

Applied Statistics for Data Science

Zusammenfassung

Stephan Stofer

23. September 2020

Inhaltsverzeichnis

1	R Basics	4
---	----------	---

Abbildungsverzeichnis

1 R Basics

Einführungsbeispiele in R

R als Taschenrechner

1+1

2*4

exp(3)

Kommentaren werden bei der Bearbeitung ignoriert; mit

Ein erstes Beispiel

1+1

Zuweisungen mit <-

Variable a wird Wert 3 und Variable der Wert 4 zugeordnet

Variablen können aus mehreren Buchstaben bestehen

a <- 3

bb <- 4

a

bb

Berechnungen mit Variablen

2.7*a

a <- 2.7*a

a^2

a/bb

R ist case sensitiv; a ist nicht gleich A

a <- 7

a

Variable A kennt R hier nicht, da sie nicht definiert wurde

A

Multiplikationszeichen schreiben

3a

3*a

Listen mit c(...) (englisch: concatenate)

z.1 <- c(3, 4, 4.5, -2, 7)

z.1

z.2 <- c(1, 2, 3, 4, 5)

z.2

Eine ziemlich allgemeine Liste; Einträge können mehrfach vorkommen

verschiedenes <- c(2, 'traurig', 5.3, 'JA', "schön", "traurig")

verschiedenes

Operationen mit Listen

z.1 und z.2 als Vektoren betrachten und mit denen auch rechnen.

4*z.1

Multiplikation wird komponentenweise multipliziert, also 1. Zahl mal 1. Zahl, 2. Zahl mal 2. Zahl

z.1*z.2

Sinnlose Multiplikation mit Liste verschiedenes:

3*verschiedenes

Durchschnitt der Zahlen, die in der Liste z.1

Befehl mean(...)

mean(z.1)

Oder Summe, Minimum oder Maximum der Werte, die in der Liste z.1 vorkommenden

sum(z.1)

min(z.1)

max(z.1)

Länge einer Liste mit length(...); Anzahl Einträge

length(verschiedenes)

Zugriff auf einen Eintrag in einer Liste; mit eckigen Klammern

verschiedenes[2]

Auswahl von mehreren Einträgen; 1. und 3. Eintrag

Was nicht geht, naheliegend, geht aber nicht wegen Optionen von R (später)

verschiedenes[1, 3]

Damit ein Eintrag vorhanden ist -> Liste

verschiedenes[c(1, 3)]

Spezielle Abfragen von Listen

Einträge 2,3,4,5 aus Liste abfragen; Schreibweise 2:5

verschiedenes[2:5]

Befehle mit Optionen; Beispiel

Wert NA (not available) kommt in der Statistik oft vor die aus irgendwelchen Gründen nicht vorhanden sind

Bsp: Geburtsdatum von einigen Personen unbekannt

z.3 <- c(5, 2, NA, 4)

Minimum aus z.3 bestimmen; Resultat nicht zu gebrauchen

R versucht in der Liste das Minimum zu suchen, kann aber mit dem NA nichts anfangen

min(z.3)

Minimum der vorkommenden Zahlen bestimmen; Option, die in den Funktionsklammern zusätzlich angegeben wird

na.rm=TRUE; die Option ist hier na.rm=\\dots und steht "für na remove"

Option TRUE gesetzt, so werden die \\Rcom{NA}'s bei der Bestimmung des Minimums entfernt

FALSE werden sie in der Liste belassen; Standardeinstellung ist FALSE

min(z.3, na.rm=TRUE)

Alle Optionen sind mit einer Standardeinstellung versehen; über ?min einsehbar

?min

Datensätze zweidimensional

Einlesen eines Datensatzes

Working directory festlegen; dort wo sich Ihr File befindet

setwd("...")

Einlesen einer .csv-Datei und der Variable temp zuordnen

temp <- **read.csv**("weather.csv")

temp

Oder Datei direkt mit Pfad einlesen

temp <- **read.csv**("...weather.csv")

temp

Mit read.table einlesen; allgemeinerer Einlesebefehl

temp <- **read.table**("weather.csv")

temp

Resultat sieht nicht schön aus; Fehler, da read.table verlangt, da Einträge durch Leerzeichen getrennt sind

Option sep=","

temp <- **read.table**("weather.csv", sep = ",")

temp

Zeilen und Spalten herauslesen

Wichtige Bemerkung: erste Zeile und die erste Spalte gehören nicht zur Tabelle

Sie werden verwendet um einzelne Werte, ganze Spalten oder andere Teile der Tabelle herauszulesen.

Befehl head(...); Gibt standardmässig die ersten sechs Zeilen einer Tabelle aus

head(temp)

Anzahl Zeilen und Spalten; dim(...); immer zuerst Zeilen und dann Spalten

dim(temp)

Spaltennamen; col für columns

colnames(temp)

Zeilenamen; row für Zeile

rownames(temp)

Erster Ueberblick über die Daten

summary(temp)

Einzelne Werte herauslesen

temp[2, 4]

Praktischer

temp["Mai", "Basel"]

Ganze Zeilen und Spalten herauslesen

temp["Jun",]

temp[, "Basel"]

Nur bei Spalten möglich

```
temp["Basel"]
```

Mittelwert einer Spalten

```
mean(temp[, "Luzern"])
```

Teile einer Tabelle einer Tabelle herauslesen und als \texttt{.csv}--File abspeichern

```
temp1 <- temp[c("Jan", "Mai"), c("Chur", "Zürich")]
```

```
temp1
```

```
write.csv(temp1, file = "wetter1.csv")
```

Comma separated file

```
write.table(temp1, file = "wetter1.csv", sep = ",")
```

Beispiel einer Graphik

Liste von Farben, die nachher verwendet werden

```
farben <- c('green', 'blue', 'red', 'brown')
```

Graphen der Spalten der Tabelle; Option col=... steht für colors

Hier col = farben weiter oben gesetzt

```
matplot(temp, col = farben, type = "l", lty = 1)
```

Legende links oben

Zahlen 1 und 25 bilden die Koordinaten, wo die Legende zu liegen kommen soll

Option legend = colnames(temp) bezeichnet die Namen der Linien; (natürlicherweise) mit den Spaltennamen der Tabelle

```
legend(x = 1, y = 25, legend = colnames(temp), col = farben, lty = 1)
```