

Angewandte Datenanalyse mit R

Tag 1 - Einführung in Programmierung mit R

Andreas Mock

Abteilung für Medizinische Onkologie & Abteilung für Translationale Medizinische Onkologie, Nationales
Centrum für Tumorerkrankungen (NCT) Heidelberg

Sommersemester 2019



NATIONAL CENTER
FOR TUMOR DISEASES
HEIDELBERG



HEIDELBERG
UNIVERSITY
HOSPITAL



GERMAN
CANCER RESEARCH CENTER
IN THE HELMHOLTZ ASSOCIATION

... * * * *

Research for a Life without Cancer



DKTK

German Cancer
Consortium

Hands-On Praktikum: Angewandte Datenanalyse mit R

Organisatorisches - Sommersemester 2019

Kontakt: andreas.mock@nct-heidelberg.de

Ort: Frauenklinik

Kurszeiten

- ▶ Termin 1: 11.06.2019 - 15:30-17:00 Uhr - Raum 00.252
- ▶ Termin 2: 18.06.2019 - 15:30-17:00 Uhr - Raum 00.252
- ▶ Termin 3: 25.06.2019 - 15:30-17:00 Uhr - Raum 00.273
- ▶ Termin 4: 02.07.2019 - 15:30-17:00 Uhr - Raum 00.250

Hands-On Praktikum: Angewandte Datenanalyse mit R

Kursinhalte

- ▶ Tag 1: Einführung in Programmierung mit R
- ▶ Tag 2: Datentransformation und -visualisierung
- ▶ Tag 3: Datenimport und -modellierung
- ▶ Tag 3: Datenimport und -modellierung

Kursunterlagen

<http://andreasmock.github.io/teaching>

Tag 1	Tag 2	Tag 3	Tag 4
Präsentation	Präsentation	Präsentation	Präsentation
Übungen	Übungen	Übungen	Übungen
Lösungen	Lösungen	Lösungen	Lösungen

Die Lösungen zu den Übungen werden am Ende des jeweiligen Tages online gestellt.

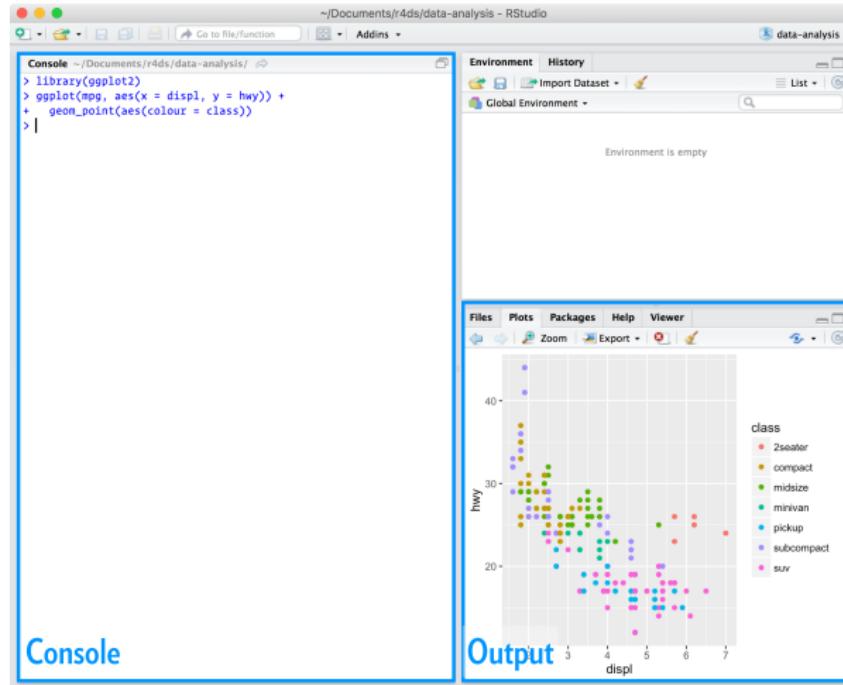


Was ist R?

- ▶ R ist eine **freie Programmiersprache** für statistische Berechnungen und Grafiken.
- ▶ Obwohl R bereits alt ist (Erscheinungsjahr 1993) gilt diese als **Standardsprache** für statistische Problemstellungen in Wirtschaft und Wissenschaft
- ▶ > **11.000 Formelsammlungen** für spezifische Fragestellungen (sog. Pakete)
- ▶ Hoch-qualitative und individuelle **Grafiken** - viele Wissenschaftler benutzen R nur hierzu
- ▶ Sowohl R, als auch alle Pakete sind **kostenlos!!**

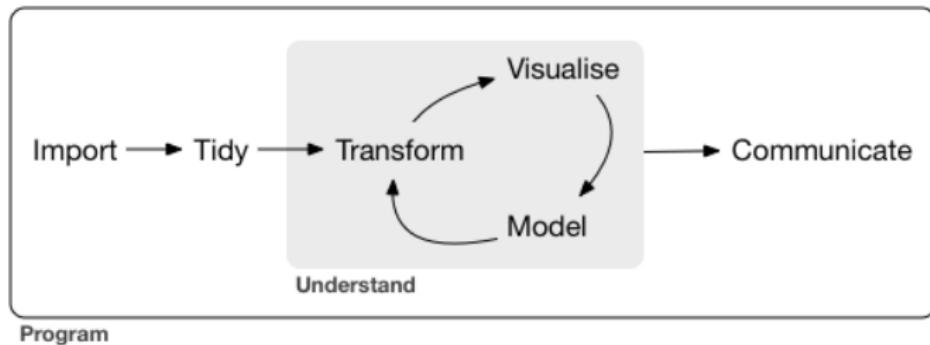
RStudio

Grafische Benutzeroberfläche und Entwicklungsumgebung für R



R for Data Science, Hadley Wickham & Garrett Grolemund 2016

Data Science



R for Data Science, Hadley Wickham & Garrett Grolemund 2016

Weiterführende Literatur

Programmieren mit R

Hands-On Programming with R, Garrett Grolemund

- Kapitel 1, 2, 3, 4, 5, 7, 9

[Link](#)

Die “Bibel” für R User

R for Data Science, Hadley Wickham & Garrett Grolemund

[Link](#)

Kochbuch für R Plots

R Graphics Cookbock, Winston Chang

[Link](#)

R Coding 101

R Architektur

R ist eine Objekt-orientierte Programmiersprache. Es dreht sich daher im Grunde alles darum Objekte herzustellen, zu manipulieren und zu visualisieren.

```
# Zuweisung von Zahlen zu einem Objekt  
alter <- 67
```

R ist *case-sensitive* - Groß- und Kleinschreibung ist wichtig!

```
# Objekt `alter` ausgeben  
alter  
  
## [1] 67  
  
# .. entspricht nicht  
Alter  
  
## Error in eval(expr, envir, enclos): object 'Alter' not found
```

R Objekttypen

Vektor

Def: Sammlung mehrerer Objekte gleicher Art (Länge 1 ist möglich).

```
# numerischer Vektor mit Länge 1  
x1 <- 5  
  
# Charaktervektor mit Länge 1 (Text wird in "" gesetzt)  
x2 <- "green"  
  
# Vektoren mit Länge 3  
y1 <- c(1,3,9)  
y2 <- c("gene1", "gene2", "gene3")
```

Subsetting:

```
# um den vollständigen Vektor y2 auszugeben  
y2  
  
## [1] "gene1" "gene2" "gene3"  
  
# um nur die ersten beiden Einträge des Vektors y2 auszugeben  
y2[1:2]  
  
## [1] "gene1" "gene2"
```

R Objekttypen

Matrix

Kombination mehrerer Vektoren gleichen Typs (numerisch oder Charakter). Die Matrix kann Zeilen- und Spaltennamen haben.

```
matrix <- cbind(y1, y1, y1)
rownames(matrix) <- y2
colnames(matrix) <- c("sample1", "sample2", "sample3")
matrix

##          sample1 sample2 sample3
## gene1      1       1       1
## gene2      3       3       3
## gene3      9       9       9
```

R Objekttypen

Matrix

Ein Subset kann man sich mit der folgenden Syntax anzeigen lassen:

```
matrix[Zeile,Spalte]
```

Bespiele hierfür sind:

```
matrix[1,]
```

```
## sample1 sample2 sample3  
##      1      1      1
```

```
matrix[,3]
```

```
## gene1 gene2 gene3  
##      1      3      9
```

```
matrix[1:2,]
```

```
##          sample1 sample2 sample3  
## gene1      1      1      1  
## gene2      3      3      3
```

R Objekttypen

Dataframe

Im Gegensatz zu Matrizen können in *Dataframes* Vektoren verschiedenen Typs (z.B. numerischer Vektor und Charaktervektor) miteinander kombiniert werden.
Wichtig: Die Vektoren müssen die gleiche Länge haben.

```
df <- data.frame(age=c(50,47,87),  
                  gender=c("male","male","female"))  
df  
  
##   age gender  
## 1  50   male  
## 2  47   male  
## 3  87 female
```

Somit eignen sich *Dataframes* insbesondere für die Analyse von Patientenmetadaten im Rahmen von molekularbiologischen Experimenten oder klinischen Studien.

R Objekttypen

Dataframe

```
df  
  
##   age gender  
## 1  50   male  
## 2  47   male  
## 3  87 female
```

Eine Besonderheit von *Dataframes* ist die Möglichkeit einzelne Spalten durch den Spaltennamen zu selektieren.

```
df$age  
  
## [1] 50 47 87
```

Dies entspricht der folgenden Matrixnotation

```
df[,1]  
  
## [1] 50 47 87
```

R Objekttypen

Vektor



1 Spalte bzw. Reihe
1 Typ (numerisch oder Text)

Matrix



Multiple Spalten / Reihen
1 Typ (numerisch oder Text)

Dataframe



Multiple Spalten / Reihen
mehrere Typen (z.B. numerisch und Text)

Die Funktion `class` ermöglicht es den Typ eines Objektes zu eruieren:

```
class(df)  
  
## [1] "data.frame"
```

R Objekttypen

Funktionen

Die Grundsyntax einer jeden Funktion ist

```
function(Objekt, Argumente)
```

Die Argumente sind hierbei fakultativ. R besitzt eine Vielzahl von Funktionen, ohne dass zusätzliche Pakete geladen werden müssen.

```
sum(y1)
```

```
## [1] 13
```

```
mean(y1)
```

```
## [1] 4.333333
```

Die Funktion `help` öffnet die Dokumentation in RStudio und zeigt die notwendigen Objekte und Argumente zu jeder Funktion an. Als Beispiel, was genau macht die Funktion `cbind`?

```
help(cbind)
```

R Pakete

Ein Paket ist nicht anderes als eine wohldurchdachte Formelsammlung, die für eine spezifische wissenschaftliche Fragestellung (z.B. die Analyse von Sequenzierungsdaten) entwickelt wurde.

Installation

```
install.packages("tidyverse")
```

Ins Environment laden

```
library(tidyverse)
```

Installations- und Kaffeepause

R Datenexploration 101

Bespieldaten des Kurses

Metadaten des *The Cancer Genome Atlas (TCGA)* zur Analyse von Kopf-Hals-Tumoren (head and neck squamous cell carcinoma; HNSCC). Der Datensatz fasst die wichtigsten klinisch-pathologischen Charakteristika der Studienkohorte (n=279) zusammen.

[Link zur Originalpublikation](#)

```
#Datensatz in R laden  
load(url("http://andreasmock.github.io/data/hnscc.RData"))
```

Bespieldaten des Kurses

```
hnscc
```

```
## # A tibble: 279 x 11
##   id      age alcohol days_to_death gender neoplasm_site grade pack_years
##   <chr> <int> <chr>          <int> <chr>    <chr>     <chr>      <dbl>
## 1 TCGA~    69 YES           461 MALE   Oral Tongue  G3       51
## 2 TCGA~    39 YES           415 MALE   Larynx     G2       30
## 3 TCGA~    45 YES          1134 FEMALE Base of Tong~ G2       30
## 4 TCGA~    83 NO            276 MALE   Larynx     G2       75
## 5 TCGA~    47 YES           248 MALE   Floor of Mou~ G2       60
## 6 TCGA~    72 YES           190 MALE   Buccal Mucosa G1       20
## 7 TCGA~    56 YES           845 MALE   Alveolar Rid~ G2       NA
## 8 TCGA~    51 YES          1761 MALE   Tonsil      G2       NA
## 9 TCGA~    54 YES           186 MALE   Larynx     G2       62
## 10 TCGA~   58 YES          179 FEMALE Floor of Mou~ G3       60
## # ... with 269 more rows, and 3 more variables: tabacco_group <chr>,
## #   tumor_stage <chr>, vital_status <chr>
```

Exploration des hnscc Datensatzes

```
colnames(hnscc)

## [1] "id"          "age"         "alcohol"      "days_to_death"
## [5] "gender"       "neoplasm_site" "grade"        "pack_years"
## [9] "tabacco_group" "tumor_stage"   "vital_status"

head(hnscc$age)

## [1] 69 39 45 83 47 72

summary(hnscc$age)

##    Min. 1st Qu. Median    Mean 3rd Qu.    Max.
## 19.00  53.00  61.00  61.32  69.00  90.00
```

Exploration des hnscc Datensatzes

```
head(hnscc$alcohol)

## [1] "YES" "YES" "YES" "NO"  "YES" "YES"

table(hnscc$alcohol)

##
##  NO YES
##  85 188

table(is.na(hnscc$alcohol))

##
## FALSE  TRUE
## 273     6
```

Exploration des hnscc Datensatzes

```
summary(hnscc$days_to_death)
```

```
##      Min. 1st Qu. Median     Mean 3rd Qu.    Max.    NA's
##      0.0   218.8  443.0   789.0  999.2 6416.0       1
```

```
table(hnscc$gender)
```

```
##
## FEMALE    MALE
##    76     203
```

```
table(hnscc$neoplasm_site)
```

```
##
## Alveolar Ridge Base of Tongue Buccal Mucosa Floor of Mouth Hard Palate
##                 7             12            8            26            5
## Hypopharynx          Larynx          Lip        Oral Cavity Oral Tongue
##                 2              72            1            49            76
## Oropharynx           Tonsil
##                 2              19
```

```
table(hnscc$grade)
```

```
##
## G1  G2  G3  G4  GX
## 23 176  71   1   8
```

Exploration des hnscc Datensatzes

```
summary(hnscc$pack_years)

##      Min.    1st Qu.     Median      Mean    3rd Qu.      Max.    NA's
## 0.01685 30.00000 45.00000 50.62485 60.00000 300.00000      125

table(hnscc$tabacco_group)

##
## Current reformed smoker for < or = 15 years
##                                         81
## Current reformed smoker for > 15 years
##                                         49
## Current smoker
##                                         90
## Lifelong Non-smoker
##                                         52
```

Exploration des hnscc Datensatzes

```
table(hnscc$tumor_stage)

##
##    Stage I   Stage II Stage III Stage IVA Stage IVB
##        14         44       38      139        5

table(hnscc$vital_status)

##
## DECEASED    LIVING
##     116      163
```

Funktionen zur Exploration von Datensätzen

```
# Tabelle von kategorialen Daten





```

R Plotting 101

ggplot2 Paket

Visualisierungen mit dem ggplot2 Paket (Teil des tidyverse Pakets) neuer Standard in R.

Prinzip: Malen eines Gemäldes - Schicht für Schicht.

Metadaten des Bespieldatensatzes, die wir explorieren können:

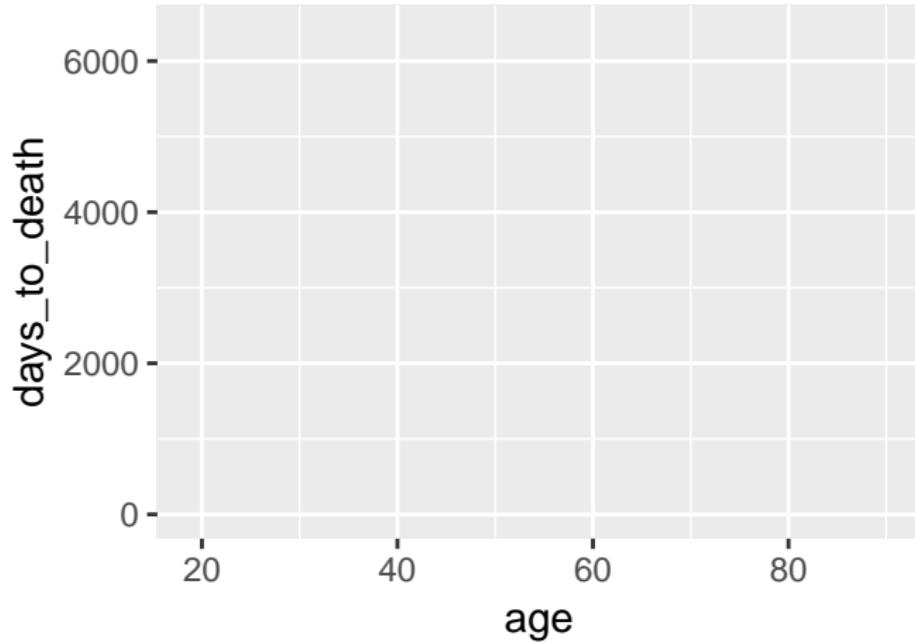
```
colnames(hnscc)

## [1] "id"          "age"         "alcohol"      "days_to_death"
## [5] "gender"       "neoplasm_site" "grade"        "pack_years"
## [9] "tabacco_group" "tumor_stage"   "vital_status"
```

Funktionsweise der ggplot Funktion

Leere Leinwand. age auf der x-Achse und days_to_death auf der y-Achse.

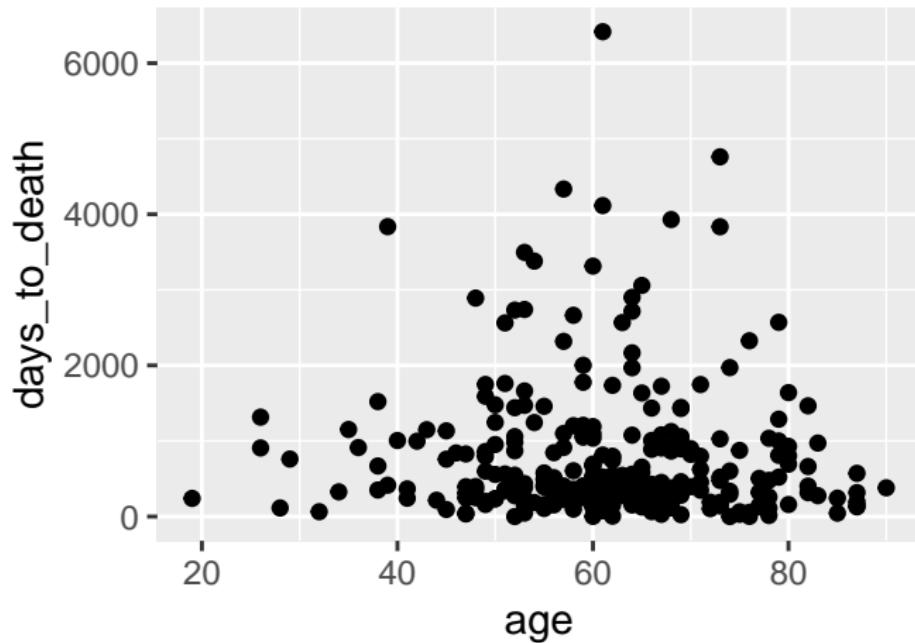
```
ggplot(hnscc, aes(x=age, y=days_to_death))
```



Funktionsweise der ggplot Funktion

Dotplot

```
ggplot(hnscc, aes(x=age, y=days_to_death)) +  
  geom_point()
```



Fragen?