

# Assignment 1

Alexia Salomons, Nathan Maxwell Jones, Yauheniya Makarevich, group 71

27 February 2023

## Exercise 1.

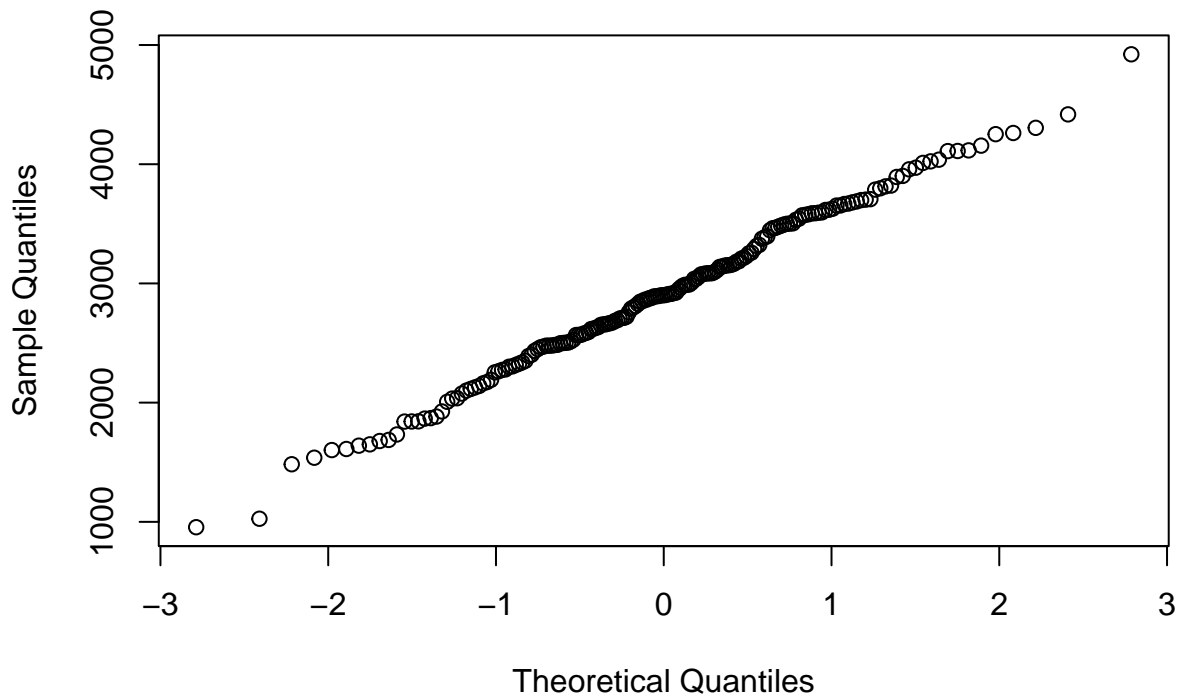
```
birthweight_mean <- mean(birthweight)
print(paste("Mean of the sample: ", birthweight_mean))
```

```
## [1] "Mean of the sample: 2913.29255319149"
```

a) Check normality of the data. Assuming normality (irrespective of your conclusion about normality), construct a bounded 96%-CI for  $\mu$ . Evaluate the sample size needed to provide that the length of the 96%-CI is at most 100. Compute a bootstrap 96%-CI for  $\mu$  and compare it to the above CI.

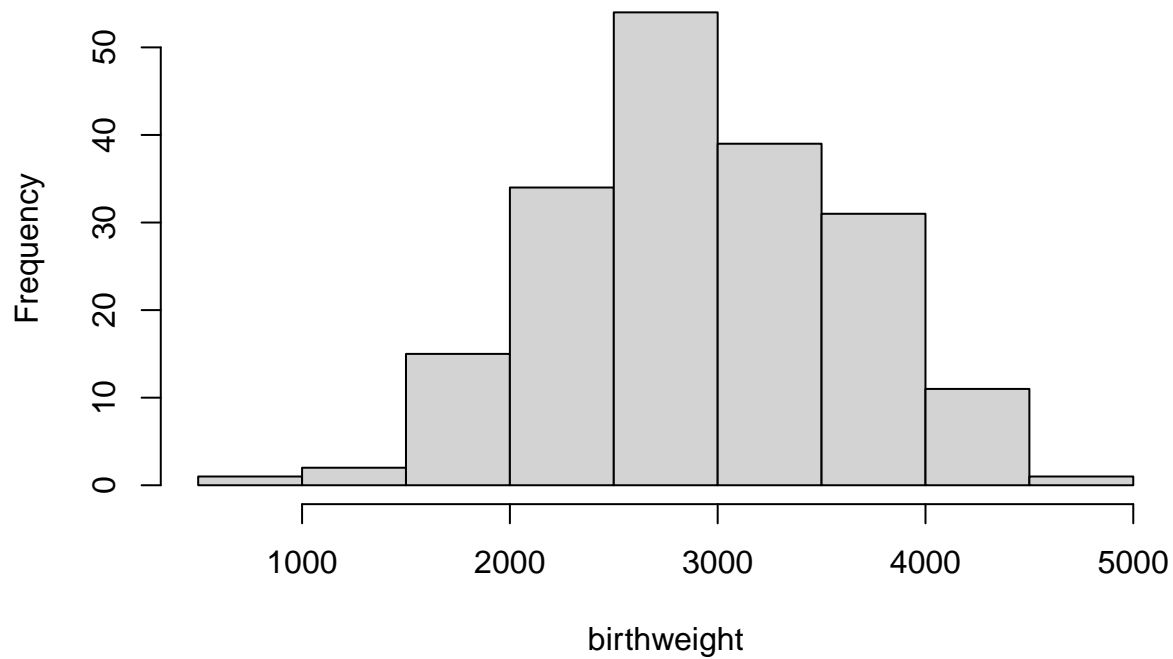
```
qqnorm(y = birthweight)
```

Normal Q-Q Plot



```
hist(birthweight)
```

## Histogram of birthweight



```
# add density line
# lines(density(birthweight), col="blue", lwd=2)
```

```
shapiro.test(birthweight)
```

```
##
## Shapiro-Wilk normality test
##
## data: birthweight
## W = 1, p-value = 0.9
```

```
# H0 - normal distribution, H1 - not normal
```

NORMAL DISTRIBUTION!

Let's go for CI-96%:

```
t.test(birthweight, conf.level = 0.96)
```

```
##
## One Sample t-test
##
## data: birthweight
## t = 57, df = 187, p-value <2e-16
## alternative hypothesis: true mean is not equal to 0
## 96 percent confidence interval:
## 2808 3019
## sample estimates:
## mean of x
```

```
##      2913
B <- 1000
alpha <- 0.04
T_star <- numeric(B)

for(i in 1:B) {
  X_star <- sample(birthweight, replace = TRUE)
  T_star[i] <- mean(X_star)
}

T_star_q2 <- quantile(T_star, alpha/2)
T_star_q98 <- quantile(T_star, 1 - alpha/2)

c(2*birthweight_mean - T_star_q98, 2*birthweight_mean - T_star_q2)

## 98% 2%
## 2808 3015

sum(T_star<T_star_q2)

## [1] 20
```

b) An expert claims that the mean birthweight is bigger than 2800 gram. Verify this claim by using a relevant t-test, explain the meaning of the CI in the R-output for this test. Also propose and perform a suitable sign tests for this problem.

```
t.test(birthweight, alternative = "greater", mu=2800)

##
## One Sample t-test
##
## data: birthweight
## t = 2, df = 187, p-value = 0.01
## alternative hypothesis: true mean is greater than 2800
## 95 percent confidence interval:
## 2829 Inf
## sample estimates:
## mean of x
##      2913
```

We reject  $H_0(p=0.01337)$ , so  $H_1$  is true and mean of the sample is bigger than 2800. CI is infinite on right side, since the test is one-sided.

Binom test

$H_0$ : mean  $\leq$  2800,  $H_1$ : mean  $>$  2800.

```
greater_weight <- as.integer(birthweight > 2800)
binom.test(sum(greater_weight), length(greater_weight), p=0.5, alt="g")

##
## Exact binomial test
```

```
##
## data:  sum(greater_weight) and length(greater_weight)
## number of successes = 107, number of trials = 188, p-value = 0.03
## alternative hypothesis: true probability of success is greater than 0.5
## 95 percent confidence interval:
##  0.507 1.000
## sample estimates:
## probability of success
##                0.569
```

We reject  $H_0(p=0.03868)$ , so  $H_1$  is true and mean of the sample is bigger than 2800.

Both test confirmed the hypothesis that mean of the sample is bigger than 2800.

c) Propose a way to compute the powers of the t-test and sing test from b) at some  $\mu > 2800$ , comment.

d) Let  $p$  be the probability that birthweight of a newborn baby is less than 2600 gram. Using asymptotic normality, the expert computed the left end  $\hat{p} = 0.25$  of the confidence interval  $[\hat{p}_l, \hat{p}_r]$  for  $p$ . Recover the whole confidence interval and its confidence level.

e) The expert also reports that there were 34 male and 28 female babies among 62 who weighted less than 2600 gram, and 61 male and 65 female babies among the remaining 126 babies. The expert claims that the mean weight is different for male and female babies. Verify this claim by an appropriate test.

success:  $w > 2600$

```
prop.test(c(61, 65), c(95, 93))
```

```
##
## 2-sample test for equality of proportions with continuity correction
##
## data:  c(61, 65) out of c(95, 93)
## X-squared = 0.5, df = 1, p-value = 0.5
## alternative hypothesis: two.sided
## 95 percent confidence interval:
## -0.2016  0.0879
## sample estimates:
## prop 1 prop 2
##  0.642  0.699
```

We accept  $H_0: p_1-p_2=0$ , where  $p_1, p_2$  - proportions of the success in population.

## Exercise 2

A study tested whether cholesterol was reduced after using a certain brand of margarine as part of a low fat low cholesterol diet. The data set cholesterol.txt contains information on 18 people using margarine to reduce cholesterol: columns Before and After8weeks contain the cholesterol level (mmol/L) respectively before the diet and after 8 weeks on the diet.

```
df <- as.data.frame(read.table("data/cholesterol.txt", header=TRUE))
head(df)
```

```
## Before After8weeks
## 1 6.42 5.75
## 2 6.76 6.13
## 3 6.56 5.71
## 4 4.80 4.15
## 5 8.43 7.67
## 6 7.49 7.05
```

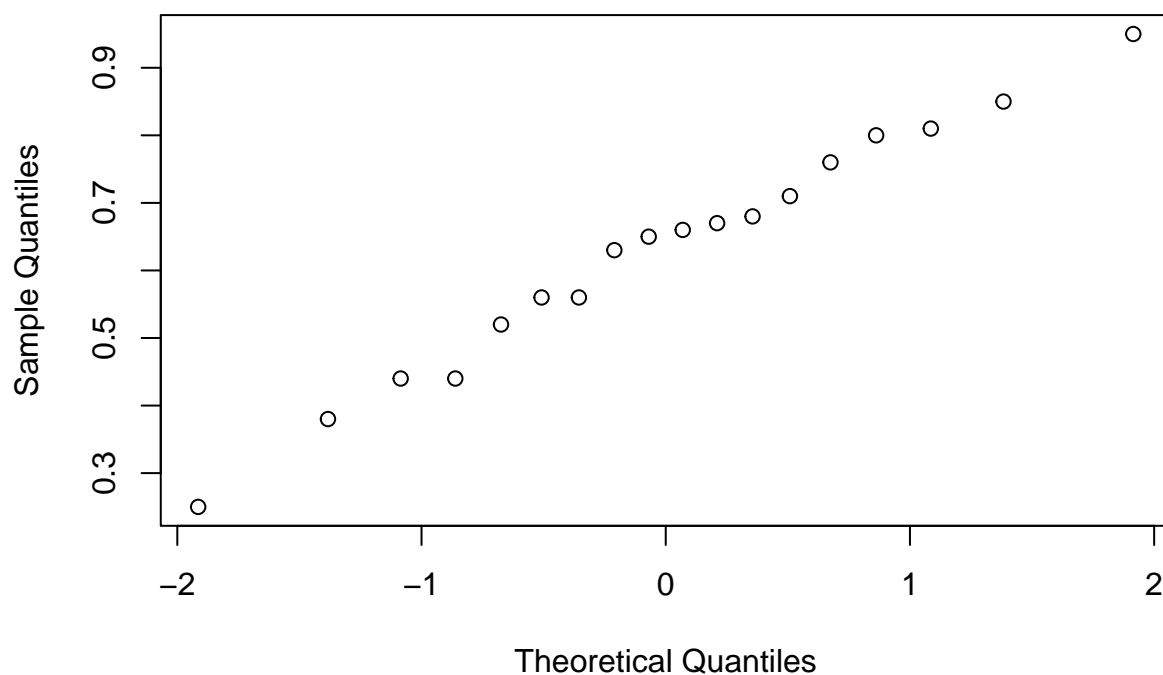
a) Make some relevant plots of this data set, comment on normality. Are there any inconsistencies in the data? Investigate whether the columns Before and After8weeks are correlated.

```
diffs <- df[, 1] - df[, 2]
diffs
```

```
## [1] 0.67 0.63 0.85 0.65 0.76 0.44 0.95 0.38 0.44 0.25 0.81 0.80 0.66 0.71 0.52
## [16] 0.56 0.68 0.56
```

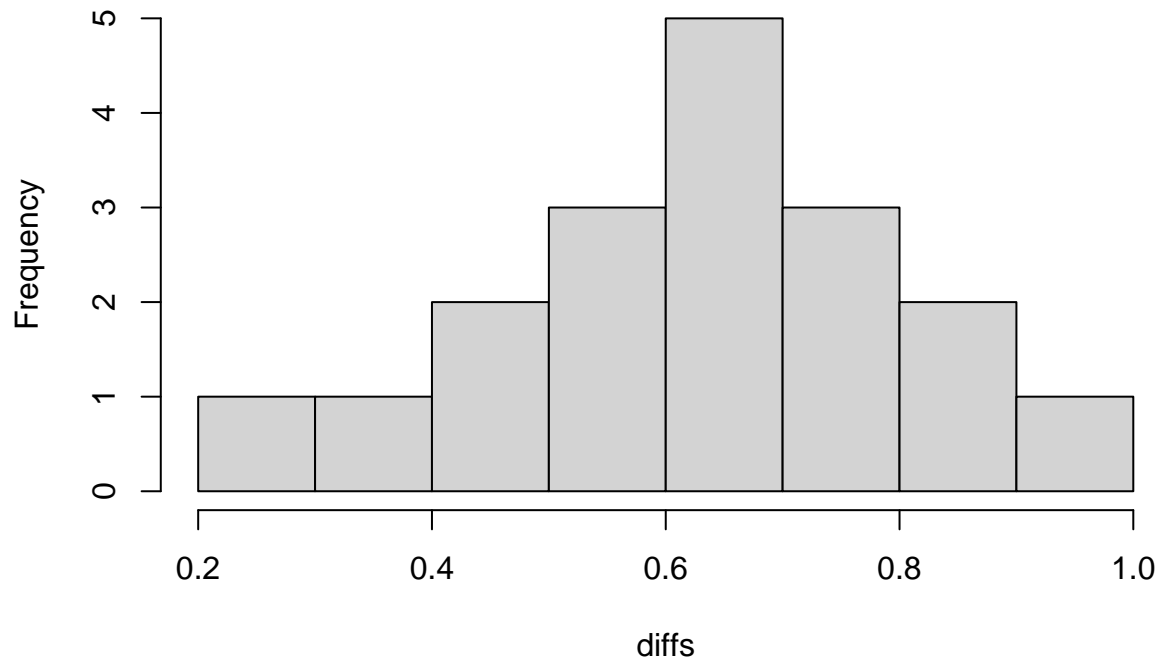
```
qqnorm(diffs)
```

**Normal Q-Q Plot**



```
hist(diffs)
```

## Histogram of diffs



```
shapiro.test(diffs)
```

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

Differences are normally distributed.

```
shapiro.test(df[, 1])
```

```
##  
## Shapiro-Wilk normality test  
##  
## data:  df[, 1]  
## W = 1, p-value = 1
```

```
shapiro.test(df[, 2])
```

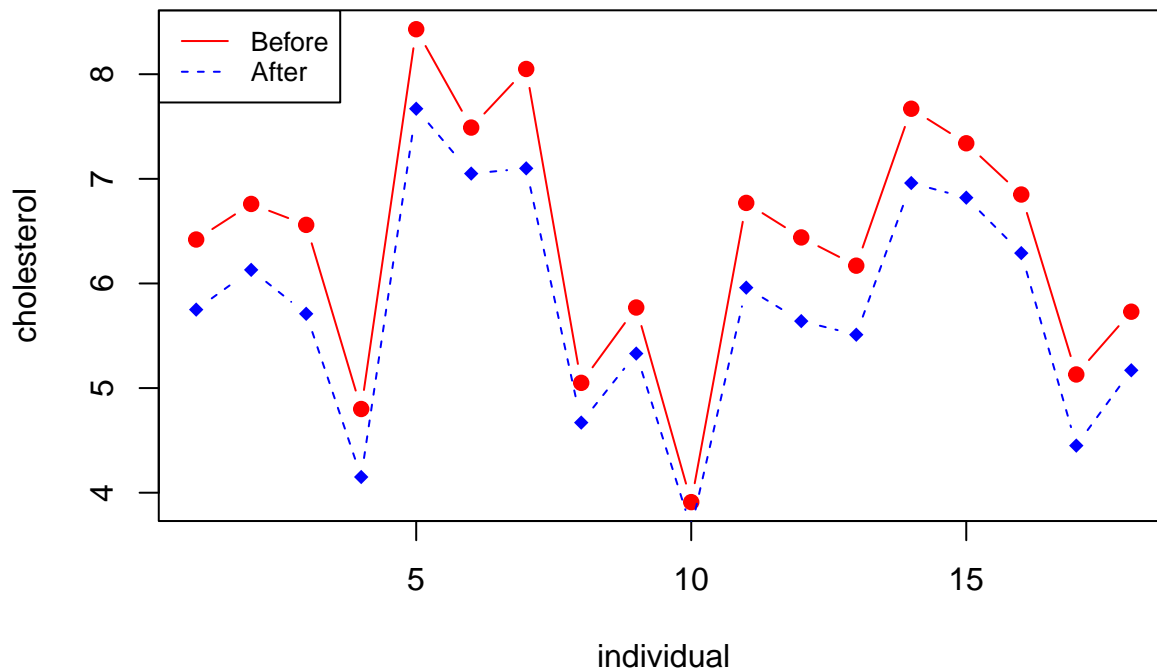
```
##  
## Shapiro-Wilk normality test  
##  
## data:  df[, 2]  
## W = 1, p-value = 0.9
```

```
cor.test(df[, 1], df[, 2], method="pearson")
```

```
##
```

```
## Pearson's product-moment correlation
##
## data: df[, 1] and df[, 2]
## t = 29, df = 16, p-value = 2e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.975 0.997
## sample estimates:
## cor
## 0.991

# Create a first line
plot(1:length(df[, 1]), df[, 1], type = "b", pch=19, col = "red", xlab = "individual", ylab = "cholesterol")
lines(1:length(df[, 2]), df[, 2], pch=18, col = "blue", type = "b", lty=2)
legend("topleft", legend=c("Before", "After"), col=c("red", "blue"), lty = 1:2, cex=0.8)
```



b) Apply two relevant tests (cf. Lectures 2, 3) to verify whether the diet with low fat margarine has an effect (argue whether the data are paired or not). Is a permutation test applicable?

Data is paired since it is two different measurements of the same person and two samples are correlated. Relevant test:

1. T-test paired test

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

```
##
## Paired t-test
##
## data: df[, 1] and df[, 2]
## t = 15, df = 17, p-value = 3e-11
```

```
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  0.540 0.718
## sample estimates:
## mean difference
##           0.629
```

-> based on p-value we reject null-hypothesis that samples have the same mean. there is a difference between these two samples.

2. Permutation test - it is applicable because we are only testing for a difference between mean, not how they relate to each other.

```
diff_mean <- function(x, y) {
  return(mean(x-y))
}

# original dataframe
stats <- diff_mean(df[, 1], df[, 2])
stats

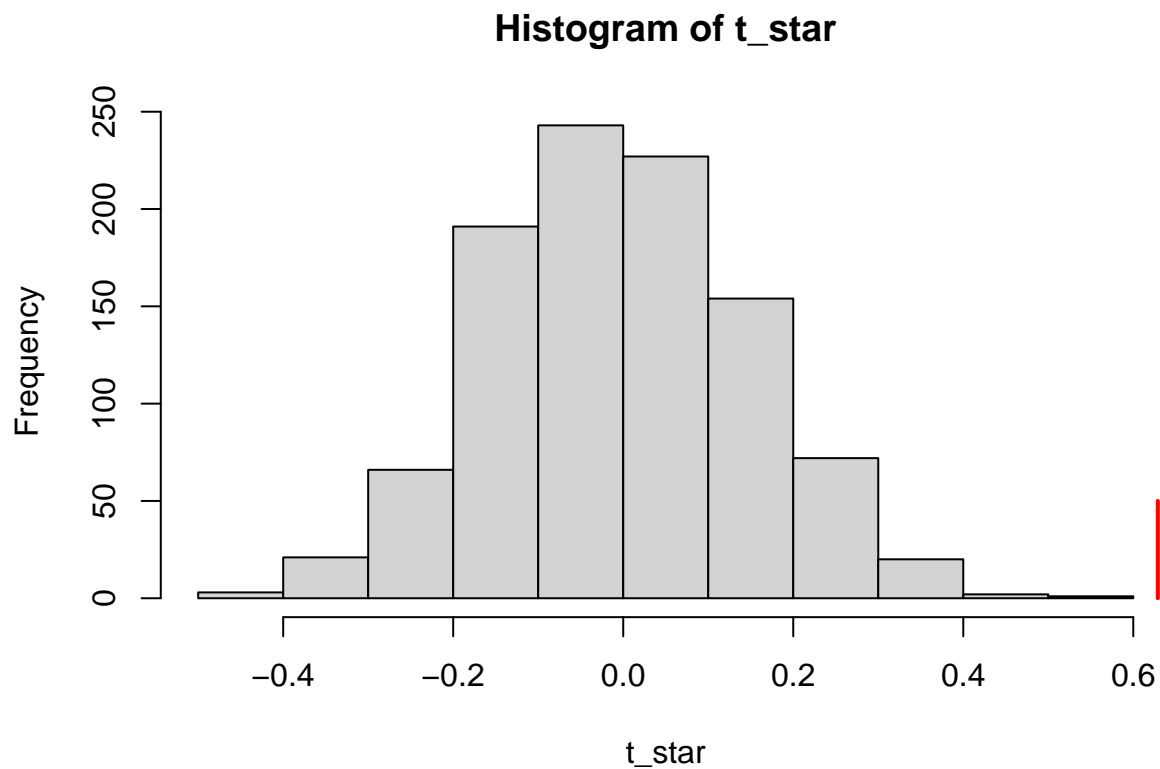
## [1] 0.629

B <- 1000
t_star <- numeric(B)

for (i in 1:B) {
  diff_star <- t(apply(cbind(df[, 1], df[, 2]), 1, sample))
  t_star[i] <- diff_mean(diff_star[, 1], diff_star[, 2])
}

hist(t_star)
# plot(rep(stats, 2), c(0, 50), col="red", lwd=2)
lines(rep(stats, 2), c(0, 50), col="red", lwd=2)
```





```
# calculating p-value
pl <- sum(t_star < stats) / B
pr <- sum(t_star > stats) / B

p <- 2*min(pl, pr)
p
```

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

c)  $\mu = (\theta + 3)/2, \theta > 3$

```
mu <- mean(df[, 2])
mu # bar{x}
```

```
## [1] 5.78
```

```
s <- sd(df[, 2])
s
```

```
## [1] 1.1
```

```
theta_hat <- 2*mu - 3
theta_hat
```

```
## [1] 8.56
```

```
alpha <- 0.05
```

```
n <- length(df[, 2])
t_alpha <- qt(1 - alpha/2, df=n)
```

```
t_alpha
```

```
## [1] 2.1
```

```
theta_l <- theta_hat - t_alpha*s/sqrt(n)
theta_r <- theta_hat + t_alpha*s/sqrt(n)
c("[", theta_l, ",", theta_r, "]")
```

```
## [1] "[" "8.01211959907622" ", " "9.10343595647934"
## [5] "]"
```

We can improve the CI by having more individuals in the samples.

d)

```
t <- max(df[, 2])
t
```

```
## [1] 7.67
```

```
n <- length(df[, 2])
n
```

```
## [1] 18
```

```
for (theta in 3:12) {
  B <- 1000
  t_star <- numeric(B)

  for (i in 1:B) {
    x_star <- runif(n, min = 3, max = theta)
    t_star[i] <- max(x_star)
  }

  pl <- sum(t_star < t)/B
  pr <- sum(t_star > t)/B

  p <- 2* min(pl, pr)
  print(paste("Theta =", theta, ", ", p))
}
```

```
## [1] "Theta = 3 , 0"
## [1] "Theta = 4 , 0"
## [1] "Theta = 5 , 0"
## [1] "Theta = 6 , 0"
## [1] "Theta = 7 , 0"
## [1] "Theta = 8 , 0.66"
## [1] "Theta = 9 , 0.014"
## [1] "Theta = 10 , 0.002"
## [1] "Theta = 11 , 0"
## [1] "Theta = 12 , 0"
```

e) Medium and proportion tests

```
less_chol <- as.integer(df[, 2] < 6)
binom.test(sum(less_chol), length(less_chol), p=0.5, alt="g")

##
## Exact binomial test
##
## data: sum(less_chol) and length(less_chol)
## number of successes = 11, number of trials = 18, p-value = 0.2
## alternative hypothesis: true probability of success is greater than 0.5
## 95 percent confidence interval:
## 0.392 1.000
## sample estimates:
## probability of success
## 0.611

less_chol <- as.integer(df[, 2] < 4.5)
binom.test(sum(less_chol), length(less_chol), p=0.25, alt="l")

##
## Exact binomial test
##
## data: sum(less_chol) and length(less_chol)
## number of successes = 3, number of trials = 18, p-value = 0.3
## alternative hypothesis: true probability of success is less than 0.25
## 95 percent confidence interval:
## 0.000 0.377
## sample estimates:
## probability of success
## 0.167
```

### Exercise 3

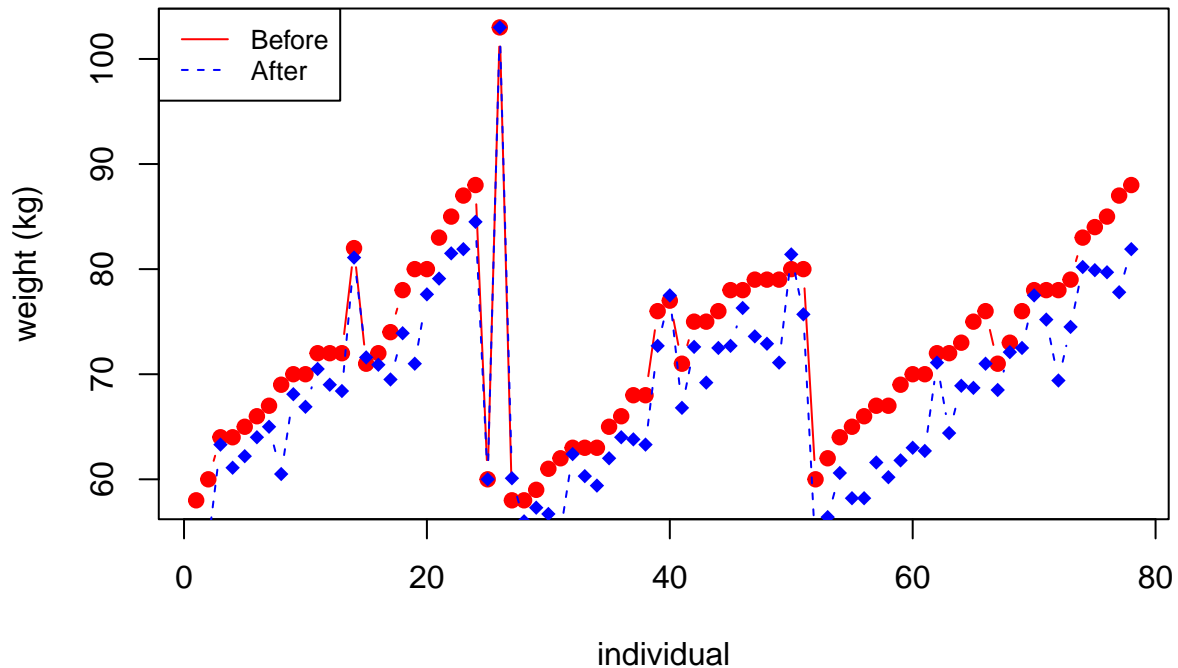
```
df <- as.data.frame(read.table("data/diet.txt", header=TRUE))
df["weight.lost"] <- df["preweight"] - df["weight6weeks"]
head(df)

## person gender age height preweight diet weight6weeks weight.lost
## 1 1 0 22 159 58 1 54.2 3.8
## 2 2 0 46 192 60 1 54.0 6.0
## 3 3 0 55 170 64 1 63.3 0.7
## 4 4 0 33 171 64 1 61.1 2.9
## 5 5 0 50 170 65 1 62.2 2.8
## 6 6 0 50 201 66 1 64.0 2.0
```

a)

```
# Create a first line
plot(1:length(df[,5]), df[,5], type = "b", pch=19, col = "red", xlab = "individual", ylab = "w
```

```
lines(1:length(df[,7]), df[,7], pch=18, col = "blue", type = "b", lty=2)
legend("topleft", legend=c("Before", "After"), col=c("red", "blue"), lty = 1:2, cex=0.8)
```



```
t.test(df[,5], df[,7], paired=TRUE)
```

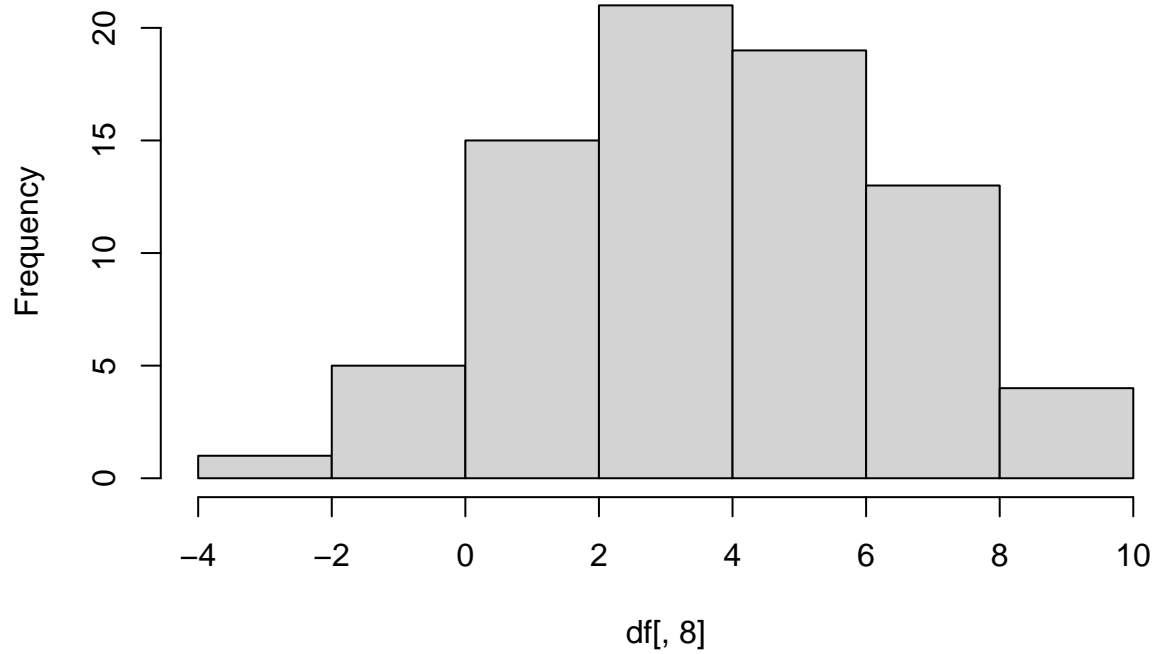
```
##
## Paired t-test
##
## data: df[, 5] and df[, 7]
## t = 13, df = 77, p-value <2e-16
## alternative hypothesis: true mean difference is not equal to 0
## 95 percent confidence interval:
##  3.27 4.42
## sample estimates:
## mean difference
##          3.84
```

Therefore diet does have an effect on weight loss.

Now check assumptions (normality)

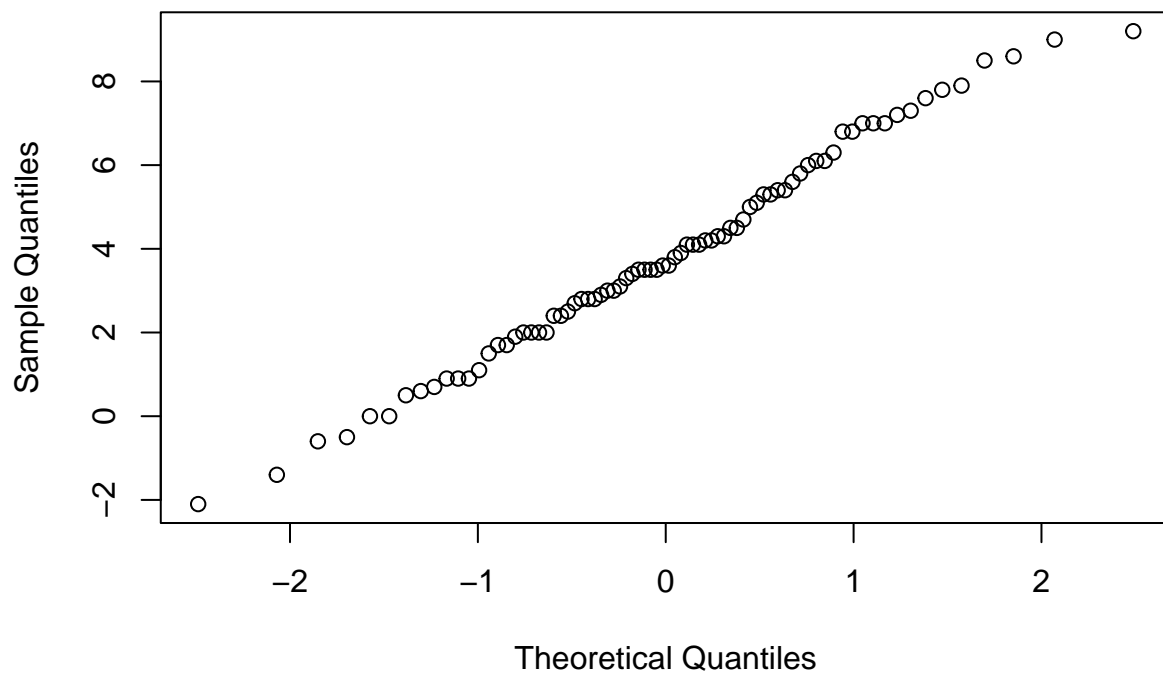
```
hist(df[,8])
```

**Histogram of df[, 8]**



```
qqnorm(df[, 8])
```

**Normal Q-Q Plot**



```
shapiro.test(df[, 8])
```

```
##
```

```
## Shapiro-Wilk normality test
```

```
##
```

```
## data: df[, 8]
```

```
## W = 1, p-value = 0.8
```

```
Data is normal :)
```

```
b) Fromat data
```

```
df$diet <- as.factor(df$diet)
```

```
dietaov=lm(weight.lost~diet,data=df)
```

```
anova(dietaov)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: weight.lost
```

```
##           Df Sum Sq Mean Sq F value Pr(>F)
```

```
## diet        2      71    35.5     6.2 0.0032 **
```

```
## Residuals  75     430      5.7
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(dietaov)
```

```
##
```

```
## Call:
```

```
## lm(formula = weight.lost ~ diet, data = df)
```

```
##
```

```
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
```

```
## -5.126 -1.381  0.176  1.652  5.700
```

```
##
```

```
## Coefficients:
```

```
##           Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)   3.300      0.489    6.75 2.7e-09 ***
```

```
## diet2        -0.274      0.672   -0.41  0.6845
```

```
## diet3         1.848      0.672    2.75  0.0075 **
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
```

```
## Residual standard error: 2.39 on 75 degrees of freedom
```

```
## Multiple R-squared:  0.142, Adjusted R-squared:  0.119
```

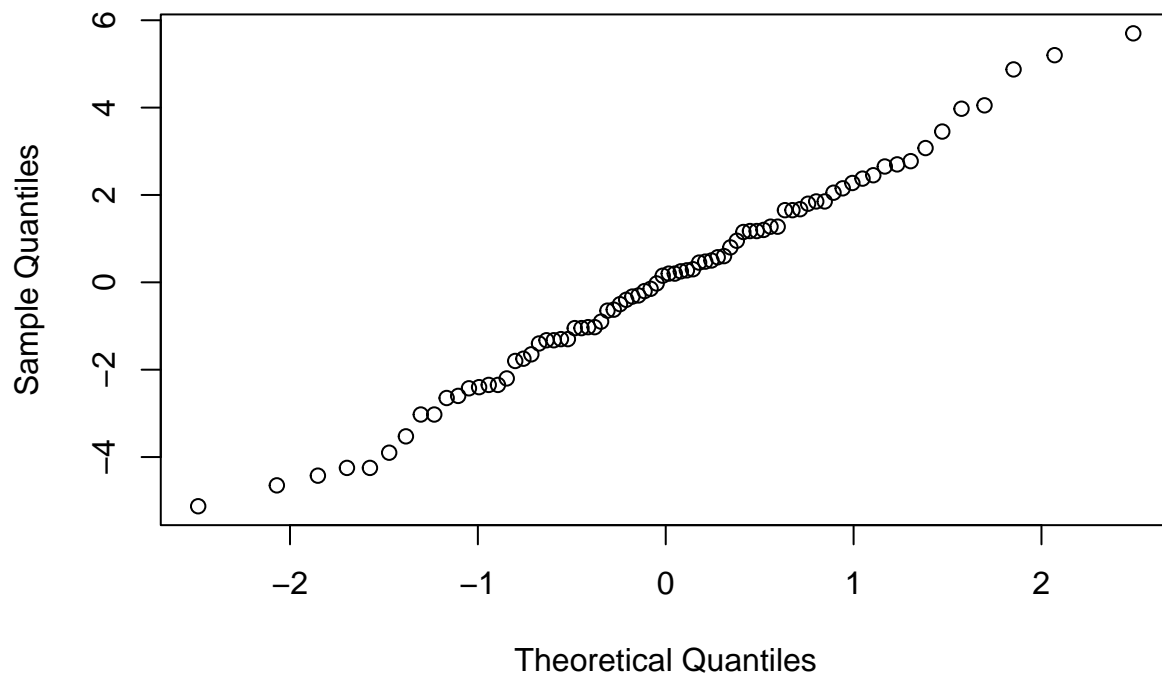
```
## F-statistic:  6.2 on 2 and 75 DF,  p-value: 0.00323
```

```
The best diet is number 3.
```

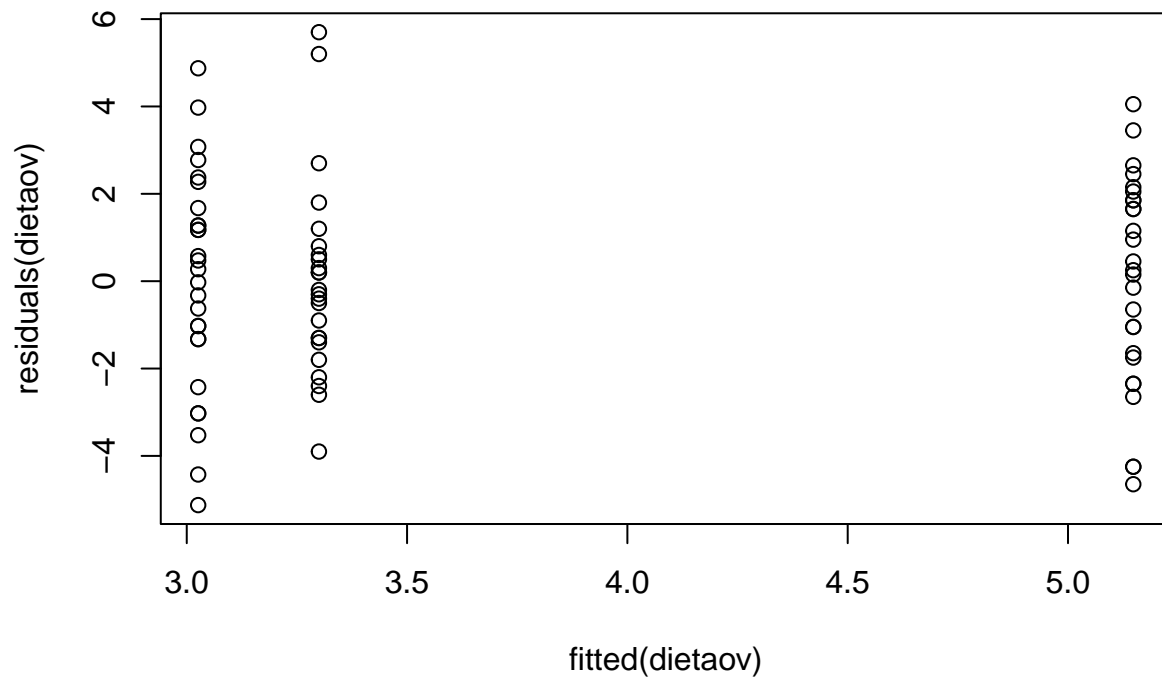
```
# Checking assumptions
```

```
qqnorm(residuals(dietaov))
```

## Normal Q-Q Plot



```
plot(fitted(dietaov), residuals(dietaov))
```



```
kruskal.test(df$weight.lost, df$diet)
```

```
##  
##  Kruskal-Wallis rank sum test  
##  
## data:  df$weight.lost and df$diet
```

```
## Kruskal-Wallis chi-squared = 10, df = 2, p-value = 0.005
```

This supports the ANOVA result.

c)

```
df$gender <- as.factor(df$gender)
dietgenderaov <- lm(weight.lost~gender*diet,data=df)
anova(dietgenderaov)

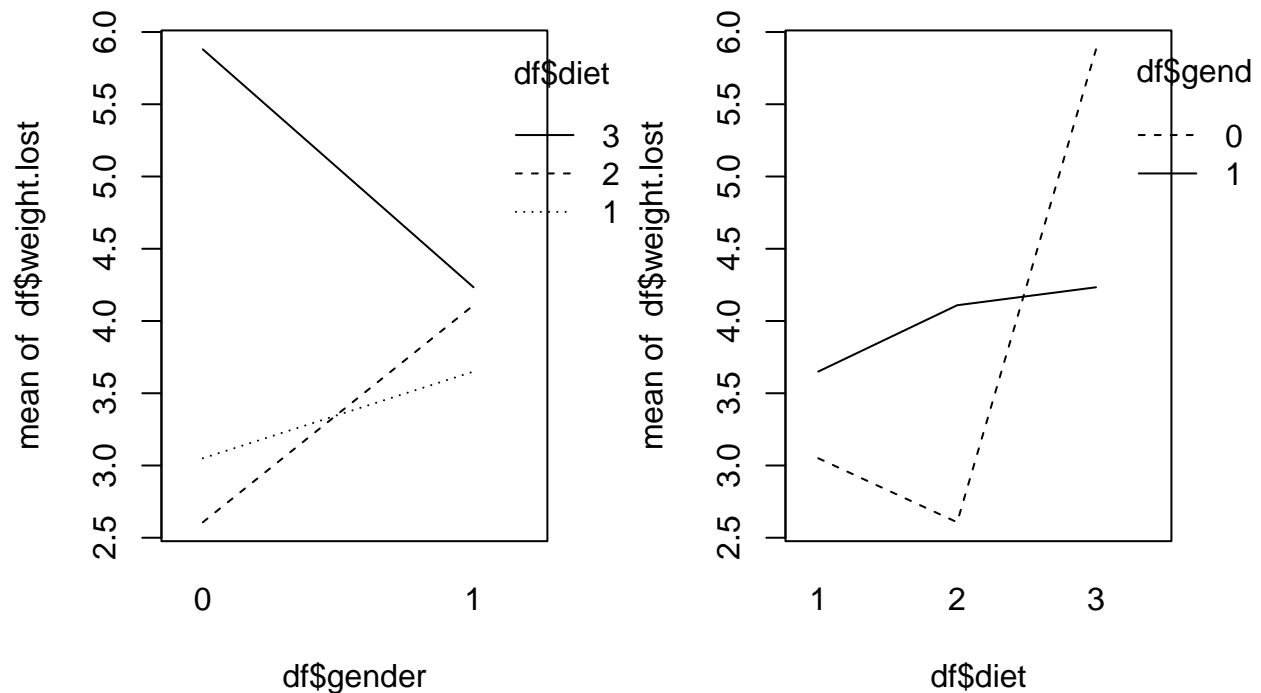
## Analysis of Variance Table
##
## Response: weight.lost
##           Df Sum Sq Mean Sq F value Pr(>F)
## gender      1      0    0.28    0.05 0.8206
## diet        2     60   30.21    5.62 0.0055 **
## gender:diet  2     34   16.95    3.15 0.0488 *
## Residuals   70    376    5.38
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

summary(dietgenderaov)

##
## Call:
## lm(formula = weight.lost ~ gender * diet, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.51  -1.30   0.07   1.22   5.45
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.050      0.620   4.92 5.5e-06 ***
## gender1          0.600      0.960   0.62  0.5340
## diet2          -0.443      0.876  -0.51  0.6149
## diet3           2.830      0.862   3.28  0.0016 **
## gender1:diet2    0.902      1.340   0.67  0.5030
## gender1:diet3   -2.247      1.315  -1.71  0.0919 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.32 on 70 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.201, Adjusted R-squared:  0.144
## F-statistic: 3.52 on 5 and 70 DF, p-value: 0.00677

par(mfrow=c(1, 2))
interaction.plot(df$gender, df$diet, df$weight.lost)
interaction.plot(df$diet, df$gender, df$weight.lost)
```





Assumption: diet depends on gender, we can see that effect of diet varies inside women.

```
genderaov <- lm(weight.lost~gender,data=df)
anova(genderaov)
```

```
## Analysis of Variance Table
##
## Response: weight.lost
##          Df Sum Sq Mean Sq F value Pr(>F)
## gender    1      0    0.28    0.04  0.83
## Residuals 74    471    6.36
```

```
summary(genderaov)
```

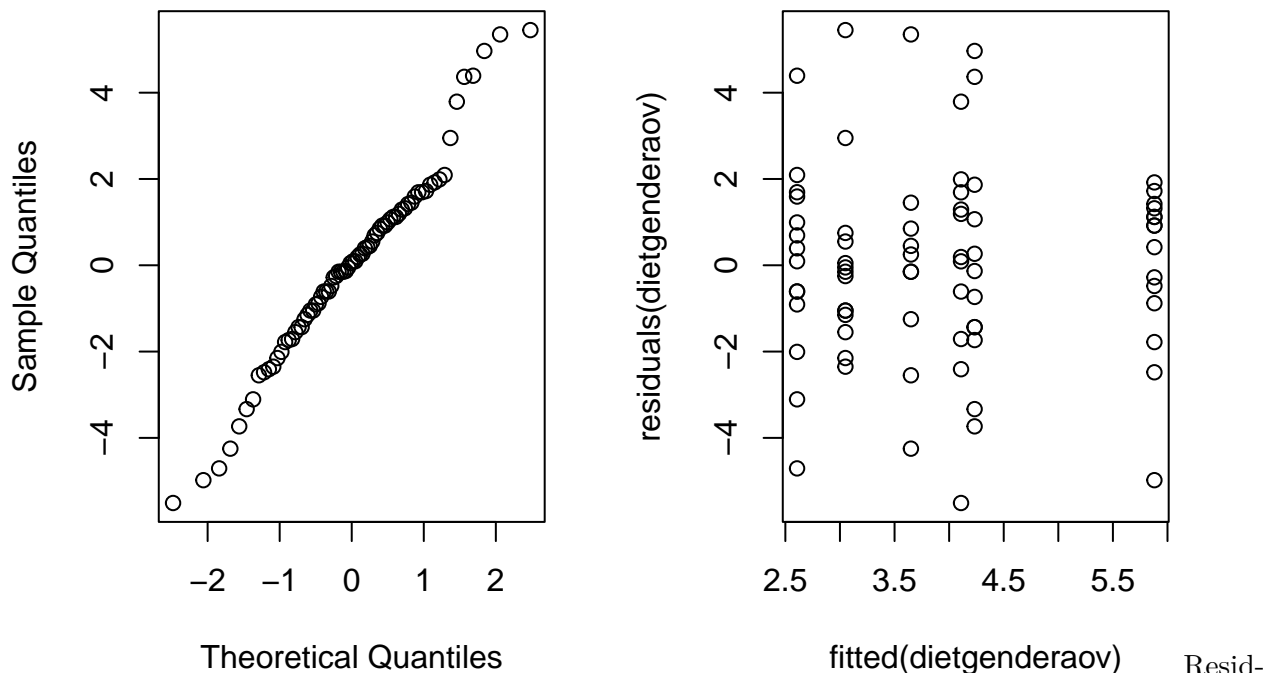
```
##
## Call:
## lm(formula = weight.lost ~ gender, data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.993 -1.685 -0.204  1.726  5.185
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.893     0.385   10.12 1.3e-15 ***
## gender1         0.122     0.584    0.21  0.83
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 2.52 on 74 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.000591, Adjusted R-squared: -0.0129
## F-statistic: 0.0438 on 1 and 74 DF, p-value: 0.835
```

```
# Checking assumptions
```

```
par(mfrow=c(1, 2))
qqnorm(residuals(dietgenderaov))
plot(fitted(dietgenderaov), residuals(dietgenderaov))
```

## Normal Q-Q Plot



Residuals do not look like something that is normally distributed, which violates our assumptions about normality and makes

e) We prefer b) as c) looks irrelevant for the weight loss.

```
print(paste('Diet 1:', dietaov$coefficients[1]))
```

```
## [1] "Diet 1: 3.3"
```

```
print(paste('Diet 2:', dietaov$coefficients[1] + dietaov$coefficients[2]))
```

```
## [1] "Diet 2: 3.02592592592593"
```

```
print(paste('Diet 3:', dietaov$coefficients[1] + dietaov$coefficients[3]))
```

```
## [1] "Diet 3: 5.14814814814815"
```

## Exercise 4

a)

```

B <- 6; P <- 4; T <- 3

process <- c()
for (i in 1:B) {
  block <- c()
  for (tr in 1:T) {
    block <- cbind(block, as.numeric(sample(1:P) > 2))
  }
  process <- rbind(process, block)
}

process <- t(process)
rownames(process) <- c("N", "P", "K")
colnames(process) <- paste0(rep(1:6, each=4), paste0(".", rep(1:4, 6)))
process

```

```

##   1.1 1.2 1.3 1.4 2.1 2.2 2.3 2.4 3.1 3.2 3.3 3.4 4.1 4.2 4.3 4.4 5.1 5.2 5.3
## N   0   1   0   1   0   1   0   1   1   0   0   1   1   0   0   1   1   0   0
## P   0   1   1   0   0   1   0   1   0   1   1   0   0   1   0   1   1   0   0
## K   0   0   1   1   1   1   0   0   1   1   0   0   1   1   0   0   1   0   0
##   5.4 6.1 6.2 6.3 6.4
## N   1   0   0   1   1
## P   1   0   1   1   0
## K   1   1   0   1   0

```

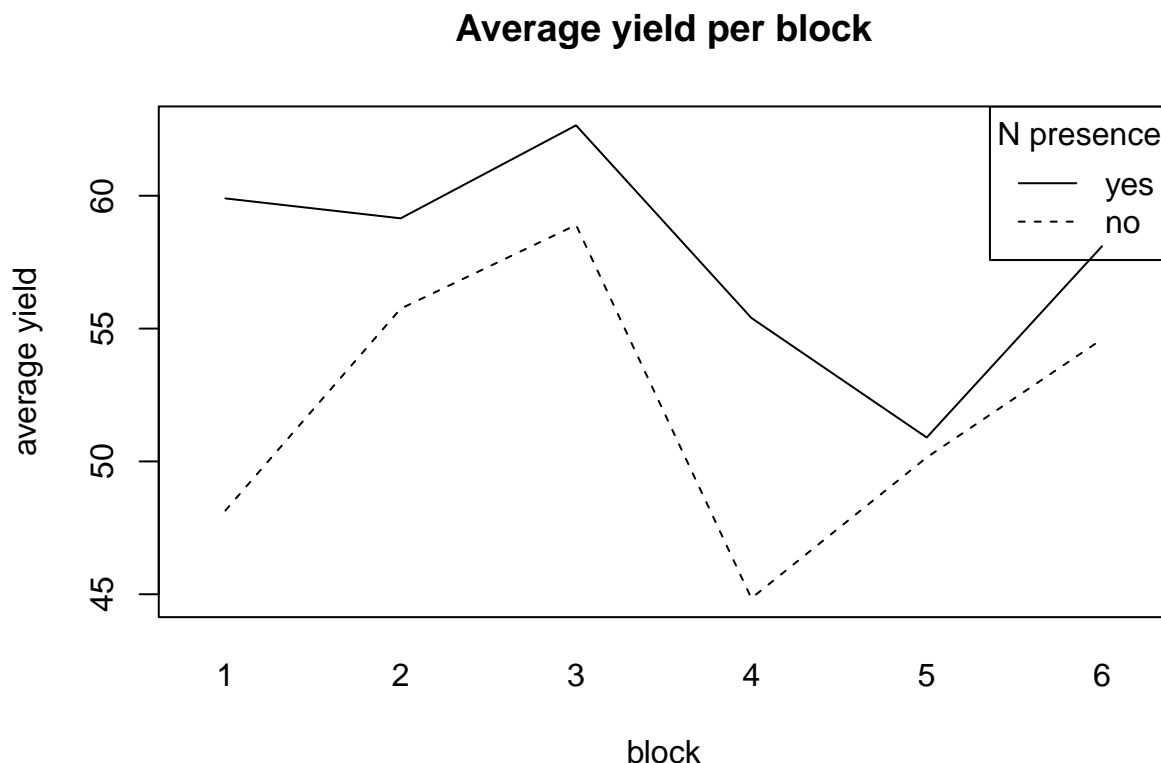
In the table rows represent every soil additive and columns represent 6 blocks, each with 4 plots (first number - block, second number - plot). As you can see, every additive appears twice in each block.

b)

```

interaction.plot(npk$block, npk$N, npk$yield, ylab='average yield', xlab='block', main='Average yield vs N presence',
legend("topright", c("yes", "no"), title="N presence", lwd=1, lty=c(1, 2))

```



We have reason to believe that *block* may affect *yield*. This could happen because of slightly different environmental conditions: sun exposure, soil composition, etc. The plot support this idea since it appears that average yield varies depending on the block when N is both present and absent.

c)

```
n_block_lm <- lm(yield~block*N, data=npk)
anova(n_block_lm)
```

```
## Analysis of Variance Table
##
## Response: yield
##           Df Sum Sq Mean Sq F value Pr(>F)
## block      5   343    68.7    3.36  0.04 *
## N           1   189   189.3    9.26  0.01 *
## block:N     5    99    19.7    0.96  0.48
## Residuals 12   245    20.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It is seen from the results that *N* and *block* are significant, but their interaction is not. Let's check summary for every level to see how each block individually affect the *yield*.

```
summary(n_block_lm)
```

```
##
## Call:
## lm(formula = yield ~ block * N, data = npk)
##
```

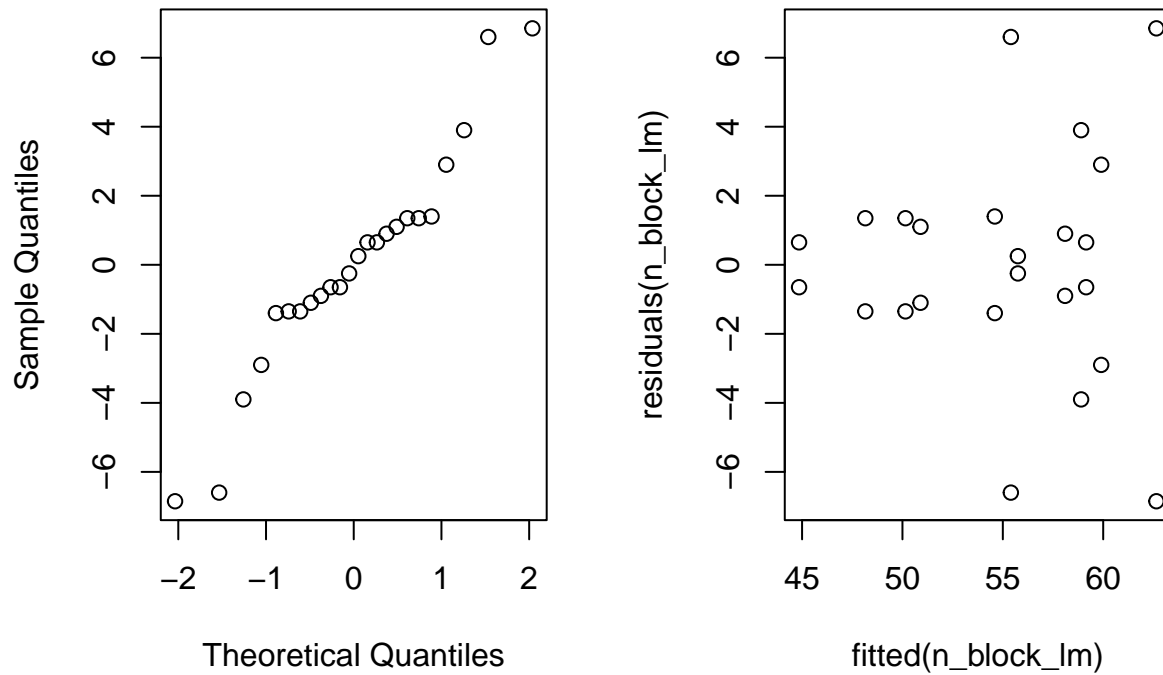
```

## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.85  -1.35    0.00    1.35    6.85
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    48.15      3.20   15.06 3.7e-09 ***
## block2         7.60      4.52    1.68   0.119
## block3        10.75      4.52    2.38   0.035 *
## block4        -3.30      4.52   -0.73   0.479
## block5         2.00      4.52    0.44   0.666
## block6         6.45      4.52    1.43   0.179
## N1            11.75      4.52    2.60   0.023 *
## block2:N1      -8.35      6.39   -1.31   0.216
## block3:N1      -8.00      6.39   -1.25   0.235
## block4:N1      -1.20      6.39   -0.19   0.854
## block5:N1     -11.00      6.39   -1.72   0.111
## block6:N1      -8.25      6.39   -1.29   0.221
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.52 on 12 degrees of freedom
## Multiple R-squared:  0.72,    Adjusted R-squared:  0.464
## F-statistic: 2.81 on 11 and 12 DF,  p-value: 0.0449

```

We can see that the difference between blocks is not significant except for block 3. Additionally, it seen that presence of *N* in the soil makes significant difference to the *yield*.

## Normal Q-Q Plot



Changes in blocks are not significant so we can consider that block variable doesn't affect

Friedman test: - we have two observation for each combination - we can't use it

d)

```
npklm1 <- lm(yield~P + K + block*N, data=npk)
npklm2 <- lm(yield~N + K + block*P, data=npk)
npklm3 <- lm(yield~N + P + block*K, data=npk)
npklm4 <- lm(yield~block + N + P + K, data=npk)
npklm5 <- lm(yield~N + P + K, data=npk)
```

```
print('Y ~ P + K + block*N')
```

```
## [1] "Y ~ P + K + block*N"
```

```
anova(npklm1)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

```
##          Df Sum Sq Mean Sq F value Pr(>F)
## P          1      8      8.4    0.59 0.4590
## K          1     95     95.2    6.72 0.0268 *
## block      5    343     68.7    4.85 0.0164 *
## N          1    189    189.3   13.36 0.0044 **
## block:N     5     99     19.7    1.39 0.3066
## Residuals 10    142     14.2
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print('Y ~ N + K + block*P')
```

```
## [1] "Y ~ N + K + block*P"
```

```
anova(npk1m2)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
N	1	189	189.3	11.21	0.0074 **
K	1	95	95.2	5.64	0.0389 *
block	5	343	68.7	4.07	0.0282 *
P	1	8	8.4	0.50	0.4966
block:P	5	71	14.3	0.85	0.5473
Residuals	10	169	16.9		

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print('Y ~ P + N + block*K')
```

```
## [1] "Y ~ P + N + block*K"
```

```
anova(npk1m3)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
N	1	189	189.3	11.14	0.0075 **
P	1	8	8.4	0.49	0.4980
block	5	343	68.7	4.04	0.0288 *
K	1	95	95.2	5.60	0.0395 *
block:K	5	70	14.1	0.83	0.5583
Residuals	10	170	17.0		

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print('Y ~ P + K + N + block')
```

```
## [1] "Y ~ P + K + N + block"
```

```
anova(npk1m4)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
block	5	343	68.7	4.29	0.0127 *
N	1	189	189.3	11.82	0.0037 **

```
## P          1      8      8.4    0.52 0.4800
## K          1     95     95.2    5.95 0.0277 *
## Residuals 15    240     16.0
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
print('Y ~ P + K + N')
```

```
## [1] "Y ~ P + K + N"
```

```
anova(npk1m5)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

```
##          Df Sum Sq Mean Sq F value Pr(>F)
## N          1    189    189.3     6.49  0.019 *
## P          1      8      8.4     0.29  0.597
## K          1     95     95.2     3.26  0.086 .
## Residuals 20    583     29.2
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
npk1m6 <- lm(yield~block + N + K, data=npk)
```

```
anova(npk1m6)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

```
##          Df Sum Sq Mean Sq F value Pr(>F)
## block     5    343     68.7     4.42  0.010 *
## N          1    189    189.3    12.18  0.003 **
## K          1     95     95.2     6.13  0.025 *
## Residuals 16    249     15.5
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
npk1m7 <- lm(yield~P*N*K, data=npk)
```

```
anova(npk1m7)
```

```
## Analysis of Variance Table
```

```
##
```

```
## Response: yield
```

```
##          Df Sum Sq Mean Sq F value Pr(>F)
## P          1      8      8.4     0.27  0.608
## N          1    189    189.3     6.16  0.025 *
## K          1     95     95.2     3.10  0.097 .
## P:N        1     21     21.3     0.69  0.418
## P:K        1      0      0.5     0.02  0.902
## N:K        1     33     33.1     1.08  0.314
## P:N:K      1     37     37.0     1.20  0.289
```



```
## Residuals 16    492    30.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We keep additive model for all factors, since pair-wise interactions were insignificant for all the factors. This model is our favorite as it shows also significance of each factor.

e)

```
require(lme4)
```

```
## Loading required package: lme4
```

```
## Loading required package: Matrix
```

```
npklmer <- lmer(yield~N+(1|block), REML=FALSE, data=npk)
npklmer1 <- lmer(yield~(1|block), REML=FALSE, data=npk)
anova(npklmer1, npklmer)
```

```
## Data: npk
```

```
## Models:
```

```
## npklmer1: yield ~ (1 | block)
```

```
## npklmer: yield ~ N + (1 | block)
```

```
##          npar AIC BIC logLik deviance Chisq Df Pr(>Chisq)
```

```
## npklmer1      3 159 163  -76.7      153
```

```
## npklmer       4 154 158  -72.7      146   7.9  1      0.005 **
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nitrogen has a significant effect.