Final_Project

February 19, 2019

```
# Performing  statistical and exploratory analysis using R.

# Data Set Info:Churn Dataset

##The data set contains 20 predictors worth of information about 3333 customers, along with the
##target variable, churn, an indication of whether that customer churned (left the company) or not.


#The variables are as follows:
#State: Categorical, for the 50 states and the District of Columbia.
#Account length: Integer-valued, how long account has been active.
#Area code: Categorical
#Phone number: Essentially a surrogate for customer ID.
#International plan: Dichotomous categorical, yes or no.
#Voice mail plan: Dichotomous categorical, yes or no.
#Number of voice mail messages: Integer-valued.
#Total day minutes: Continuous, minutes customer used service during the day.
#Total day calls: Integer-valued.
#Total day charge: Continuous, perhaps based on above two variables.
#Total eve minutes: Continuous, minutes customer used service during the evening.
#Total eve calls: Integer-valued.
#Total eve charge: Continuous, perhaps based on above two variables.
#Total night minutes: Continuous, minutes customer used service during the night.
#Total night calls: Integer-valued.
#Total night charge: Continuous, perhaps based on above two variables.
#Total international minutes: Continuous, minutes customer used service to make
#international calls.
#Total international calls: Integer-valued.
#Total international charge: Continuous, perhaps based on above two variables.
#Number of calls to customer service: Integer-valued.
#Churn: Target. Indicator of whether the customer has left the company (true or false).



library(ggplot2)
library(gridExtra)
#install.packages("dplyr")
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:gridExtra':
##
##     combine
```

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

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

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
#install.packages("caret")
library(caret)
```

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

```
library(rpart)
```

```
# Importing and reading data

library(readr)

churn <- read.csv("C:/Users/Rahul/Downloads/churn.txt", stringsAsFactors=TRUE)

View(churn)
```

```
#To start with first read the file and take a look at the field values for few records.

# Show the first ten records
churn[1:10,]
```

```
##    State Account.Length Area.Code    Phone Int.l.Plan VMail.Plan
## 1     KS            128       415 382-4657         no        yes
## 2     OH            107       415 371-7191         no        yes
## 3     NJ            137       415 358-1921         no         no
## 4     OH             84       408 375-9999        yes         no
## 5     OK             75       415 330-6626        yes         no
## 6     AL            118       510 391-8027        yes         no
## 7     MA            121       510 355-9993         no        yes
## 8     MO            147       415 329-9001        yes         no
## 9     LA            117       408 335-4719         no         no
## 10    WV            141       415 330-8173        yes        yes
##    VMail.Message Day.Mins Day.Calls Day.Charge Eve.Mins Eve.Calls
## 1             25    265.1       110      45.07    197.4        99
## 2             26    161.6       123      27.47    195.5       103
## 3              0    243.4       114      41.38    121.2       110
## 4              0    299.4        71      50.90     61.9        88
## 5              0    166.7       113      28.34    148.3       122
## 6              0    223.4        98      37.98    220.6       101
## 7             24    218.2        88      37.09    348.5       108
## 8              0    157.0        79      26.69    103.1        94
## 9              0    184.5        97      31.37    351.6        80
## 10            37    258.6        84      43.96    222.0       111
##    Eve.Charge Night.Mins Night.Calls Night.Charge Intl.Mins Intl.Calls
## 1       16.78      244.7          91        11.01      10.0          3
## 2       16.62      254.4         103        11.45      13.7          3
## 3       10.30      162.6         104         7.32      12.2          5
## 4        5.26      196.9          89         8.86       6.6          7
## 5       12.61      186.9         121         8.41      10.1          3
## 6       18.75      203.9         118         9.18       6.3          6
```

```
## 7       29.62      212.6         118         9.57       7.5          7
## 8        8.76      211.8          96         9.53       7.1          6
## 9       29.89      215.8          90         9.71       8.7          4
## 10      18.87      326.4          97        14.69      11.2          5
##    Intl.Charge CustServ.Calls Churn.
## 1         2.70              1 False.
## 2         3.70              1 False.
## 3         3.29              0 False.
## 4         1.78              2 False.
## 5         2.73              3 False.
## 6         1.70              0 False.
## 7         2.03              3 False.
## 8         1.92              0 False.
## 9         2.35              1 False.
## 10        3.02              0 False.
```

```
head(churn)
```

```
##   State Account.Length Area.Code    Phone Int.l.Plan VMail.Plan
## 1    KS            128       415 382-4657         no        yes
## 2    OH            107       415 371-7191         no        yes
## 3    NJ            137       415 358-1921         no         no
## 4    OH             84       408 375-9999        yes         no
## 5    OK             75       415 330-6626        yes         no
## 6    AL            118       510 391-8027        yes         no
##   VMail.Message Day.Mins Day.Calls Day.Charge Eve.Mins Eve.Calls
## 1            25    265.1       110      45.07    197.4        99
## 2            26    161.6       123      27.47    195.5       103
## 3             0    243.4       114      41.38    121.2       110
## 4             0    299.4        71      50.90     61.9        88
## 5             0    166.7       113      28.34    148.3       122
## 6             0    223.4        98      37.98    220.6       101
##   Eve.Charge Night.Mins Night.Calls Night.Charge Intl.Mins Intl.Calls
## 1      16.78      244.7          91        11.01      10.0          3
## 2      16.62      254.4         103        11.45      13.7          3
## 3      10.30      162.6         104         7.32      12.2          5
## 4       5.26      196.9          89         8.86       6.6          7
## 5      12.61      186.9         121         8.41      10.1          3
## 6      18.75      203.9         118         9.18       6.3          6
##   Intl.Charge CustServ.Calls Churn.
## 1        2.70              1 False.
## 2        3.70              1 False.
## 3        3.29              0 False.
## 4        1.78              2 False.
## 5        2.73              3 False.
## 6        1.70              0 False.
```

```
# Show last ten records

tail(churn)
```

```
##      State Account.Length Area.Code    Phone Int.l.Plan VMail.Plan
## 3328    SC             79       415 348-3830         no         no
## 3329    AZ            192       415 414-4276         no        yes
## 3330    WV             68       415 370-3271         no         no
## 3331    RI             28       510 328-8230         no         no
```

```
## 3332    CT            184       510 364-6381        yes         no
## 3333    TN             74       415 400-4344         no        yes
##      VMail.Message Day.Mins Day.Calls Day.Charge Eve.Mins Eve.Calls
## 3328             0    134.7        98      22.90    189.7        68
## 3329            36    156.2        77      26.55    215.5       126
## 3330             0    231.1        57      39.29    153.4        55
## 3331             0    180.8       109      30.74    288.8        58
## 3332             0    213.8       105      36.35    159.6        84
## 3333            25    234.4       113      39.85    265.9        82
##      Eve.Charge Night.Mins Night.Calls Night.Charge Intl.Mins Intl.Calls
## 3328      16.12      221.4         128         9.96      11.8          5
## 3329      18.32      279.1          83        12.56       9.9          6
## 3330      13.04      191.3         123         8.61       9.6          4
## 3331      24.55      191.9          91         8.64      14.1          6
## 3332      13.57      139.2         137         6.26       5.0         10
## 3333      22.60      241.4          77        10.86      13.7          4
##      Intl.Charge CustServ.Calls Churn.
## 3328        3.19              2 False.
## 3329        2.67              2 False.
## 3330        2.59              3 False.
## 3331        3.81              2 False.
## 3332        1.35              2 False.
## 3333        3.70              0 False.
```

```
# observing structure of the dataset
str(churn)
```

```
## 'data.frame':    3333 obs. of  21 variables:
##  $ State         : Factor w/ 51 levels "AK","AL","AR",..: 17 36 32 36 37 2 20 25 19 50 ...
##  $ Account.Length: int  128 107 137 84 75 118 121 147 117 141 ...
##  $ Area.Code     : int  415 415 415 408 415 510 510 415 408 415 ...
##  $ Phone         : Factor w/ 3333 levels "327-1058","327-1319",..: 1927 1576 1118 1708 111 2254 1048
##  $ Int.l.Plan    : Factor w/ 2 levels "no","yes": 1 1 1 2 2 2 1 2 1 2 ...
##  $ VMail.Plan    : Factor w/ 2 levels "no","yes": 2 2 1 1 1 1 2 1 1 2 ...
##  $ VMail.Message : int  25 26 0 0 0 0 24 0 0 37 ...
##  $ Day.Mins      : num  265 162 243 299 167 ...
##  $ Day.Calls     : int  110 123 114 71 113 98 88 79 97 84 ...
##  $ Day.Charge    : num  45.1 27.5 41.4 50.9 28.3 ...
##  $ Eve.Mins      : num  197.4 195.5 121.2 61.9 148.3 ...
##  $ Eve.Calls     : int  99 103 110 88 122 101 108 94 80 111 ...
##  $ Eve.Charge    : num  16.78 16.62 10.3 5.26 12.61 ...
##  $ Night.Mins    : num  245 254 163 197 187 ...
##  $ Night.Calls   : int  91 103 104 89 121 118 118 96 90 97 ...
##  $ Night.Charge  : num  11.01 11.45 7.32 8.86 8.41 ...
##  $ Intl.Mins     : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 8.7 11.2 ...
##  $ Intl.Calls    : int  3 3 5 7 3 6 7 6 4 5 ...
##  $ Intl.Charge   : num  2.7 3.7 3.29 1.78 2.73 1.7 2.03 1.92 2.35 3.02 ...
##  $ CustServ.Calls: int  1 1 0 2 3 0 3 0 1 0 ...
##  $ Churn.        : Factor w/ 2 levels "False.","True.": 1 1 1 1 1 1 1 1 1 1 ...
```

```
# Observing the structure
summary(churn)
```

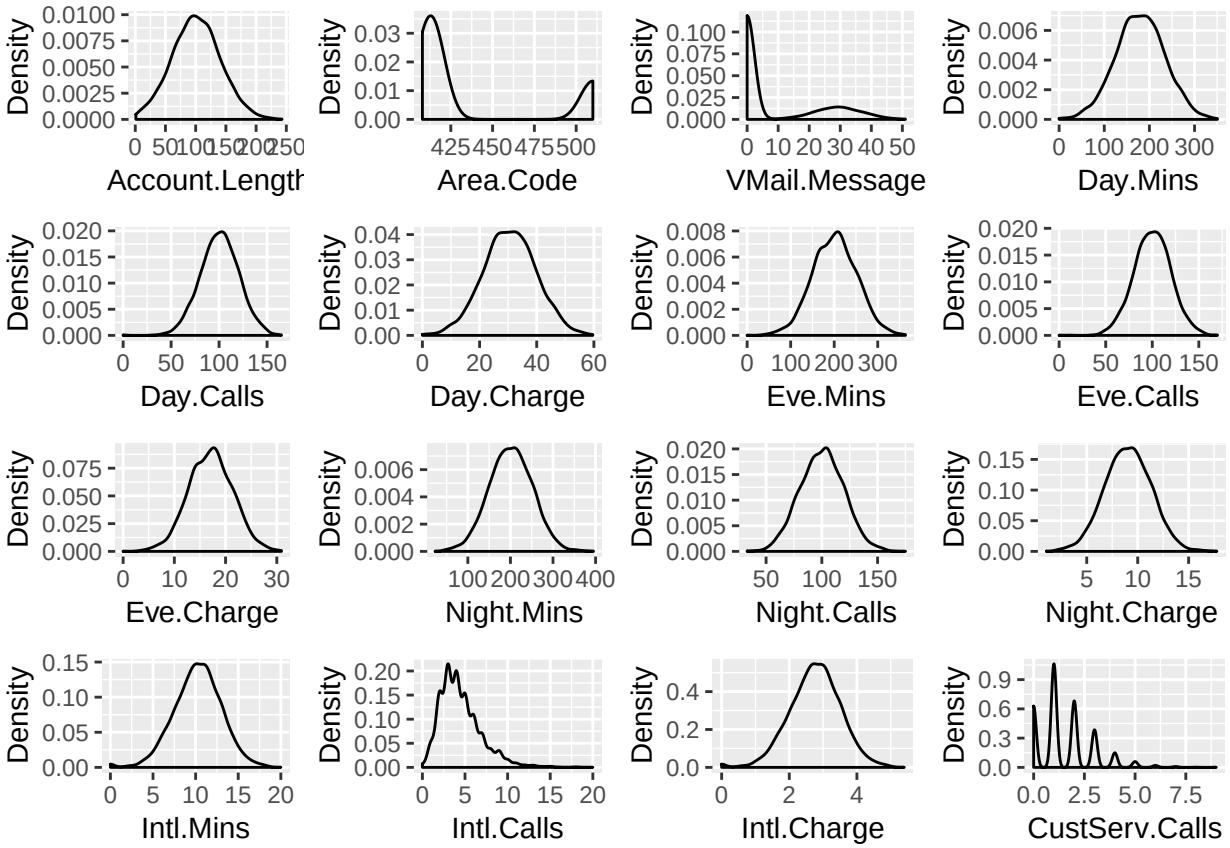
```
##      State      Account.Length    Area.Code          Phone      Int.l.Plan
##  WV     : 106   Min.   :  1.0   Min.   :408.0   327-1058:   1   no :3010
##  MN     :  84   1st Qu.: 74.0   1st Qu.:408.0   327-1319:   1   yes: 323
```

```
##  NY     :  83   Median :101.0   Median :415.0   327-3053:   1
##  AL     :  80   Mean   :101.1   Mean   :437.2   327-3587:   1
##  OH     :  78   3rd Qu.:127.0   3rd Qu.:510.0   327-3850:   1
##  OR     :  78   Max.   :243.0   Max.   :510.0   327-3954:   1
##  (Other):2824                                   (Other) :3327
##  VMail.Plan VMail.Message       Day.Mins       Day.Calls
##  no :2411   Min.   : 0.000   Min.   :  0.0   Min.   :  0.0
##  yes: 922   1st Qu.: 0.000   1st Qu.:143.7   1st Qu.: 87.0
##             Median : 0.000   Median :179.4   Median :101.0
##             Mean   : 8.099   Mean   :179.8   Mean   :100.4
##             3rd Qu.:20.000   3rd Qu.:216.4   3rd Qu.:114.0
##             Max.   :51.000   Max.   :350.8   Max.   :165.0
##
##    Day.Charge       Eve.Mins       Eve.Calls       Eve.Charge
##  Min.   : 0.00   Min.   :  0.0   Min.   :  0.0   Min.   : 0.00
##  1st Qu.:24.43   1st Qu.:166.6   1st Qu.: 87.0   1st Qu.:14.16
##  Median :30.50   Median :201.4   Median :100.0   Median :17.12
##  Mean   :30.56   Mean   :201.0   Mean   :100.1   Mean   :17.08
##  3rd Qu.:36.79   3rd Qu.:235.3   3rd Qu.:114.0   3rd Qu.:20.00
##  Max.   :59.64   Max.   :363.7   Max.   :170.0   Max.   :30.91
##
##    Night.Mins     Night.Calls     Night.Charge      Intl.Mins
##  Min.   : 23.2   Min.   : 33.0   Min.   : 1.040   Min.   : 0.00
##  1st Qu.:167.0   1st Qu.: 87.0   1st Qu.: 7.520   1st Qu.: 8.50
##  Median :201.2   Median :100.0   Median : 9.050   Median :10.30
##  Mean   :200.9   Mean   :100.1   Mean   : 9.039   Mean   :10.24
##  3rd Qu.:235.3   3rd Qu.:113.0   3rd Qu.:10.590   3rd Qu.:12.10
##  Max.   :395.0   Max.   :175.0   Max.   :17.770   Max.   :20.00
##
##    Intl.Calls      Intl.Charge    CustServ.Calls     Churn.
##  Min.   : 0.000   Min.   :0.000   Min.   :0.000   False.:2850
##  1st Qu.: 3.000   1st Qu.:2.300   1st Qu.:1.000   True. : 483
##  Median : 4.000   Median :2.780   Median :1.000
##  Mean   : 4.479   Mean   :2.765   Mean   :1.563
##  3rd Qu.: 6.000   3rd Qu.:3.270   3rd Qu.:2.000
##  Max.   :20.000   Max.   :5.400   Max.   :9.000
##
```

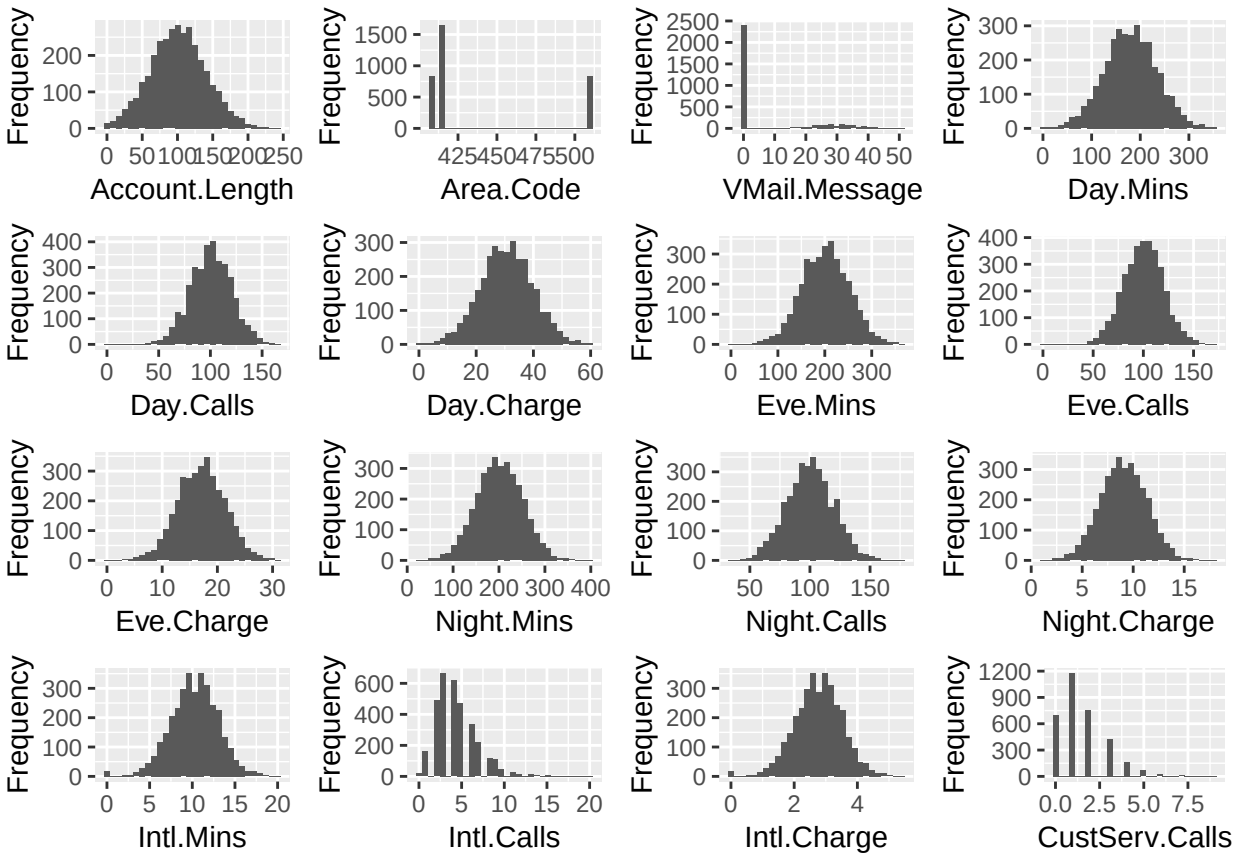
```
# Summarize the Churn variable
sum.churn <- summary(churn$Churn)
sum.churn
```

```
## False.  True.
##   2850    483
```

```
# Plotting some useful plots to explore the data

#install.packages("DataExplorer")
library(DataExplorer)
plot_density(churn)
```

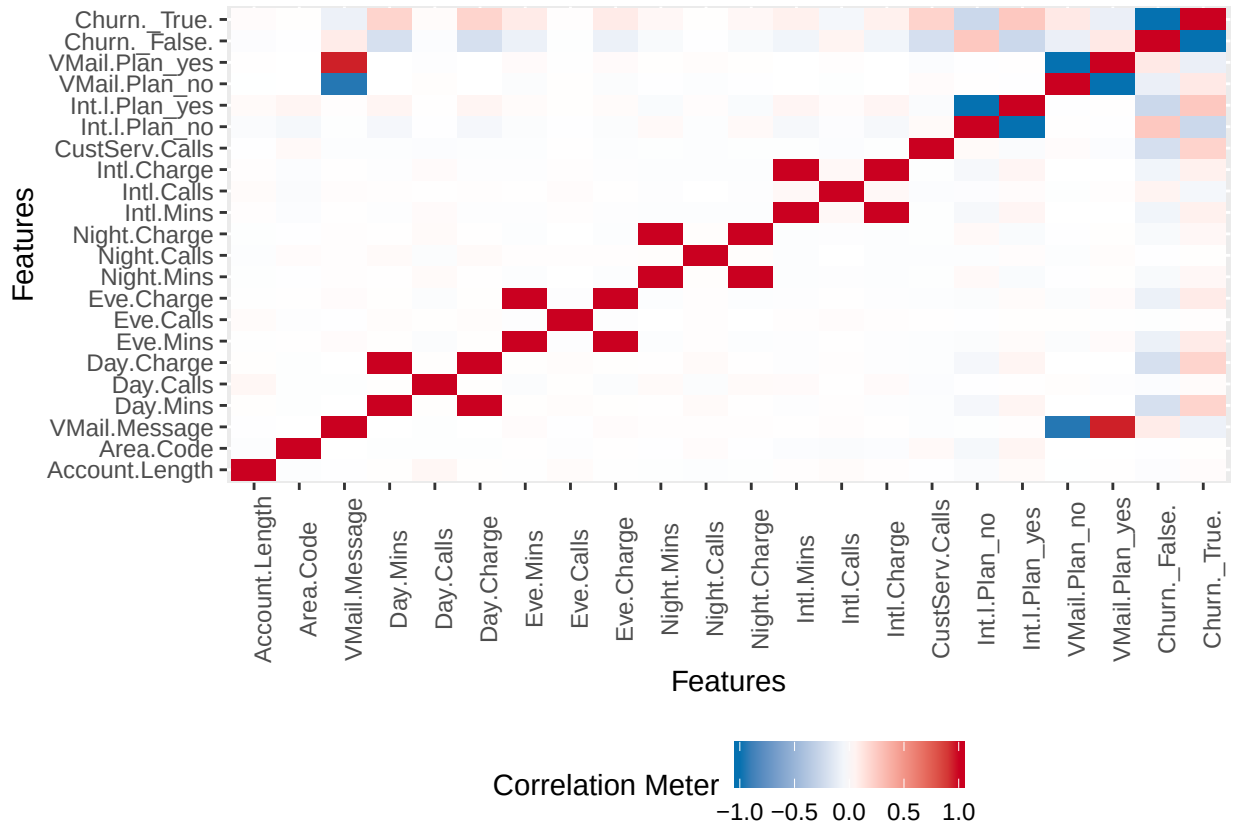



```
plot_histogram(churn)
```




```
plot_correlation(churn)
```

```
## 2 features with more than 20 categories ignored!
## State: 51 categories
## Phone: 3333 categories
```




```
# Calculate proportion of churners
prop.churn <- sum(churn$Churn == "True") / length(churn$Churn)
prop.churn
```

```
## [1] 0
```

```
#Checking missing values

sum(is.na(churn))
```

```
## [1] 0
```

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

```
##          State Account.Length      Area.Code          Phone     Int.l.Plan
##              0              0              0              0              0
##     VMail.Plan  VMail.Message       Day.Mins      Day.Calls     Day.Charge
##              0              0              0              0              0
##       Eve.Mins      Eve.Calls     Eve.Charge     Night.Mins    Night.Calls
##              0              0              0              0              0
##   Night.Charge      Intl.Mins     Intl.Calls    Intl.Charge CustServ.Calls
##              0              0              0              0              0
##         Churn.
##              0
```

```
# We can see that there are no NA values in the dataset
```

```
# Adding NAs:
```

```r
churn.proc <- churn[, c(2,4,8,9,21)]
head(churn.proc)
```

```
##   Account.Length    Phone Day.Mins Day.Calls Churn.
## 1            128 382-4657    265.1       110 False.
## 2            107 371-7191    161.6       123 False.
## 3            137 358-1921    243.4       114 False.
## 4             84 375-9999    299.4        71 False.
## 5             75 330-6626    166.7       113 False.
## 6            118 391-8027    223.4        98 False.
```

```r
churn.proc[2,5] <- churn.proc[2,3] <- NA
head(churn.proc)
```

```
##   Account.Length    Phone Day.Mins Day.Calls Churn.
## 1            128 382-4657    265.1       110 False.
## 2            107 371-7191       NA       123   <NA>
## 3            137 358-1921    243.4       114 False.
## 4             84 375-9999    299.4        71 False.
## 5             75 330-6626    166.7       113 False.
## 6            118 391-8027    223.4        98 False.
```

```r
# Replacement with constants

churn.proc[2,3] <- 0
churn.proc[2,5] <- "missing"
```

```
## Warning in `[<-.factor`(`*tmp*`, iseq, value = "missing"): invalid factor
## level, NA generated
```

```r
head(churn.proc)
```

```
##   Account.Length    Phone Day.Mins Day.Calls Churn.
## 1            128 382-4657    265.1       110 False.
## 2            107 371-7191      0.0       123   <NA>
## 3            137 358-1921    243.4       114 False.
## 4             84 375-9999    299.4        71 False.
## 5             75 330-6626    166.7       113 False.
## 6            118 391-8027    223.4        98 False.
```

```r
# Replacement with mean, mode
churn.proc[2,3] <- mean(na.omit(churn.proc$Day.Mins))
our_table <- table(churn.proc$Churn.)
our_mode <- names(our_table)[our_table == max(our_table)]
churn.proc[2,5] <- our_mode
head(churn.proc)
```

```
##   Account.Length    Phone Day.Mins Day.Calls Churn.
## 1            128 382-4657 265.1000       110 False.
## 2            107 371-7191 179.7266       123 False.
## 3            137 358-1921 243.4000       114 False.
## 4             84 375-9999 299.4000        71 False.
## 5             75 330-6626 166.7000       113 False.
## 6            118 391-8027 223.4000        98 False.
```

```r
# Generating random variables
```

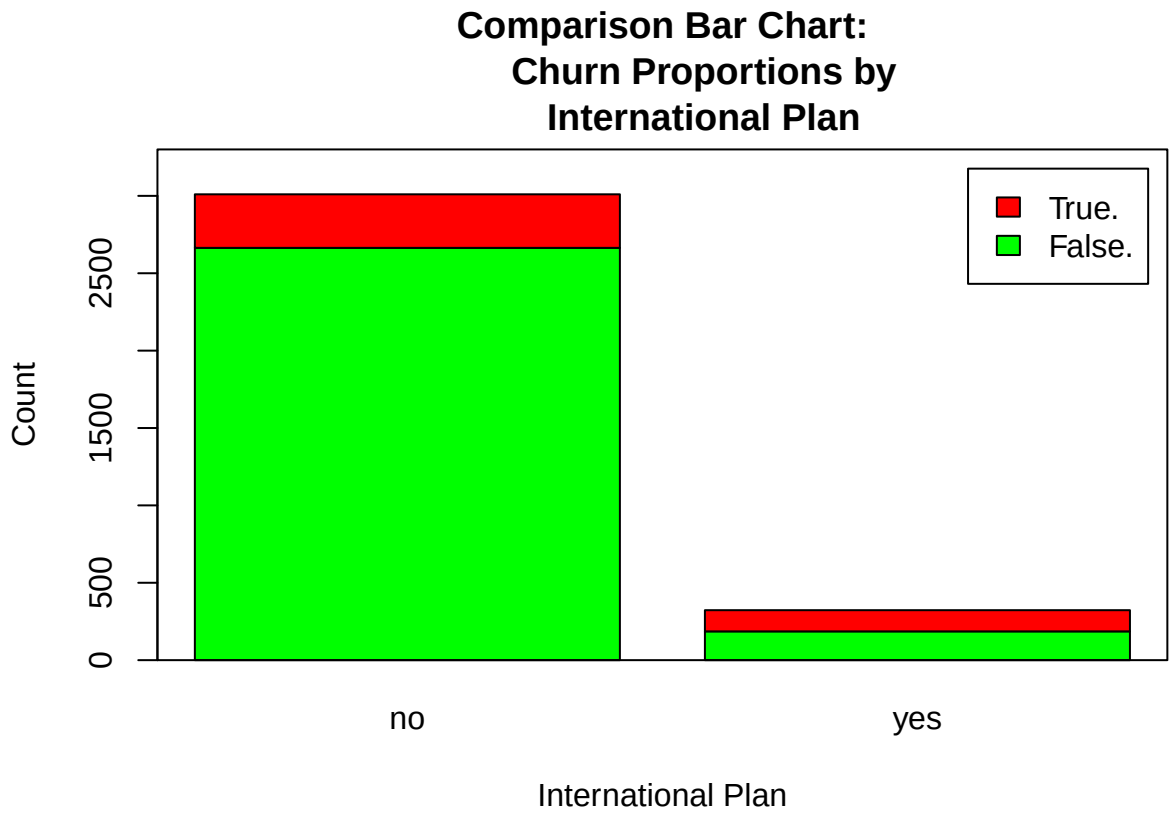
```r
gen_daymin <- sample(na.omit(churn.proc$Day.Mins), 1)
gen_churn <- sample(na.omit(churn.proc$Churn.), 1)
churn.proc[2,3] <- gen_daymin
churn.proc[2,5] <- gen_churn
head(churn.proc)
```

```
##   Account.Length    Phone Day.Mins Day.Calls Churn.
## 1            128 382-4657    265.1       110 False.
## 2            107 371-7191    239.3       123  True.
## 3            137 358-1921    243.4       114 False.
## 4             84 375-9999    299.4        71 False.
## 5             75 330-6626    166.7       113 False.
## 6            118 391-8027    223.4        98 False.
```

```r
# Make a table for counts of Churn and International Plan

counts <- table(churn$Churn,
                churn$Int.l.Plan,
                dnn=c("Churn", "International Plan"))
counts
```

```
##         International Plan
## Churn      no  yes
##   False. 2664  186
##   True.   346  137
```

```r
#Overlayed bar chart

barplot(counts,
        legend = rownames(counts),
        col = c("green", "red"),
        ylim = c(0, 3300),
        ylab = "Count",
        xlab = "International Plan",
        main = "Comparison Bar Chart:
        Churn Proportions by
        International Plan")
box(which = "plot",
    lty = "solid",
    col="black")
```




```
# Create a table with sums for both variables

sumtable <- addmargins(counts,
                       FUN = sum)
```

```
## Margins computed over dimensions
## in the following order:
## 1: Churn
## 2: International Plan
```

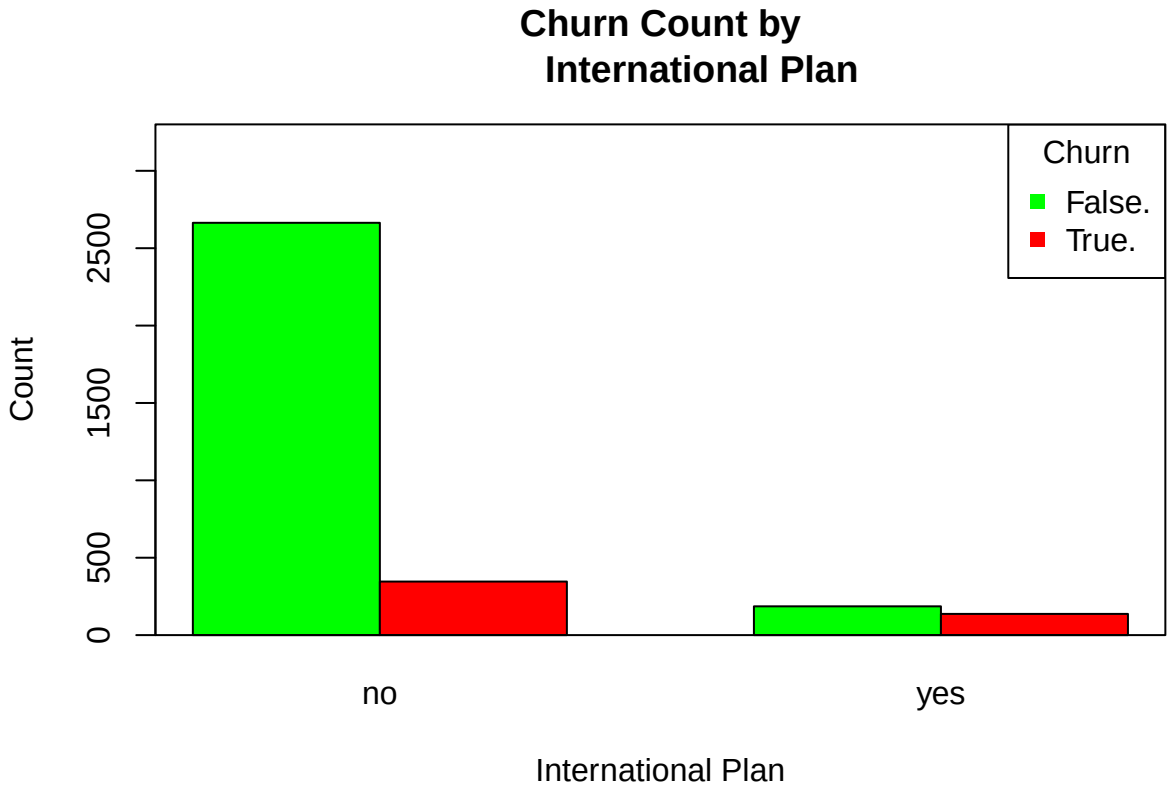
```
sumtable
```

```
##         International Plan
## Churn      no  yes  sum
##   False. 2664  186 2850
##   True.   346  137  483
##   sum    3010  323 3333
```

```
# Create a table of proportions over rows
row.margin <- round(prop.table(counts,
                               margin = 1),4)*100
row.margin
```

```
##         International Plan
## Churn       no   yes
##   False. 93.47  6.53
##   True.  71.64 28.36
```

```r
# Clustered Bar Chart, with legend

barplot(counts,
        col = c("green", "red"),
        ylim = c(0, 3300),
        ylab = "Count",
        xlab = "International Plan",
        main = "Churn Count by
        International Plan",
        beside = TRUE)
legend("topright",
       c(rownames(counts)),
       col = c("green", "red"),
       pch = 15,
       title = "Churn")
box(which = "plot",
    lty = "solid",
    col="black")# Create a table of proportions over columns
```

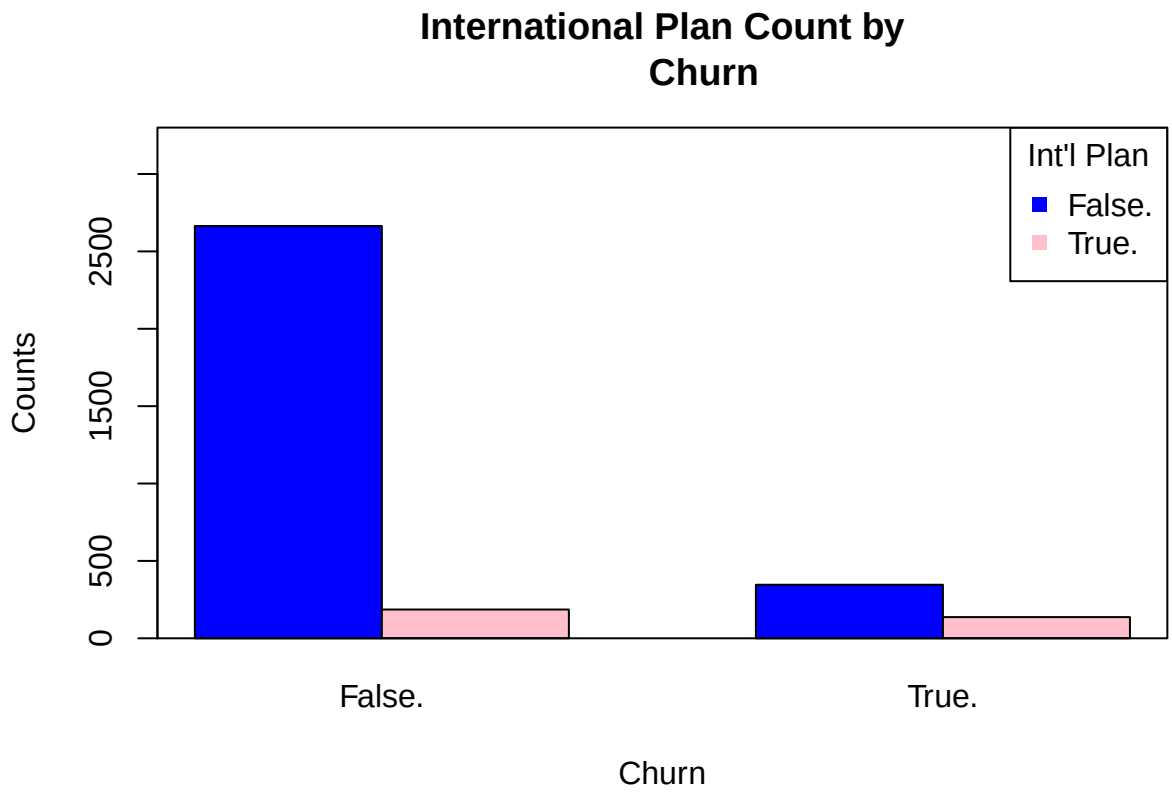



```r
col.margin <- round(prop.table(counts,
                               margin = 2), 4)*100
col.margin
```

```
##         International Plan
## Churn       no   yes
##   False. 88.50 57.59
```

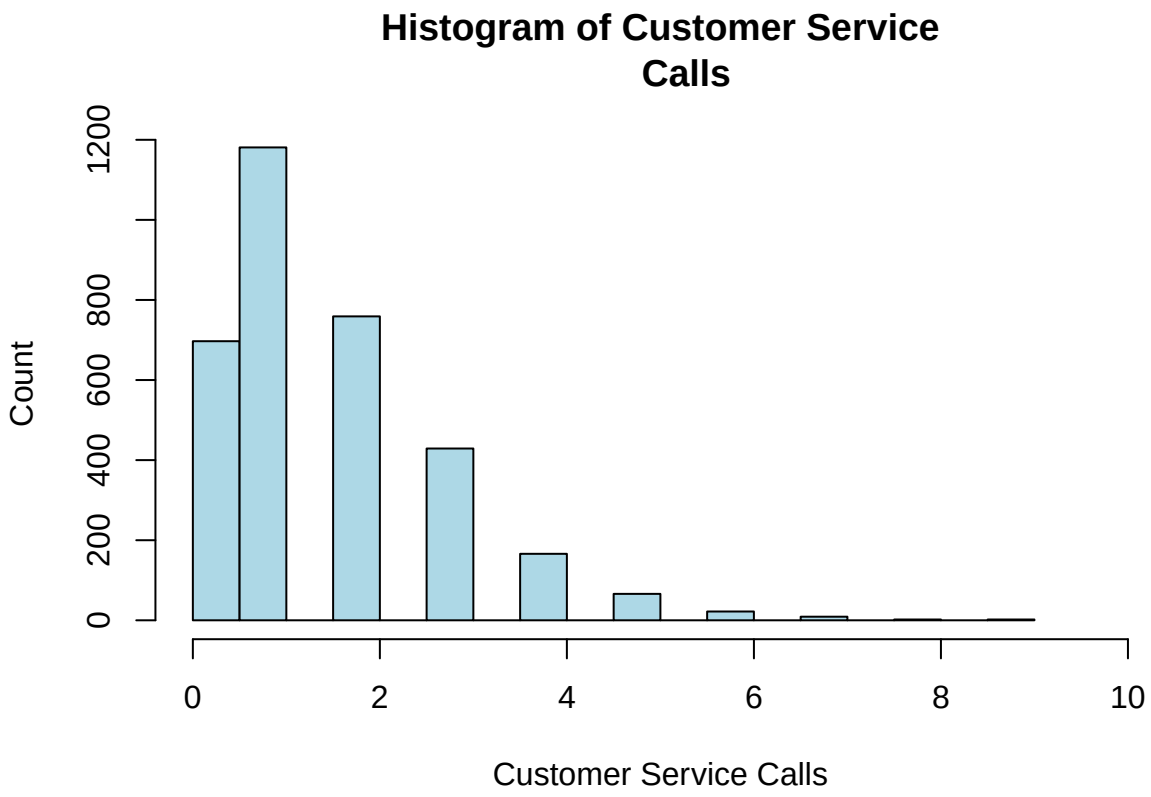
```
##   True.  11.50 42.41
```

```
# Clustered Bar Chart of Churn and International Plan with legend

barplot(t(counts),
        col = c("blue", "pink"),
        ylim = c(0, 3300),
        ylab = "Counts",
        xlab = "Churn",
        main = "International Plan Count by
        Churn",
        beside = TRUE)
legend("topright",
       c(rownames(counts)),
       col = c("blue", "pink"),
       pch = 15,
       title = "Int'l Plan")
box(which = "plot",
    lty = "solid",
    col="black")
```

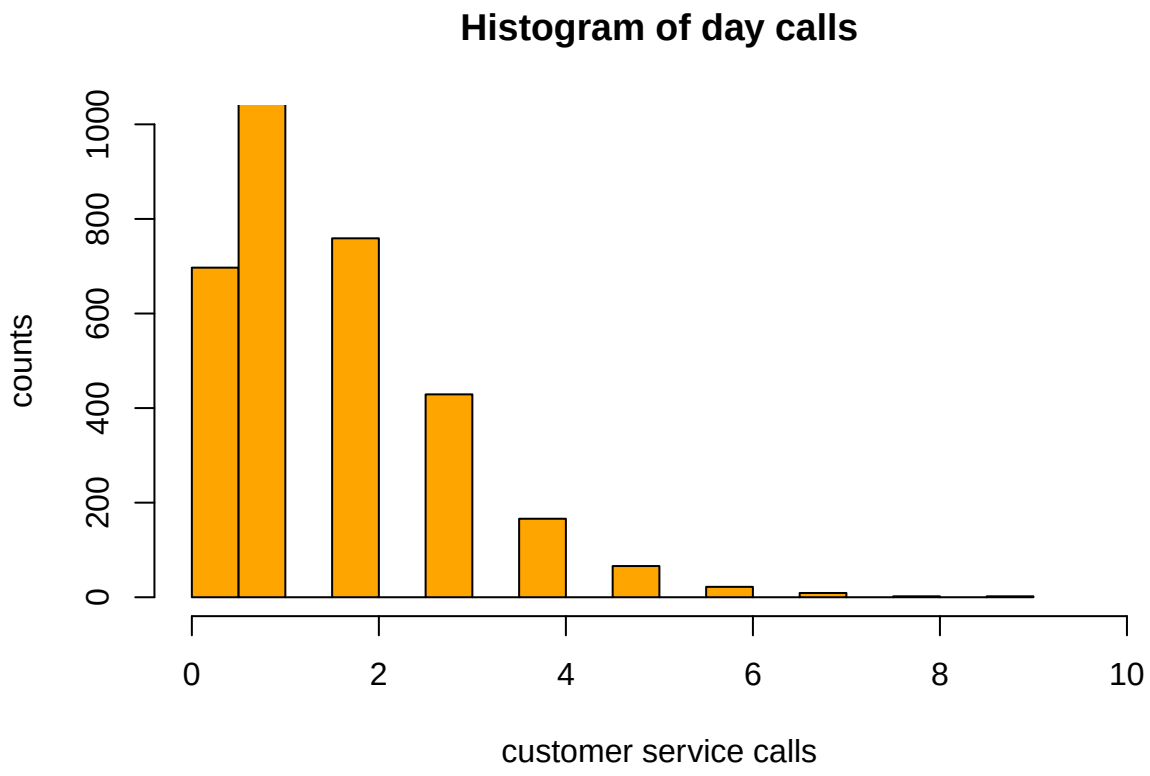



```
# Histogram of non-overlayed Customer Service Calls

hist(churn$CustServ.Calls,
     xlim = c(0,10),
     col = "lightblue",
     ylab = "Count",
```

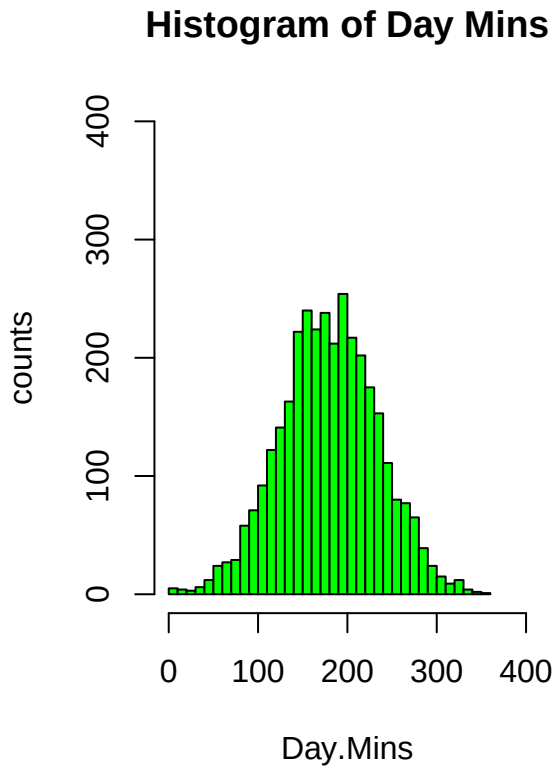
```
     xlab = "Customer Service Calls",
     main = "Histogram of Customer Service
     Calls")
```

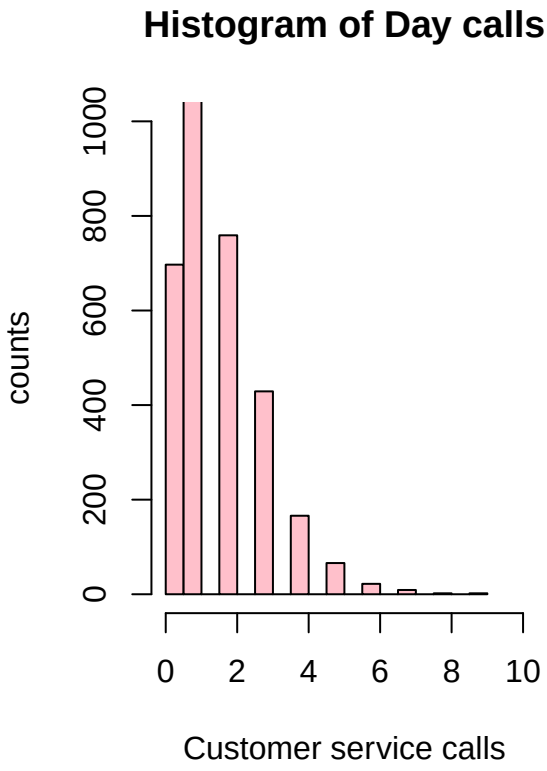



```
# Graphical Representation of Variables

#install.packages("ggplot2")
library(ggplot2)

# Create histogram of literacy
hist(churn$CustServ.Calls,
     breaks = 30,
     xlim = c(0,10),
     col = "orange",
     border = "black",
     ylim = c(0,1000),
     xlab = "customer service calls",
     ylab = "counts",
     main = "Histogram of day calls")
```

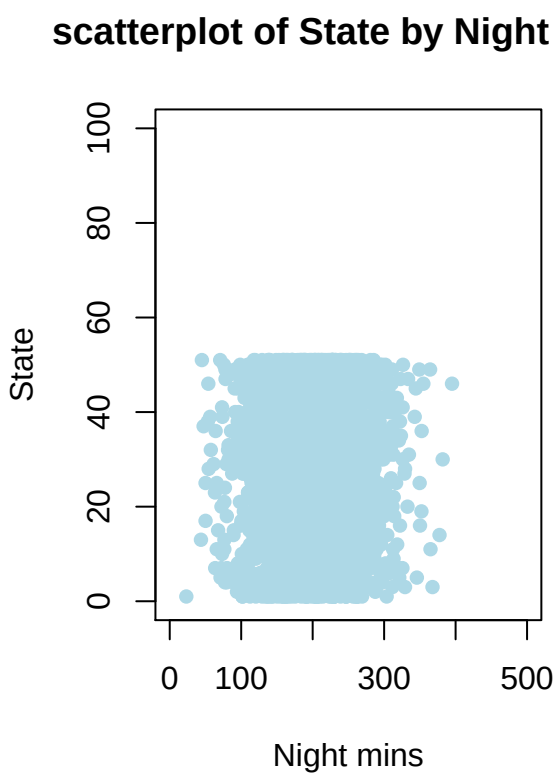


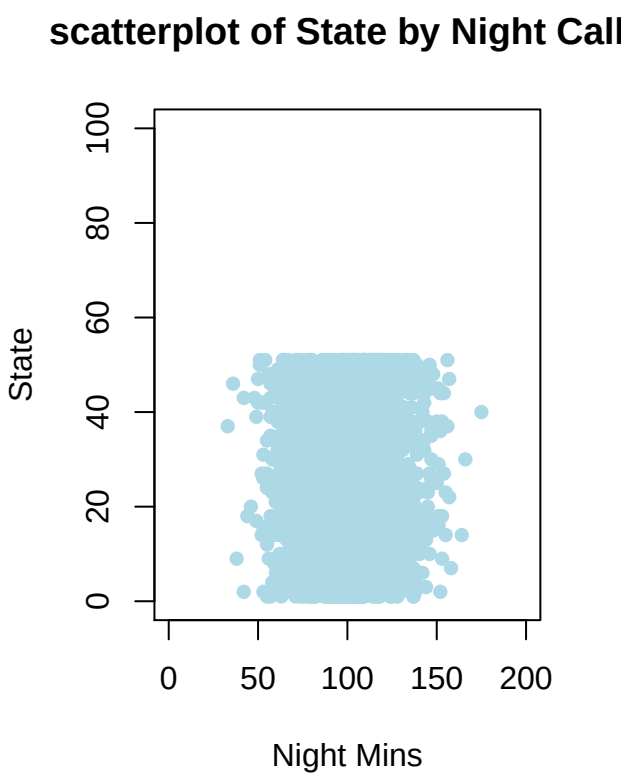

```
par(mfrow = c(1,2))
hist(churn$Day.Mins,
     breaks = 30,
     xlim = c(0, 400),
     col = "green",
     border = "black",
     ylim = c(0, 400),
     xlab = "Day.Mins",
     ylab = "counts",
     main = "Histogram of Day Mins")


hist(churn$CustServ.Calls,
     breaks = 30,
     xlim = c(0,10),
     col = "pink",
     border = "black",
     ylim = c(0,1000),
     xlab = "Customer service calls",
     ylab = "counts",
     main = "Histogram of Day calls")
```



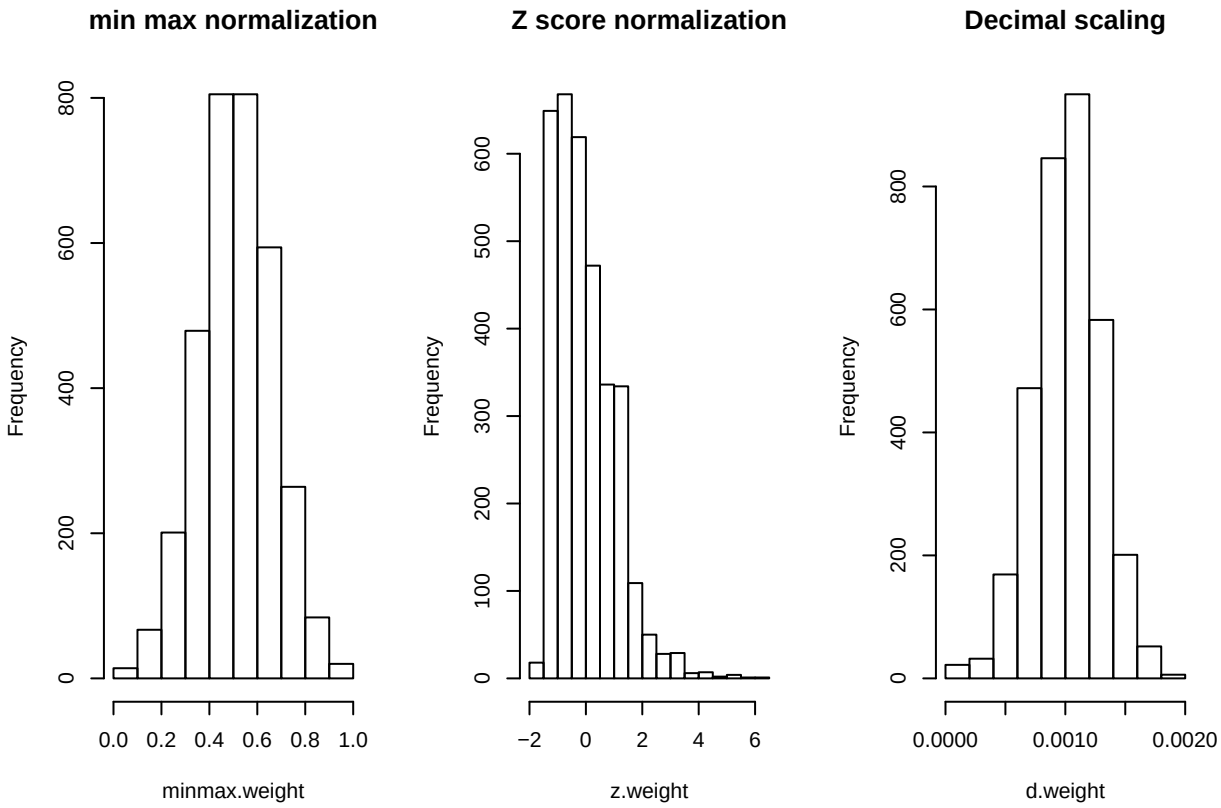



```
# creating a scatterplot

par(mfrow = c(1,2))
plot(churn$Night.Mins,
     churn$State,
     xlim = c(0,500),
     ylim = c(0,100),
     xlab = "Night mins",
     ylab = "State",
     main = "scatterplot of State by Night Mins",
     type = "p",
     pch = 16,
     col = "light bLUE")

plot(churn$Night.Calls,
     churn$State,
     xlim = c(0,200),
     ylim = c(0,100),
     xlab = "Night Mins",
     ylab = "State",
     main = "scatterplot of State by Night Calls",
     type = "p",
     pch = 16,
     col = "light blue")
```

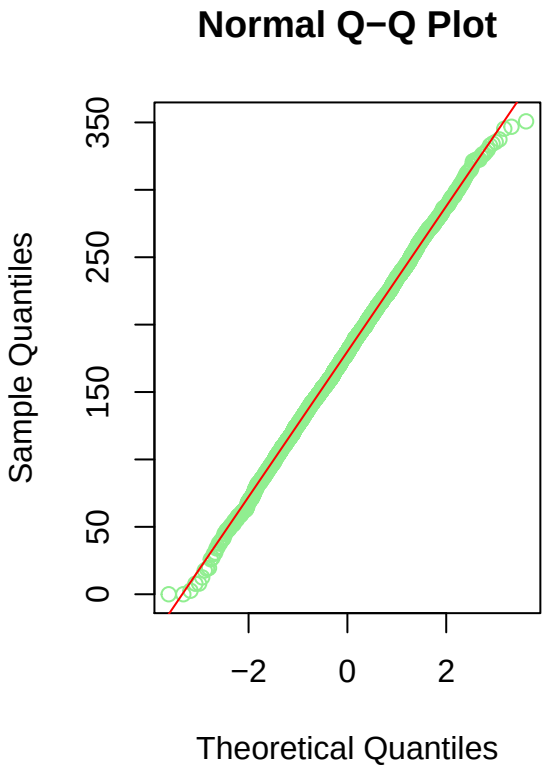


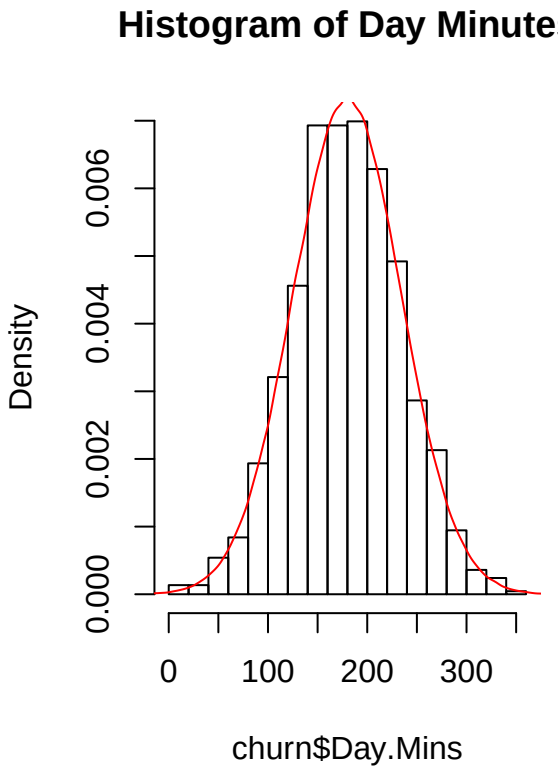



```r
# Transformation

#min-max normalisation
par(mfrow = c(1,3))
mi <- min(churn$Day.Mins)
ma <- max(churn$Day.Mins)
minmax.weight <- (churn$Day.Mins - mi)/(ma - mi)
hist(minmax.weight,
     main = "min max normalization")


# Z score normalization

m <- mean(churn$Intl.Calls)
s <- sd(churn$Intl.Calls)
z.weight <- (churn$Intl.Calls - m)/s
hist(z.weight,
     main = "Z score normalization")


# Decimal Scaling

d.weight <- churn$Intl.Mins / (10^4)
hist(d.weight,
     main = "Decimal scaling")
```

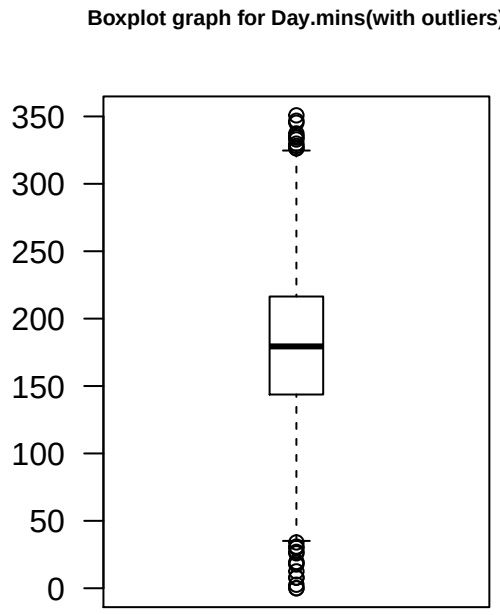


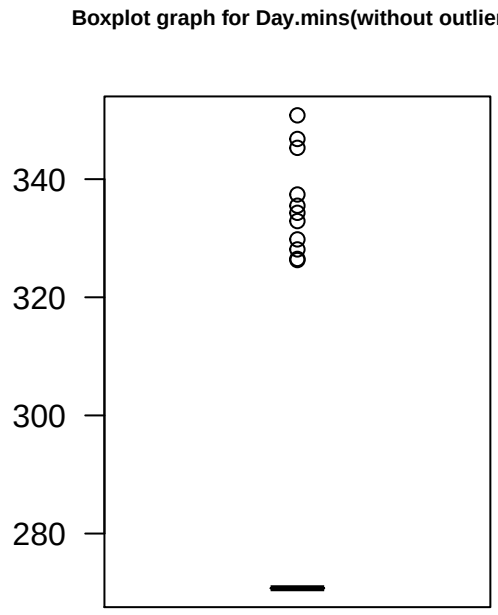

```
# Outliers:

par(mfrow = c(1,2))
# Detecting Outlier
qqnorm(churn$Day.Mins, col = "light green")
qqline(churn$Day.Mins, col = "red")
# Histogram
x <- rnorm(1000000, mean = mean(churn$Day.Mins), sd = sd(churn$Day.Mins))

hist(churn$Day.Mins, prob = T, main = "Histogram of Day Minutes")
lines(density(x), col = "red")
```






```r
# Eliminating outliers

x <- churn$Day.Mins
qnt <- quantile(x, probs=c(.25, .75))
caps <- quantile(x, probs=c(.05, .95))
h <- 1.5 * IQR (x)
x[x < (qnt[1] - h)] <- caps[1]
x[x < (qnt[2] + h)] <- caps[2]
churn.daymin.nooutlier <- x
par(mfrow = c(1,2))

# box plot

boxplot(churn$Day.Mins,
        las = 2,
        boxwex = 0.3,
        main = "Boxplot graph for Day.mins(with outliers)",
        cex.main = 0.7)

boxplot(churn.daymin.nooutlier,
        las = 2,
        boxwex = 0.3,
        main = "Boxplot graph for Day.mins(without outliers)",
        cex.main = 0.7)
```



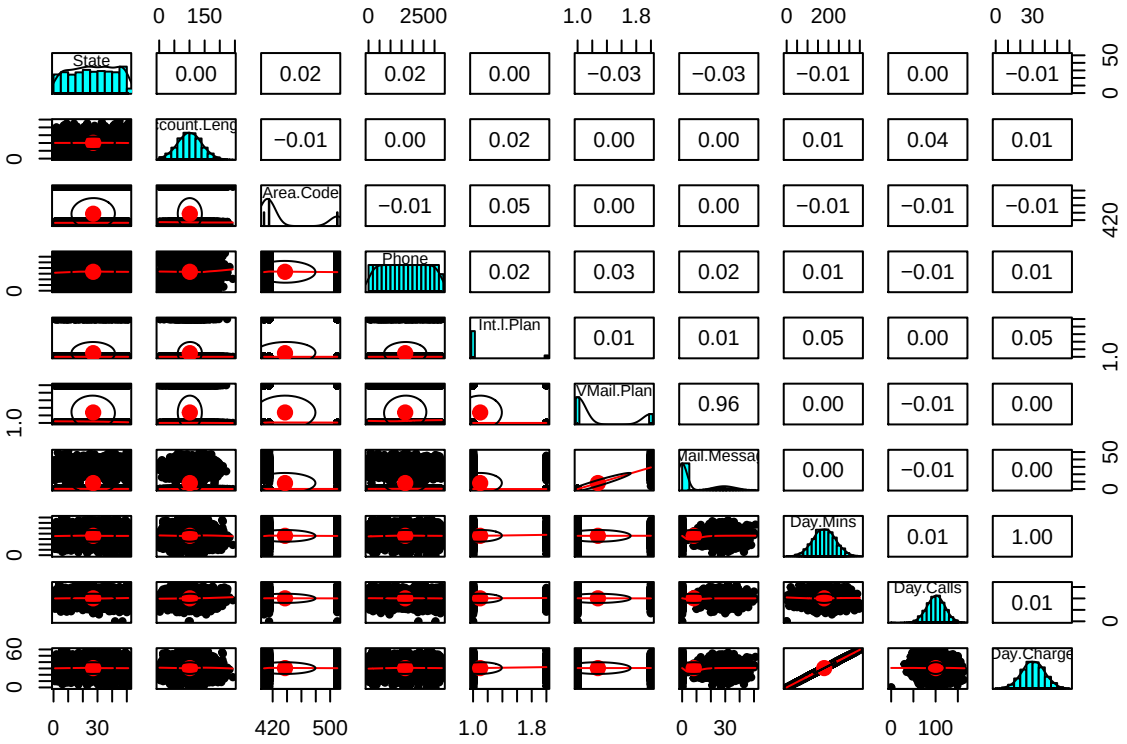



```
# Distribution of Variables

# Creating Pairs
#install.packages("lattice")
require(lattice)

#install.packages("psych")
library("psych")
```

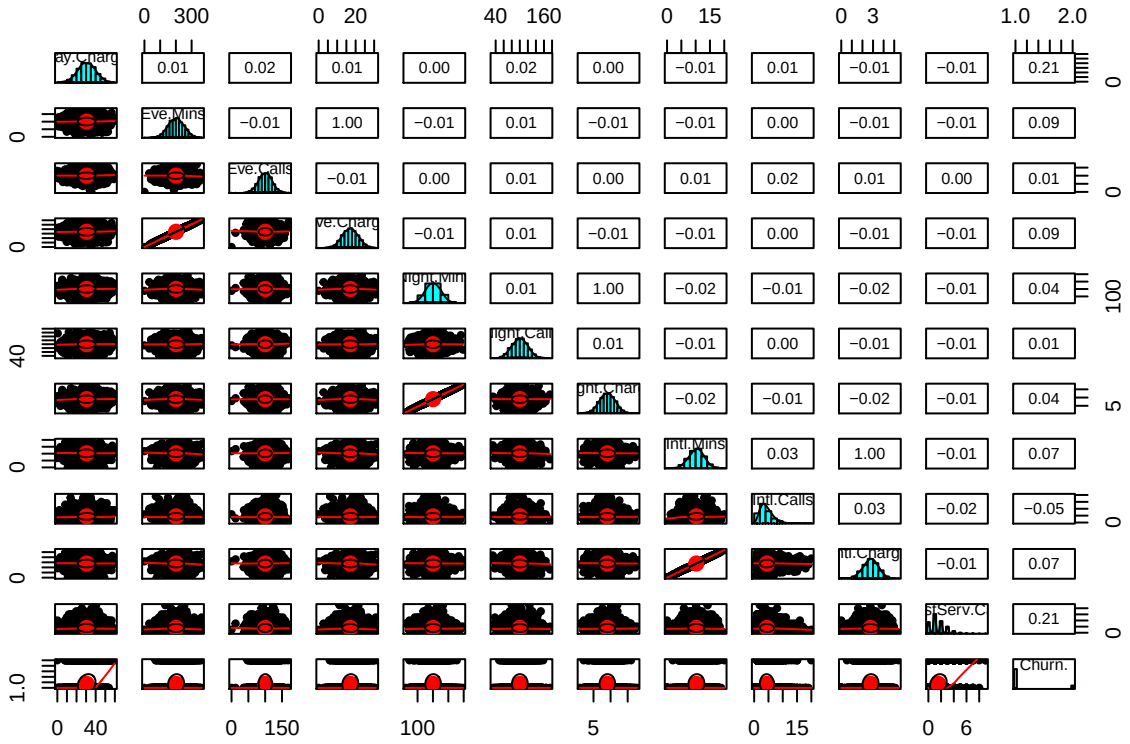
```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

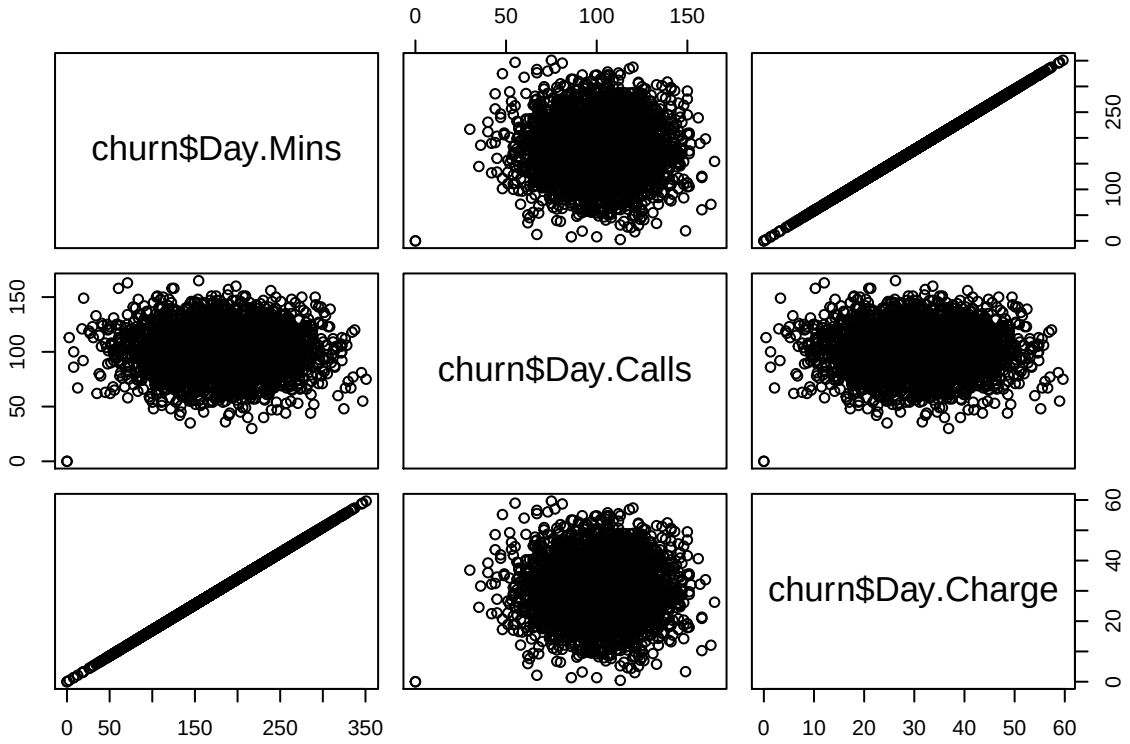
```
pairs.panels(churn[1:10])
```



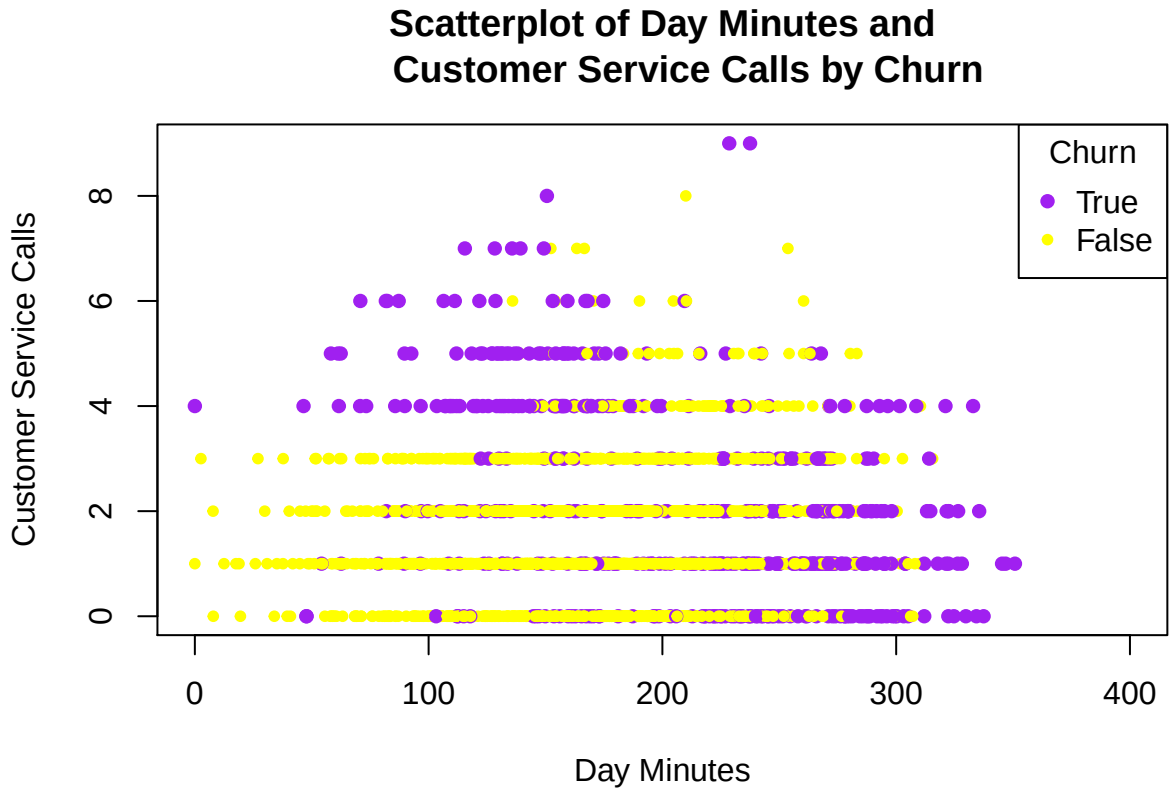

```
pairs.panels(churn[10:21])
```




```
# Scatterplot matrix

pairs(~churn$Day.Mins+
        churn$Day.Calls+
        churn$Day.Charge)
```




```
# Scatter plot of Day Minutes and Customer Service Calls, colored by Churn

plot(churn$Day.Mins,
     churn$CustServ.Calls,
     xlim = c(0, 400),
     xlab = "Day Minutes",
     ylab = "Customer Service Calls",
     main = "Scatterplot of Day Minutes and
     Customer Service Calls by Churn",
     col = ifelse(churn$Churn.=="True.",
                  "purple",
                  "yellow"),
     pch = ifelse(churn$Churn.=="True.",
                  16, 20))
legend("topright",
       c("True",
         "False"),
       col = c("purple",
               "yellow"),
       pch = c(16, 20),
       title = "Churn")
```




```
# Tabloical Representation

table.international <- table(churn$Int.l.Plan, churn$State)
table.international
```

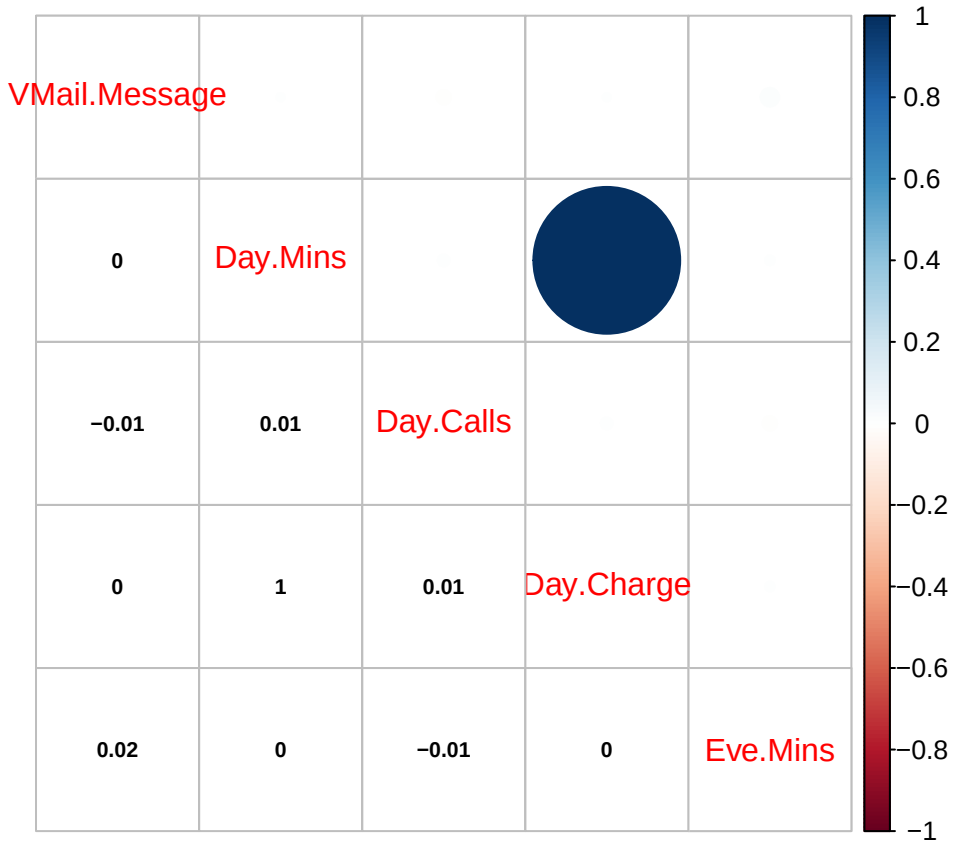
```
##
##       AK AL AR AZ CA CO CT DC DE FL GA HI IA ID IL IN KS KY LA MA MD ME MI
##   no  48 72 47 61 30 62 66 49 51 55 50 50 44 67 43 68 62 55 50 57 60 56 64
##   yes  4  8  8  3  4  4  8  5 10  8  4  3  0  6 15  3  8  4  1  8 10  6  9
##
##       MN MO MS MT NC ND NE NH NJ NM NV NY OH OK OR PA RI SC SD TN TX UT VA
##   no  76 57 57 66 57 54 58 52 62 57 57 75 69 56 71 39 55 58 55 47 66 66 68
##   yes  8  6  8  2 11  8  3  4  6  5  9  8  9  5  7  6 10  2  5  6  6  6  9
##
##       VT WA WI WV WY
##   no  67 62 70 99 67
##   yes  6  4  8  7 10
```

```
# Corelated Variables

churn.corelation <- cor(churn[,c(7:11)], use="complete.obs", method="kendall")
churn.corelation
```

```
##                VMail.Message    Day.Mins    Day.Calls  Day.Charge
## VMail.Message   1.000000000 0.003087796 -0.009573189 0.003087796
## Day.Mins        0.003087796 1.000000000  0.006333184 1.000000000
## Day.Calls      -0.009573189 0.006333184  1.000000000 0.006333184
## Day.Charge      0.003087796 1.000000000  0.006333184 1.000000000
```

```
## Eve.Mins         0.016106727 0.004202887 -0.009551729 0.004202887
##                    Eve.Mins
## VMail.Message   0.016106727
## Day.Mins        0.004202887
## Day.Calls      -0.009551729
## Day.Charge      0.004202887
## Eve.Mins        1.000000000
```

```
# Corplot

library(corrplot)
corrplot.mixed(churn.corelation, lower.col = "black", number.cex = .7)
```




```
# Regression of Day Charge vs Day Minutes
names(churn)
```

```
##  [1] "State"          "Account.Length" "Area.Code"      "Phone"
##  [5] "Int.l.Plan"     "VMail.Plan"     "VMail.Message"  "Day.Mins"
##  [9] "Day.Calls"      "Day.Charge"     "Eve.Mins"       "Eve.Calls"
## [13] "Eve.Charge"     "Night.Mins"     "Night.Calls"    "Night.Charge"
## [17] "Intl.Mins"      "Intl.Calls"     "Intl.Charge"    "CustServ.Calls"
## [21] "Churn."
```

```
fit <- lm(churn$Day.Charge ~ churn$Day.Mins)
summary(fit)
```

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

```
## lm(formula = churn$Day.Charge ~ churn$Day.Mins)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0045935 -0.0025391  0.0004326  0.0024587  0.0045224
##
## Coefficients:
##                  Estimate Std. Error   t value Pr(>|t|)
## (Intercept)    6.134e-04  1.711e-04 3.585e+00 0.000341 ***
## churn$Day.Mins 1.700e-01  9.108e-07 1.866e+05  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.002864 on 3331 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 3.484e+10 on 1 and 3331 DF,  p-value: < 2.2e-16
```

```
# Correlation values, with p-values

days <- cbind(churn$Day.Mins,
              churn$Day.Calls,
              churn$Day.Charge)
MinsCallsTest <- cor.test(churn$Day.Mins,
                          churn$Day.Calls)
MinsChargeTest <- cor.test(churn$Day.Mins,
                           churn$Day.Charge)
CallsChargeTest <- cor.test(churn$Day.Calls,
                            churn$Day.Charge)
round(cor(days),4)
```

```
##        [,1]   [,2]   [,3]
## [1,] 1.0000 0.0068 1.0000
## [2,] 0.0068 1.0000 0.0068
## [3,] 1.0000 0.0068 1.0000
```

```
MinsCallsTest$p.value
```

```
## [1] 0.6968515
```

```
MinsChargeTest$p.value
```

```
## [1] 0
```

```
CallsChargeTest$p.value
```

```
## [1] 0.6967428
```

```
# Correlation values and p-values in matrix form
names(churn)
```

```
##  [1] "State"          "Account.Length" "Area.Code"      "Phone"
##  [5] "Int.l.Plan"     "VMail.Plan"     "VMail.Message"  "Day.Mins"
##  [9] "Day.Calls"      "Day.Charge"     "Eve.Mins"       "Eve.Calls"
## [13] "Eve.Charge"     "Night.Mins"     "Night.Calls"    "Night.Charge"
## [17] "Intl.Mins"      "Intl.Calls"     "Intl.Charge"    "CustServ.Calls"
## [21] "Churn."
```

```r
# Collect variables of interest
corrdata <-
  cbind(churn$Account.Length,
        churn$VMail.Message,
        churn$Day.Mins,
        churn$Day.Calls,
        churn$CustServ.Calls)

# Declare the matrix
corrpvalues <- matrix(rep(0, 25),
                      ncol = 5)
```

```r
# Fill the matrix with correlations
for (i in 1:4) {
  for (j in (i+1):5) {
    corrpvalues[i,j] <-
      corrpvalues[j,i] <-
      round(cor.test(corrdata[,i],
                     corrdata[,j])$p.value,4)
  }}

round(cor(corrdata), 4)
```

```
##         [,1]    [,2]    [,3]    [,4]    [,5]
## [1,]  1.0000 -0.0046  0.0062  0.0385 -0.0038
## [2,] -0.0046  1.0000  0.0008 -0.0095 -0.0133
## [3,]  0.0062  0.0008  1.0000  0.0068 -0.0134
## [4,]  0.0385 -0.0095  0.0068  1.0000 -0.0189
## [5,] -0.0038 -0.0133 -0.0134 -0.0189  1.0000
```

```r
corrpvalues
```

```
##        [,1]   [,2]   [,3]   [,4]   [,5]
## [1,] 0.0000 0.7894 0.7198 0.0264 0.8266
## [2,] 0.7894 0.0000 0.9642 0.5816 0.4440
## [3,] 0.7198 0.9642 0.0000 0.6969 0.4385
## [4,] 0.0264 0.5816 0.6969 0.0000 0.2743
## [5,] 0.8266 0.4440 0.4385 0.2743 0.0000
```

```r
#From the analysis we observed that the features of Day, Evening, night and
#international calls data on the effect of churn rate in the analysis and also
#that most of the customers are not churned, given the observed states on
#their expenses.
# As we got Rsquared value 1 and the p-value: < 2.2e-16 which are great for our model.
#The account length has a positive effect on the churn rate,
#as well as customer service calls. These things we observed from the
#churn data set using histograms corplot and many more.
```