Assignment 1

Rinus van Grunsven (st.number 10755373), Florens Douwes (11254483)

24 September 2022

Exercises

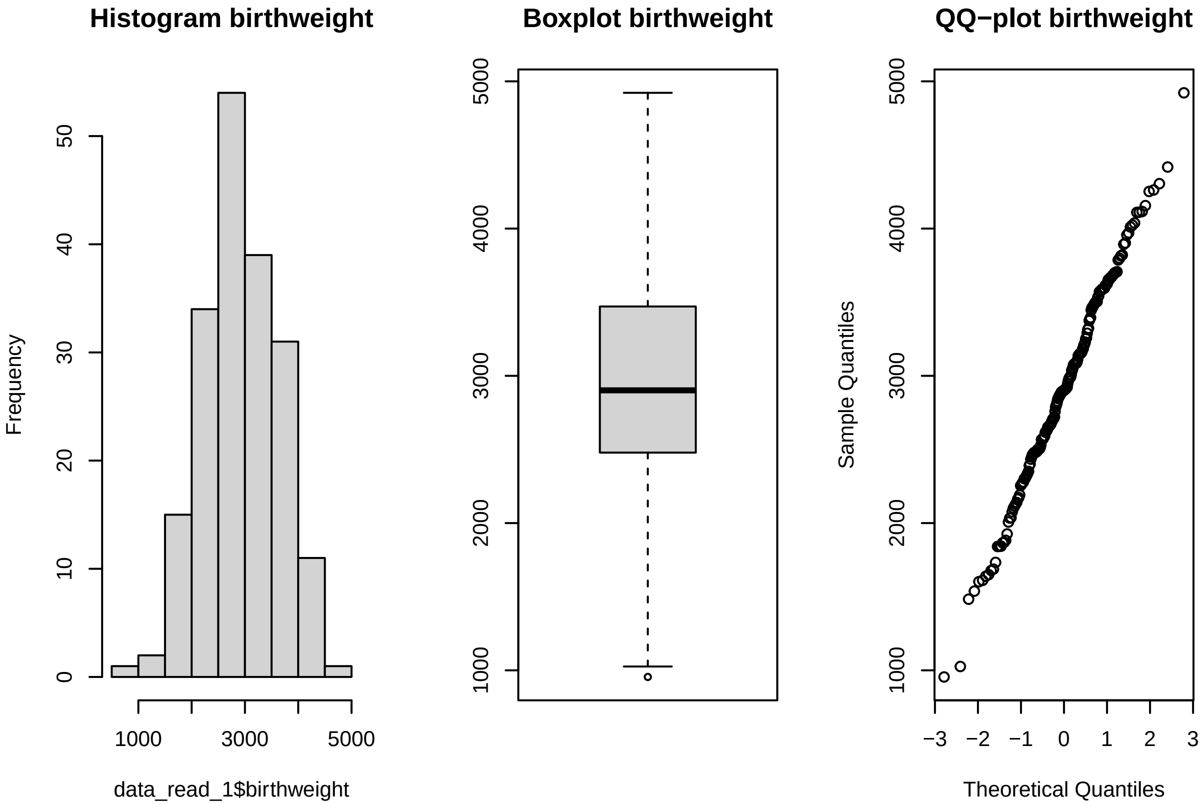
Exercise 1.1

The data that is being used in the exercise is retrieved via the following command: # {r} # data_read_1 = read.table("/Users/rinusvangrunsven/Documents/Study/UvA/SSO/week2/birthweight.tx #

```
data_read_1 = read.table("./birthweight.txt",header=TRUE)
```

(a) Below are the histogram, boxplot, and QQ-plot of the data.

```
par(mfrow=c(1,3));hist(data_read_1$birthweight, main="Histogram birthweight");boxplot(data_rea
```



The figures above all show indications of a normal distribution: the histogram is symmetrical and bell-shaped, the boxplot has roughly equally distanced whiskers, and the qq-plot has an almost straight line from the bottom left to top right corner.

Below a summary of the data

```
summary(data_read_1)
```

```
##   birthweight  
##  Min.   : 955  
##  1st Qu.:2479  
##  Median :2902  
##  Mean   :2913  
##  3rd Qu.:3468  
##  Max.   :4922
```

The mean, rounded to three decimals

```
m = round(mean(data_read_1$birthweight),2)
m
```

```
## [1] 2913.29
```

(b)

```
mu=0.2
n = length(data_read_1$birthweight); n # calculates the length
```

```
## [1] 188
```

```
s = round(sd(data_read_1$birthweight), 3); s # calculates standard deviation
```

```
## [1] 697.5
```

```
t = round(qt(0.95,df=length(data_read_1$birthweight)-1),3); t   # calculates test statistic
```

```
## [1] 1.653
```

```
ci = round(c(m-t*s/sqrt(n),m+t*s/sqrt(n)), 3); ci # 90% confidence interval
```

```
## [1] 2829.201 2997.379
```

The length is: 188. This is calculated with the following formula:

```
n = length(data_read_1$birthweight)
n
```

```
## [1] 188
```

The standard deviation is: 698. This is calculated with the following formula:

```
s = round(sd(data_read_1$birthweight), 3)
s
```

```
## [1] 697.5
```

The test statistic is: 1.6530429. This is calculated with the following formula:

```
t = round(qt(0.95,df=length(data_read_1$birthweight)-1),3)
t
```

```
## [1] 1.653
```

The confidence interval is then calculated with the formula: c(m-t*s/sqrt(n),m+t*s/sqrt(n))

```
round(c(m-t*s/sqrt(n),m+t*s/sqrt(n)), 3)
```

```
## [1] 2829.201 2997.379
```

(c) H0: mean birthweight is equal or smaller than 2800 H1: mean birthweight is bigger than 2800

```
t.test(data_read_1$birthweight,mu=2800,alt="g")
```

```
##
##  One Sample t-test
##
## data:  data_read_1$birthweight
## t = 2.2271, df = 187, p-value = 0.01357
## alternative hypothesis: true mean is greater than 2800
## 95 percent confidence interval:
##  2829.202      Inf
## sample estimates:
## mean of x
##  2913.293
```

The p-value is smaller than 0,05. This means that there is enough statistical evidence to reject H0 and thus to state that H1 is true.

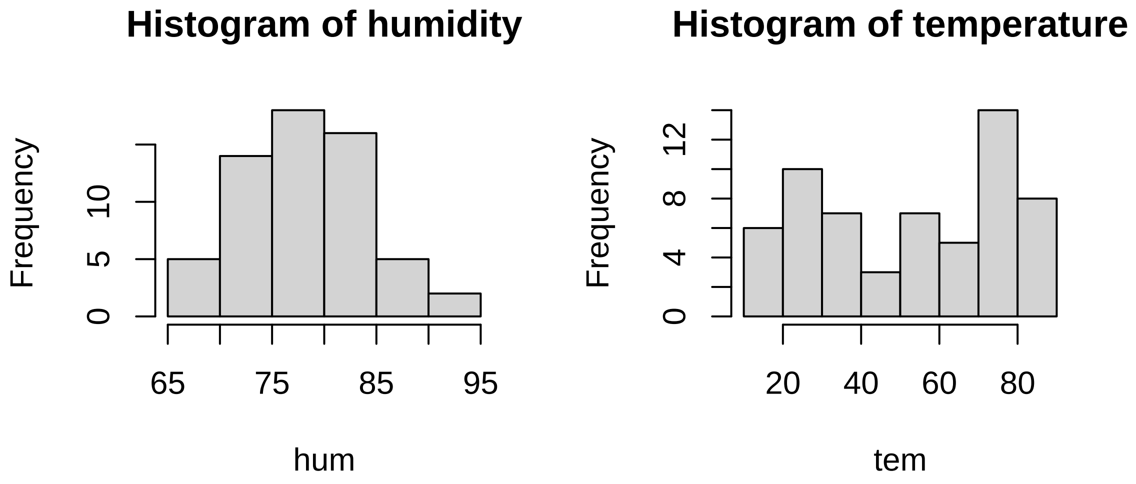
(d) It's different because in b) we calculated a confidence interval with a 90% confidence level, while the T-test that was conducted at c) calculated a confidence interval with a 95% confidence level. A confidence interval is a range of values that has a upper and lower bound below and above the mean of your statistics. The population parameter that you are trying to find would have to fall in this range based on the confidence level. A confidence level is the degree of certainty that the population value would fall in this range. The more you are certain that the population parameter will fall in this range, the wider the range will get. So a 90% confidence level will have a smaller range and thus a smaller confidence interval than a 95% confidence level would have. It is one-sided because we wanted to check if the mean birthweight is bigger than 2800. Therefore, the upper bound of the confidence interval does not have a value but goes to infinity as it does not matter which value it is. The null hypothesis is rejected as long as the value is bigger than the lower bound.

Exercise 1.3

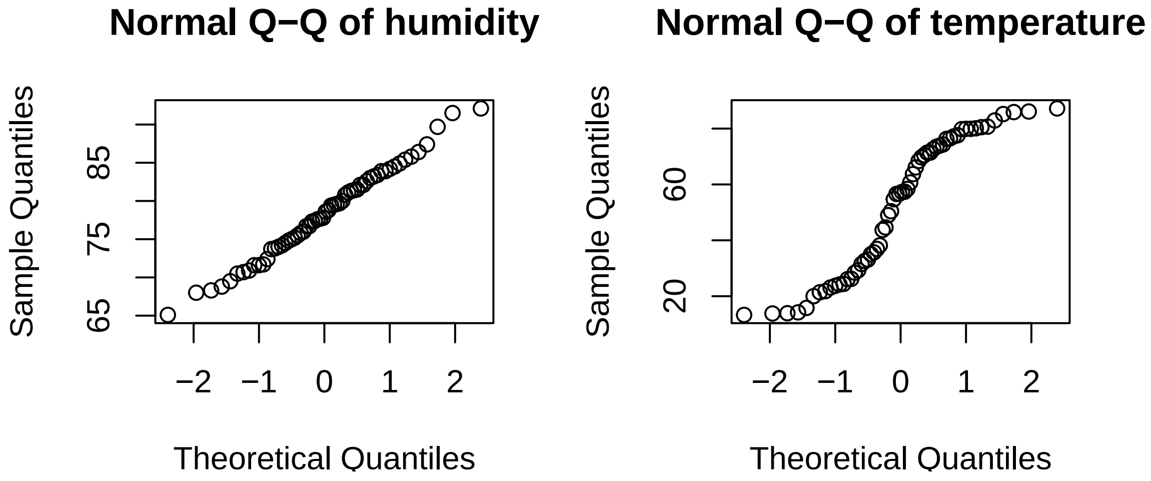
a)

```
hum=weather$humidity
tem=weather$temperature

par(mfrow=c(1,2))
hist(hum, main="Histogram of humidity")
hist(tem, main="Histogram of temperature")
```



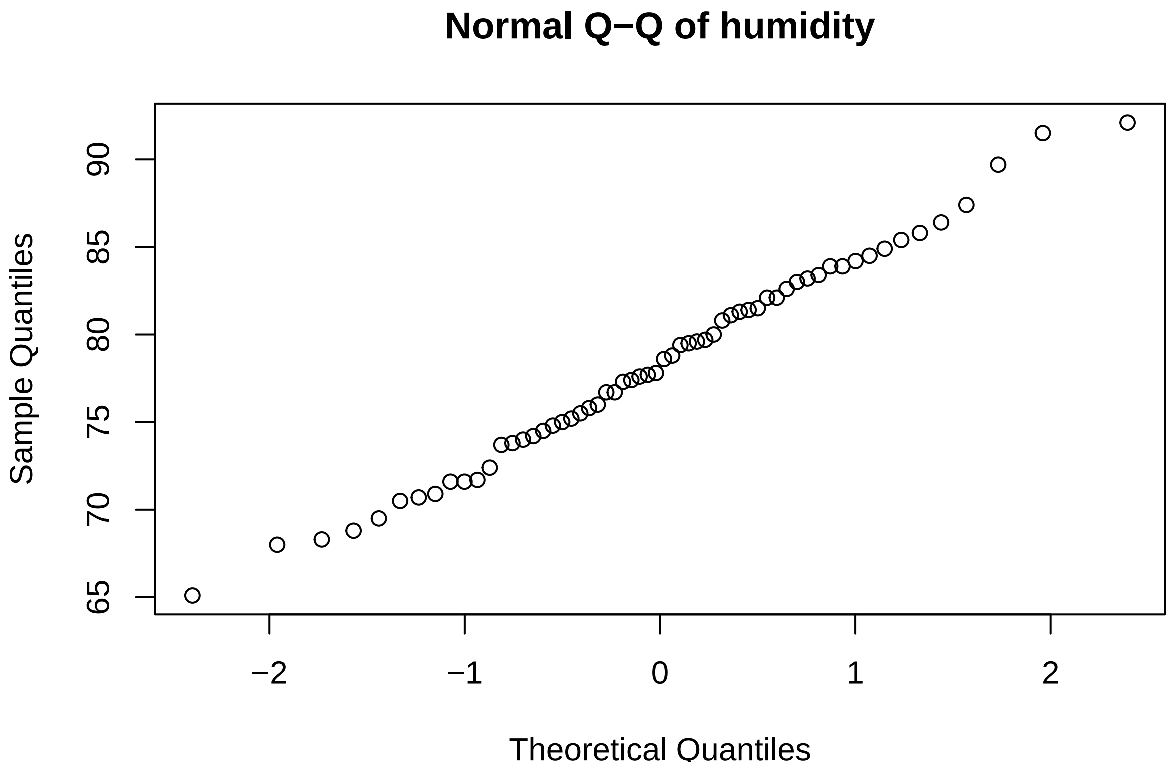
```
qqnorm(hum, main="Normal Q-Q of humidity")
qqnorm(tem, main="Normal Q-Q of temperature")
```



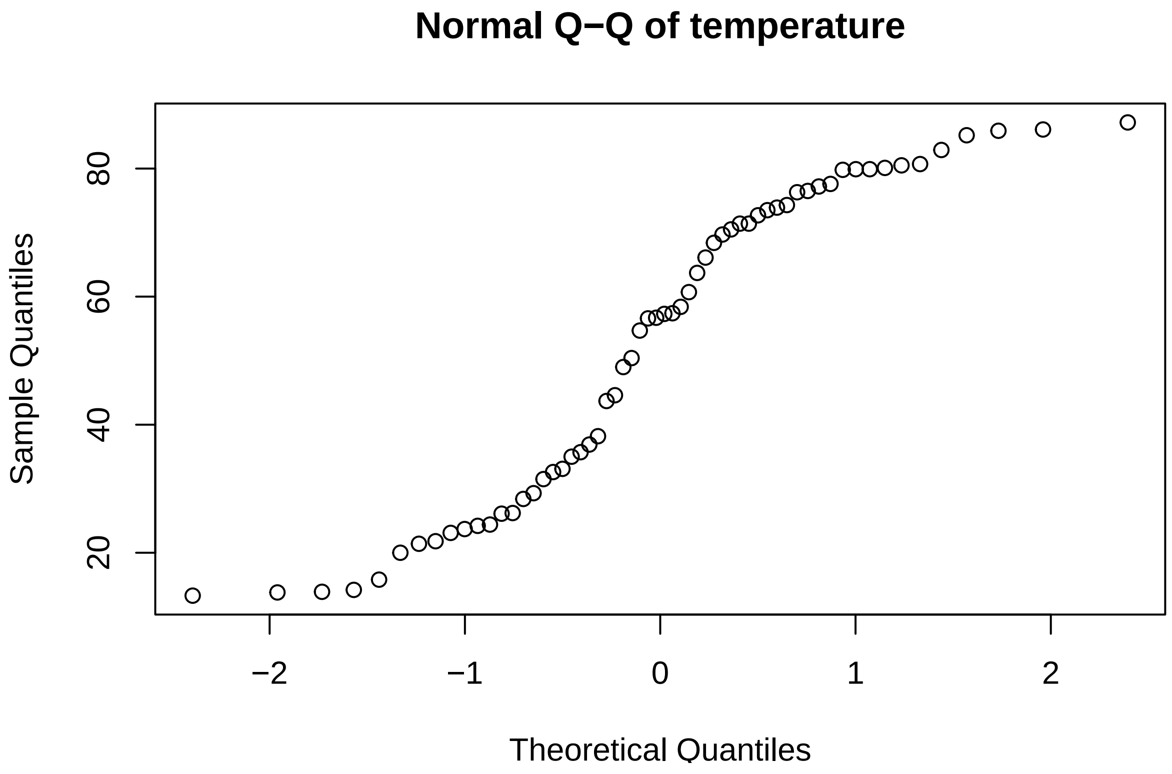
65.1, 74.15, 78.2, 78.3433333, 82.7, 92.1 13.3, 29.075, 57, 52.725, 74.8, 87.2

b)

```
qqnorm(hum, main="Normal Q-Q of humidity")
```



```r
qqnorm(tem, main="Normal Q-Q of temperature")
```



With this plot we can see that the temperature data does not follow a pattern seen with normal distributed samples. It is not a straight line in the Q-Q plot.

c)

The mean is unknown, as is the standard deviation. Therefore we will estimate the confidence interval using the t-distribution.

The formula to calculate the confidence interval then is $\bar{X} \pm t_{a/2,n-1}\frac{s}{\sqrt{n}}$.

With the following R code we can calculate the 90% confidence interval.

```
alpha = 0.10
n = length(weather$temperature)
m = mean(weather$temperature)
s = sd(weather$temperature)

t = qt(1 - alpha / 2, df=n - 1)
c(m - t * s / sqrt(n), m + t * s / sqrt(n))
```

```
## [1] 47.48704 57.96296
```

```
z = qnorm(0.95)
```

This means that the true mean is (with a confidence interval of 90%) between 47.49 and 57.96.

d)

This means that the margin of error should be a maximum 2% of humidity (that is, a value of 2).

n must satisfy the following formula:

$$t_{a/2,n-1}\frac{s}{\sqrt{n}} \leq E = \sqrt{n} \geq \frac{t_{a/2,n-1^s}}{E} \approx \frac{(z_{a/2})^2 s^2}{E^2} = \frac{(1.64)^2(24.28)^2}{4} = 396.40$$

Which would mean we would need at least 397 samples.

Exercise 1.4

The following data is being used for this exercise:

```
data_read_4 = read.table("./austen.txt",header=TRUE)
```

(a) It is unclear by whom Sense and Emma are written. For the other two novels it is not: Sand1 is written by Austen and Sand2 is written by an admirer. Our guess is that Sense and Emma ar written by Austen as well and we will use this assumption while conducting the other exercises also. Furthermore, we presumed that the test would be performed in order to check to what extend the novels are all written in the same way. According to us, a test for homogeneity is the most suitable option to use. This test is able to check if the distribution of words is evenly spread across columns when comparing rows with each other. The distribution should be the same across rows in case the writing style is the same in all the novels.

(b) We start with making a matrix of only the novels that Austen wrote. The matrix is shown below.

```
Sense = data_read_4$Sense
Emma = data_read_4$Emma
```

```
Sand1 = data_read_4$Sand1
x=as.data.frame(matrix(c(Sense, Emma, Sand1),ncol=3,nrow=6))
dimnames(x)=list(c("a","an", "this", "that", "with", "without"),c("Sense","Emma", "Sand1"))
x
```

```
##         Sense Emma Sand1
## a         147  186   101
## an         26   25    11
## this       32   38    16
## that       98  105    37
## with       59   76    28
## without    20   10    10
```

Now that we only have the data of Austens novels, we can run a chi-squared test to see if Austen herself was consistent in her writing style or not. We will use the following hypotheses: H0: the distribution of words is the same for each column H1: the distribution of words is not the same for each column This is performed in R with the command below

```
z = chisq.test(x)
z
```

```
##
##  Pearson's Chi-squared test
##
## data:  x
## X-squared = 14.274, df = 10, p-value = 0.1609
```

The p-value is 0.1308 and this is greater than 0.05 (or 5%). This means that there is not enough statistical evidence to reject H0, and thus we conclude that the distribution of words is the same for each column. As a result of this, we conclude that Austen was consistent in her writing style.

Shall we include the parts below?

```
z$expected
```

```
##             Sense      Emma     Sand1
## a       161.74439 186.30244 85.953171
## an       23.10634  26.61463 12.279024
## this     32.05073  36.91707 17.032195
## that     89.44390 103.02439 47.531707
## with     60.74732  69.97073 32.281951
## without  14.90732  17.17073  7.921951
```

```
z$observed
```

```
##         Sense Emma Sand1
## a         147  186   101
## an         26   25    11
## this       32   38    16
## that       98  105    37
## with       59   76    28
## without    20   10    10
```

```
X2=sum((z$observed-z$expected)^2/z$expected)
X2
```

```
## [1] 14.27373
```

And this also?

```
1-pchisq(X2,df=(6-1)*(3-1)) # same is the same as the p-value we got from the Chi-squared test
```

```
## [1] 0.1608682
```

```
residuals(z)
```

```
##                Sense       Emma      Sand1
## a       -1.159343692 -0.0221579  1.6229833
## an       0.601979480 -0.3129781 -0.3650034
## this    -0.008961083  0.1782319 -0.2501074
## that     0.904691184  0.1946396 -1.5275909
## with    -0.224185835  0.7207861 -0.7536363
## without  1.319006371 -1.7304899  0.7383115
```

(c) Since we are going to use all data, we have to add the data from the novel written by the admirer to our current matrix. This is done with the code below.

```
Sand2 = data_read_4$Sand2
x=as.data.frame(matrix(c(Sense, Emma, Sand1, Sand2),ncol=4,nrow=6))
dimnames(x)=list(c("a","an", "this", "that", "with", "without"),c("Sense","Emma", "Sand1", "Sa
x
```

```
##         Sense Emma Sand1 Sand2
## a         147  186   101    83
## an         26   25    11    19
## this       32   38    16    15
## that       98  105    37    41
## with       59   76    28    39
## without    20   10    10     4
```

The matrix now consists of data from all novels. We will again use a chi-squared test in order to test whether the admirer was successful in imitating Austen's style. We run a test for homogeneity since we test if the admirer was successful in imitating Austen's writing style. The hypotheses will therefore be: H0: the distribution of words is the same for each column H1: the distribution of words is not the same for each column The chi-squared test is performed in R with the command below

```
z = chisq.test(x)
z
```

```
##
##  Pearson's Chi-squared test
##
## data:  x
## X-squared = 21.528, df = 15, p-value = 0.1208
```

The p-value is 0.1208 and this is greater than 0.05 (or 5%). This means that there is not enough statistical evidence to reject H0, and thus we conclude that the distribution of words is the same for each column. As a result of this, we conclude that the admirer was successful in imitating Austen's writing style.

Shall we include the code below as well?

We also ran a second test in order to confirm if our conclusion stated above. For this second test we only took one novel of Austen and compared this with the one from the admirer. A new matrix was made for this, see below.

```
sand2 = data_read_4$Sand2
x=as.data.frame(matrix(c(Sand1, Sand2),ncol=2,nrow=6))
dimnames(x)=list(c("a","an", "this", "that", "with", "without"),c("Sand1", "Sand2"))
x
```

```
##         Sand1 Sand2
## a         101    83
## an         11    19
## this       16    15
## that       37    41
## with       28    39
## without    10     4
```

A chi-squared test was conducted on these numbers. The result is shown below.

```
z = chisq.test(x)
z
```

```
##
##  Pearson's Chi-squared test
##
## data:  x
## X-squared = 8.4993, df = 5, p-value = 0.1308
```

The p-value is 0.1308 and this is again greater than 0.05 (or 5%). This means that there is not enough statistical evidence to reject H0, and thus we conclude that the distribution of words is the same for each column. As a result of this, we conclude that the admirer was successful in imitating Austen's writing style.

Shall we include the code below as well?

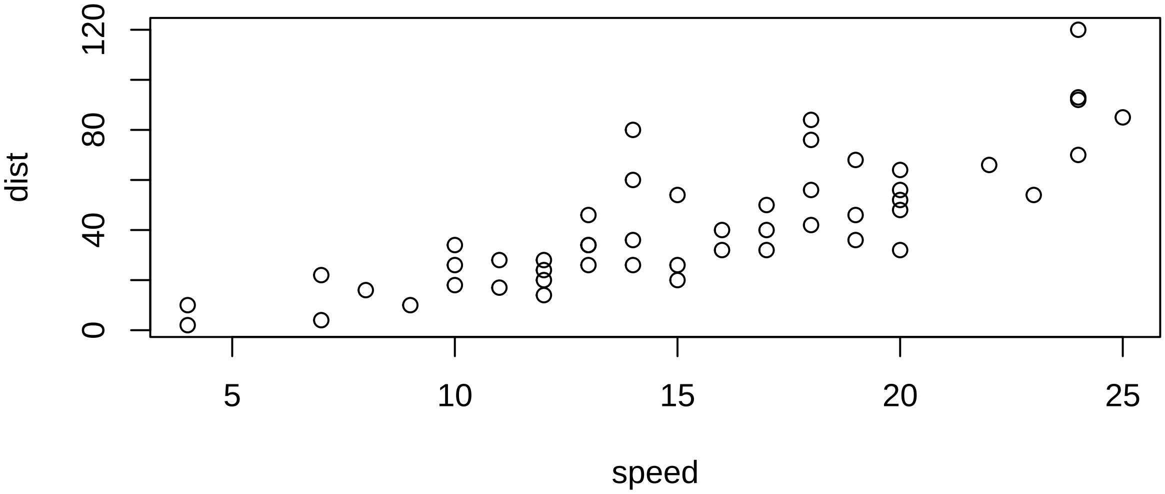
Introduction to R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. R Markdown files permit you to interweave R code with ordinary text to produce well-formatted data analysis reports that are easy to modify. The R Markdown file itself shows the readers exactly how you got the results in your report. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button, a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. For inline R code, surround code with back ticks and r. R replaces inline code with its results. For example, two plus one is 3; for the build-in R dataset `cars`, there were 50 cars studied. You can embed an R code chunk like this:

Figures

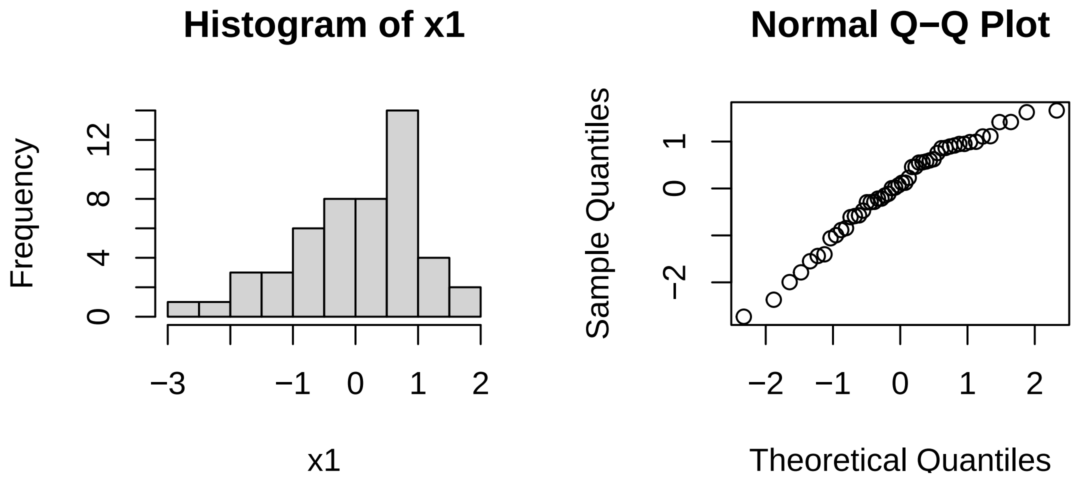
You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot. Use knitr options to style the output of a chunk. Place options in brackets above the chunk. Other options with the defaults are: the `eval=FALSE` option just displays the R code (and does not run it); `warning=TRUE` whether to display warnings; `tidy=TRUE` wraps long code so it does not run off the page.

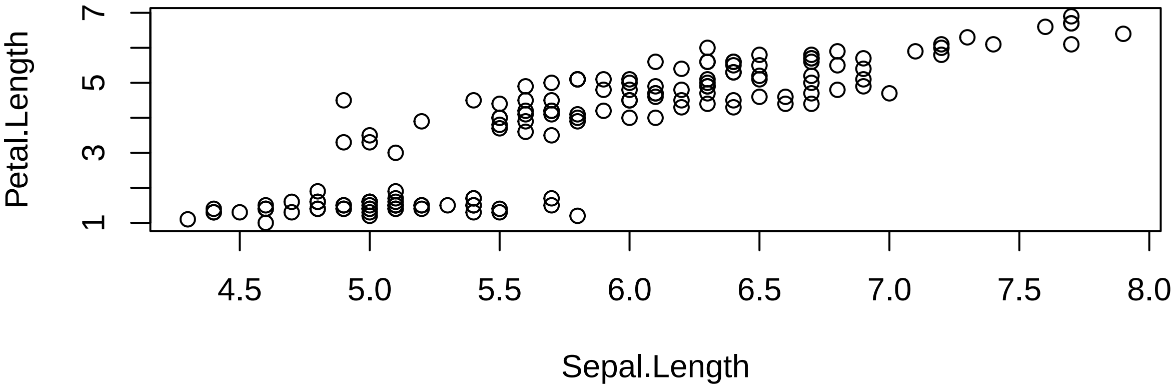
You can control the size and placement of figures. For example, you can put two figures (or more) next to each other. Use `par(mfrow=c(n,m))` to create `n` by `m` plots in one picture in R. You can adjust the proportions of figures by using the `fig.width` and `fig.height` chunk options. These are specified in inches, and will be automatically scaled down to fit within the handout margin. Chunk option `fig.align` takes values `left`, `right`, or `center` (to align figures in the output document).

```r
par(mfrow=c(1,2)); x1=rnorm(50); hist(x1); qqnorm(x1)
```



You can arrange for figures to span across the entire page by using the `fig.fullwidth` chunk option.

```
plot(iris$Sepal.Length,iris$Petal.Length,xlab="Sepal.Length",ylab="Petal.Length")
```



More about chunk options can be found at https://yihui.name/knitr/options/.

Equations

You can also include LaTeX expessions/equations in your report: inline $\frac{d}{dx}\left(\int_0^x f(u)\,du\right)=f(x)$ and in the display mode:

$$\frac{d}{dx}\left(\int_0^x f(u)\,du\right)=f(x).$$

To possibly avoid LaTeX expressions in you report (you do not really need LaTeX for your reports), remove this part (part **Equations**) and try to knit your Rmd-file.

Footnotes

Here is the use of a footnote[1].

Tables

Want a table? This will create one (note that the separators *do not* have to be aligned).

Table Header	Second Header
Table Cell	Cell 2
Cell 3	Cell 4

You can also make table by using knit's `kable` function:

Table 2: A knit kable.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1

[1] This is a footnote.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2

Block quote

> This will create a block quote, if you want one.

Verbatim

```
This text is displayed verbatim/preformatted.
```

Links

Links: http://example.com, in-text link to Google, linked phrase.

Italicized and embolded text

- Single asterisks italicize text *like this.*
- Double asterisks embolden text **like this**.

One more way: *italic* and **bold**.

Exercises

Exercise 1

(i) Here are some consequitive R-commands.

```
x=rep(c("A","B"),each=5); x
```

```
##  [1] "A" "A" "A" "A" "A" "B" "B" "B" "B" "B"
```

```
sample(x)
```

```
##  [1] "B" "A" "A" "A" "A" "B" "A" "B" "B" "B"
```

```
x=rnorm(100)
```

Now the same code chunk but with all the output collapsed into signle block.

```
x=rep(c("A","B"),each=5); x
##  [1] "A" "A" "A" "A" "A" "B" "B" "B" "B" "B"
sample(x)
##  [1] "A" "B" "A" "A" "B" "B" "B" "B" "A" "A"
x=rnorm(100)
```

(ii) Below we perform a one sample t-test for the artificial date (that we generated ourselves).

```
mu=0.2
x=rnorm(100,mu,1) # creating artificial data
t.test(x,mean=0)   # t.test(x,alternative=c("two.sided"),conf.level=0.95,mu=10)
```

```
##
##  One Sample t-test
##
## data:  x
## t = 1.8997, df = 99, p-value = 0.06039
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.007969277  0.366066555
## sample estimates:
## mean of x
## 0.1790486
```

(iii) We often do not need to report the whole output of R-commands, only certain values of the output. For example, below we perform a two-sample t-test and report only the (appropriately rounded) values of t-statistics and the p-pavue.

```
mu=0;nu=0.5
x=rnorm(50,mu,1); y=rnorm(50,nu,1) # creating artificial data
ttest=t.test(x,y)
```

The value of t-statistics in the above evaluation is
-1.21 and the p-value is 0.2278.