Data Exploration 3G

10/22/2020

```
options(digits = 3, warn = -1)

# Load libraries
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------------------------
------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.0     v purrr   0.3.3
## v tibble  3.0.1     v dplyr   0.8.5
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------------------------
- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(e1071)
library(ggplot2)
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
require(gridExtra)
```

```
## Loading required package: gridExtra
```

```
##
## Attaching package: 'gridExtra'
```

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

This sample contains **30 observations** of several navigation performance features, collected over **3G connection**.

Factors (Perfume.js metrics): 1. **headerSizeAvg** - Average size of the header 2. **timeToFirstByteAvg** (ms) - Time to First Byte (TTFB) - Average amount of time ittakes for the server to send the first payload to the client 3. **downloadTimeAvg** - Download Time (ms) - Average response time only (download) 4. **totalTimeAvg** - Total time (ms) - Average request plus response time (network only) 1. **headerSizeNormalized** - Normalized size of the header 2. **timeToFirstByteNormalized** (ms) - Time to First Byte (TTFB) - Normalized amount of time it takes for the server to send the first payload to the client 3. **downloadTimeNormalized** - Download Time (ms) - Normalized response time only (download) 4. **totalTimeNormalized** - Total time (ms) - Normalized request plus response time (network only) 5. **navigationMean** - Performance score from the perspective of navigation timing 6. **performanceLvl** - Ordinal performance score

Numerical outcome (dependent variable): **batteryStatsAvg** - Mean energy efficiency measured with battery stats (Joules)

All averages represent the mean of the values measured for 11 runs, for each subject.

The **navigationMean** (the performance score from the perspective of navigation timing) is an average of the normalized navigation timing metrics (i.e., headerSize, timeToFirstByte, downloadTime, totalTime).

```
threeG_nav_df =  read_csv("./NavigationTiming_threeG.csv", col_names = TRUE)
```

```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   subject = col_character(),
##   navigationMean = col_double(),
##   headerSizeAvg = col_double(),
##   timeToFirstByteAvg = col_double(),
##   fetchTimeAvg = col_double(),
##   downloadTimeAvg = col_double(),
##   totalTimeAvg = col_double(),
##   navigationMean.1 = col_double(),
##   headerSizeNormalized = col_double(),
##   timeToFirstByteNormalized = col_double(),
##   fetchTimeNormalized = col_double(),
##   downloadTimeNormalized = col_double(),
##   totalTimeNormalized = col_double(),
##   batteryStatsAvg = col_double(),
##   performanceLvl = col_character(),
##   website = col_character()
## )
```

```
threeG_nav_df = select(threeG_nav_df, -c(X1))

View(threeG_nav_df)
```

Summary of energy efficiency:

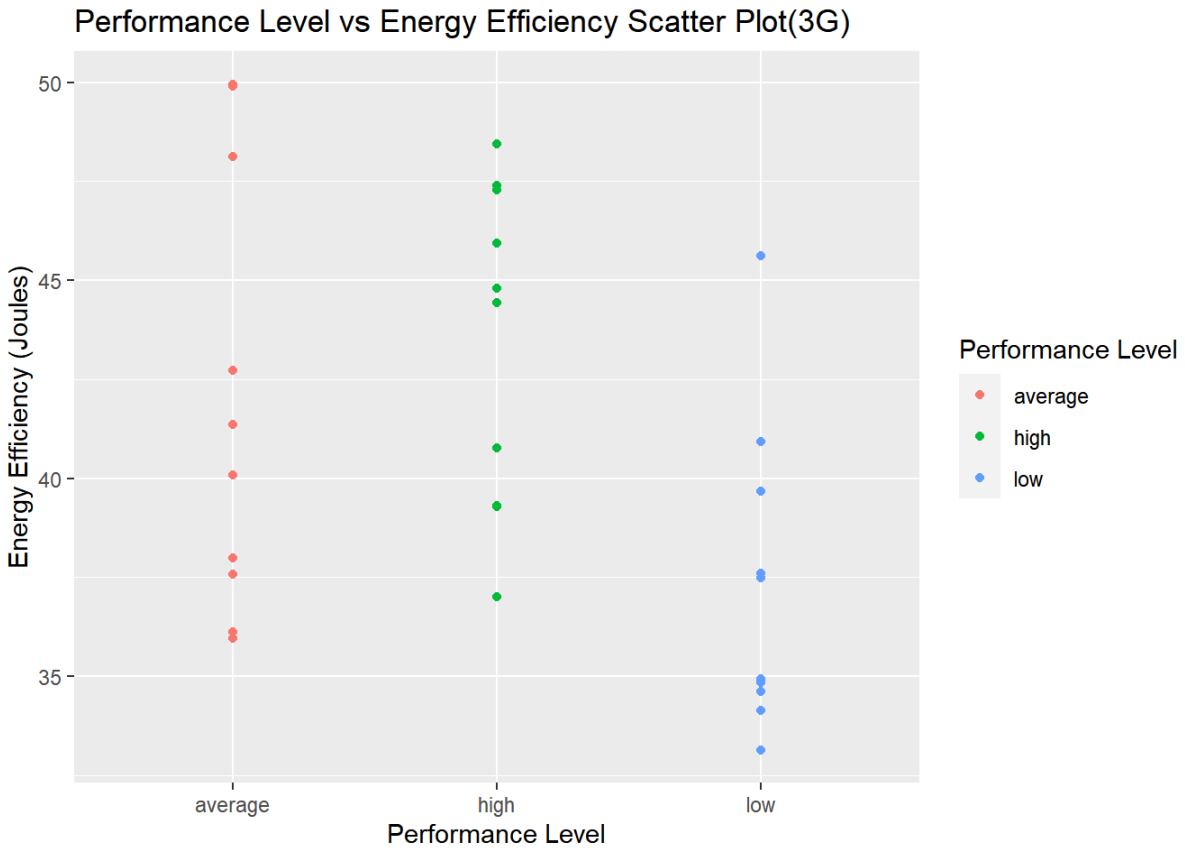
```
summary(threeG_nav_df$batteryStatsAvg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    33.1    37.1    39.9    40.9    45.4    50.0
```

Exploring the data

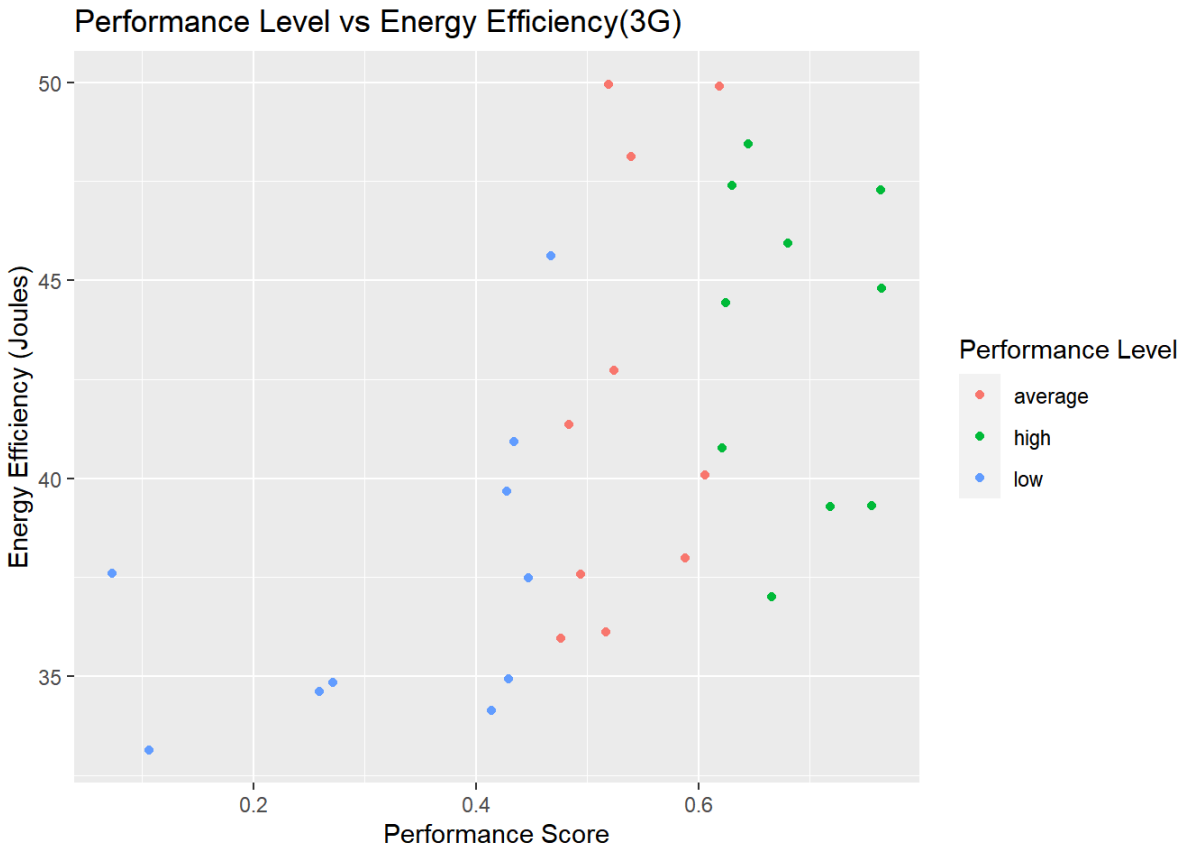
Performance Levels vs Energy Efficiency Scatter Plot

```
ggplot(data.frame(y = threeG_nav_df$batteryStatsAvg), aes(x = threeG_nav_df$performanceLvl, y = y, sample = y, color = threeG_nav_df$performanceLvl)) +
  labs(title="Performance Level vs Energy Efficiency Scatter Plot(3G)", x="Performance Level",  y = "Energy Efficiency (Joules)", color = "Performance Level")+
  geom_point()
```



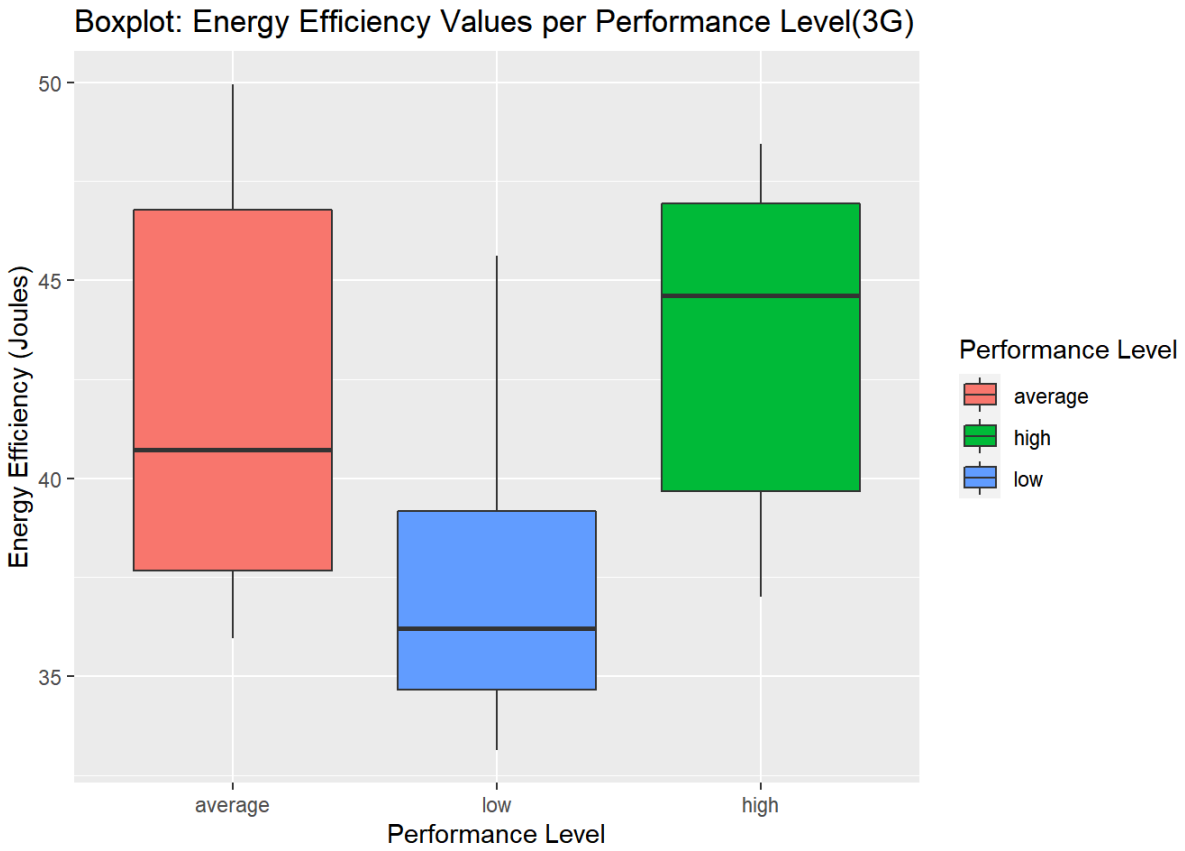
Performance score vs Energy Efficiency Scatter Plot

```
ggplot(data.frame(y = threeG_nav_df$batteryStatsAvg), aes(x = threeG_nav_df$navigationMean, y=y, sample = y, color = threeG_nav_df$performanceLvl)) +
  labs(title="Performance Level vs Energy Efficiency(3G)", x="Performance Score",  y = "Energy Efficiency (Joules)", color = "Performance Level")+
  geom_point()
```



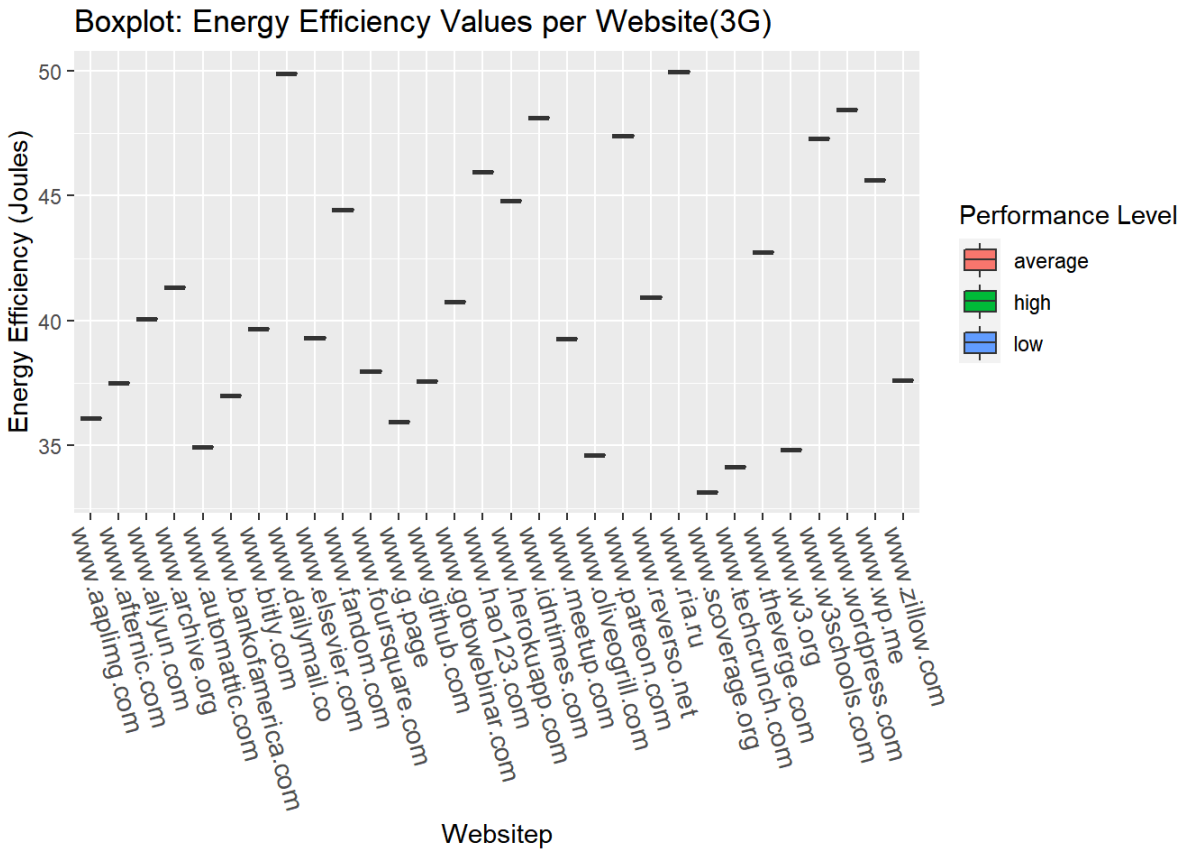
Box plot: Energy Efficiency vs Performance Level

```
ggplot(data.frame(y = threeG_nav_df$batteryStatsAvg), aes(x = rev(threeG_nav_df$performanceLvl), y=y, sample = y, fill = (threeG_nav_df$performanceLvl))) +
  labs(title="Boxplot: Energy Efficiency Values per Performance Level(3G)", x="Performance Level",  y = "Energy Efficiency (Joules)", fill = "Performance Level")+
  geom_boxplot() + scale_x_discrete(breaks=c("high", "average", "low"),
                                    labels=c("low", "average", "high"))
```



Box Plot: Website vs Energy Efficiency

```
ggplot(data.frame(y = threeG_nav_df$batteryStatsAvg), aes(x = threeG_nav_df$website, y=y, sample = y, fill = threeG_nav_df$perf
ormanceLvl)) +
  theme(axis.text.x = element_text(angle= -75, hjust = 0, size = 11))+
  labs(title="Boxplot: Energy Efficiency Values per Website(3G)", x="Websitep",  y = "Energy Efficiency (Joules)", fill = "Perf
ormance Level")+
  geom_boxplot() + scale_x_discrete(limits= rev(levels(threeG_nav_df$website)))
```



Investigating the normality of the dependent variable

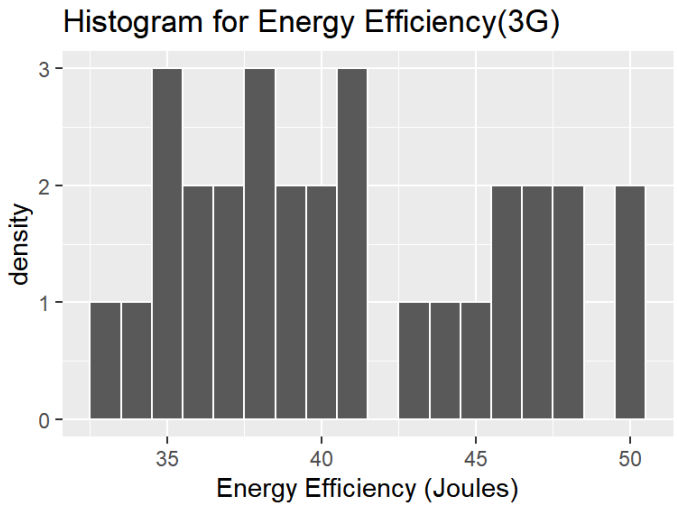
Raw energy efficiency

We analyze the fit of the sample energy efficiency to the normal distribution.

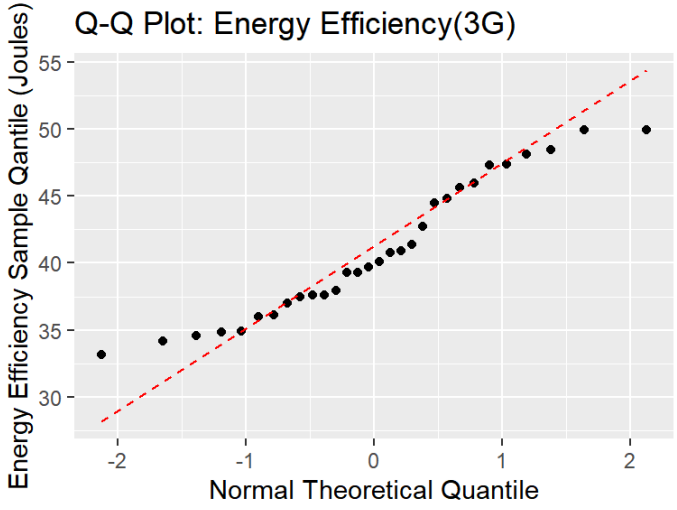
Visualize using histogram and QQ-plot:

```
histoRaw <-qplot(threeG_nav_df$batteryStatsAvg, geom="histogram", main="Histogram for Energy Efficiency(3G)", xlab = "Energy Efficiency (Joules)", ylab = "density", col = I("White"), binwidth=1)
qqplotRaw <-ggplot(data.frame(y = threeG_nav_df$batteryStatsAvg), aes(sample = y)) + stat_qq() + stat_qq_line(col="red", lty=2) + ylab("Energy Efficiency Sample Qantile (Joules)") + xlab("Normal Theoretical Quantile") + ggtitle("Q-Q Plot: Energy Efficiency(3G)")

grid.arrange(histoRaw, ncol=1)
```




```
grid.arrange(qqplotRaw, ncol=1)
```



The **histogram** clearly shows the shape and the spread of the distributions of the data. However, it is more difficult/reliable to assessing normality for smaller sample sizes, such as ours (with 30 observations). This happens because the appearance of the histogram relies on the number of data points (observations) and the number of bars in the plot. As a result, other assessment methods, such as QQ-plots or box plots can offer a better insight regarding the normality of the data.

However, by looking at the histogram, energy efficiency does not seem to fit the normal distribution.

The **QQ-plot** shows all of the observations against a standard normal distribution (with mean 0 and standard deviation 1) and the same number of points. In other words, the actual values of X against the theoretical values of X under the normal distribution are represented in this plot. If the points fall right on the line when normality has been met.

The QQ-plot shows the whole data is fluctuate that the tail part is above the standard normal distribution, while the middle part is lower than the normal distribution. At last, we can say that the data is far from the normality. But by given the small sample size (only 30 observations), it is difficult to draw conclusions about it.

Shapiro-Wilk normality test

This test is suitable for small sample sizes (< 50 samples).

Hypothesis:

H_0: energy efficiency data is normally distributed

H_1: energy efficiency data is NOT normally distributed

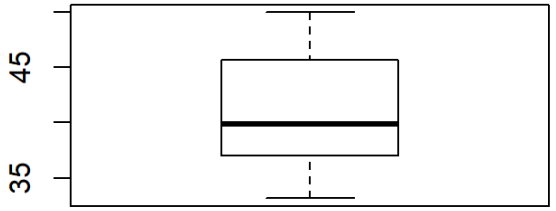
```
shapiro.test(threeG_nav_df$batteryStatsAvg)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  threeG_nav_df$batteryStatsAvg
## W = 0.9, p-value = 0.08
```

Outcome: The p-value for testing H_0 is greater than 0.05, hence we cannot reject the null hypothesis that energy efficiency data is normally distributed.

Skewness

```
boxplot(threeG_nav_df$batteryStatsAvg,names=c("3G Nav Mean"))
```



The box plot shows that the data is potentially right-skewed. This may be due to the small number of measurements (30 observations).

Test for skewness

```
skewness(threeG_nav_df$batteryStatsAvg)
```

```
## [1] 0.303
```

The coefficient of skewness is greater than 0 i.e. $\gamma_1 = 0.303 > 0$, hence data is positively skewed, with the majority of data values less than the mean 40.9. This means most of the values are concentrated on the left side of the graph.

Attempting to fix skewness:

```
batteryStatsSquared = threeG_nav_df$batteryStatsAvg ^ 2
```

Square of the energy efficiency

There is no visible improvement.

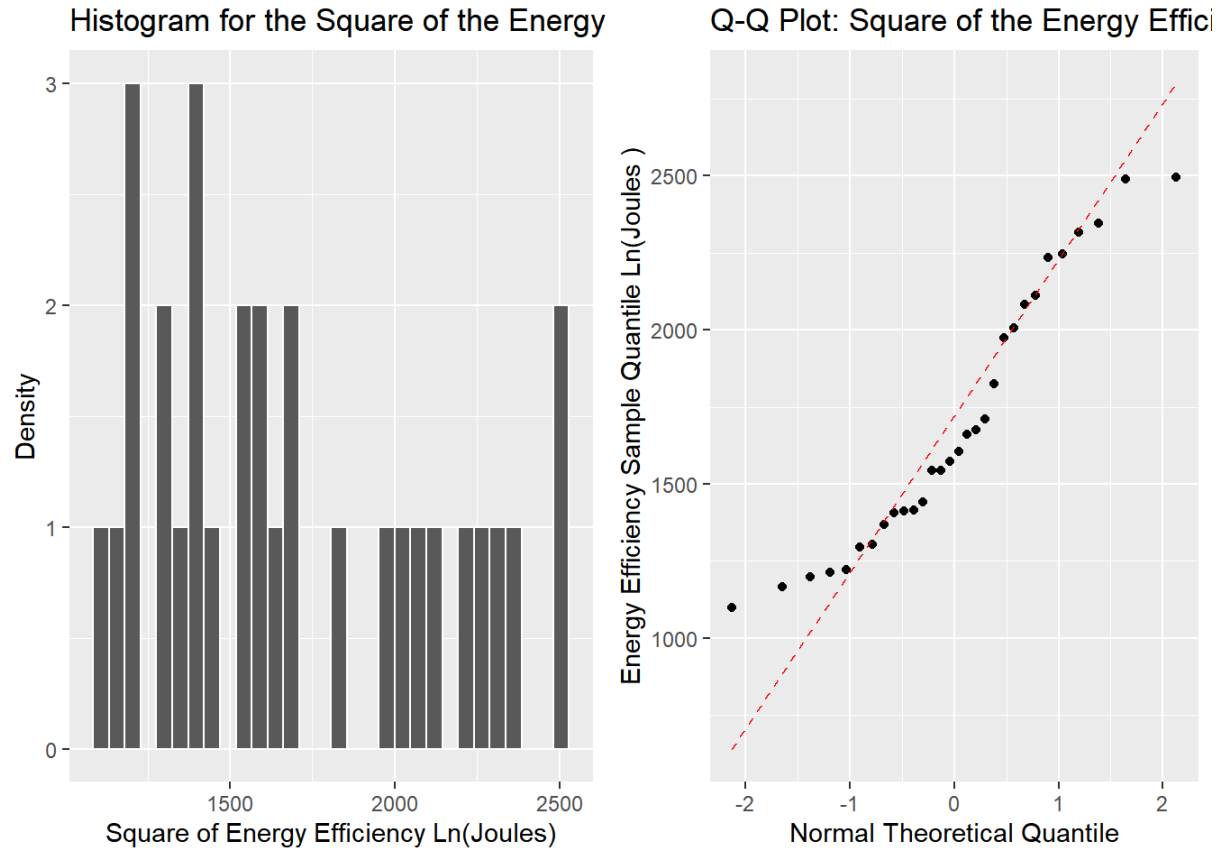
Shapiro-Wilk normality test: The p-value for testing H_0 is less than 0.05, hence the null hypothesis that the data is normally distributed is rejected.

There is no improvement in the skewness, it is bigger than for raw values when considering the square of the data.

```
histLog <- qplot(batteryStatsSquared, geom="histogram", main="Histogram for the Square of the Energy Efficiency(3G) ", xlab =
"Square of Energy Efficiency Ln(Joules)", ylab = "Density", col=I("White"))
qqplotLog <- ggplot(data.frame(y = batteryStatsSquared), aes(sample = y)) + stat_qq() + stat_qq_line(col="red", lty=2) + ylab(
"Energy Efficiency Sample Quantile Ln(Joules )") +
xlab("Normal Theoretical Quantile") + ggtitle("Q-Q Plot: Square of the Energy Efficiency Values(3G)")

grid.arrange(histLog, qqplotLog, ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# Test the data for normality of the square
# Shapiro-Wilk normality test
shapiro.test(batteryStatsSquared)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  batteryStatsSquared
## W = 0.9, p-value = 0.04
```

```
# Test for skewness -- Higher than for raw energy efficiency, positively skewed
skewness(batteryStatsSquared)
```

```
## [1] 0.429
```

Reciprocal of energy efficiency

There is no visible improvement.

Shapiro-Wilk normality test: The p-value for testing H_0 is higher than 0.05, hence the null hypothesis that the data is normally distributed is accepted.

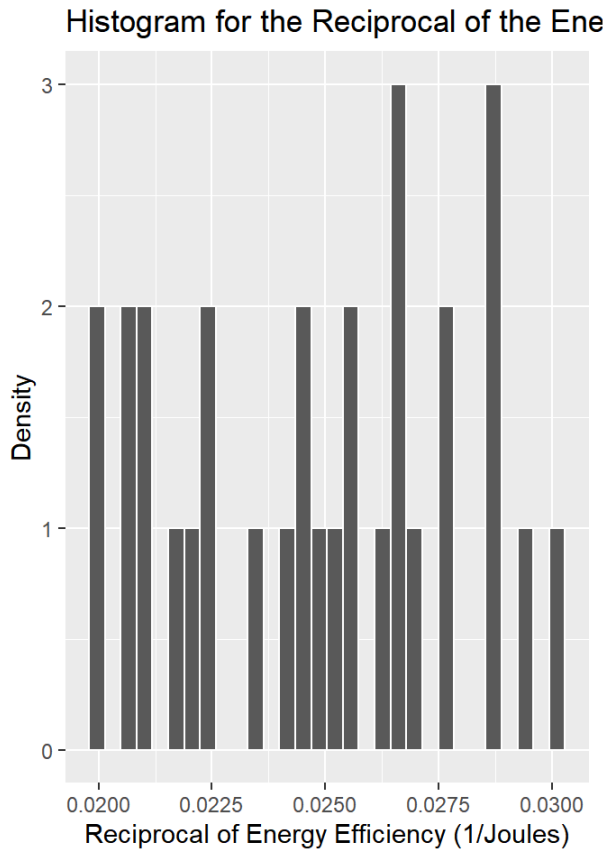
Data is negatively skewed for the reciprocal of energy efficiency.

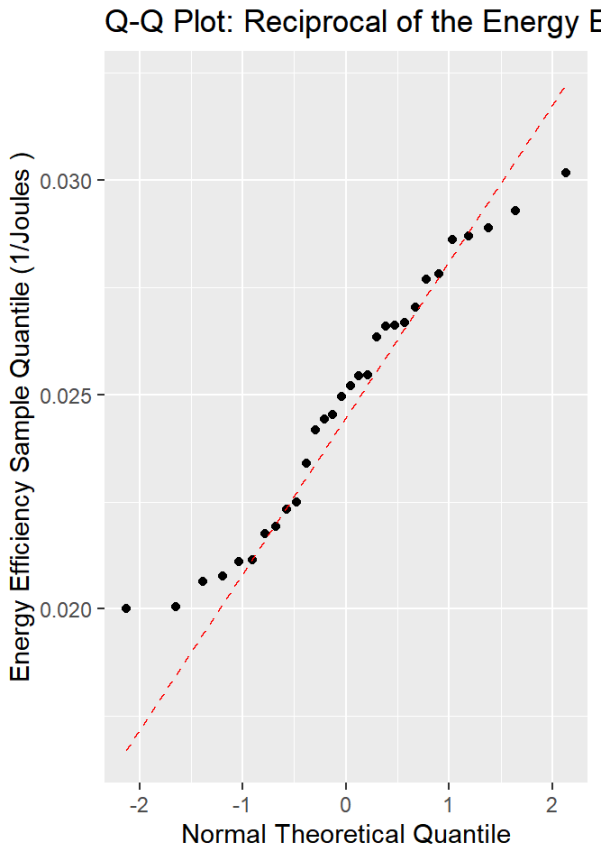
```
batteryStatsReciprocal <- 1/ threeG_nav_df$batteryStatsAvg

# Visualize the reciprocal of the data -- No major improvement
histLog <- qplot(batteryStatsReciprocal, geom="histogram", main="Histogram for the Reciprocal of the Energy Efficiency(3G) ", x
lab = "Reciprocal of Energy Efficiency (1/Joules)", ylab = "Density", col=I("White"))
qqplotLog <- ggplot(data.frame(y = batteryStatsReciprocal), aes(sample = y)) + stat_qq() + stat_qq_line(col="red", lty=2) + yla
b("Energy Efficiency Sample Quantile (1/Joules )") +
xlab("Normal Theoretical Quantile") + ggtitle("Q-Q Plot: Reciprocal of the Energy Efficiency Values(3G)")

grid.arrange(histLog, qqplotLog, ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```






```
# Test the data for normality of the reciprocal
# Shapiro-Wilk normality test
shapiro.test(batteryStatsReciprocal)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  batteryStatsReciprocal
## W = 1, p-value = 0.2
```

```
# Test for skewness -- Higher than for raw energy efficiency, negatively skewed
skewness(batteryStatsReciprocal)
```

```
## [1] -0.0331
```

Log of energy efficiency

The data visualizations look promising.

Shapiro-Wilk normality test: The p-value for testing H_0 is higher than 0.05, hence the null hypothesis that the data is normally distributed is accepted.

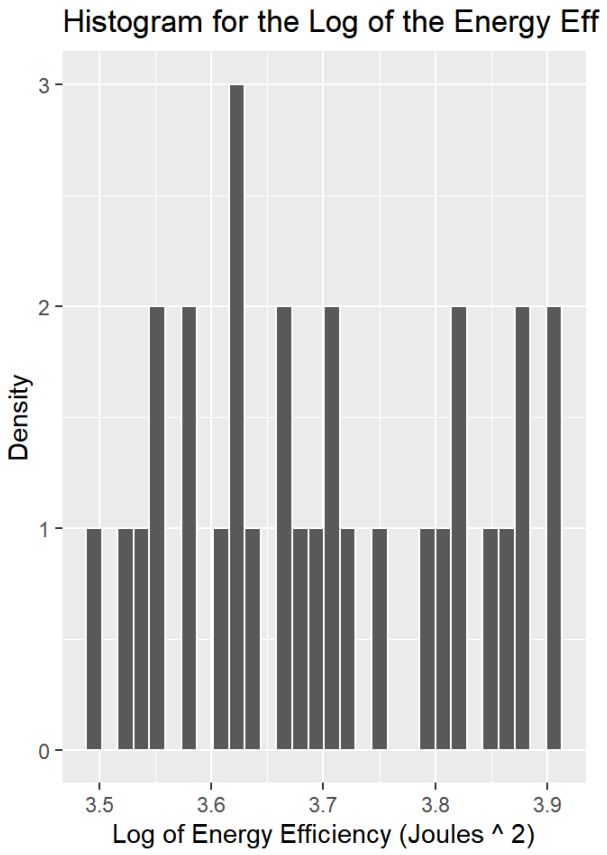
In this case, there is an improvement in skewness, but we cannot assume a normal distribution according to the results of the Shapiro-Wilk test.

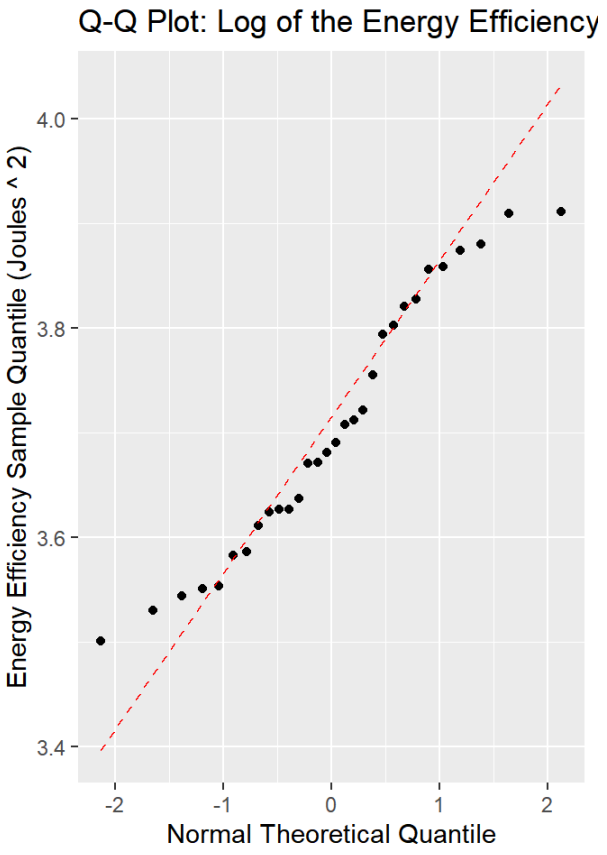
```
batteryStatsLog <- log(threeG_nav_df$batteryStatsAvg)

# Visualize the log of the data -- Promising
histLog <- qplot(batteryStatsLog, geom="histogram", main="Histogram for the Log of the Energy Efficiency(3G) ", xlab = "Log of
 Energy Efficiency (Joules ^ 2)", ylab = "Density", col=I("White"))
qqplotLog <- ggplot(data.frame(y = batteryStatsLog), aes(sample = y)) + stat_qq() + stat_qq_line(col="red", lty=2) + ylab("Ener
gy Efficiency Sample Quantile (Joules ^ 2)") +
xlab("Normal Theoretical Quantile") + ggtitle("Q-Q Plot: Log of the Energy Efficiency Values(3G)")

grid.arrange(histLog, qqplotLog, ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```






```
# Test the data for normality of the log
# Shapiro-Wilk normality test
shapiro.test(batteryStatsLog)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  batteryStatsLog
## W = 0.9, p-value = 0.1
```

```
# Test for skewness -- Substantially lower but still skewed
skewness(batteryStatsLog)
```

```
## [1] 0.17
```

As a result, we consider the raw energy efficiency when performing statistical tests to test the hypothesis. We can perform parametric tests in this case, taking into account the above observations and the results of the Shapiro-Wilk test. In terms of skewness, we consider 0.30 an acceptable value (i.e., moderate skewness) for the data being normally distributed.

Hypothesis testing

$H_0 : \mu 1_{high} = \mu 1_{average} = \mu 1_{low}$

i.e.,: The mean energy efficiency does not significantly differ among web apps having different navigation timing levels

$H_a : \mu 1_{high} \neq \mu 1_{average} \vee \mu 1_{average} \neq \mu_{low} \vee \mu 1_{low} \neq \mu_{high}$

i.e.,: The mean energy efficiency significantly differs among web apps having different levels of navigation timing for at least one pair of navigation timing levels.

Test used and motivation: We have one factor, >2 treatments, hence we perform Analysis of Variance (ANOVA) to test for effect of the treatments (navigation timing levels) on the mean energy efficiency.

One-way ANOVA can be used in the analysis to test the effect of each performance category (i.e., navigation timing, load speed) on energy efficiency. In this case, we assume: - The dependent variable (energy efficiency) is continuous - TRUE - The samples are independent - TRUE (guaranteed by the experimental design) - Normal distribution of the dependent variable between the independent groups - TRUE - Residuals (i.e., errors in the sample) should be normally distributed - TO BE CHECKED AFTER FITTING THE MODEL - Homoscedasticity (variance between groups should be the same) - TO BE CHECKED AFTER FITTING THE MODEL

```
# quantitative variable, representing the average power consumed in each round of the experiment
efficiency = as.numeric(threeG_nav_df$batteryStatsAvg)
# the treatment, categorical variable with 3 levels - high, average, low
performance = as.factor(threeG_nav_df$performanceLvl)
summary(efficiency) # obtain a numeric summary
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    33.1    37.1    39.9    40.9    45.4    50.0
```

```
summary(performance) # listed as categorical variable, with the number of observations at each level
```

```
## average    high     low
##      10      10      10
```

```
threeG_nav_df.aov <- aov(efficiency~performance, data = threeG_nav_df)
summary(threeG_nav_df.aov)
```

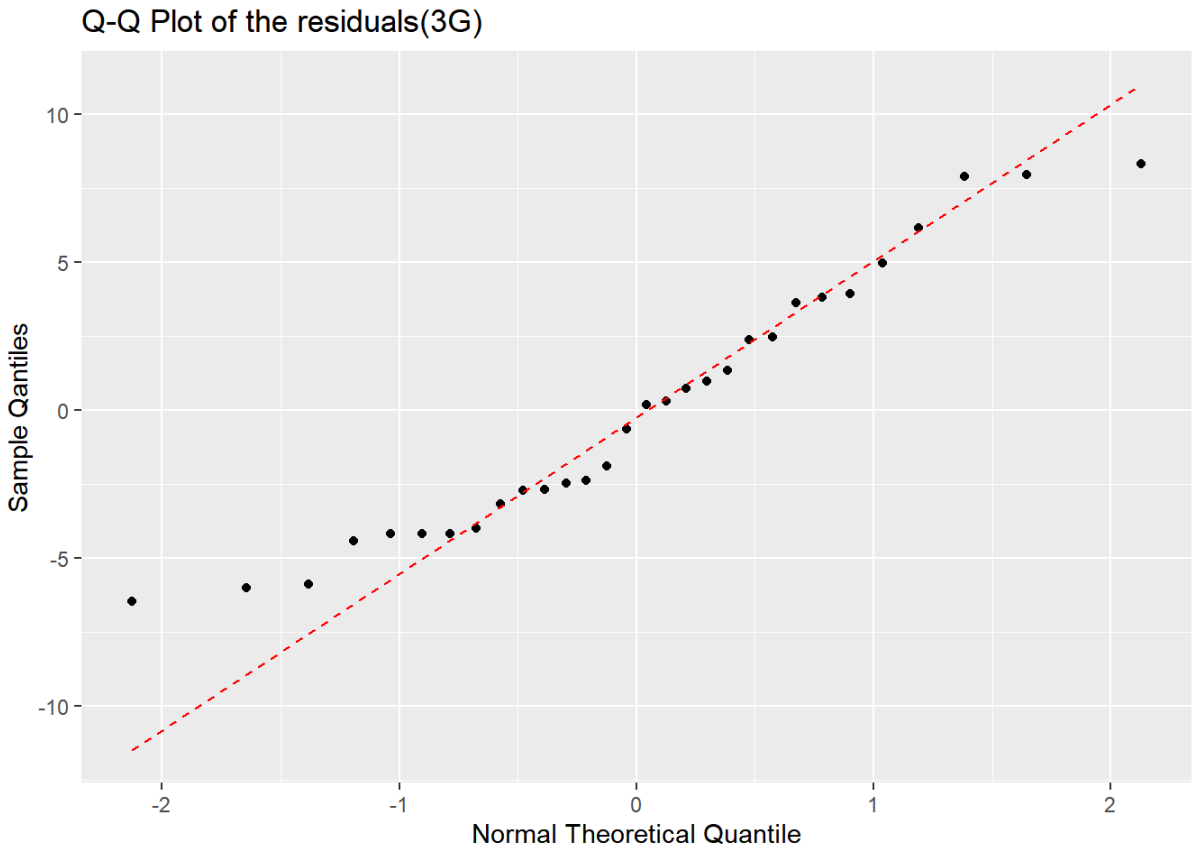
```
##             Df Sum Sq Mean Sq F value Pr(>F)
## performance  2    207   103.5    5.02  0.014 *
## Residuals   27    557    20.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Outcome: The case factor is giving a p-value lower than 0.05, meaning that we can reject the null hypothesis, stating that the mean energy efficiency does not significantly differ among web apps having different navigation timing levels. Hence, there is statistical significance between the variables.

Check normality of residuals

QQ-plot

```
ggplot(data.frame(y = residuals(threeG_nav_df.aov)), aes(sample = y)) + stat_qq() + stat_qq_line(col="red", lty=2) + ylab("Samp
le Qantiles") + xlab("Normal Theoretical Quantile") + ggtitle("Q-Q Plot of the residuals(3G)")
```



Shapiro-Wilk normality test

Hypothesis:

H_0: energy efficiency data is normally distributed

H_1: energy efficiency data is NOT normally distributed

```
shapiro.test(residuals(threeG_nav_df.aov))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(threeG_nav_df.aov)
## W = 0.9, p-value = 0.1
```

Outcome: The p-value for testing H_0 is greater than 0.05, hence we cannot reject the null hypothesis that energy efficiency data is normally distributed.

Check homoscedacity

The spread of the data points between the groups must be the same.

Levene's test with one independent variable

Hypothesis:

H_0: the population variances are equal

H_1: the population variances are NOT equal

```
leveneTest(efficiency~performance, data = threeG_nav_df)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  2    0.78   0.47
##       27
```

Outcome:

The p-value of Levene's test is higher than the significance level (0.05), hence the differences in sample variances are likely to have occurred based on random sampling from a population with equal variances. Thus, we cannot reject the null hypothesis of equal variances.

##Hence, the assumptions of ANOVA are all valid.

##We conclude that there is no evidence of statistical significance between performance and energy efficiency.