Assignment 3, EDDA 2017

Martin de la Riva(11403799) and Kieran O'Driscoll(11426438), Group 23

24 April 2017

Assignment 2

Exercise 1

1.

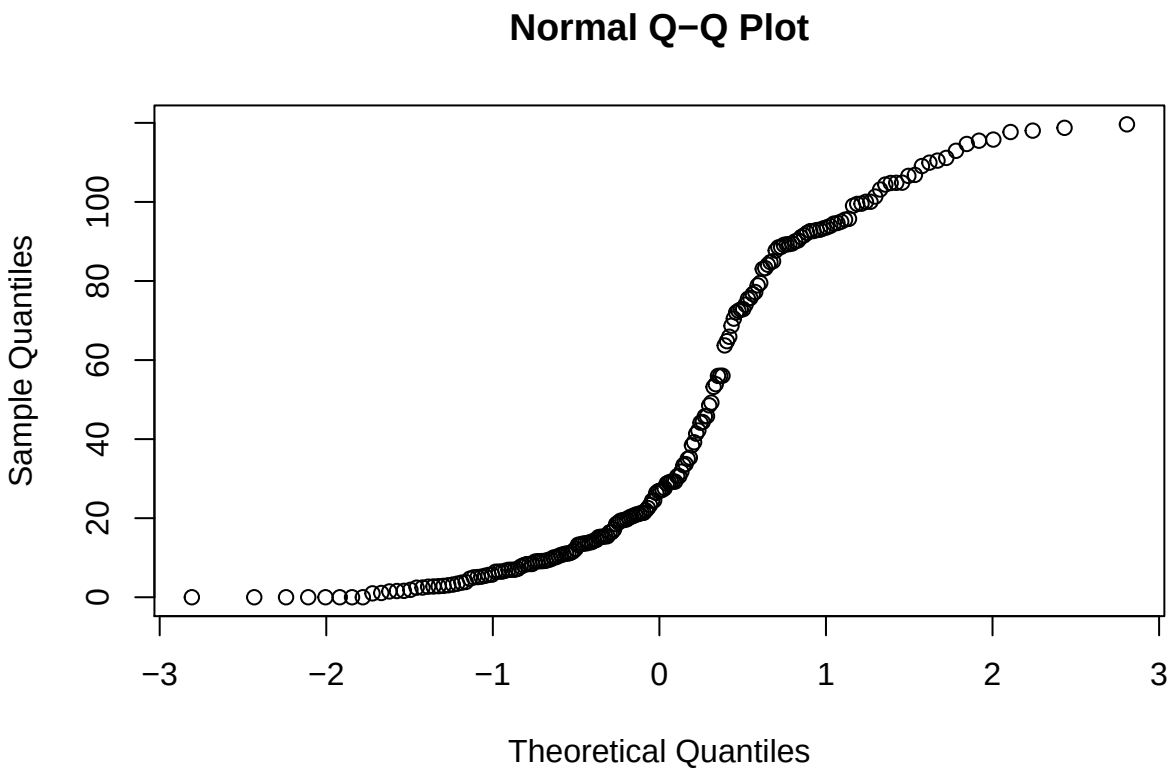
For this exercise we first create an exponential distribution of a significant amount of elements with $\lambda = 0.035$. With this, we get a precise value fo what the median of such distribution is.

```
telephone=read.table("telephone.txt",header=TRUE)

# exponential distribution of 20000 elements
exp_d = rexp(20000, rate=0.035)
exp_d_median = median(exp_d)
```

Then, we perform a Wilcoxon test to see if the median is close to the previously mentioned distribution. In order to perform this test, we first have to assume that the data is a random sample from a symmetric population with a certain median m. By looking at the QQ-plot of the dataset, We can make the symmetric assumption and therefore go on with our Wilcoxon test.

```
# Test if data may come from a Symmetric population
qqnorm(telephone[,"Bills"])
```



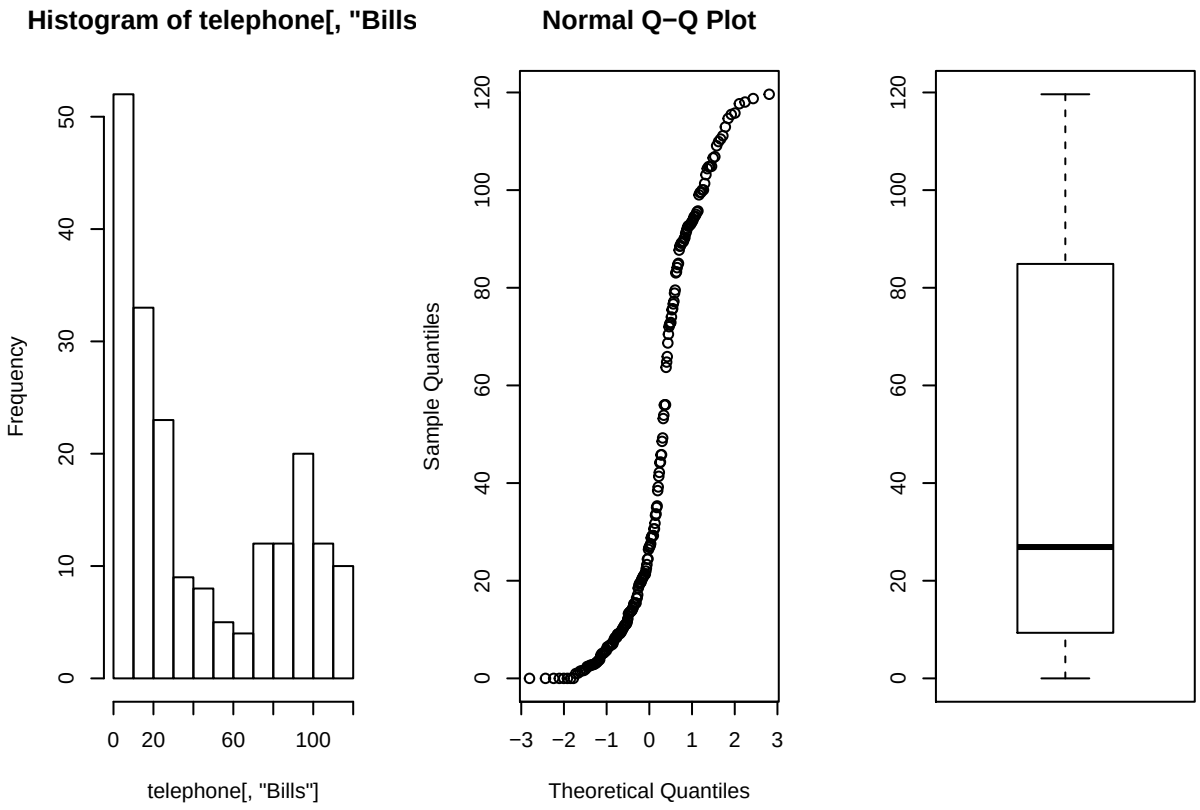

```r
# Rejected, median is not close to exponential median
wilcox.test(telephone[,"Bills"], mu = exp_d_median)
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  telephone[, "Bills"]
## V = 14584, p-value = 3.172e-08
## alternative hypothesis: true location is not equal to 19.78243
```

This results in a really low p-value, therefore rejecting the hypothesis that the data stems from an exponential distribution with rate $\lambda = 0.035$.

2.

```r
par(mfrow=c(1,3))
hist(telephone[,"Bills"])
qqnorm(telephone[,"Bills"])
boxplot(telephone[,"Bills"])
```



It can be seen that most people pay around 0 to 20$. If you wanna be competitive you should start pricing around that value. 30-70$ is an unsuccesfull pricing, as it has the lowest clients, while it starts to increase at more than 70$.

In Conclusion:
Either offer a cheap service of less than 30$; or create a high quality service with a price lower than 80$, so you win clients that are used to pay more.

Exercise 2

Firstly, we have different measures in the datasets. We need to transform these values in order to have the 3 datasets with the same measures and criteria.
We will use the 1879-1882 criteria, which is in km/s, and substracting 299000 to it. In order to do this we will have to transform light3 into this criteria. We know light3 was calculated measuring how many microseconds it takes to perform 7442 km, and afterwards substracting 24.8 milliseconds.

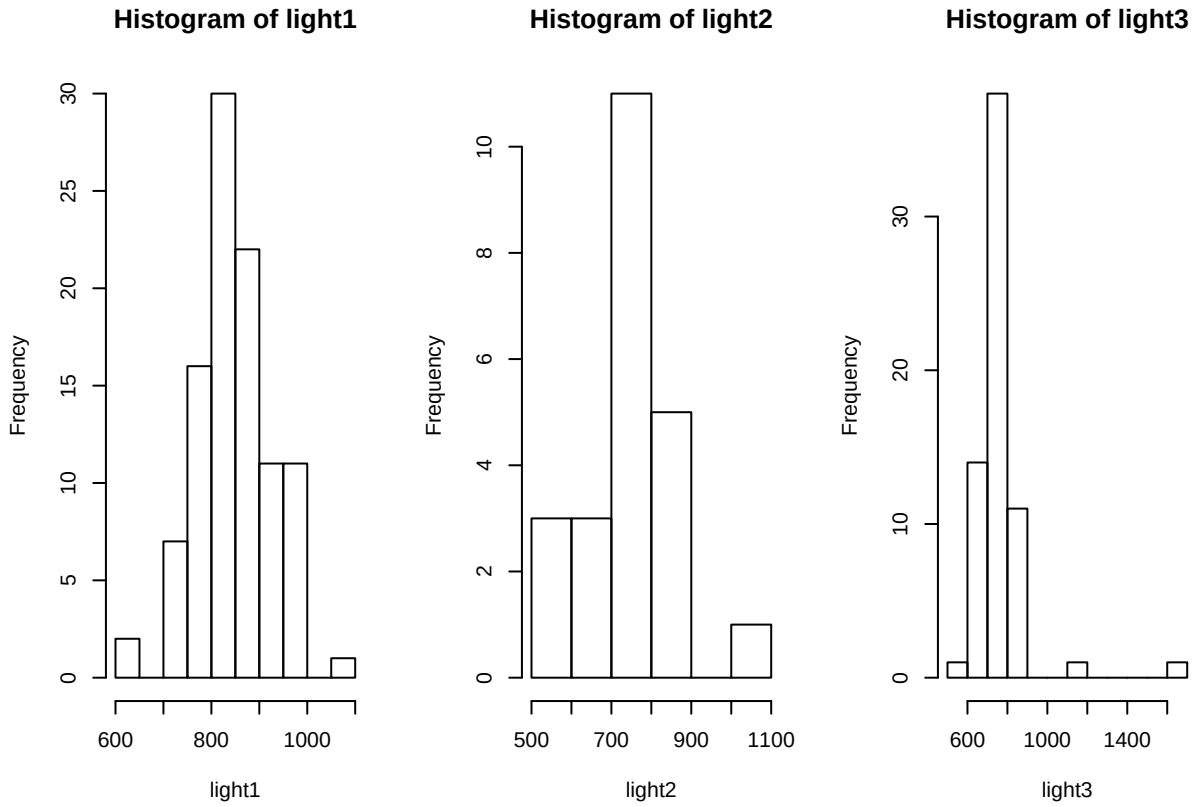
NOTE: We assumed the assignment means milliseconds when talking about Newcomb measurements (24.8 substracted), as after multiplying by 1000 you do get microseconds, and otherways the measurements don't match by any means the speed of light.

```
light1 = scan("light1879.txt")
light2 = scan("light1882.txt")
light3 = scan("light.txt")
```

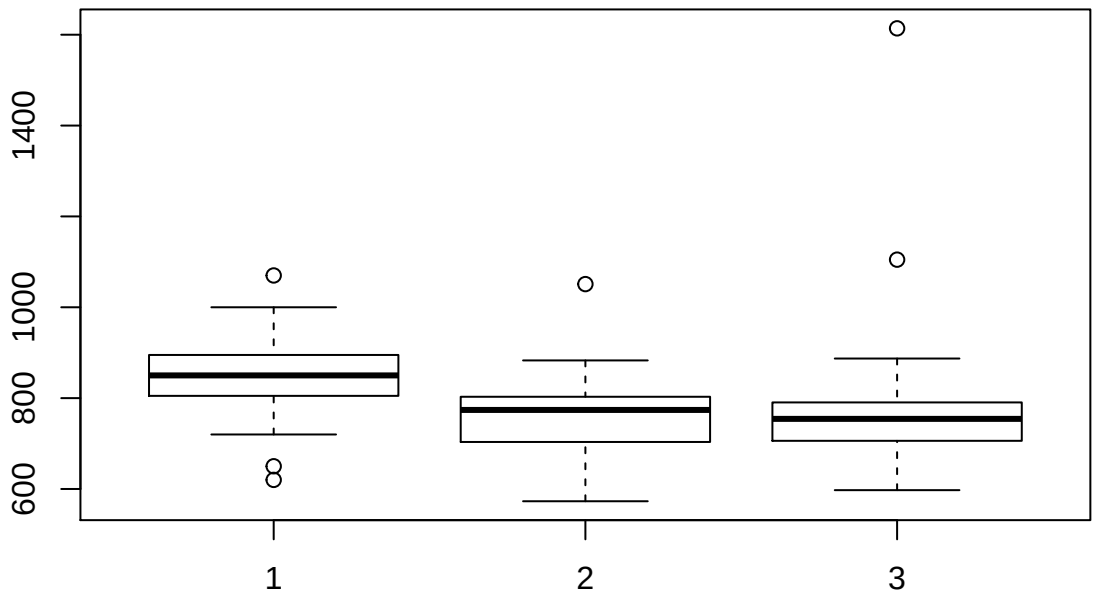
```r
# Transform light3 to 1879-1882 criteria
light3 = (light3/1000)+24.8 # milliseconds
light3 = (light3/1000) # seconds to perform 7442 km
light3 = (7442/light3) # in 1 second it will perform x km (transform to km/s)
light3 = light3 - 299000 # final 1879-1882 criteria
```

1.

```r
par(mfrow=c(1,3))
hist(light1)
hist(light2)
hist(light3)
```




```r
par(mfrow=c(1,1))
boxplot(light1, light2, light3)
```



Michelson second experiment coincides with Newcomb measurements, while Michelson's first experiment (light1) differs from the others.
Although histograms weren't as helpful as we expected them to be, the Boxplots were really useful to see the similarities and differences between the datasets.

2.

```
t.test(light1)$conf.int
```

```
## [1] 836.7226 868.0774
## attr(,"conf.level")
## [1] 0.95
```

```
t.test(light2)$conf.int
```

```
## [1] 709.8976 802.5372
## attr(,"conf.level")
## [1] 0.95
```

```
t.test(light3)$conf.int
```

```
## [1] 731.9112 795.8250
## attr(,"conf.level")
## [1] 0.95
```

NOTE: We maintained 1879-1882 criteria (km/sec - 299000). We also tried with only km/sec, but the confidence intervals where exactly the same although adding 299000 to them.

3.

Again, 2 and 3 coincide (light3 interval is inside light2 confidence interval values), while 1 differs, having a confidence interval with higher values, and being completely outside of the two other ones.

4.

```
most_precise_speed_of_light = 299792.458 - 299000
most_precise_speed_of_light
```

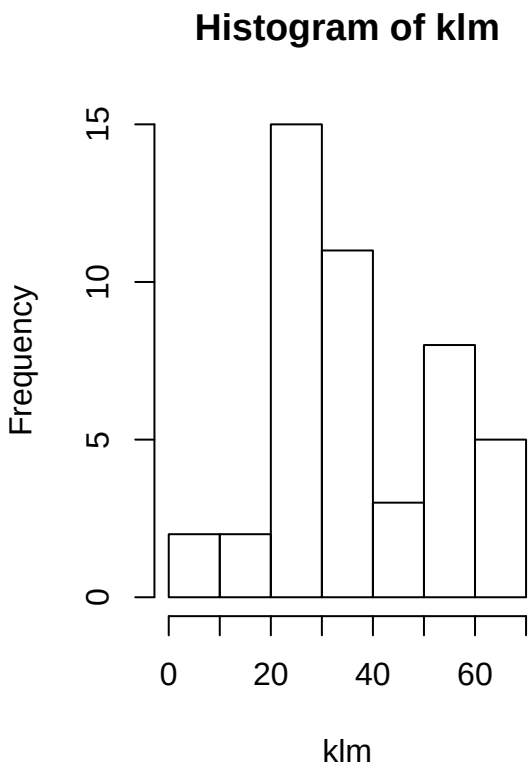
```
## [1] 792.458
```

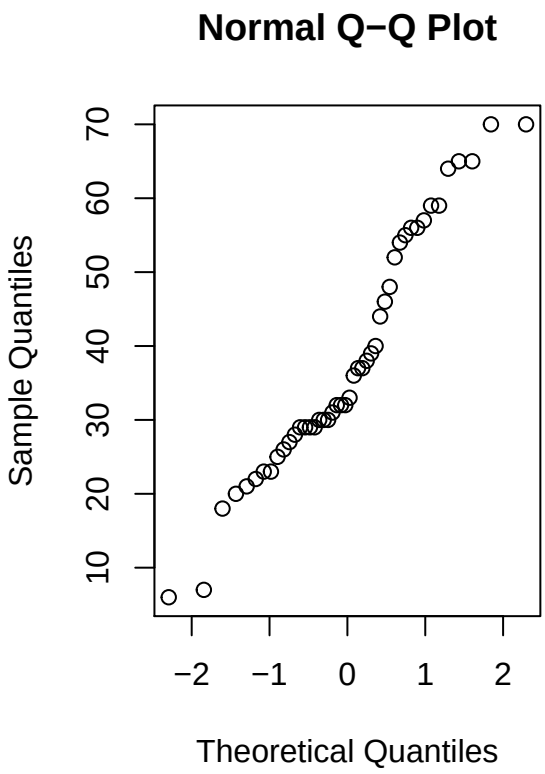
The most precise actual speed of light shows that Michelson's first experiment measures were clearly off, while his second and Newcomb's were much closer to the current accurate solution.

Exercise 3

1.

```
klm = scan("klm.txt")
# First, we check distribution of the data.
klm <- klm[klm<71]
par(mfrow=c(1,2))
hist(klm)
qqnorm(klm)
```





Assuming the maximum delivery duration of the parts is 70 days, we first ignore outliers that are out of that assumption.
In order to test the median we can know two methods, a sign test or a Wilcoxon test. Looking at the histogram and QQ-plot, we can assume the data to stem from a symmetric population with a certain median. Therefore, we can use Wilcoxon test, which is preferred as it is based on more information about the dataset.
In order to test wether the median is 32 or less, we will firstly test if it's equal to 32, and if it is rejected, we will check the lower values to see if it can have a lower median.

```
wilcox.test(klm,mu=32)
```

```
## Warning in wilcox.test.default(klm, mu = 32): cannot compute exact p-value
## with ties
```

```
## Warning in wilcox.test.default(klm, mu = 32): cannot compute exact p-value
## with zeroes
```

```
##
##  Wilcoxon signed rank test with continuity correction
##
## data:  klm
## V = 638.5, p-value = 0.04627
## alternative hypothesis: true location is not equal to 32
```

We get a low p-value, and therefore the hypothesis that the median is equal to 32 is rejected. When

testing lower values, it can be seen how the p-value decreases, and therefore we also reject the hypothesis that the median of the population may be lower than 32.

```
wilcox.test(klm,mu=31)[[3]]
```

```
## Warning in wilcox.test.default(klm, mu = 31): cannot compute exact p-value
## with ties
```

```
## Warning in wilcox.test.default(klm, mu = 31): cannot compute exact p-value
## with zeroes
```

```
## [1] 0.02354406
```

```
wilcox.test(klm,mu=30)[[3]]
```

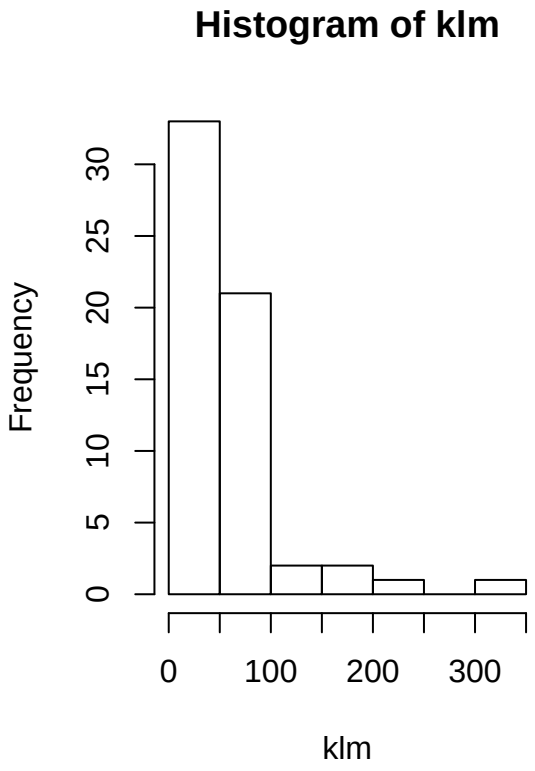
```
## Warning in wilcox.test.default(klm, mu = 30): cannot compute exact p-value
## with ties
```

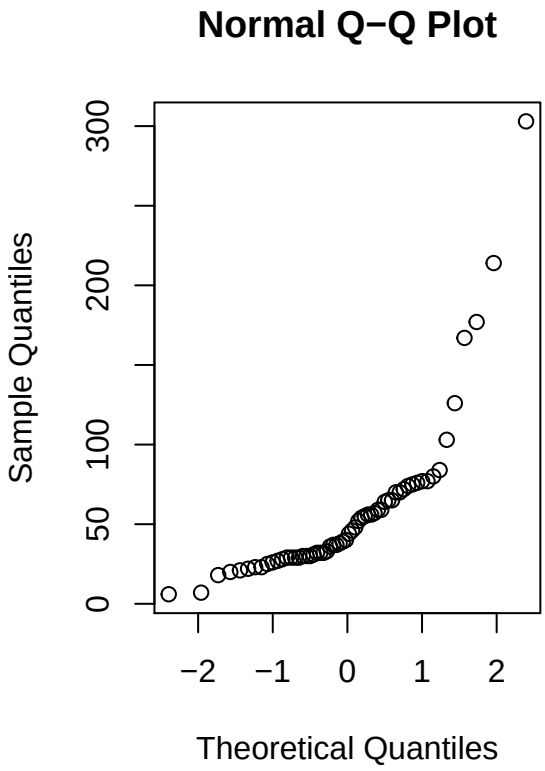
```
## Warning in wilcox.test.default(klm, mu = 30): cannot compute exact p-value
## with zeroes
```

```
## [1] 0.004790985
```

2.

```
klm = scan("klm.txt")
par(mfrow=c(1,2))
hist(klm)
qqnorm(klm)
```





As it can be seen in the plots, in this case we cannot make the symmetric assumption, and therefore we have to perform a sign Binomial test.

```
binom.test(sum(klm>70),sum(!!klm),p=0.1)
```

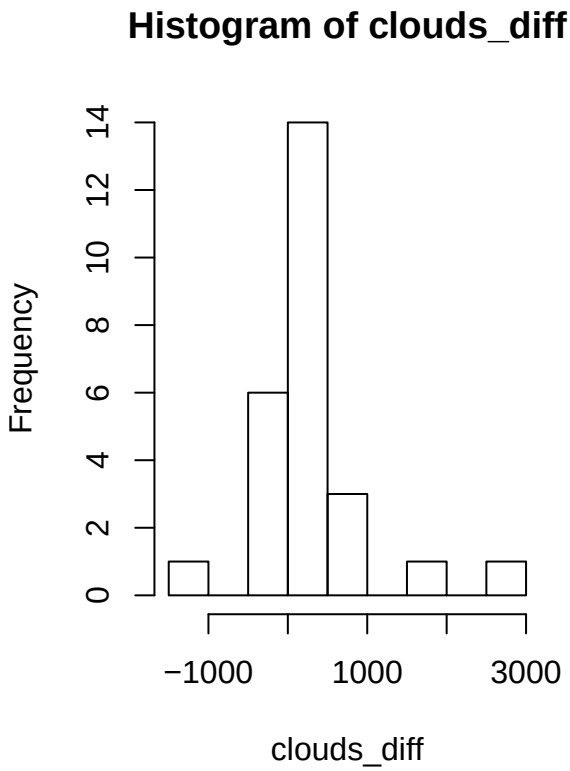
```
##
##  Exact binomial test
##
## data:  sum(klm > 70) and sum(!(!klm))
## number of successes = 14, number of trials = 60, p-value =
## 0.002028
## alternative hypothesis: true probability of success is not equal to 0.1
## 95 percent confidence interval:
##  0.1338373 0.3603828
## sample estimates:
## probability of success
##              0.2333333
```

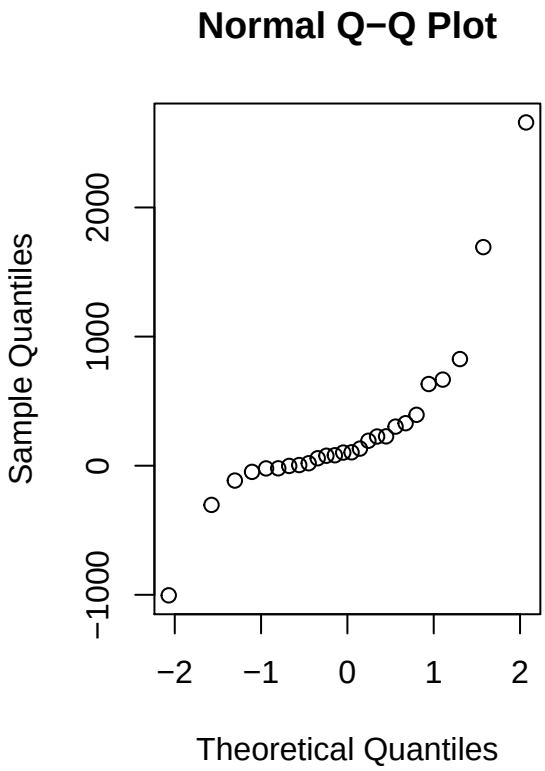
The test returns a confidence interval between 13.38% and 36.04%, and therefore KLM criterion (at most 10% of parts take more than 70 days) is not met.

Exercise 4

```
par(mfrow=c(1,2))
clouds=read.table("clouds.txt",header=TRUE)

clouds_diff = clouds[,1]-clouds[,2]
hist(clouds_diff)
qqnorm(clouds_diff)
```





Two-paired test

Data is clearly paired as it arises from the same individual (cloud) at different points in time.

```
t.test(clouds[,1],clouds[,2],paired=TRUE)
```

```
##
##  Paired t-test
##
## data:  clouds[, 1] and clouds[, 2]
## t = 2.1204, df = 25, p-value = 0.04407
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##    7.961957 546.883428
## sample estimates:
## mean of the differences
##                277.4227
```

According to a paired t-test, silver nitrate does have an effect, giving a confidence interval of difference of (8, 547). As it is a confidence interval with positive values, silver nitrate would result in more rain.
Although this is the result we expected, a Two-Paired test assumes that the sample comes from a normal distribution, which we cannot as it can be seen in the previous plots.

Mann-Whitney test

```
wilcox.test(clouds[,1],clouds[,2])
```

```
## Warning in wilcox.test.default(clouds[, 1], clouds[, 2]): cannot compute
## exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  clouds[, 1] and clouds[, 2]
## W = 473, p-value = 0.01383
## alternative hypothesis: true location shift is not equal to 0
```

In order to perform a Mann-Whitney test, we assume that the first sample stems for population F and the second sample from population G, and we test if F=G. As we can make this assumption, the test is appropiate.
As the p-value is low, we can reject the hypothesis that seeded and unseeded clouds come from the same population. This means that there is certainly a difference between clouds with silver nitrate and without it.
The underlying distribution of seeded clouds is shifted to the right from that of unseeded. (i.e. seeded clouds have bigger values than unseeded)

Kolmogorov-Smirnov test

```
ks.test(clouds[,1],clouds[,2])
```

```
## Warning in ks.test(clouds[, 1], clouds[, 2]): cannot compute exact p-value
## with ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  clouds[, 1] and clouds[, 2]
## D = 0.42308, p-value = 0.01905
## alternative hypothesis: two-sided
```

Again, the assumption for Kolmogorov is similar to Mann-Whitney's and therefore we can perform the test. The p-value is low again and therefore we can conclude that the samples come from different populations, being the mean of seeded clouds higher than those unseeded.

2.

```
sqrt_clouds_1 = sqrt(clouds[,1])
sqrt_clouds_2 = sqrt(clouds[,2])
t.test(sqrt_clouds_1,sqrt_clouds_2,paired=TRUE)
```

```
##
##  Paired t-test
##
## data:  sqrt_clouds_1 and sqrt_clouds_2
## t = 2.682, df = 25, p-value = 0.01278
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##   1.656326 12.617061
## sample estimates:
## mean of the differences
##                7.136693
```

```
wilcox.test(sqrt_clouds_1,sqrt_clouds_2)
```

```
## Warning in wilcox.test.default(sqrt_clouds_1, sqrt_clouds_2): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  sqrt_clouds_1 and sqrt_clouds_2
## W = 473, p-value = 0.01383
## alternative hypothesis: true location shift is not equal to 0
```

```
ks.test(sqrt_clouds_1,sqrt_clouds_2)
```

```
## Warning in ks.test(sqrt_clouds_1, sqrt_clouds_2): cannot compute exact p-
## value with ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  sqrt_clouds_1 and sqrt_clouds_2
## D = 0.42308, p-value = 0.01905
## alternative hypothesis: two-sided
```

For the Two-Paired test we get an smaller confidence interval, but the hypothesis is still rejected, with an even smaller p-value. For Mann-Whitney and Kolmogorov we get the same exact values, for p-values, W (in Mann-Whitney) and D (in Kolmogorov)

3.

```
sqrt_clouds_1 = sqrt(sqrt(clouds[,1]))
sqrt_clouds_2 = sqrt(sqrt(clouds[,2]))
t.test(sqrt_clouds_1,sqrt_clouds_2,paired=TRUE)
```

```
##
##  Paired t-test
##
## data:  sqrt_clouds_1 and sqrt_clouds_2
## t = 2.8413, df = 25, p-value = 0.008811
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.2673422 1.6759523
## sample estimates:
## mean of the differences
##               0.9716472
```

```
wilcox.test(sqrt_clouds_1,sqrt_clouds_2)
```

```
## Warning in wilcox.test.default(sqrt_clouds_1, sqrt_clouds_2): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  sqrt_clouds_1 and sqrt_clouds_2
## W = 473, p-value = 0.01383
## alternative hypothesis: true location shift is not equal to 0
```

```
ks.test(sqrt_clouds_1,sqrt_clouds_2)
```

```
## Warning in ks.test(sqrt_clouds_1, sqrt_clouds_2): cannot compute exact p-
## value with ties
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  sqrt_clouds_1 and sqrt_clouds_2
## D = 0.42308, p-value = 0.01905
## alternative hypothesis: two-sided
```

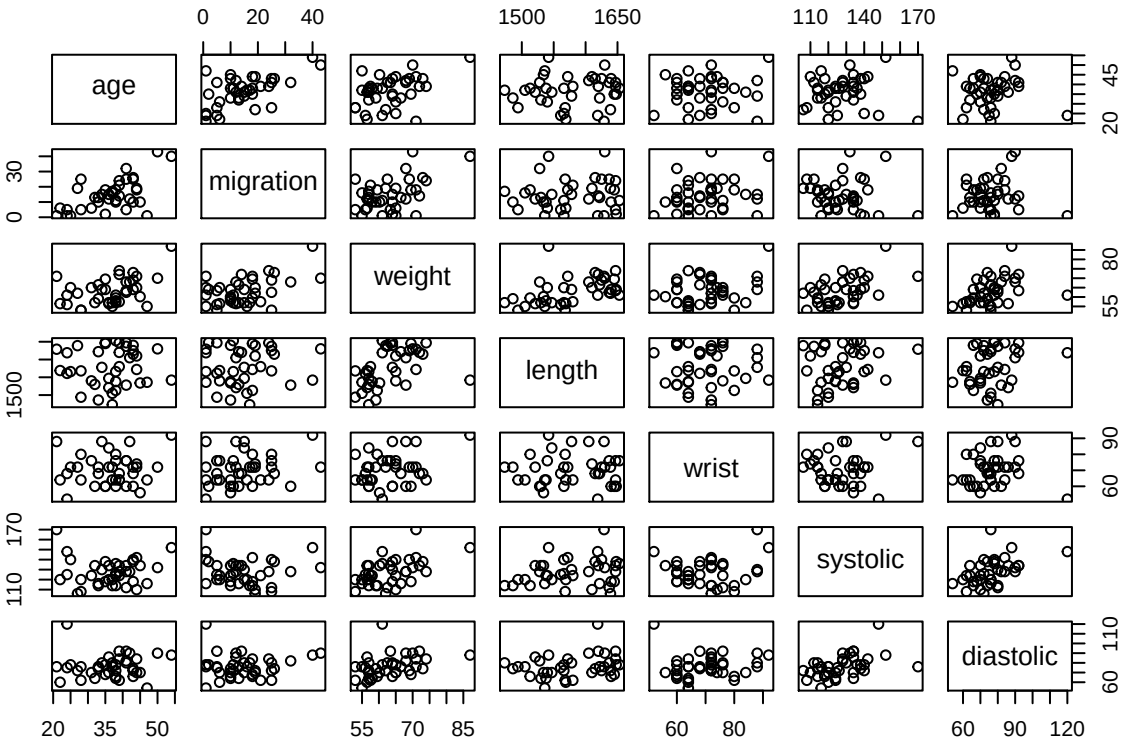
Again, we get a smallet confidence interval for Two-Paired test with an even smaller p-value, while Mann-Whitney and Kolmogorov maintain their same exact values.

Assignment 3

Exercise 1

Q1

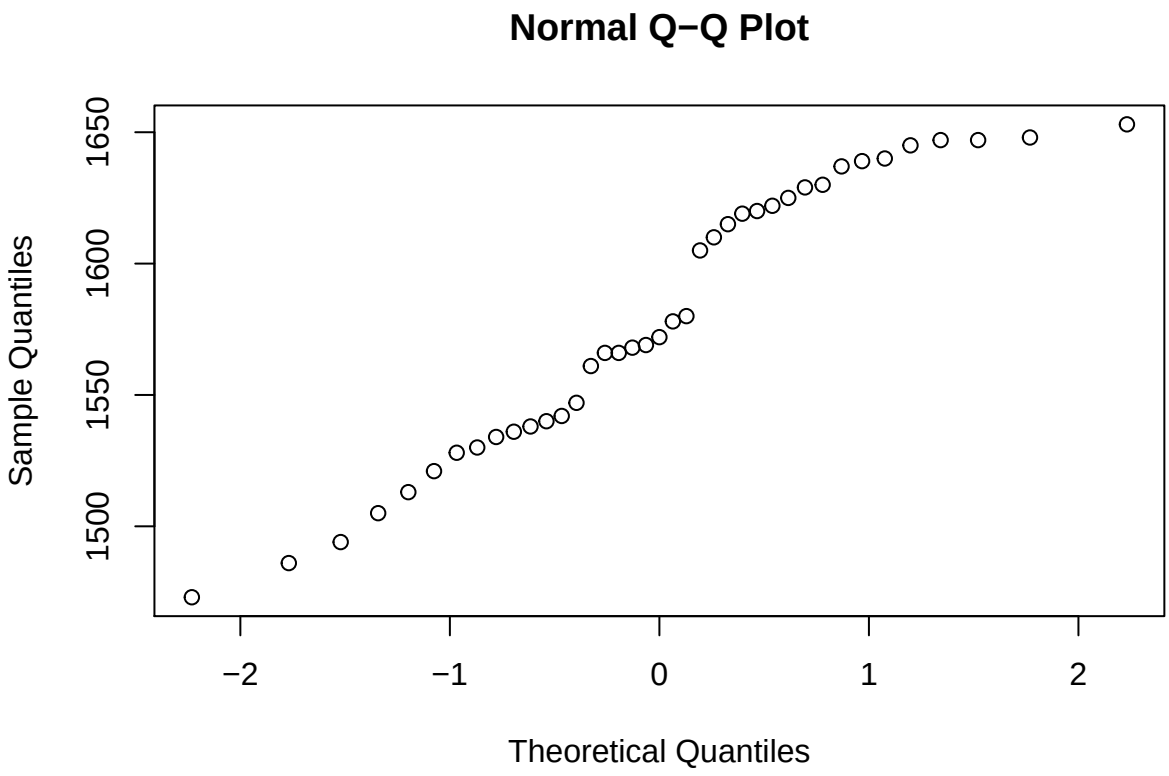
```
data = read.table("data/peruvians.txt", header=TRUE)
pairs(data[,-c(5,6,7)])
```



Based on the diagram above, age, weight and perhaps a case can be made for diastolic. These were chosen because their scatter plots show a cluster that shows the values are in porportion with migration whereas the othe graphs have a scatter plot that don't show any connection with migration.

Q2

```
qqnorm(data[['length']])
```




```
attach(data)
```

Peruvians is not drawn from a normal distribution so therefore will use Spearmen correlation test to test for correlation.

```
cor.test(age,migration,method="spearman")
```

```
## Warning in cor.test.default(age, migration, method = "spearman"): Cannot
## compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  age and migration
## S = 5176.6, p-value = 0.002189
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.4760575
```

The p-value recived is 0.002189 which falls under the 0.05 signicant level. Therefore the value is correlated with migration. Since the value is quite low, it can be said there is a high positive correlation as the correlation increases as values get bigger.

```r
cor.test(length,migration,method="spearman")
```

```
## Warning in cor.test.default(length, migration, method = "spearman"): Cannot
## compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  length and migration
## S = 9044.3, p-value = 0.6087
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## 0.08458432
```

With a P-value of 0.6087 it's clear that lenght and migration do not have a correlation.

```r
cor.test(wrist,migration,method="spearman")
```

```
## Warning in cor.test.default(wrist, migration, method = "spearman"): Cannot
## compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  wrist and migration
## S = 7712.8, p-value = 0.1797
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.2193498
```

Wrist gets a p-value of 0.1797 which means there is no correlation.

```r
cor.test(systolic,migration,method="spearman")
```

```
## Warning in cor.test.default(systolic, migration, method = "spearman"):
## Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  systolic and migration
## S = 11544, p-value = 0.3054
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## -0.1684286
```

Systolic has no correlation with migration, p-value= 0.3054

```r
cor.test(diastolic,migration,method="spearman")
```

```
## Warning in cor.test.default(diastolic, migration, method = "spearman"):
## Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  diastolic and migration
## S = 9137.6, p-value = 0.6494
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## 0.07514098
```

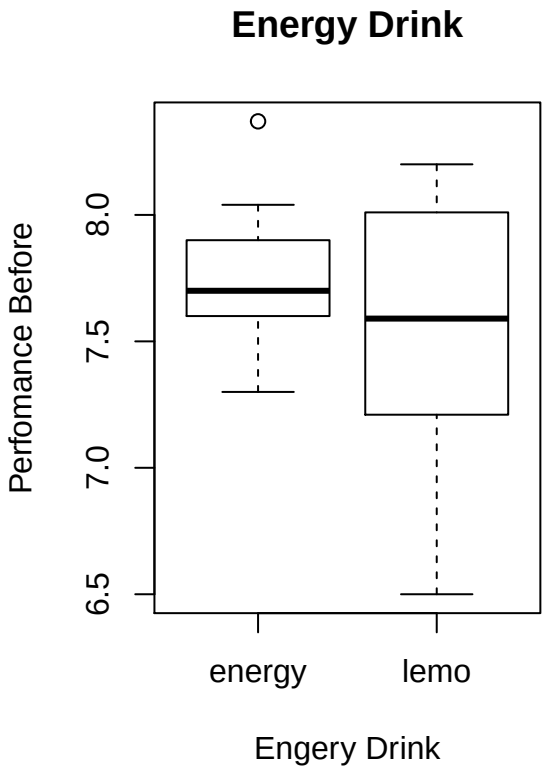
Diastolic has no correlation, the p-value was 0.6494

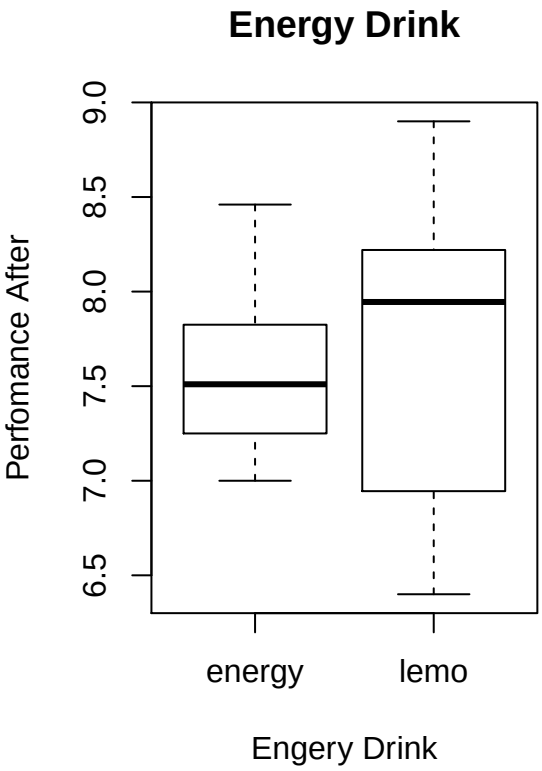
```r
cor.test(weight,migration,method="spearman")
```

```
## Warning in cor.test.default(weight, migration, method = "spearman"): Cannot
## compute exact p-value with ties
```

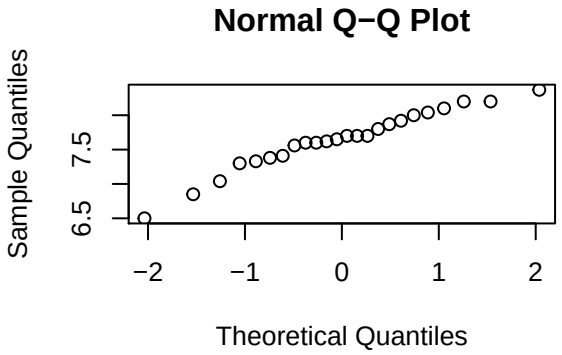
```
##
##  Spearman's rank correlation rho
##
## data:  weight and migration
## S = 6415.1, p-value = 0.02861
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.3506956
```

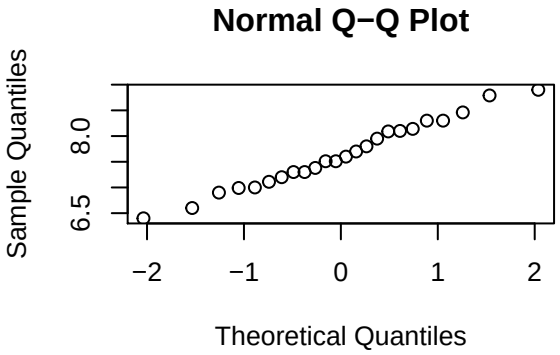
Lastly, weight is correlated with migration having a p-value of 0.02861. Two of the tree values predicted in question one were correlated, bith weight and age. ##Exercise 1 ##Q1

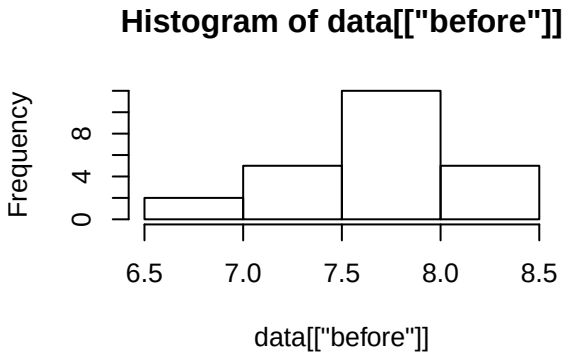


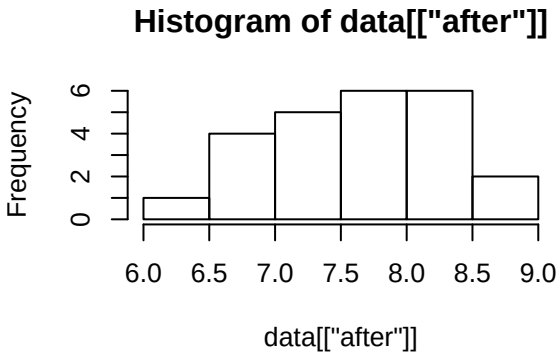


From the boxplots it indicates that the engery actaully makes students worse after 30 mins and the lemonade makes students better.









Data looks like it was drawn from a normal distribution. With the exception of the histogram for "after" but there are only 12 entries per drink type which is a small amount to guage whether this is drawn from a normal distribution

Q2

Using a two paired sample test

```
##
##  Paired t-test
##
## data:  data["before"][filter] and data["after"][filter]
## t = 1.6538, df = 11, p-value = 0.1264
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.05101059  0.35934392
## sample estimates:
## mean of the differences
##               0.1541667
```

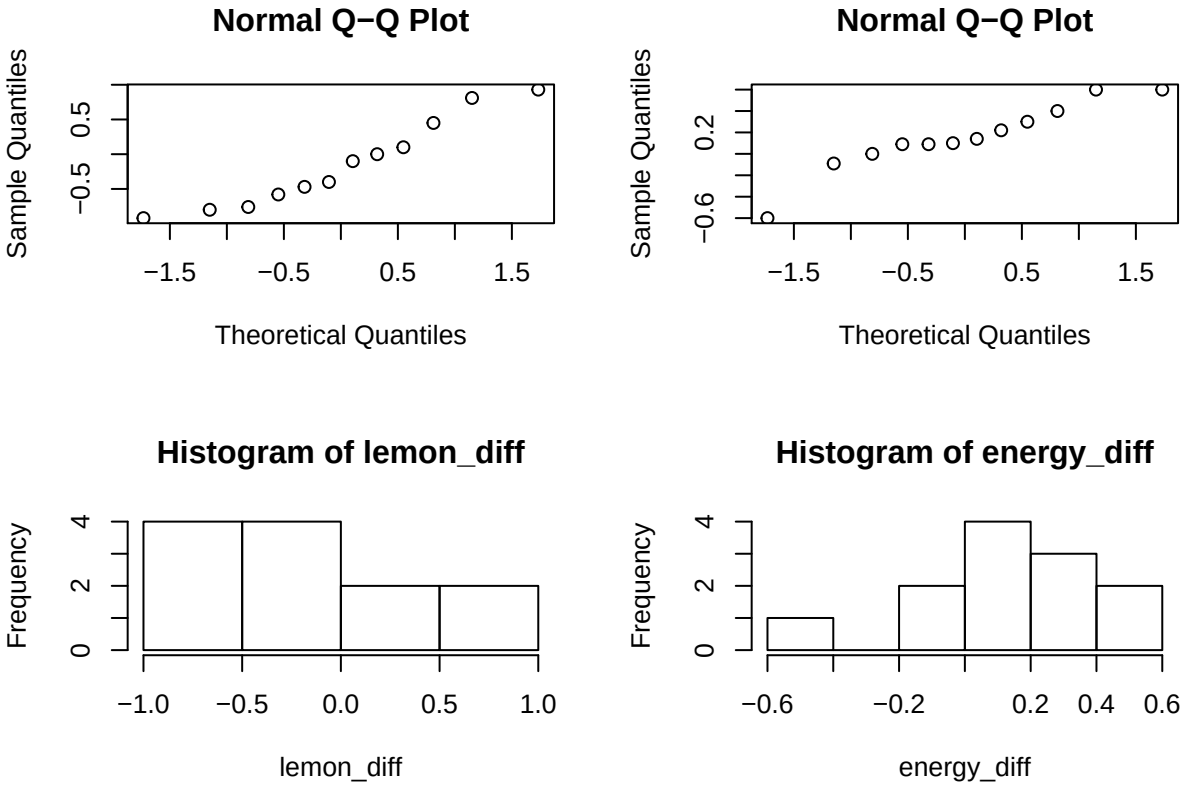
```
##
##  Paired t-test
##
## data:  data["before"][filter] and data["after"][filter]
```

```
## t = -0.80596, df = 11, p-value = 0.4373
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5409781  0.2509781
## sample estimates:
## mean of the differences
##                  -0.145
```

For both drinks the p-values obtained are 0.1264 and 0.4373 for "energy" and "lemo" respectively. These p-values fall above the 0.05 range so therefore we cannot reject the hypothesis nor say that there is increase from the before to the after.

Q3

Using permuation test permutation for independent samples.




```
##
##  Welch Two Sample t-test
##
## data:  lemon_diff and energy_diff
## t = -1.4764, df = 16.509, p-value = 0.1586
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
```

```
##  -0.7276409  0.1293076
## sample estimates:
##  mean of x  mean of y
## -0.1450000  0.1541667
```

With a high p-value of 0.1586 the null hypothesis can not be rejected therfore there in no meaningful difference between both energy and lemonade.

Q4

There is only a small sample size, each drink gets allocated 12 people, it could be the case that one group were faster on average therefore would be faster before and after whereas some people might need more than 30 minutes to recover. Also, perhaps a bigger margin then 30 minutes for the gap between before and after.

Q5

It would have affected the time difference if people were still tired after the first run.

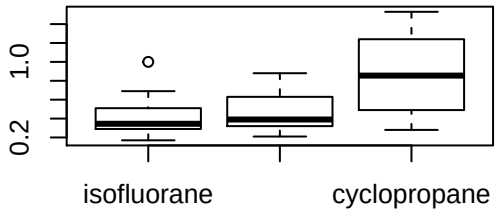
Q6

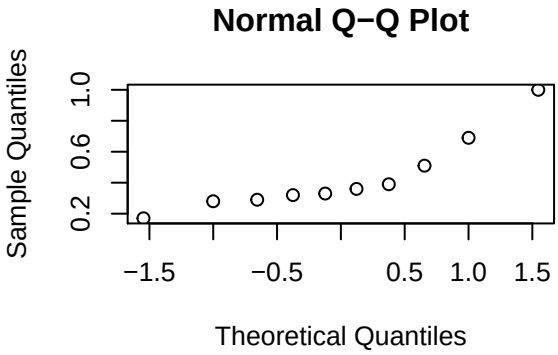
The conclusion is that both the differences for lemonade and energy are drawn from a normal distrubution. The qq plots can be seen in question 3.

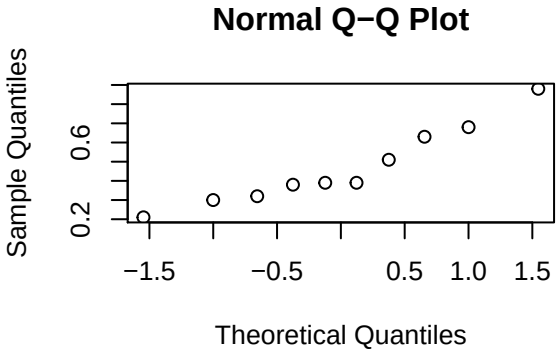
Exercise 3

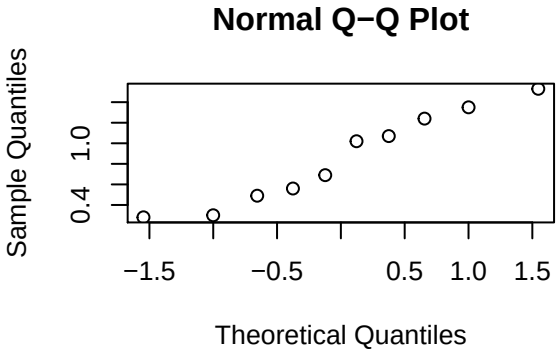
Q1

```
data = read.table("data/dogs.txt", header=TRUE)
par(mfrow=c(2,2))
boxplot(data,data=data)
qqnorm(data[['isofluorane']])
qqnorm(data[['halothane']])
qqnorm(data[['cyclopropane']])
```









Each drug type has 10 examples. All qqplots are close to a normal distribution with the exception of isofluorane which is displayed in top right. It can be presummed that by adding more data that the qqplots would more closely resembles a qqplot of a normal distubution. In this case it is reasonable to assume sample were taken from normal distribution.

Q2

Using anova.

```
treats = data.frame(dog=as.vector(as.matrix(data)),treatment=factor(rep(1:3,each=10)))
attach(treats)
pvcaov=lm(dog~treatment,data=treats)
anova(pvcaov)
```

```
## Analysis of Variance Table
##
## Response: dog
##           Df Sum Sq Mean Sq F value Pr(>F)
## treatment  2 1.0808 0.54040   5.355  0.011 *
## Residuals 27 2.7247 0.10092
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The hypothesis is rejected as the p-value is 0.011 and therefore there is a difference between the

treatments.

```
summary(pvcaov)
```

```
##
## Call:
## lm(formula = dog ~ treatment, data = treats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5730 -0.1608 -0.0790  0.2000  0.6770
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.4340     0.1005   4.320 0.000189 ***
## treatment2    0.0350     0.1421   0.246 0.807266
## treatment3    0.4190     0.1421   2.949 0.006504 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3177 on 27 degrees of freedom
## Multiple R-squared:  0.284,  Adjusted R-squared:  0.231
## F-statistic: 5.355 on 2 and 27 DF,  p-value: 0.011
```

The mean estimation for $\mu_i sofluorane$ is 0.434 and with a p-value of 0.000189. The estimation for $\mu_h alo - \mu_i so$ is 0.035 and with a p-value of 0.807266. The estimation for $\mu_c yclo - \mu_i so$ is 0.419 and with a p-value of 0.006504.

```
confint(pvcaov)
```

```
##                 2.5 %   97.5 %
## (Intercept)  0.227879 0.640121
## treatment2  -0.256499 0.326499
## treatment3   0.127501 0.710499
```

Those are the 95% confidence intervals for $\mu_i sofluorane$, $\mu_h alo - \mu_i so$ and $\mu_c yclo - \mu_i so$, respectively.

Q3

```
kruskal.test(dog,treatment,data=treats)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  dog and treatment
## Kruskal-Wallis chi-squared = 5.6442, df = 2, p-value = 0.05948
```

The p-value is really close to 0.05, but it is slightly above it, so we cannot reject the hypothesis and therefore we cannot assume that the samples come from different populations. This contrasts with the findings we found with anova test.

```
treats = data.frame(dog=as.vector(as.matrix(data)),treatment=factor(rep(1:3,each=10)))
attach(treats)
```

```
## The following objects are masked from treats (pos = 3):
##
##     dog, treatment
```

```
pvcaov=lm(dog~treatment,data=treats)
anova(pvcaov)
```

```
## Analysis of Variance Table
##
## Response: dog
##           Df Sum Sq Mean Sq F value Pr(>F)
## treatment  2 1.0808 0.54040   5.355  0.011 *
## Residuals 27 2.7247 0.10092
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
qqnorm(pvcaov$residuals)
```