

Excercise Section

Contents

About	1
Chapter 1	1
Chapter 2	2
Datensätze	6

About

The Excercises follow the course “Einführung in die Datenanalyse mit R”.

A useful Markdown Cheatsheet can be found here on Github.

Chapter 1

1.3 Help

```
# install.packages("tidyverse")
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

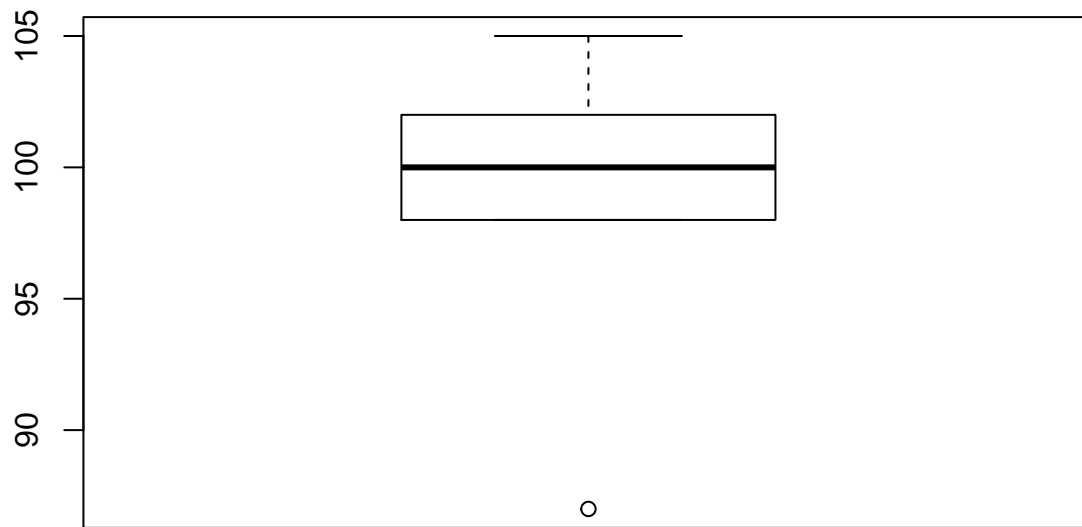
## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

?dplyr
```

1.4 Arbeiten mit RStudio

```
x <- c(101, 105, 99, 87, 102, 98)
boxplot(x)
```



Chapter 2

2.1 Operatoren und Funktionen

```
#1
(1/3)*(sum(1,3,5,7,2)/sum(3,5,4))
```

```
## [1] 0.5
```

```
#2
exp(1)
```

```
## [1] 2.718282
```

```
#3
sqrt(2)
```

```
## [1] 1.414214
```

```
#4
3^(1/8)
```

```
## [1] 1.147203
```

```
#5
sin(2*pi)
```

```
## [1] -2.449294e-16
```

```
#6
log2(8)
```

```
## [1] 3
```

```
log(8)/log(2)
```

```
## [1] 3
```

```

#1
seq(from=0, to=100,by=5)

## [1] 0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80
## [18] 85 90 95 100

#2
vector <- c(1,3,4,7,11,2)
mean(vector)

## [1] 4.666667

#3
max(vector)-min(vector)

## [1] 10

#4
sum(vector)

## [1] 28

#5
scale(vector)

##           [,1]
## [1,] -0.9846580
## [2,] -0.4475718
## [3,] -0.1790287
## [4,] 0.6266005
## [5,] 1.7007728
## [6,] -0.7161149
## attr("scaled:center")
## [1] 4.666667
## attr("scaled:scale")
## [1] 3.723797

#6
sample(c(0,1), size = 100, replace = T)

## [1] 1 1 0 1 0 1 0 1 1 0 0 1 0 1 1 0 0 0 0 1 0 0 0 1 0 0 1 1 1 0 1 1 0 0 1
## [36] 1 0 0 1 0 1 1 1 0 1 1 1 1 1 0 1 0 1 1 0 1 1 1 1 0 0 0 0 1 0 1 0 1
## [71] 1 0 0 1 1 0 0 0 1 1 1 0 1 1 1 0 0 0 1 1 0 0 0 0 1 1 1 1 1 1

#7
sample(c(0,1), size = 100, replace = T,prob = c(.25,.75))

## [1] 1 1 1 0 1 1 0 1 0 1 1 1 1 1 1 0 0 1 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1
## [36] 0 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 0 0 0 1 0
## [71] 1 1 0 1 1 1 0 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 0 1

#8
rep(3,100)

## [1] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
## [36] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
## [71] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3

```

2.2 Variablen definieren 1. Just DO it! 2. round(x=digits=) 3. rnorm(n, mean = 0, sd = 0) 4. seq(from = 1, to = 1, by = ((to - from)/(length.out - 1)), length.out = NULL, along.with = NULL, ...) 5.

```
seq()
```

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

```
seq(1, 10)
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
seq(1, 10, 2)
```

```
## [1] 1 3 5 7 9
```

```
#seq(1, 10, 2, 20) #--> too many Arguments
```

```
seq(1, 10, length.out = 20)
```

```
## [1] 1.000000 1.473684 1.947368 2.421053 2.894737 3.368421 3.842105
```

```
## [8] 4.315789 4.789474 5.263158 5.736842 6.210526 6.684211 7.157895
```

```
## [15] 7.631579 8.105263 8.578947 9.052632 9.526316 10.000000
```

2.5 Übungsaufgaben

```
#Zahlen runden
```

```
x <- rnorm(10, mean = 1, sd = 0.5)
```

```
round(x = x, digits = 0)
```

```
## [1] 1 1 1 1 1 0 0 1 2 1
```

```
round(x = x, digits = 3)
```

```
## [1] 0.719 0.758 0.628 0.773 0.654 0.203 0.194 1.081 1.792 0.592
```

```
(zahl <- 3.45263)
```

```
## [1] 3.45263
```

```
ceiling(zahl)
```

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

```
floor(zahl)
```

```
## [1] 3
```

```
#Mittelwert berechnen
```

```
df <- data_frame(geschlecht = sample(c("male", "female"),size = 24,replace = TRUE), alter = runif(24, m
```

```
mean(df$alter)
```

```
## [1] 32.43684
```

```
summary(df)
```

```
## geschlecht alter
```

```
## Length:24 Min. :19.82
```

```
## Class :character 1st Qu.:26.33
```

```
## Mode :character Median :32.69
```

```
## Mean :32.44
```

```
## 3rd Qu.:38.68
```

```
## Max. :44.62
```

```
#Matrizen
```

```
##1
```

```

m1 <- matrix(rnorm(48, mean = 110, sd = 5), ncol = 4)
m2 <- matrix(rnorm(48, mean = 100, sd = 10), ncol = 4)
m3 <- rbind(m1, m2)

```

```
##2
```

```
m3[1:12,]
```

```

##      [,1]      [,2]      [,3]      [,4]
## [1,] 108.0184 109.7103 115.0869 114.7510
## [2,] 115.0432 105.7658 114.4151 107.7136
## [3,] 104.0622 108.4615 100.7761 104.6772
## [4,] 119.2901 108.1481 114.8561 103.6908
## [5,] 112.1457 106.8386 111.3219 109.3283
## [6,] 116.3367 106.2751 111.4422 105.3279
## [7,] 110.6727 107.9805 104.8990 103.1462
## [8,] 113.9921 108.2898 107.7823 112.4254
## [9,] 113.9843 112.0892 102.2051  99.9914
## [10,] 100.3329 107.8115 113.4493 120.5926
## [11,] 118.4559 108.2560 107.4697 113.4479
## [12,] 106.7541 111.2311 112.2812 110.3523

```

```
m3[1:12,] == m1
```

```

##      [,1] [,2] [,3] [,4]
## [1,] TRUE TRUE TRUE TRUE
## [2,] TRUE TRUE TRUE TRUE
## [3,] TRUE TRUE TRUE TRUE
## [4,] TRUE TRUE TRUE TRUE
## [5,] TRUE TRUE TRUE TRUE
## [6,] TRUE TRUE TRUE TRUE
## [7,] TRUE TRUE TRUE TRUE
## [8,] TRUE TRUE TRUE TRUE
## [9,] TRUE TRUE TRUE TRUE
## [10,] TRUE TRUE TRUE TRUE
## [11,] TRUE TRUE TRUE TRUE
## [12,] TRUE TRUE TRUE TRUE

```

```
#Character vectors
```

```

ID <- c(1, 2, 3, 4, 5)
Initialen <- c("RS", "MM", "PD", "PG", "DK")
Alter <- c(44, 78, 22, 34, 59)

```

```
personen <- paste(ID, Initialen, Alter, sep = "-")
```

```
#Data Frame
```

```

library(dplyr)
library(tidyr)

```

```

kein_alkohol <- c(64, 58, 64)
placebo <- c(74, 79, 72)
anti_placebo <- c(71, 69, 67)
alkohol <- c(69, 73, 74)

```

```

alk_aggr <- data_frame(kein_alkohol = kein_alkohol,
                      placebo = placebo,
                      anti_placebo = anti_placebo,
                      alkohol = alkohol)

alk_aggr <- alk_aggr %>%
  gather(key = alkoholbedingung, value = aggressivitaet) %>%
  mutate(alkoholbedingung = factor(alkoholbedingung))

levels(alk_aggr$alkoholbedingung)

## [1] "alkohol"      "anti_placebo" "kein_alkohol" "placebo"

levels(alk_aggr$alkoholbedingung) <- c("placebo",
                                         "anti_placebo",
                                         "kein_alkohol",
                                         "alkohol")

levels(alk_aggr$alkoholbedingung)

## [1] "placebo"      "anti_placebo" "kein_alkohol" "alkohol"

alk_aggr$alkoholbedingung <- relevel(alk_aggr$alkoholbedingung, ref = "placebo")
levels(alk_aggr$alkoholbedingung)

## [1] "placebo"      "anti_placebo" "kein_alkohol" "alkohol"

#Fortgeschrittene Aufgaben:
##1
x <- seq(1, 20, by = 1)
x[x %% 2==0]

## [1]  2  4  6  8 10 12 14 16 18 20

##2
x[x %% 2==1]

## [1]  1  3  5  7  9 11 13 15 17 19

```

Datensätze

3.1

```

kein_alkohol <- c(64, 58, 64)
placebo <- c(74, 79, 72)
anti_placebo <- c(71, 69, 67)
alkohol <- c(69, 73, 74)

kein_alkohol <- data_frame(aggressivitaet = kein_alkohol,
                          alkoholbedingung = "kein_alkohol")
placebo <- data_frame(aggressivitaet = placebo,
                     alkoholbedingung = "placebo")
anti_placebo <- data_frame(aggressivitaet = anti_placebo,
                          alkoholbedingung = "anti_placebo")
alkohol <- data_frame(aggressivitaet = alkohol,
                     alkoholbedingung = "alkohol")
alk_aggr <- bind_rows(kein_alkohol,
                     placebo,

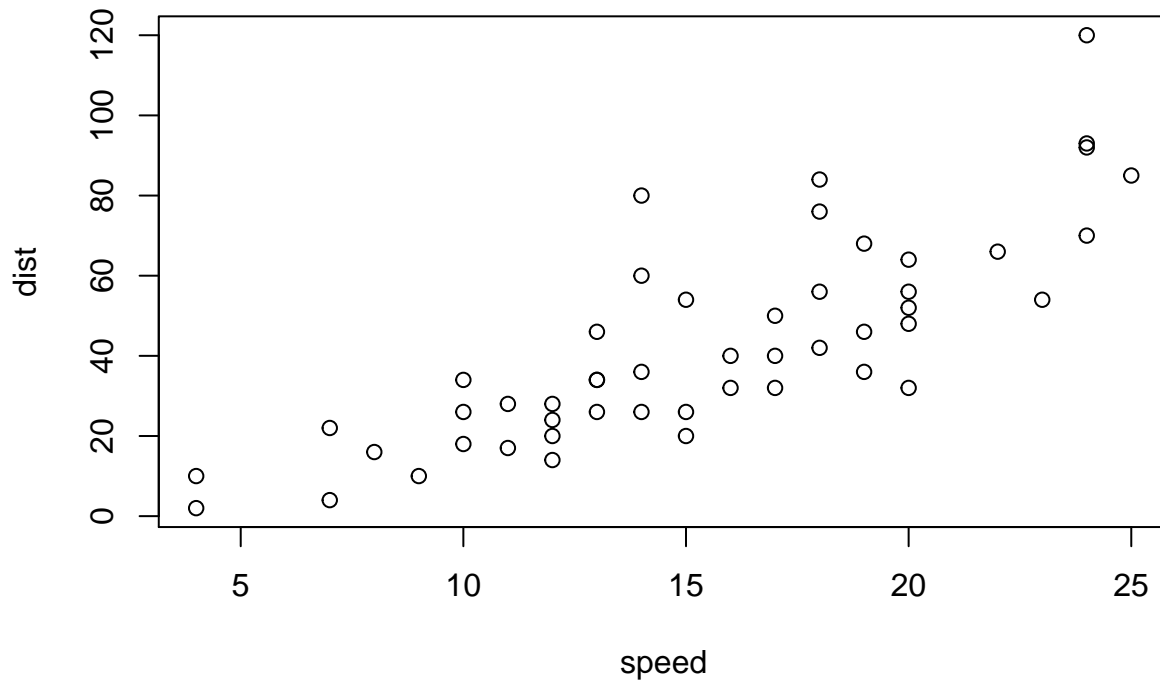
```

```
      anti_placebo,  
      alkohol)  
alk_aggr$alkoholbedingung <- factor(alk_aggr$alkoholbedingung)
```

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

```
plot(cars)
```



Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

```
x<-runif(54)  
boxplot(x)  
hist(x)
```