Assignement 1

Programmer : Edouard Dewaerheijd, Presenter : Vienne Jetten

2023-11-07

R Markdown

question 33

```
missing_values <- sapply(churn, function(x) sum(is.na(x)))
missing_values #there is no missing data after runing this!
```

```
##          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 there is no missing data.

question 34

```
area.code_frequency <- table(churn$Area.Code)
area.code_frequency
```

```
## 
##  408  415  510 
##  838 1655  840
```

```
state_frequency <- table(churn$State)
state_frequency
```

```
## 
##  AK  AL  AR  AZ  CA  CO  CT  DC  DE  FL  GA  HI  IA  ID  IL  IN  KS  KY  LA  MA 
##  52  80  55  64  34  66  74  54  61  63  54  53  44  73  58  71  70  59  51  65
## MD  ME  MI  MN  MO  MS  MT  NC  ND  NE  NH  NJ  NM  NV  NY  OH  OK  OR  PA  RI
## 70  62  73  84  63  65  68  68  62  61  56  68  62  66  83  78  61  78  45  65
## SC  SD  TN  TX  UT  VA  VT  WA  WI  WV  WY
## 60  60  53  72  72  77  73  66  78 106  77
```

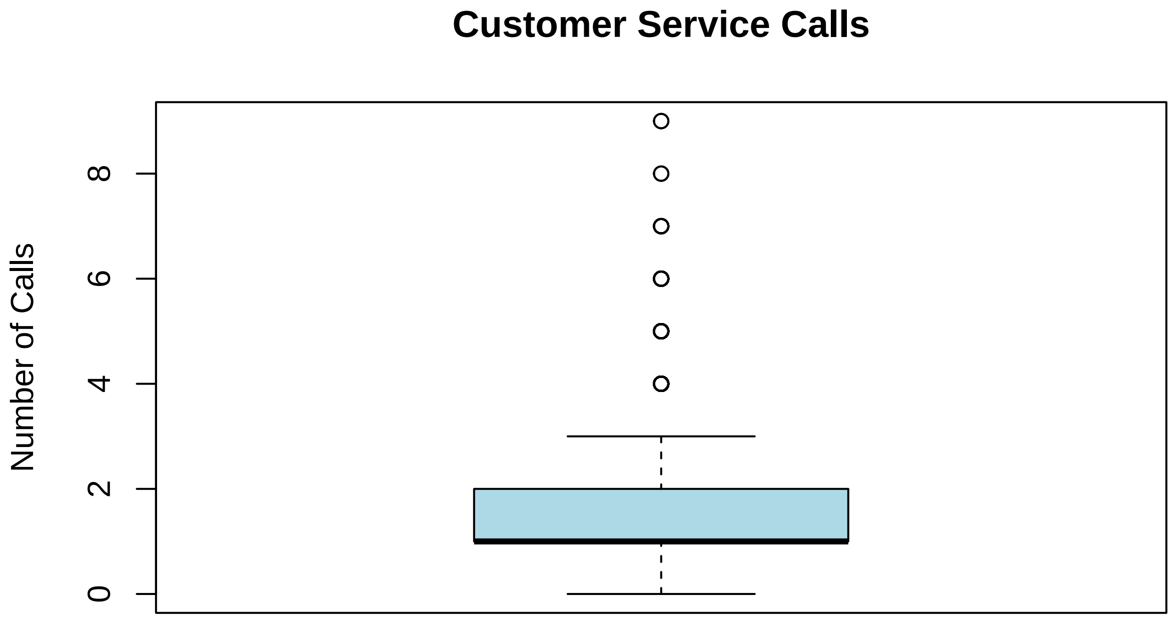
```
length(unique(churn$State))
```

```
## [1] 51
```

we can see there is only 3 different area code and 51 different states, this seems abnormal. we'd expect at least more than one code per state.

question 35

```
boxplot(churn$CustServ.Calls,
        main="Customer Service Calls",
        ylab="Number of Calls",
        col = "lightblue",
        border = "black")
```



there are quite a few outliers.

question 36

a

```
custServCalls <- churn$CustServ.Calls
z_scores <- (custServCalls - mean(custServCalls)) / sd(custServCalls)

# Identify outliers
outliers_z <- which(abs(z_scores) > 3)
outliers_z_values <- custServCalls[outliers_z]
outliers_z
```

```
##  [1]  333  523  543  695  722  779  903  909  975 1143 1274 1326 1408 1503 1639
## [16] 1695 1832 1866 1913 1920 2224 2328 2381 2388 2429 2554 2787 2954 2959 2962
## [31] 2980 3027 3082 3113 3191
```

```
range(outliers_z_values)
```

```
## [1] 6 9
```

the range of outliers is between 6 and 9.

b

```
# Calculate IQR
IQR_value <- IQR(custServCalls)
Q1 <- quantile(custServCalls, 0.25)
Q3 <- quantile(custServCalls, 0.75)

# Calculate the bounds for outliers
lower_bound <- Q1 - 1.5 * IQR_value
upper_bound <- Q3 + 1.5 * IQR_value

# Identify outliers
outliers_iqr <- which(custServCalls < lower_bound | custServCalls > upper_bound)
outliers_iqr_values <- custServCalls[outliers_iqr]
range(outliers_iqr_values)
```

```
## [1] 4 9
```

the range of outliers using the IQR methode is bewteen 4 and 9.

question 37

```
# Calculate the mean and standard deviation of the 'Day Mins' column
day_mins_mean <- mean(churn$Day.Mins)
day_mins_sd <- sd(churn$Day.Mins)

# Perform Z-score standardization
churn$Day.Mins.Z <- (churn$Day.Mins - day_mins_mean) / day_mins_sd
```

the rerun is 3 pages long of all the just all the diffrent z scores.

question 38

function to calculate the skewness

```
# Function to calculate skewness
calculate_skewness <- function(x) {
  n <- length(x)
  mean_x <- mean(x)
  sd_x <- sd(x)
  skewness <- (n / ((n - 1) * (n - 2))) * sum(((x - mean_x) / sd_x) ^ 3)
  return(skewness)
}
```

a

```
Skewness_Day.Mins <- calculate_skewness(churn$Day.Mins)
Skewness_Day.Mins
```

```
## [1] -0.02907707
```

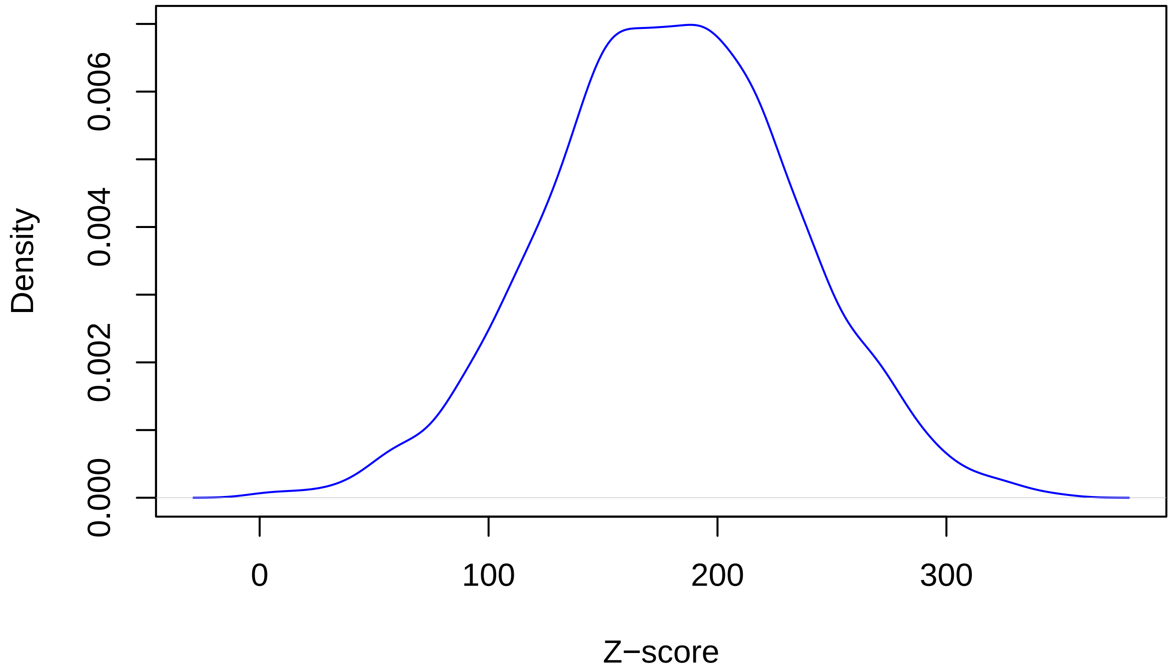
b

```
Skewness_Day.Mins._z <- calculate_skewness(churn$Day.Mins.Z)
Skewness_Day.Mins._z
```

```
## [1] -0.02907707
```

the skewness is -0.029 for both the ‘day mins’ and it is the dame for the z-score standardized version because the z-score changes the scale but not the skewnes # c

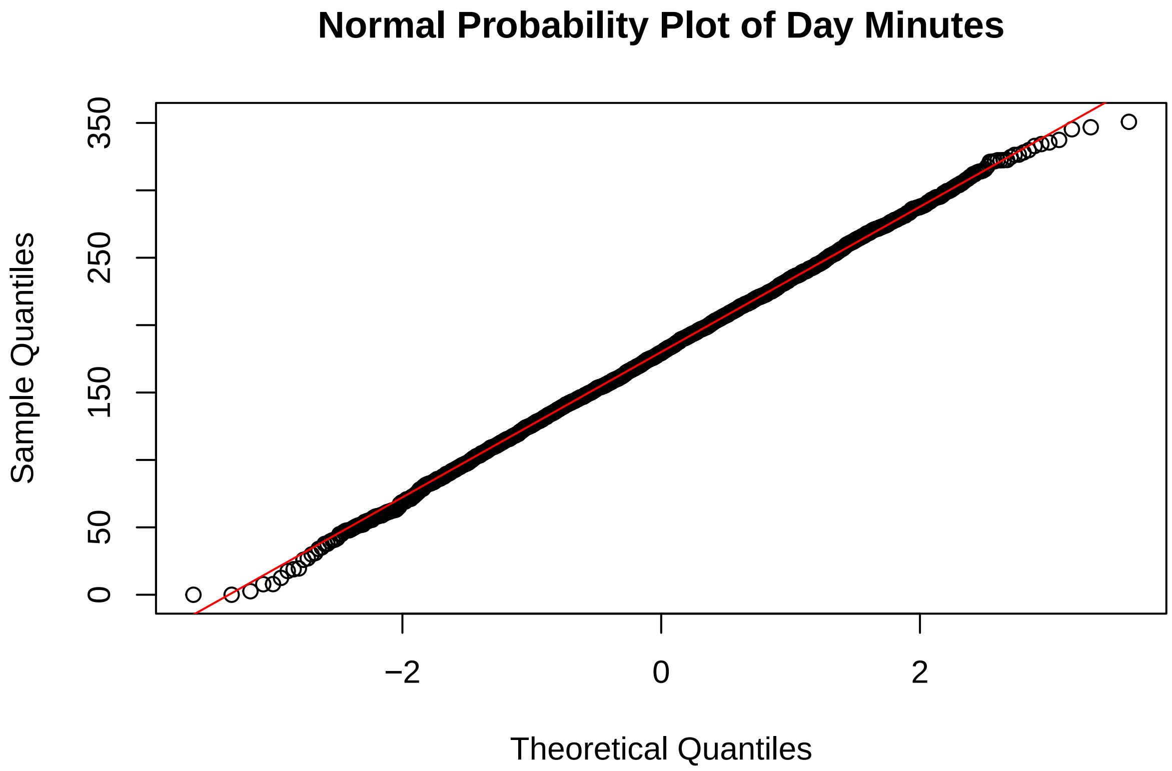
```
plot(density(churn$Day.Mins), main = "",
     xlab = "Z-score", ylab = "Density", col = "blue")
```



to conclude, -0.029 indicates that the distribution is very close to being symmetric. we also graphed the function to make sure of are answers

question 39

```
qqnorm(churn$Day.Mins, main = "Normal Probability Plot of Day Minutes")
qqline(churn$Day.Mins, col = "red")
```

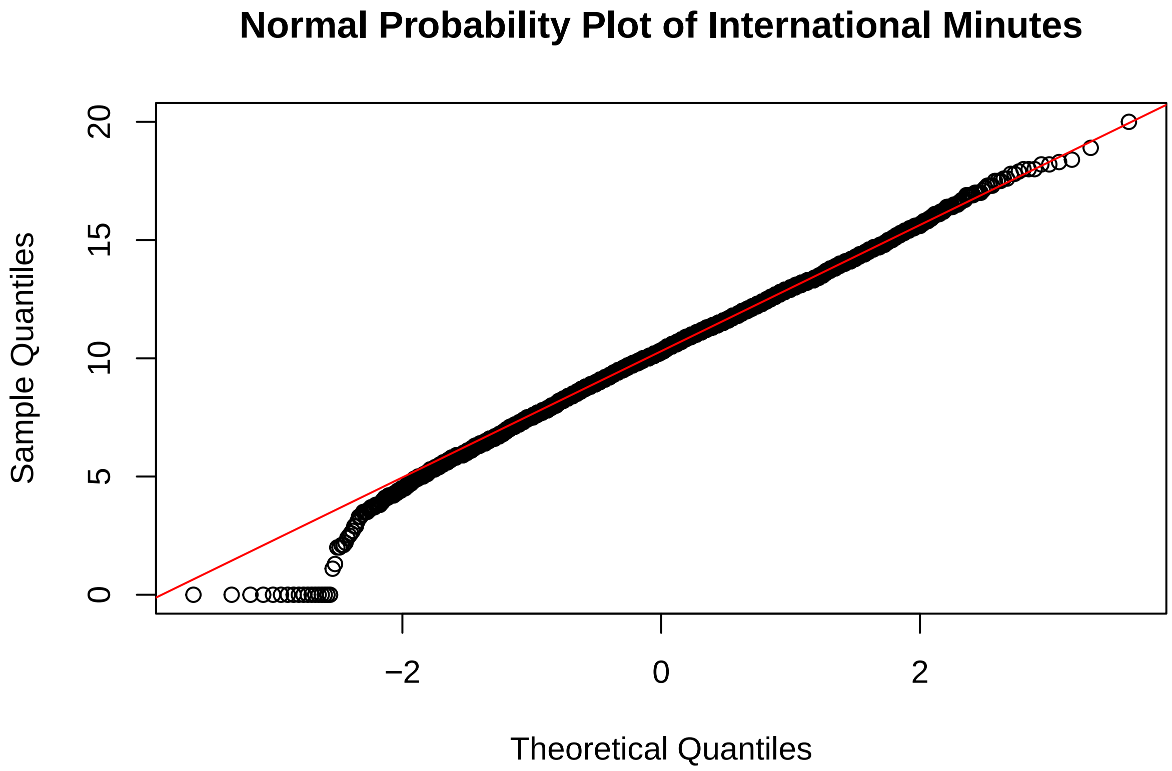


Day minutes seems normaly distributed.

question 40

a

```
qqnorm(churn$Intl.Mins, main = "Normal Probability Plot of International Minutes")
qqline(churn$Intl.Mins, col = "red")
```



b

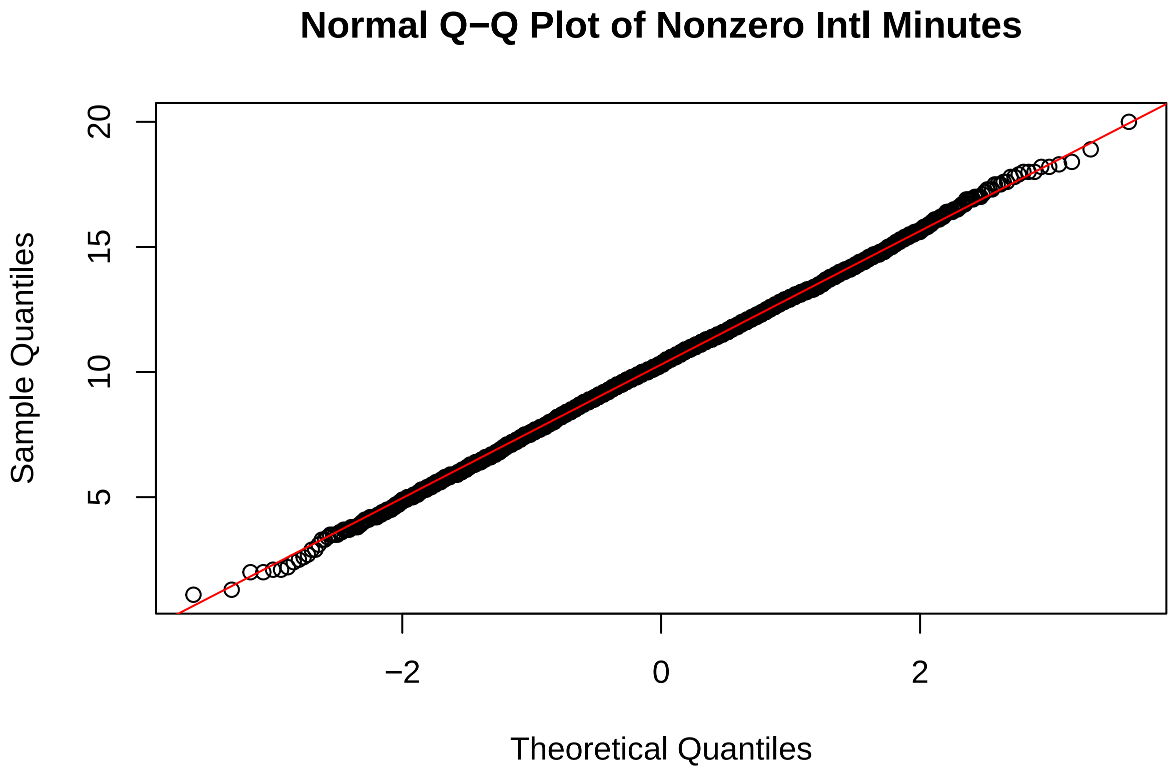
all the 0 in international minutes is causing it not to be normally distributed. # c

```
# Create a flag variable for whether there are international minutes
churn$Int.Min.Flag <- ifelse(churn$Intl.Mins > 0, 1, 0)

# Subset the data to include only rows with nonzero international minutes
nonzero_intl_mins <- subset(churn$Intl.Mins, churn$Intl.Mins > 0)
```

d

```
qqnorm(nonzero_intl_mins, main = "Normal Q-Q Plot of Nonzero Intl Minutes")
qqline(nonzero_intl_mins, col = "red")
```

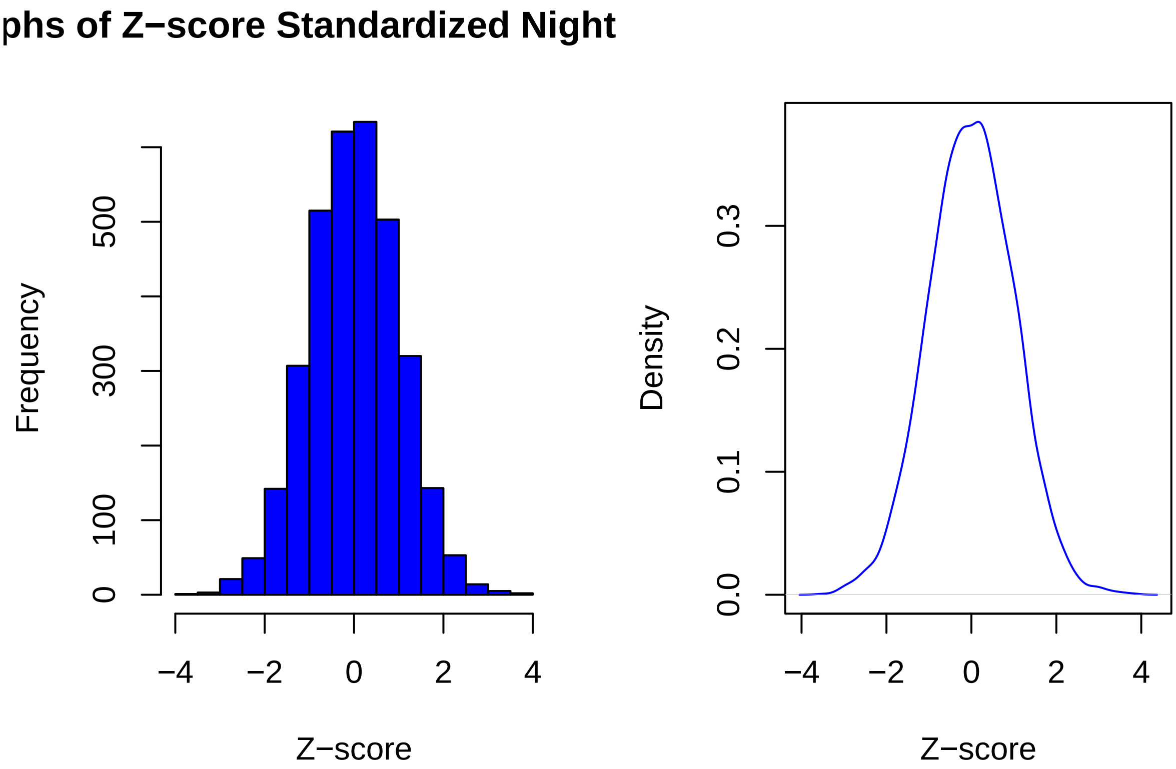


without the 0s it seems almost normally distributed.

question 41

```r
night_mins_mean <- mean(churn$Night.Mins)
night_mins_sd <- sd(churn$Night.Mins)
# Perform Z-score standardization
churn$Night.Mins.Z <- (churn$Night.Mins - night_mins_mean) / night_mins_sd
par(mfrow = c(1,2))
hist(churn$Night.Mins.Z, main = "graphs of Z-score Standardized Night Minutes",
     xlab = "Z-score", ylab = "Frequency", col = "blue")

plot(density(churn$Night.Mins.Z), main = "",
     xlab = "Z-score", ylab = "Density", col = "blue")
```


phs of Z−score Standardized Night
Frequency
0
100
300
500
−4
−2
0
2
4
Z−score
Density
0.0
0.1
0.2
0.3
−4
−2
0
2
4
Z−score