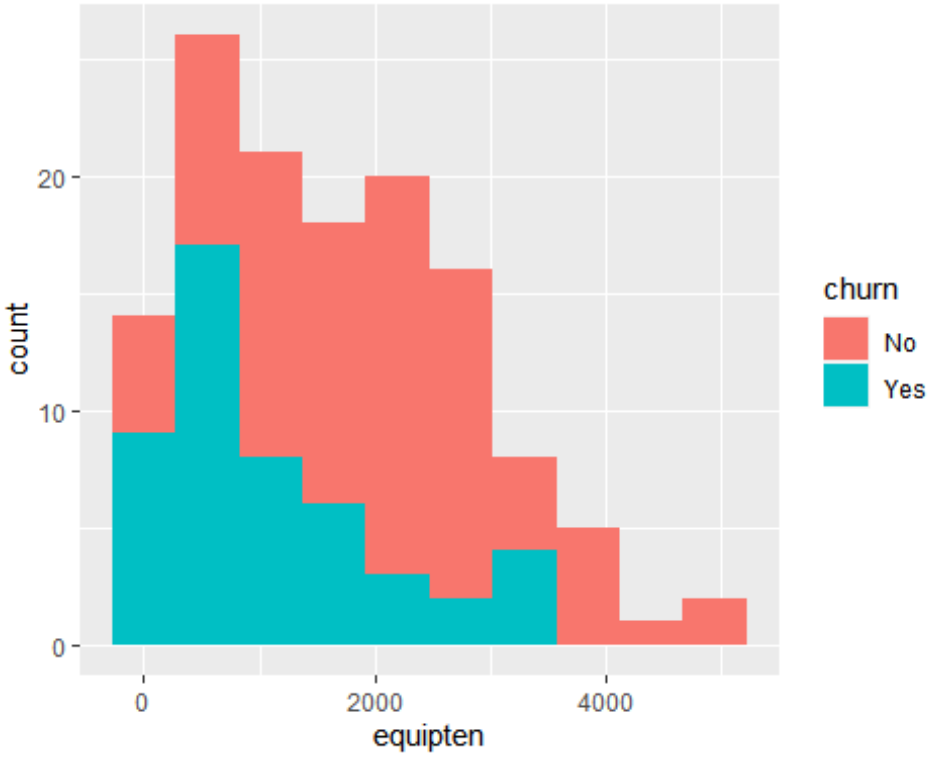

30
20
10
0
count
40
60
80
equipmon
churn
No
Yes

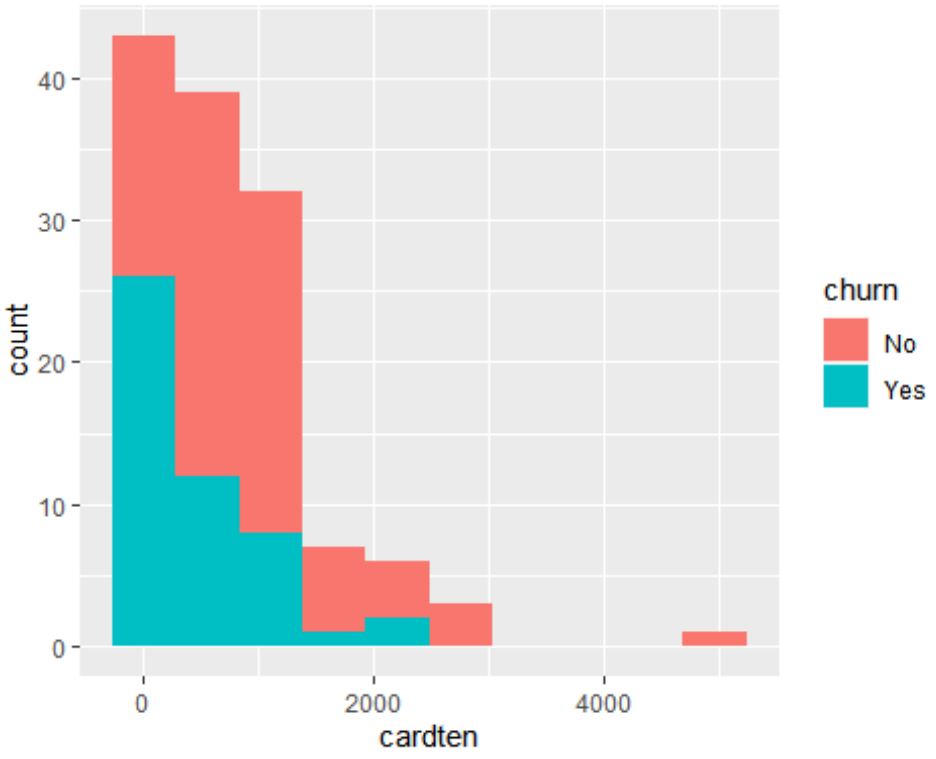

40
30
20
10
0
count
20
40
60
80
cardmon
churn
No
Yes

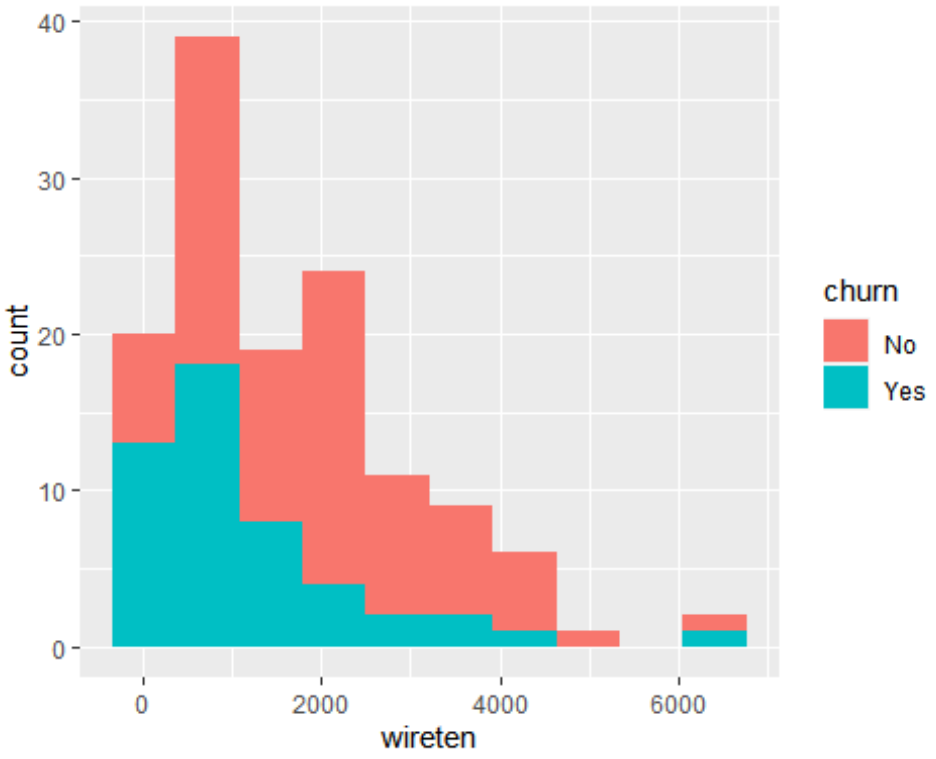

30
20
10
0
count
30
60
90
120
wiremon
churn
No
Yes

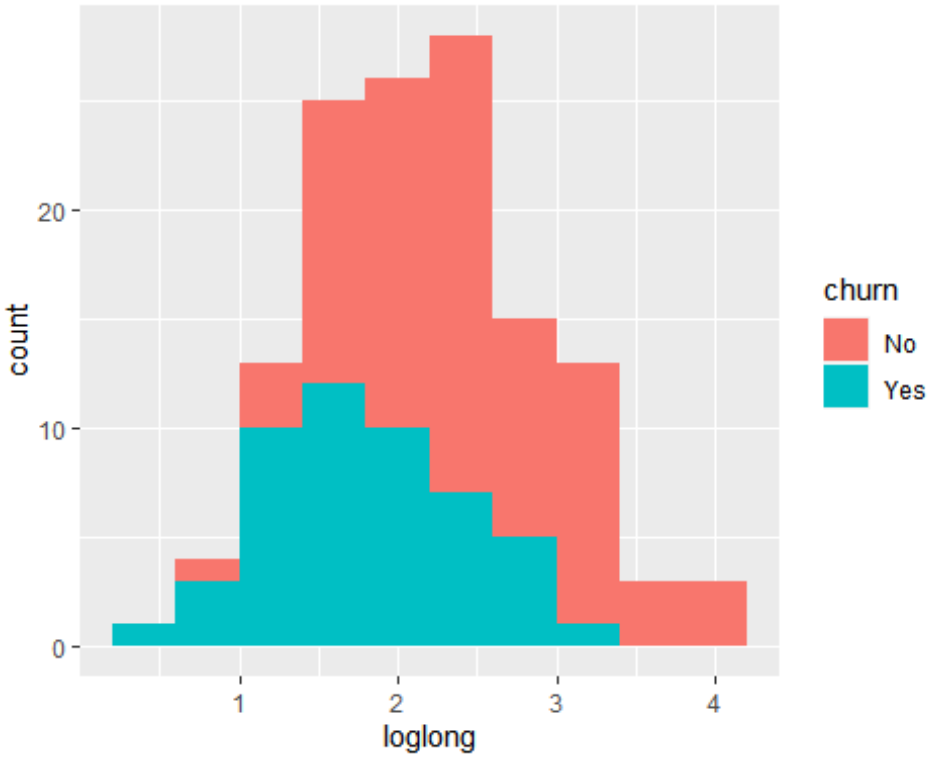

60
40
20
0
count
0
1000
2000
3000
4000
longten
churn
No
Yes

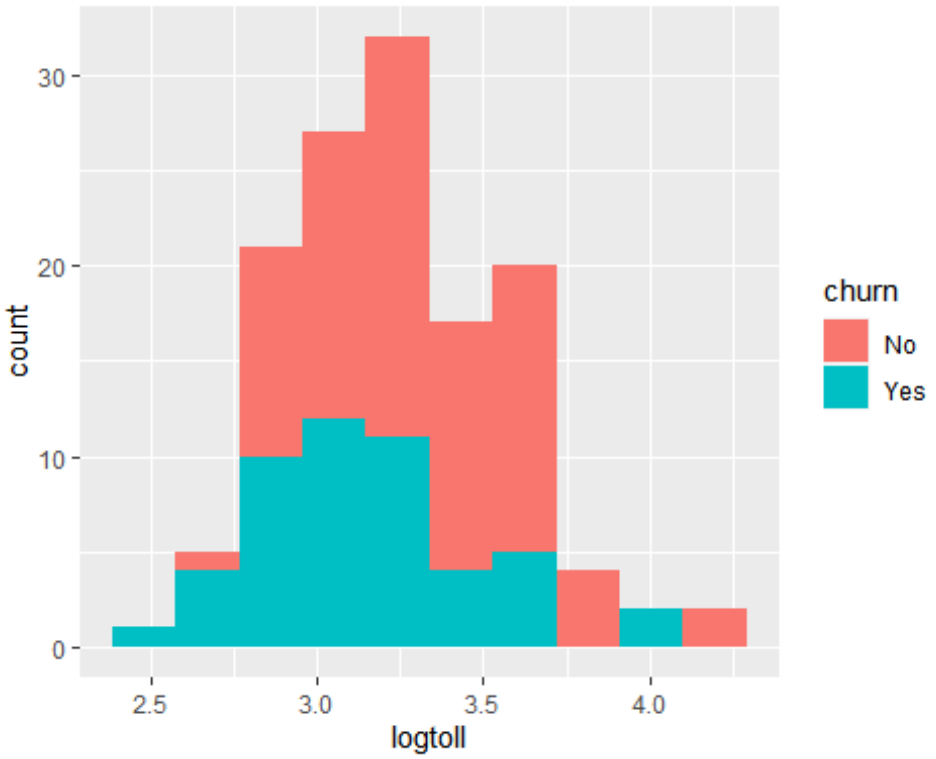

30
20
10
0
count
0
1000
2000
3000
4000
5000
tollten
churn
No
Yes

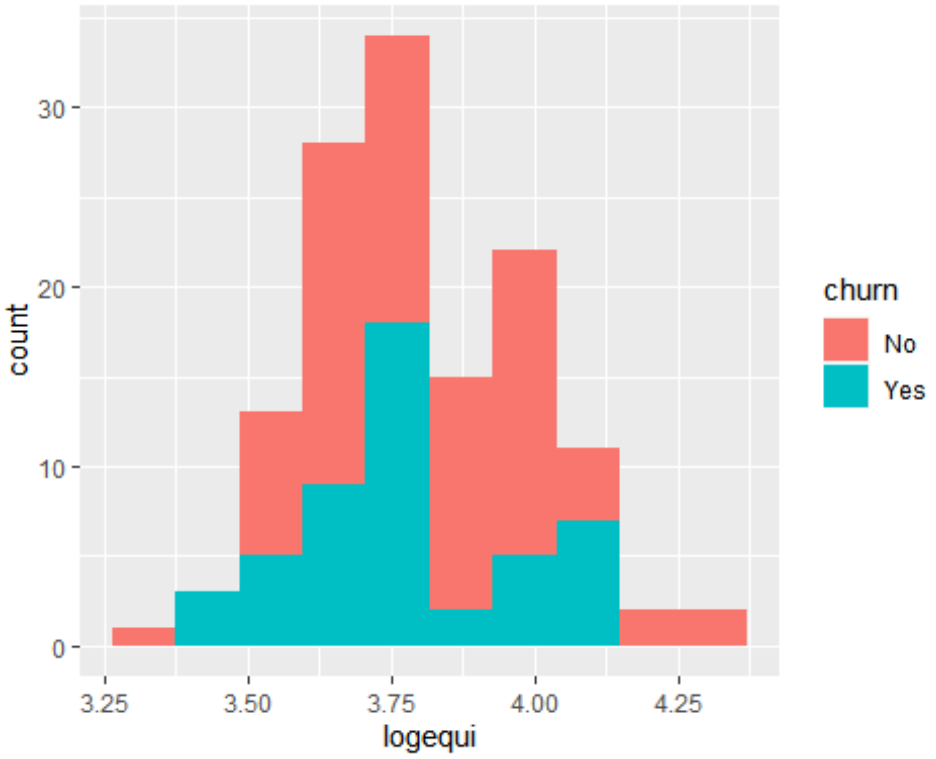

count
20
10
0
0
2000
4000
equipten
churn
No
Yes

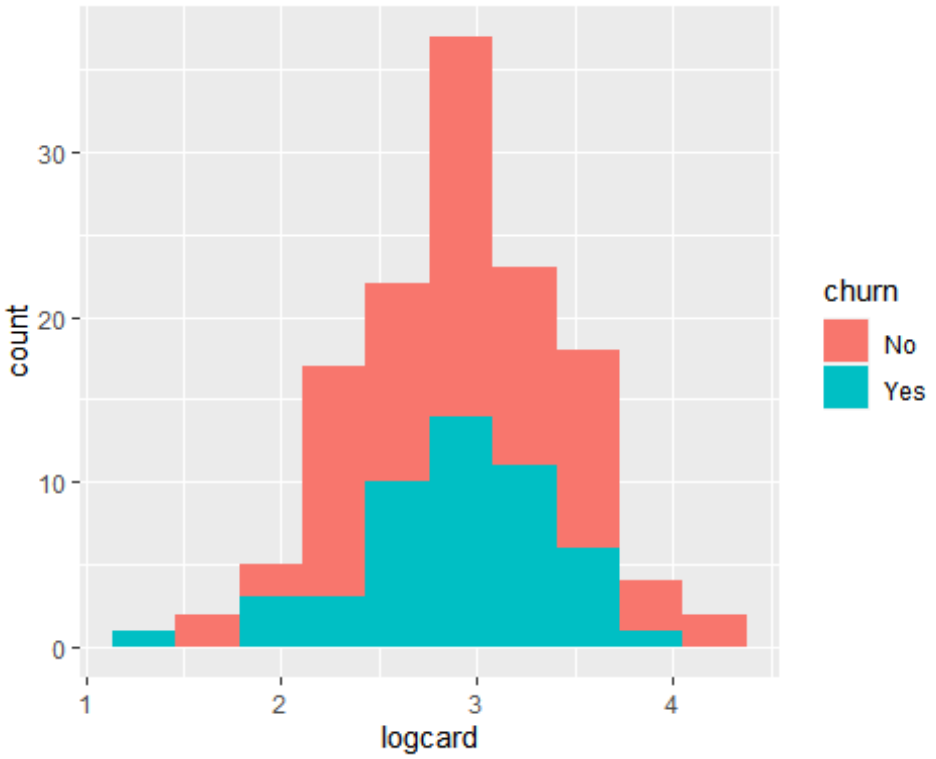

count
40
30
20
10
0
0
2000
4000
cardten
churn
No
Yes

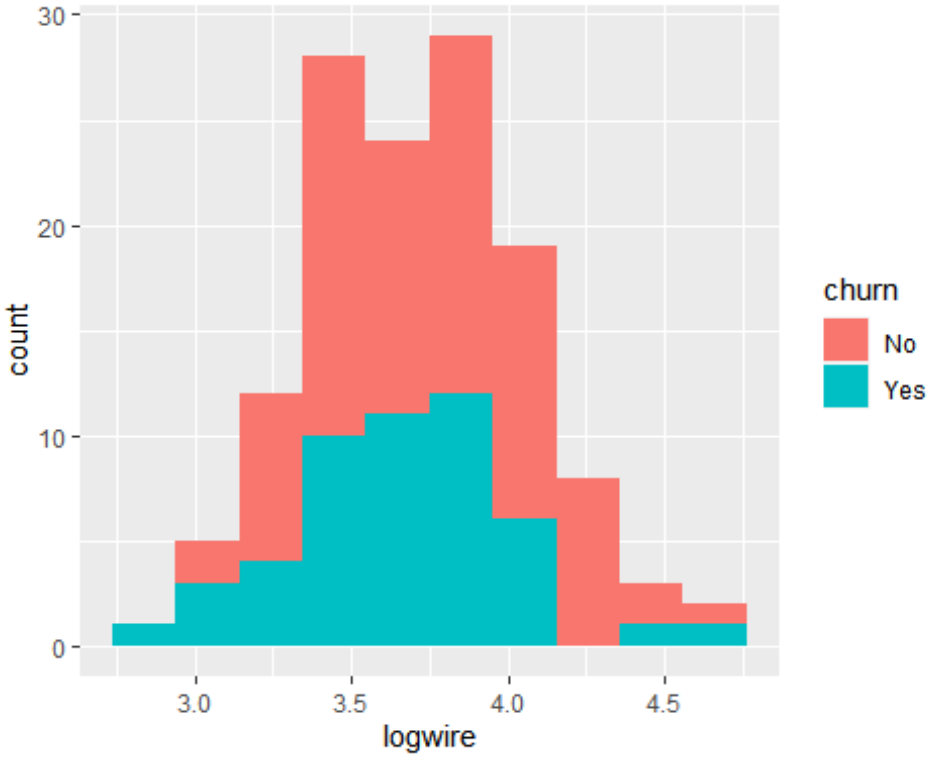

40
30
20
10
0
count
0
2000
4000
6000
wireten
churn
No
Yes

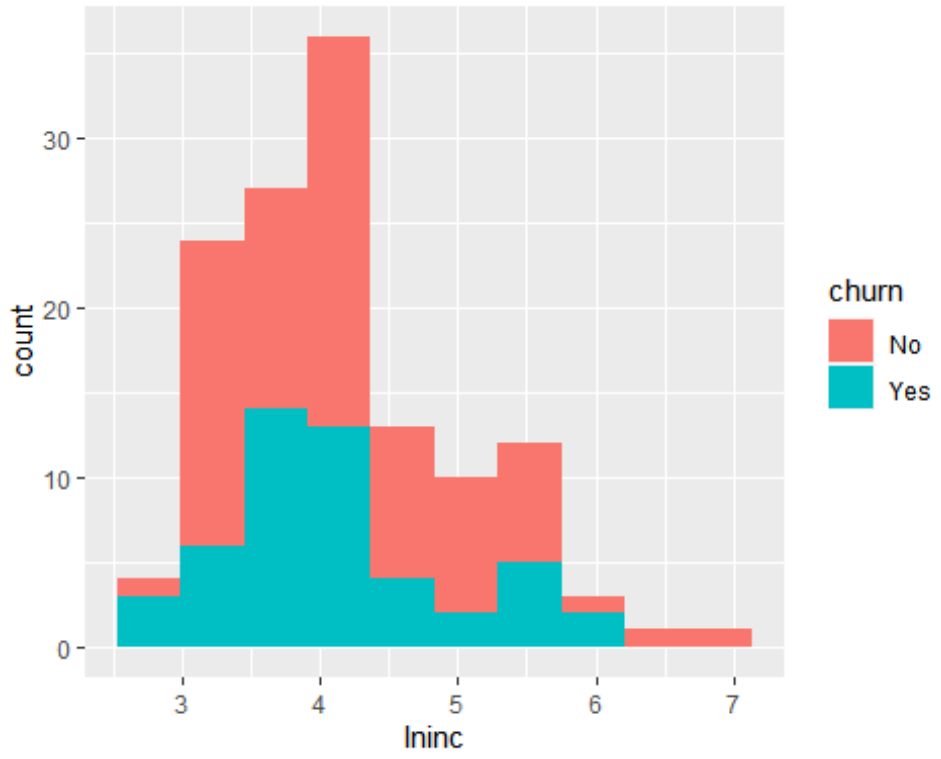

20
10
0
count
1
2
3
4
loglong
churn
No
Yes

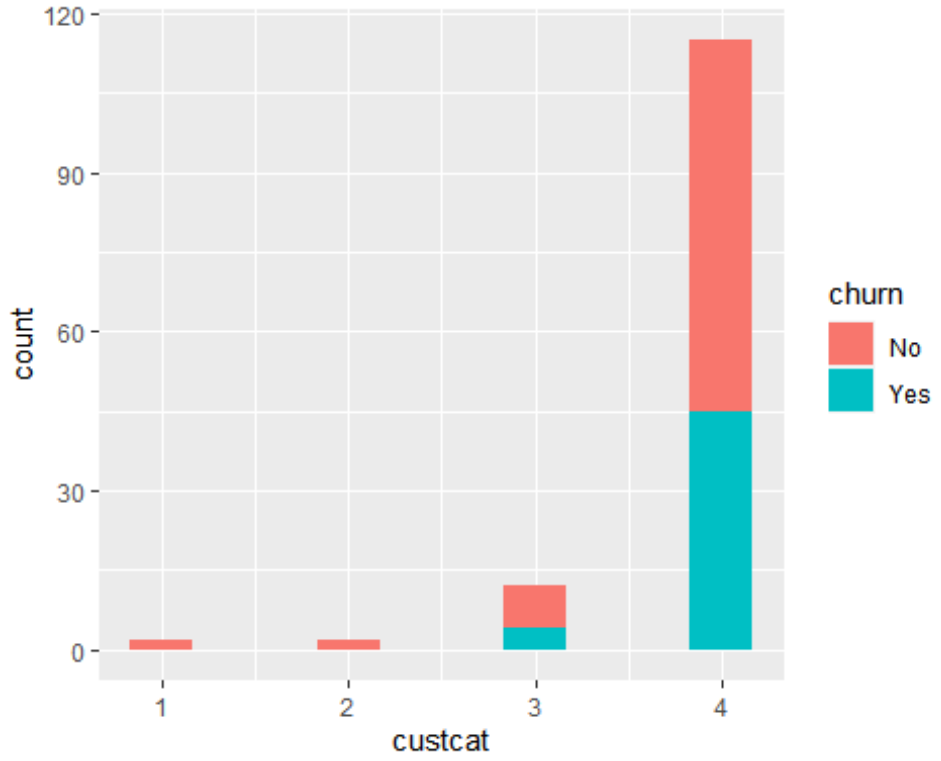

30
20
10
0
count
2.5
3.0
3.5
4.0
logtoll
churn
No
Yes


30
20
10
0
count
3.25
3.50
3.75
4.00
4.25
logequi
churn
No
Yes


count
30
20
10
0
1
2
3
4
logcard
churn
No
Yes


count
30
20
10
0
3.0
3.5
4.0
4.5
logwire
churn
No
Yes

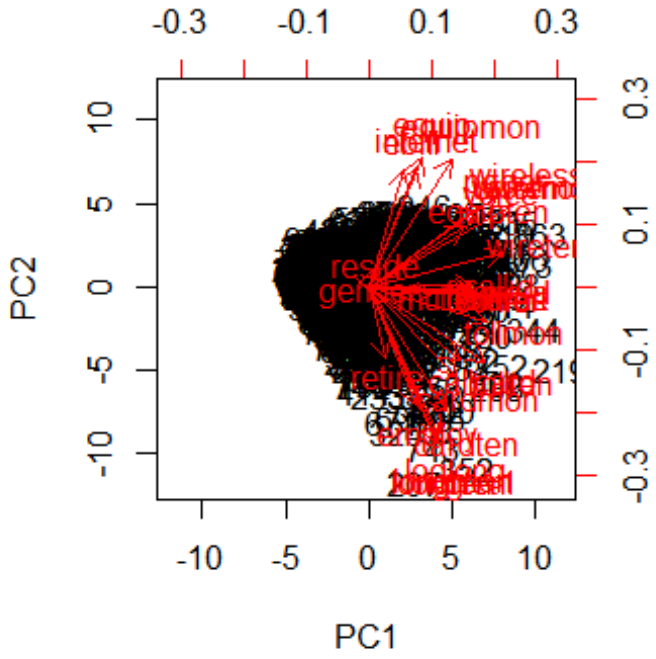




Exploratory data analysis (Dimensionality Reduction):

```
#reading data
df1 <- read.delim("C:/Users/AK DAS/Desktop/Business Analytics/telco.txt")
```

```r
data<-prcomp(df1[,c(8:35,20,21)],center=TRUE,scale.=TRUE)
library(devtools)

## Warning: package 'devtools' was built under R version 3.6.3

## Loading required package: usethis

## Warning: package 'usethis' was built under R version 3.6.3

str(data)

## List of 5
##  $ sdev    : num [1:30] 2.94 2.36 1.89 1.14 1.07 ...
##  $ rotation: num [1:30, 1:30] 0.09375 0.03141 0.00225 0.01066 0.21433 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:30] "employ" "retire" "gender" "reside" ...
##   .. ..$ : chr [1:30] "PC1" "PC2" "PC3" "PC4" ...
##  $ center  : Named num [1:30] 10.987 0.047 0.517 2.331 0.474 ...
##   ..- attr(*, "names")= chr [1:30] "employ" "retire" "gender" "reside" ...
##  $ scale   : Named num [1:30] 10.082 0.212 0.5 1.436 0.5 ...
##   ..- attr(*, "names")= chr [1:30] "employ" "retire" "gender" "reside" ...
##  $ x       : num [1:1000, 1:30] -2.953 1.892 0.704 -3.58 -2.462 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:30] "PC1" "PC2" "PC3" "PC4" ...
##  - attr(*, "class")= chr "prcomp"

biplot(data,scale=0)
```



The biplot suggests the directions of all the variables. It states that internet, Equipment last month etc. convey the same data value and explain the same variances as these vectors are in the same direction.

Summary:

• Female customers churn more than the male customers.

• Retire doesn't matter that much as there are only 3% of the customers who are retired.

• Only younger & middle aged customers are most likely to churn.

• Having or not having internet does not matter that much in customer churn as the percentage of the customers churning in both scenarios are almost 37%.

• Customers having less tenure are most likely to churn

• Customers using Pager,Wireless services,Voice Mail,Multiple lines,Calling card service,toll free service are most likely to churn

• Customers whose calls are waiting & forwarded are most likely to churn.

• People living in region 1 are most likely to churn.

• Customers having more income are most likely to churn.

• Customers belonging to only category 3 & 4 churn.

• In dimension reduction,the biplot suggests the directions of all the variables. It states that internet, Equipment last month etc. convey the same data value and explain the same variances as these vectors are in the same direction.

Telecommunication industry always suffers from a very high churn rates when one industry offers a better plan than the previous.There is a high possibility of the customer churning from the present to a better plan.In such a scenario it is very difficult to avoid losses but through prediction we can keep it to a minimal level.