

GESIS-Workshop

"Datenanalyse mit R"

Erste Schritte

Jan-Philipp Kolb

Dienstag, 20. Mai, 2014



Gliederung

R kam, sah und blieb

Warum R nutzen?

Was ist eigentlich R?

Dein Freund das GUI

Modularer Aufbau

Wie bekommt man Hilfe?

Grundlagen im Umgang mit der Sprache R

R ist eine Objekt-orientierte Sprache

Indizieren

Rein und raus – Datenimport und -export

Datemimport

Datenexport

Ein erster Eindruck – Was uns die Daten sagen

Häufigkeiten und gruppierte Kennwerte

Die apply-Familie

Gliederung

R kam, sah und blieb

Warum R nutzen?

Was ist eigentlich R?

Dein Freund das GUI

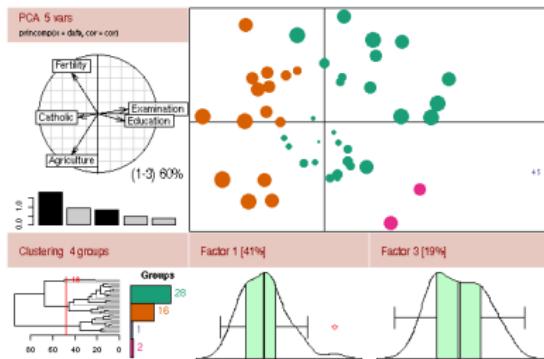
Grundlagen im Umgang mit der Sprache R

Rein und raus – Datenimport und -export

Ein erster Eindruck – Was uns die Daten sagen

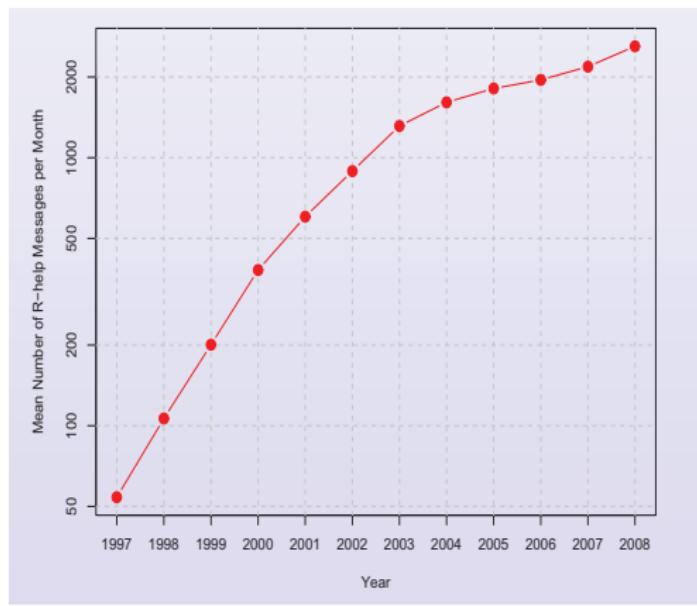
Open Source Programm R

- ▶ R ist eine freie, nicht-kommerzielle Implementierung der Programmiersprache S (von AT&T Bell Laboratories entwickelt)
- ▶ Freie Beteiligung ⇒ modularer Aufbau (immer mehr Erweiterungspakete)



www.r-project.org

Anzahl der Fragen in Hilfeforen zu R



└ R kam, sah und blieb

└ Warum R nutzen?

Warum R nutzen?

- ▶ Als Weg kreativ zu sein ...
- ▶ Graphiken, Graphiken, Graphiken
- ▶ In Kombination mit anderen Programmen nutzbar
- ▶ Zur Verbindung von Datenstrukturen
- ▶ Zum Automatisieren
- ▶ Um die Intelligenz anderer Leute zu nutzen ;-)
- ▶ ...

└ R kam, sah und blieb

└ Warum R nutzen?

R lässt sich kombinieren...

The image displays five separate screenshots arranged in a grid-like fashion, each showing a different way R can be integrated with other statistical software:

- IBM SPSS Statistics Essentials for R**: A screenshot of a web-based interface titled "IBM SPSS Statistics Essentials for R". It features a red header with "Use R!" and a yellow main content area with the title "R Through Excel". Below the main content, there's a sidebar for "Open Source Software" with links for "Users", "Download IBM SPSS Statistics Essentials for R files", "Donate money", "Project detail and discuss", and "Get support".
- R-Forge**: A screenshot of the R-Forge logo, which consists of a stylized "R" icon followed by the word "Forge" in a bold, sans-serif font.
- rPython R package**: A screenshot of the "rPython R package" logo, featuring a blue and white design with the text "rPython R package" and "Statistics and Computing".
- SASmixed**: A screenshot of the "SASmixed" logo, which features a large, stylized "R" icon followed by the word "mixed" in a bold, sans-serif font.
- R for Stata Users**: A screenshot of the "R for Stata Users" logo, featuring a blue and yellow design with the text "R for Stata Users" and "Statistics and Computing".

└ R kam, sah und blieb

└ Warum R nutzen?

R lässt sich kombinieren...

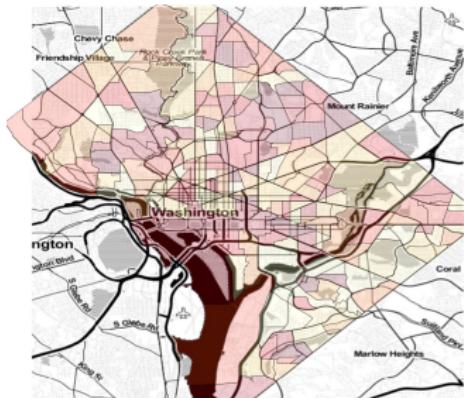
The collage consists of five separate images arranged in a grid-like layout:

- Top Left:** A book cover titled "R Through Excel" by Richard M. Heiberger and Erich Neuwirth. The cover has a red top section with the title and authors' names, and a yellow bottom section with the subtitle "R Through Excel".
- Middle Left:** A screenshot of a website for "IBM SPSS Statistics Essentials for R". It features a sidebar with "Open Source Software" and a main content area with sections for "Users", "Download IBM SPSS Statistics Essentials for R files", "Donate money", "Project detail and discuss", and "Get support".
- Middle Center:** The "R-Java" logo, which consists of the R logo followed by the word "Java".
- Middle Right:** The "SASmixed" logo, which consists of the SAS logo followed by the word "mixed".
- Bottom Right:** A book cover titled "R for Stata Users" by Robert A. Muenchen and Joseph M. Hilbe. The cover has a blue top section with the subtitle "Statistics and Computing" and a yellow bottom section with the title.

Aber das ist nicht Thema des Workshops...

└ R kam, sah und blieb

└ Warum R nutzen?



Möglichkeit schöne Karten zu gestalten

- ▶ <http://stathack.wordpress.com/>
- ▶ Mehr Beispiele unter:
www.r-bloggers.com

Erwartungen und Anforderungen

Das kann diese Schulung vermitteln:

- ▶ Eine praxisnahe Einführung in die statistische Programmiersprache **R**
- ▶ Erlernen einer Programmier-Strategie

Erwartungen und Anforderungen

Das kann diese Schulung vermitteln:

- ▶ Eine praxisnahe Einführung in die statistische Programmiersprache **R**
- ▶ Erlernen einer Programmier-Strategie
- ▶ „Guten Stil“

Erwartungen und Anforderungen

Das kann diese Schulung vermitteln:

- ▶ Eine praxisnahe Einführung in die statistische Programmiersprache **R**
- ▶ Erlernen einer Programmier-Strategie
- ▶ „Guten Stil“
- ▶ Die Vorzüge grafischer Datenanalyse

Erwartungen und Anforderungen

Das kann diese Schulung vermitteln:

- ▶ Eine praxisnahe Einführung in die statistische Programmiersprache **R**
- ▶ Erlernen einer Programmier-Strategie
- ▶ „Guten Stil“
- ▶ Die Vorzüge grafischer Datenanalyse

Erwartungen und Anforderungen

Das kann sie nicht leisten:

- ▶ Eine Einführungsveranstaltung in die Statistik geben
- ▶ Grundlegende datenanalytische Konzepte vermitteln

Erwartungen und Anforderungen

Das kann sie nicht leisten:

- ▶ Eine Einführungsveranstaltung in die Statistik geben
- ▶ Grundlegende datenanalytische Konzepte vermitteln
- ▶ Verständnis zementieren

Erwartungen und Anforderungen

Das kann sie nicht leisten:

- ▶ Eine Einführungsveranstaltung in die Statistik geben
- ▶ Grundlegende datenanalytische Konzepte vermitteln
- ▶ Verständnis zementieren
- ▶ Das „Trainieren“ abnehmen

Erwartungen und Anforderungen

Das kann sie nicht leisten:

- ▶ Eine Einführungsveranstaltung in die Statistik geben
- ▶ Grundlegende datenanalytische Konzepte vermitteln
- ▶ Verständnis zementieren
- ▶ Das „Trainieren“ abnehmen

- └ R kam, sah und blieb
- └ Was ist eigentlich R?

Wo kommen wir her und wo gehen wir hin?

„R is a **language** and environment for **statistical computing** and **graphics**. It is a **GNU** project which is similar to the **S language** and environment (...),“

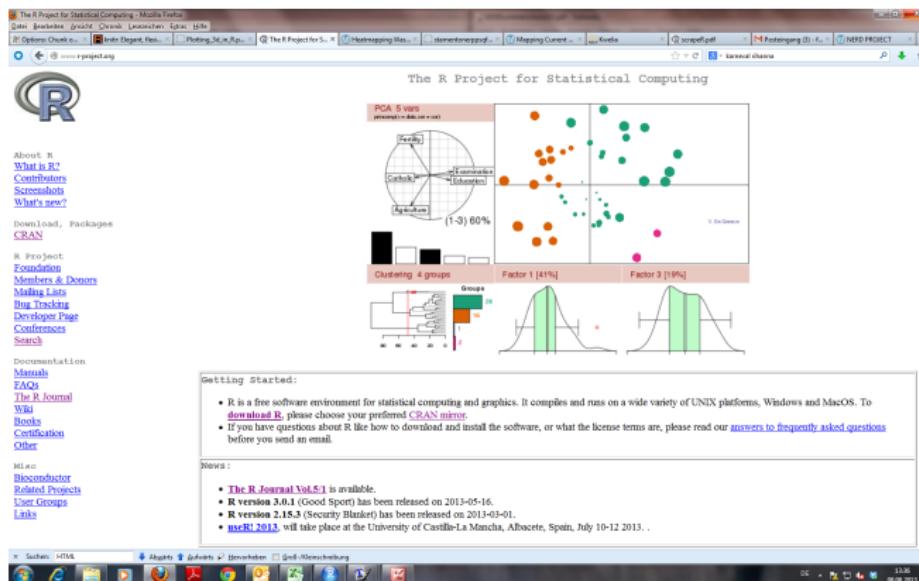
„R provides a wide variety of **statistical** (...) and **graphical techniques**, and is **highly extensible**. (...) R provides an **Open Source** route to participation in that activity.“

„One of R's strengths is the ease with which well-designed **publication-quality plots** can be produced (...) the user retains full control.“

(<http://www.r-project.org/about.html>)

- └ R kam, sah und blieb
- └ Was ist eigentlich R?

R herunterladen:



<http://www.r-project.org/>

- └ R kam, sah und blieb
- └ Was ist eigentlich R?

Das ist das Basis-R:

The screenshot shows the RGui (64-bit) application window. The title bar reads "RGui (64-bit)". The menu bar includes "Datei", "Bearbeiten", "Pakete", "Windows", and "Hilfe". The toolbar contains icons for file operations like Open, Save, and Print. The left pane is titled "R Namenslos - R Editor" and contains the text "# This is the script". The right pane is titled "R Console" and displays the following R session output:

```
R version 2.15.1 (2012-06-22) -- "Roasted I
Copyright (C) 2012 The R Foundation for St
ISBN 3-900051-07-0
Platform: x86_64-pc-mingw32/x64 (64-bit)

R ist freie Software und kommt OHNE JEGLIC
Sie sind eingeladen, es unter bestimmten B
Tippen Sie 'license()' or 'licence()' für

R ist ein Gemeinschaftsprojekt mit vielen
Tippen Sie 'contributors()' für mehr Infor
um zu erfahren, wie R oder R packages in P

Tippen Sie 'demo()' für einige Demos, 'hel
'help.start()' für eine HTML Browserschnit
Tippen Sie 'q()', um R zu verlassen.

> |
```

The taskbar at the bottom shows various application icons, and the system tray indicates the date and time as "DE 07.08.2013 15:22".

Gliederung

R kam, sah und blieb

Dein Freund das GUI

Modularer Aufbau

Wie bekommt man Hilfe?

Grundlagen im Umgang mit der Sprache R

Rein und raus – Datenimport und -export

Ein erster Eindruck – Was uns die Daten sagen

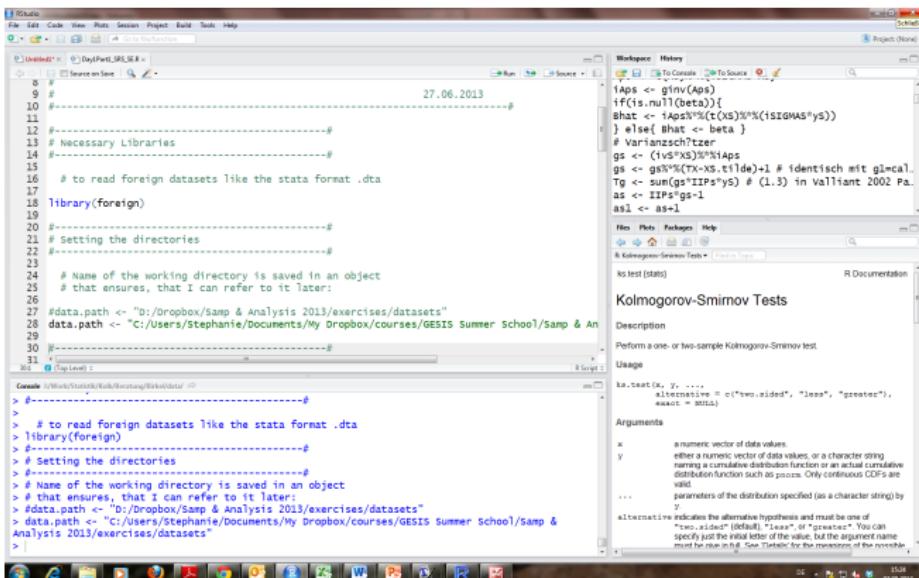
Aber die meisten Menschen nutzen einen Editor oder ein *graphical user interface (GUI)*

Aus den folgenden Gründen:

- ▶ Syntax highlighting
- ▶ Auto-Vervollständigung
- ▶ Bessere Übersicht über Graphiken, Bibliotheken

- ▶ Gedit mit R-spezifischen Add-ons für Linux
<https://projects.gnome.org/gedit/>
- ▶ Emacs
<http://www.gnu.org/software/emacs/>
- ▶ TinnR
<http://www.sciviews.org/Tinn-R/>
- ▶ ...

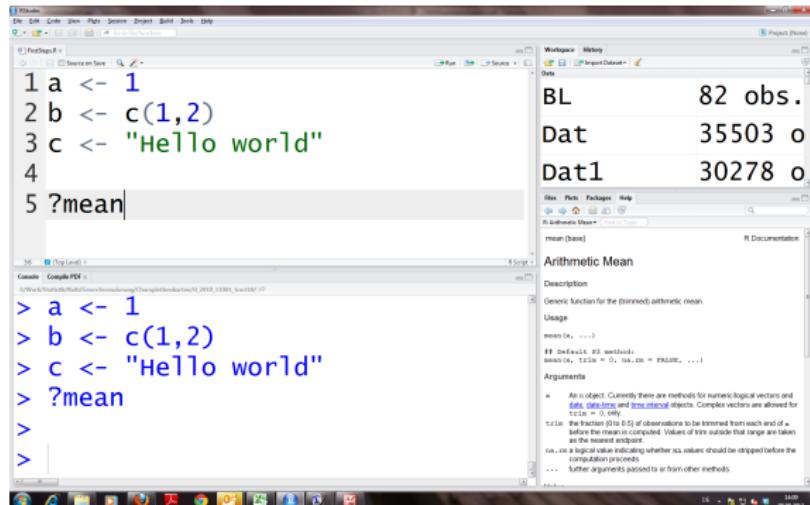
Ich nutze Rstudio!



<http://www.rstudio.com/>

Aufgabe 1 - Vorbereitung

- ▶ Prüfen Sie, ob eine Version von R auf Rechner installiert ist.
- ▶ Falls dies nicht der Fall ist, laden Sie R unter r-project.org runter und installieren Sie R.
- ▶ Prüfen Sie, ob Rstudio installiert ist.
- ▶ Falls nicht → Installieren: <http://www.rstudio.com/>.



Die verschiedenen Fenster in Rstudio

The screenshot shows the RStudio interface. On the left, the script editor contains the following R code:

```
1 a <- 1
2 b <- c(1,2)
3 c <- "Hello world"
4
5 ?mean
```

The line `?mean` is highlighted with a red rectangle. On the right, the help viewer displays the documentation for the `mean` function, specifically the "Arithmetic Mean" section. The help text describes the function as calculating the arithmetic mean of a vector or data frame. It includes examples, usage information, and details about arguments like `na.rm` and `trim`.

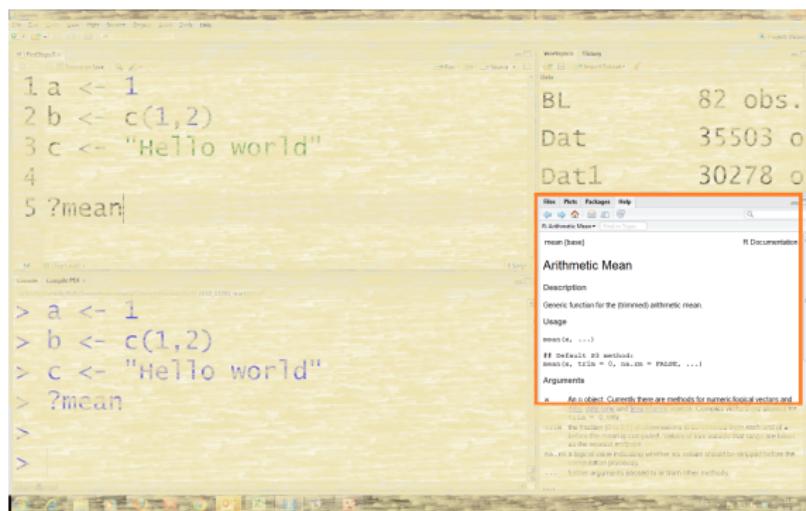
Die verschiedenen Fenster in Rstudio

Das Script



Die verschiedenen Fenster in Rstudio

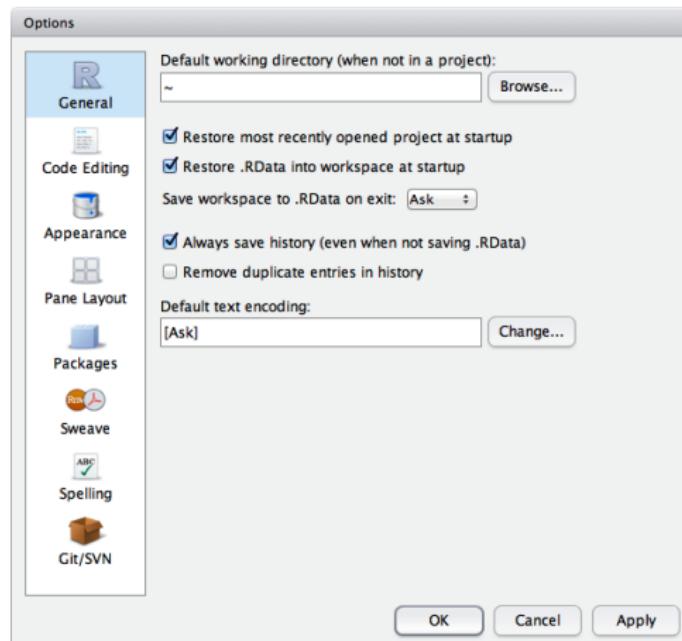
Konsole



Die verschiedenen Fenster in Rstudio

Hilfe

RStudio anpassen



<http://www.rstudio.com/ide/docs/using/customizing>

R als Taschenrechner

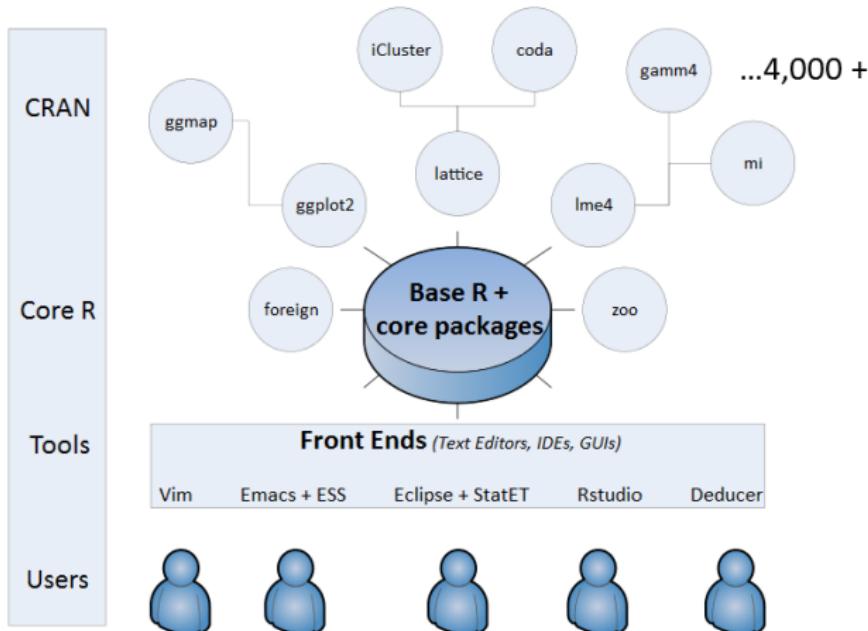
#Grundrechenarten

1+1
2-1
2*2
4/2

#Mathematische Funktionen

`log(4)`
`log(4, base=2)`
`exp(4)`
`sqrt(4)`
`2^4`

Modularer Aufbau



Quelle: <http://www.ats.ucla.edu/stat/r/seminars/intro.htm>

Modularer Aufbau

- ▶ Viele Funktionen sind im Basis-R enthalten
- ▶ Viele spezifische Funktionen sind in zusätzlichen Bibliotheken integriert
- ▶ R kann modular erweitert werden durch sog. **packages** bzw. **libraries**
- ▶ Auf **CRAN** werden die wichtigsten packages gehostet (im Moment 4567)
- ▶ Weitergehende Pakete finden sich z.B. bei www.bioconductor.org

```
install.packages("lme4")
```

```
library(lme4)
```

└ Dein Freund das GUI

└ Modularer Aufbau

Installation von Paketen

The screenshot shows the RStudio interface with several windows open:

- Code Editor:** A script named "paths.R" containing the code `setwd("D:/Projekte/R/packages/germanwebr/Rfunctions")`.
- Environment:** Shows the message "Environment is empty".
- Console:** Displays the message "R is a collaborative project with many contributors. Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in publications."
- Packages:** A window listing available packages with their descriptions and versions. The packages listed are:

Package	Description	Version
AER	Applied Econometrics with R	12.2
arules	Mining Association Rules and Frequent Itemsets	1.1-2
bitops	Bitwise Operations	10-6
boot	Bootstrap Functions (originally by Angelo Canty for S)	1.3-11
brew	Templating Framework for Report Generation	1.0-6
car	Companion to Applied Regression	2.0-19
caTools	Tools: moving window statistics, GIF, Base64, ROC AUC, etc.	1.17
class	Functions for Classification	7.3-10
cluster	Cluster Analysis Extended Rousseeuw et al.	1.15.2
codetools	Code Analysis Tools for R	0.2-8
colorspace	Color Space Manipulation	1.2-4
compiler	The R Compiler Package	3.1.0
DAAG	Data Analysis And Graphics data and functions	1.18

└ Dein Freund das GUI

└ Modularer Aufbau

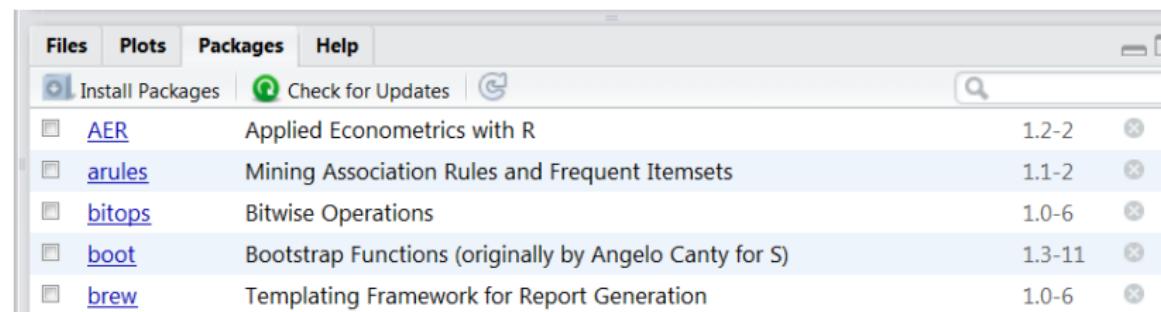
Installation von Paketen

The screenshot shows the RStudio interface with several windows open:

- Code Editor:** A script named "paths.R" containing the code `setwd("D:/Projekte/R/packages/germanwebr/Rfunctions")`.
- Environment:** Shows the message "Environment is empty".
- Console:** Displays the message "R is a collaborative project with many contributors. Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in publications."
- Packages:** A window listing available packages with their descriptions and versions. The packages listed are:

Package	Description	Version
AER	Applied Econometrics with R	12.2
arules	Mining Association Rules and Frequent Itemsets	1.1-2
bitops	Bitwise Operations	10-6
boot	Bootstrap Functions (originally by Angelo Canty for S)	1.3-11
brew	Templating Framework for Report Generation	1.0-6
car	Companion to Applied Regression	2.0-19
caTools	Tools: moving window statistics, GIF, Base64, ROC AUC, etc.	1.17
class	Functions for Classification	7.3-10
cluster	Cluster Analysis Extended Rousseeuw et al.	1.15.2
codetools	Code Analysis Tools for R	0.2-8
colorspace	Color Space Manipulation	1.2-4
compiler	The R Compiler Package	3.1.0
DAAG	Data Analysis And Graphics data and functions	1.18

Installation von Paketen



Aufgabe 2 - Interessante Zusatzpakete

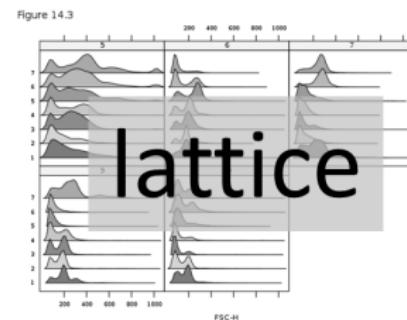
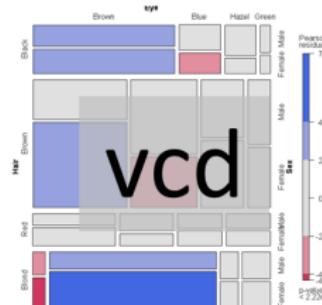
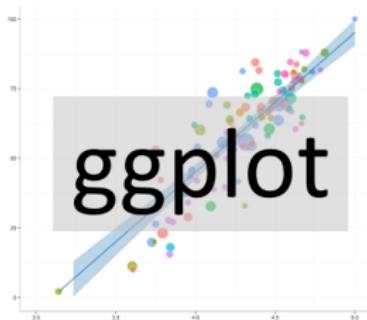
Gehen Sie auf cran.r-project.org und suchen Sie in dem Bereich, wo die Pakete vorgestellt werden, nach Paketen,...

1. die für die deskriptive Datenanalyse geeignet sind.
2. um Regressionen zu berechnen
3. um fremde Datensätze einzulesen (z.B. SPSS-Daten)
4. um mit großen Datenmengen umzugehen

Dein Freund das GUI

Modularer Aufbau

Pakete - deskriptive Datenanalyse



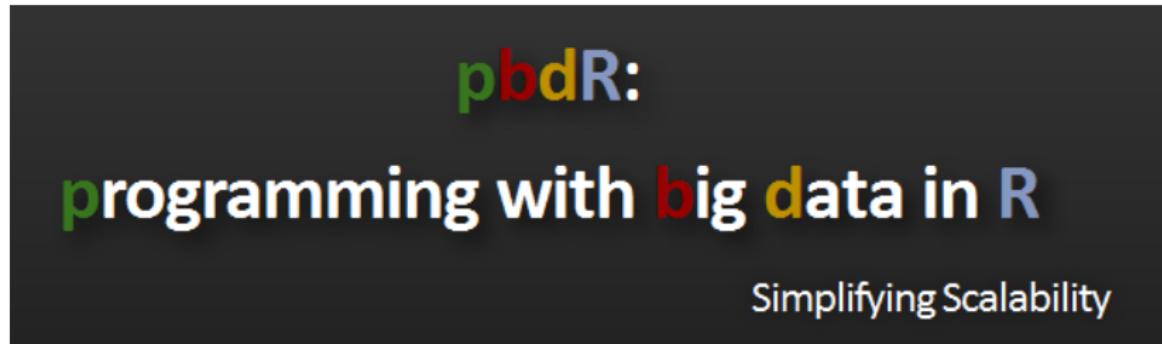
Pakete - Regression

Paket	Für was?
base{lm}	Einfache lineare Regression
base{glm}	Generalisierte Lineare Modelle
tsDyn	Autoregressive Modelle (Zeitreihen)
robustbase	Robuste Regressionen
crs	Nichtparametrische Regression
glmnet	Lasso Verfahren

Paket - fremde Datensätze (foreign)

read.spss
write.dta
read.dta write.dbf
write.foreign
read.octave
read.arff
read.xport
read.mtp
read.epiinfo
read.ssd
write.arff
read.xport
read.arff

Paket - big data (pbdR)



<http://r-pbd.org/>

Wichtige Bibliotheken

Bibliothek	Thema
foreign	Functions for reading and writing data stored by statistical packages
sampling	Functions for drawing and calibrating samples.
survey	Analysis of complex survey samples
MASS	Functions and Datasets for Venables and Ripley's Modern Applied Statistics with S'

Weitere nützliche Bibliotheken

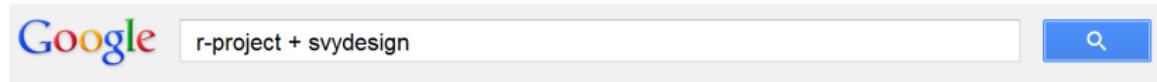
Bibliothek	Thema
xtable	Coerce data to LaTeX and HTML tables
dummies	Expands factors, characters and other eligible classes into dummy/indicator variables.
mvtnorm	Multivariate Normal and t Distributions
maptools	Tools for reading and handling spatial objects

Wie bekommt man Hilfe?

- ▶ Um generell Hilfe zu bekommen: `help.start()`
- ▶ Online Dokumentation für die meisten Funktionen:
`help(name)`
- ▶ Nutze `?` um Hilfe zu bekommen.
Beispiel: `?mean`
- ▶ `example(lm)` gibt ein Beispiel für die lineare Regression

└ Dein Freund das GUI

└ Wie bekommt man Hilfe?



- ▶ Ich nutze meistens google
- ▶ Tippe:
R-project + Was ich schon immer wissen wollte
- ▶ Das funktioniert natürlich mit jeder Suchmaschine!

- └ Dein Freund das GUI
- └ Wie bekommt man Hilfe?

The screenshot shows the top navigation bar of Stack Overflow with links for sign up, log in, tour, help, careers 2.0, and search. Below the header is the Stack Overflow logo and navigation links for Questions, Tags, Tour, and Users. A prominent 'Ask Question' button is on the right. The main content area features a large text block about the site's purpose, a 'Take the 2-minute tour' button, and a 'Here's how it works:' section with three icons: a question mark icon, a green 'A' icon, and a green 'A' icon with arrows.

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free, no registration required.

Here's how it works:

- Anybody can ask a question
- Anybody can answer
- The best answers are voted up and rise to the top

- ▶ <http://stackoverflow.com/>
- ▶ Für Fragen zum Programmieren
- ▶ Ist nicht auf R fokussiert
- ▶ Sehr detaillierte Diskussionen

Gliederung

R kam, sah und blieb

Dein Freund das GUI

Grundlagen im Umgang mit der Sprache R

R ist eine Objekt-orientierte Sprache
Indizieren

Rein und raus – Datenimport und -export

Ein erster Eindruck – Was uns die Daten sagen

Vektoren und Zuweisungen

- ▶ R ist eine Objekt-orientierte Sprache
- ▶ `<-` ist der Zuweisungsoperator
- ▶ `b <- c(1,2)` erzeugt ein Objekt mit den Zahlen 1 und 2
- ▶ Eine Funktion kann auf dieses Objekt angewendet werden:
- ▶ `mean(b)` berechnet den Mittelwert

Mit den folgenden Funktionen können wir etwas über die Eigenschaften des Objekts lernen:

- ▶ `length(b)` - b hat die Länge 2
- ▶ `str(b)` - b ist ein numerischer Vektor

- └ Grundlagen im Umgang mit der Sprache R
- └ R ist eine Objekt-orientierte Sprache

Funktionen im base-Paket

Funktion	Bedeutung	Beispiel
<code>length()</code>	Länge	<code>length(b)</code>
<code>max()</code>	Maximum	<code>max(b)</code>
<code>min()</code>	Minimum	<code>min(b)</code>
<code>sd()</code>	Standardabweichung	<code>sd(b)</code>
<code>var()</code>	Varianz	<code>var(b)</code>
<code>mean()</code>	Mittelwert	<code>mean(b)</code>
<code>median()</code>	Median	<code>median(b)</code>

Diese Funktionen brauchen nur ein Argument.

Andere Funktionen brauchen mehr:

<code>quantile()</code>	90 % Quantile	<code>quantile(b,.9)</code>
<code>sample()</code>	Stichprobe ziehen	<code>sample(b,1)</code>

Eine einführende Übersicht findet man unter:

<http://cran.r-project.org/doc/manuals/R-intro.html>

2.1 Vectors and assignment

R operates on named *data structures*. The simplest such structure is the numeric *vector*, which is a single entity consisting of an ordered collection of numbers. To set up a vector named `x`, say, consisting of five numbers, namely 10.4, 5.6, 3.1, 6.4 and 21.7, use the R command

```
> x <- c(10.4, 5.6, 3.1, 6.4, 21.7)
```

This is an *assignment* statement using the *function* `c()` which in this context can take an arbitrary number of vector *arguments* and whose value is a vector got by concatenating its arguments end to end.⁶

Aufgabe 3 - Zuweisungen und Funktionen

Erzeugen Sie einen Vektor b mit den Zahlen von 1 bis 5 und berechnen Sie...

1. den Mittelwert
2. die Varianz
3. die Standardabweichung
4. die quadratische Wurzel aus dem Mittelwert

- └ Grundlagen im Umgang mit der Sprache R
- └ R ist eine Objekt-orientierte Sprache

Verschiedene Datentypen

Datentyp	Beschreibung	Beispiel
numeric	ganze und reelle Zahlen	5, 3.462
logical	logische Werte	FALSE, TRUE
character	Buchstaben und Zeichenfolgen	"Hallo"

Quelle: R. Münnich und M. Knobelspieß (2007): Einführung in das statistische Programm Paket R

Verschiedene Datentypen

```
numeric    b <- c(1,2)
logical    log <- c(T,F)
character  char <- c("Ä","b")
factor     fac <- as.factor(c(1,2))
```

Mit `str()` bekommt man den Objekttyp.

```
> str(fac)
Factor w/ 2 levels "1","2": 1 2
|
```

Indizieren

Indizieren eines Vektors:

```
> A1 <- c(1,2,3,4)
> A1
[1] 1 2 3 4
> A1[1]
[1] 1
> A1[4]
[1] 4
> A1[1:3]
[1] 1 2 3
> A1[-4]
[1] 1 2 3
```

data.frames

Beispieldaten generieren:

```
AGE <- c(20,35,48,12)
SEX <- c("m","w","w","m")
```

Diese beiden Vektoren zu einem data.frame verbinden:

```
Daten <- data.frame(Alter=AGE, Geschlecht=SEX)
```

Anzahl der Zeilen/Spalten herausfinden

```
nrow(Daten) # Zeilen
ncol(Daten) # Spalten
```

Indizieren

Indizieren eines dataframe:

```
> AA <- 4:1
> A2 <- cbind(A1,AA)
> A2[1,1]
A1
 1
> A2[2,]
A1 AA
 2  3
> A2[,1]
[1] 1 2 3 4
> A2[,1:2]
      A1 AA
[1,]  1  4
[2,]  2  3
[3,]  3  2
[4,]  4  1
```

Matrizen und Arrays

- ▶ In Matrizen und Arrays stehen meist nur numerische Werte.
- ▶ Dadurch wird beispielsweise Matrix Multiplikation möglich.
- ▶ Anders als beim data.frame sind mehr als zwei Dimensionen möglich.

```
A <- matrix(seq(1,100), nrow = 4)
dim(A)
```

Indizieren

Indizieren eines array:

```
> A3 <- array(1:8,c(2,2,2))
> A3
, , 1

 [,1] [,2]
[1,]    1    3
[2,]    2    4

, , 2

 [,1] [,2]
[1,]    5    7
[2,]    6    8

> A3[, , 2]
 [,1] [,2]
[1,]    5    7
[2,]    6    8
```

Listen

- ▶ Eine Liste in R entspricht einem geschachtelten Array in anderen Programmiersprachen
- ▶ Listen können alles enthalten
- ▶ Listen können geschachtelt sein
- ▶ Listen sollte man sehr bedacht verwenden

Indizieren

Indizieren einer Liste:

```
> A4 <- List(A1,1)
> A4
[[1]]
[1] 1 2 3 4

[[2]]
[1] 1

> A4[[2]]
[1] 1
```

Logische Operatoren

Operator	Operation
>	größer als
<	kleiner als
==	genau gleich
!=	ungleich (Negation)
>=	größer gleich
<=	kleiner gleich
&	und
	oder
!	Negation
xor	entweder oder

Quelle: R. Münnich und M. Knobelspieß (2007): Einführung in das statistische Programm Paket R

Sequenzen

```
> 1:10
[1] 1 2 3 4 5 6 7 8 9 10
> rep(1,10)
[1] 1 1 1 1 1 1 1 1 1 1
> rep("A",10)
[1] "A" "A" "A" "A" "A" "A" "A" "A" "A" "A"
> seq(-2,8,by=1.5)
[1] -2.0 -0.5 1.0 2.5 4.0 5.5 7.0
```

Die Funktion paste

```
?paste
paste(1:4)
paste("A", 1:6, sep = "")
```

```
> paste(1:4)
[1] "1" "2" "3" "4"
> paste("A", 1:6, sep = "")
[1] "A1" "A2" "A3" "A4" "A5" "A6"
```

Gliederung

R kam, sah und blieb

Dein Freund das GUI

Grundlagen im Umgang mit der Sprache R

Rein und raus – Datenimport und -export

Datemimport

Datenexport

Ein erster Eindruck – Was uns die Daten sagen

Datenimport



Datenzugang

Public-Use-File (PUF)

Datei zur öffentlichen Nutzung -
meist stark anonymisierte Daten
viele Beispiele unter:

www.forschungsdatenzentrum.de

www.statistik-portal.de

[www.infothek.statistik.rlp.de/lis/MeineRegion/
index.asp](http://www.infothek.statistik.rlp.de/lis/MeineRegion/index.asp)

Scientific-Use-File (SUF)

Datei zur wissenschaftlichen Nutzung - anonymisierte
Daten, die zu wissenschaftlichen Zwecken und zur
Sekundäranalyse genutzt werden können.

On-Site-Nutzung

- ▶ Arbeitsplätze für Gastwissenschaftler
- ▶ Kontrollierte Datenfernverarbeitung

Datenquellen:

- ▶ Forschungsdatenzentrum
www.forschungsdatenzentrum.de/
- ▶ GDELT: Global Data on Events, Location and Tone
<http://gdelt.utdallas.edu/>
- ▶ Social security administration puf
<http://www.ssa.gov/policy/docs/data/index.html>
- ▶ National health and nutrition examination survey
library(survey) und data(nhanes)
- ▶ FAO Datenbank
<http://cran.r-project.org/web/packages/FAOSTAT/index.html>

Download Daten

The screenshot shows the header of a German statistics portal. At the top left is the logo of the Federal Statistical Office of Germany. To its right, the text "STATISTISCHE ÄMTER DES BUNDES UND DER LÄNDER FORSCHUNGSDATENZENTREN" is displayed. Below the logo is a horizontal navigation bar with four items: "Startseite", "Impressum", "Wir über uns", and "English".

[Datenangebot](#)

Datenangebot | Mikrozensus 2002

[Datenzugang](#)

[Metadaten | Ansprechpartner](#)

[Nutzungsbedingungen](#)

[Amtliche Firmendaten](#)

[CAMPUS-Files](#)

[Veranstaltungen](#)

[Veröffentlichungen](#)

[Kontakt](#)

CAMPUS-File

Das CAMPUS-File zum Mikrozensus 2002 ist eine 3,5%-Wohnungssstichprobe des Originalmaterials des Mikrozensus 2002, der speziell für Lehr- und Übungszwecke erstellt wurde. Die Daten wurden durch Vergrößerung und Löschung einzelner Merkmale anonymisiert. Weitere Informationen zur Methodik und zur Anonymisierung sind in der Methodenbeschreibung aufbereitet. Das CAMPUS-File wird in den Formaten SPSS, SAS und STATA sowie als ASCII-CSV angeboten.

Die Nutzung dieses CAMPUS-Files ist unentgeltlich.

Fälle	Variablen	Hochrechnung
25.137 Personen 11.655 Haushalte 11.788 Wohnungen	335	Bund, Länder

Metadaten zum Download

Datei	Format	Größe
Daten: SAS (mit Labels)	zip	3.050 KB

Dateiformate in R

- ▶ Von R werden quelloffene, nicht-proprietäre Formate bevorzugt
- ▶ Es können aber auch Formate von anderen Statistik Software Paketen eingelesen werden
- ▶ R speichert Objekte gerne in sog. Workspaces ab
- ▶ Auch hier jedoch gilt: (fast) alles andere ist möglich

Formate - base package

- ▶ R unterstützt von Haus aus schon einige wichtige Formate:
 - ▶ CSV (Comma Separated Values): `read.csv()`
 - ▶ FWF (Fixed With Format): `read.fwf()`
 - ▶ Tab-getrennte Werte: `read.delim()`

CSV Dateien einlesen

Liegt ein ordentlich formatierter CSV Datensatz vor, kann er mit dem `read.csv()` Befehl eingelesen werden

Datenimport mit `read.csv()`

```
read.csv(file, header = TRUE, sep = ",",
         quote = "\"", dec = ".", fill = TRUE,
         comment.char = "", ...)

read.csv2(file, header = TRUE, sep = ";",
          quote = "\"", dec = ",", fill = TRUE,
          comment.char = "", ...)

MZ02 <- read.csv2(file='MZ02.csv')
```

Import von Excel-Daten

- ▶ `library(foreign)` ist für den Import von fremden Datenformaten nötig
- ▶ Wenn Excel-Daten vorliegen - als .csv abspeichern
- ▶ Dann kann `read.csv()` genutzt werden um die Daten einzulesen.
- ▶ Bei Deutschen Daten kann es sein, dass man `read.csv2()` wegen der Komma-Separierung braucht.

Der Arbeitsspeicher

So findet man heraus, in welchem Verzeichnis man sich gerade befindet

```
getwd()
```

So kann man das Arbeitsverzeichnis ändern:

Man erzeugt ein Objekt in dem man den Pfad abspeichert

```
main.path <- "C:/"
```

Und ändert dann den Pfad mit setwd():

```
setwd(main.path)
```

Wichtig ist es Slashes anstelle von Backslashes zu verwenden.

SPSS Dateien einlesen

- ▶ Das (ebenfalls proprietäre) .sav Format lässt sich mit der Funktion `read.spss()` aus dem Paket `foreign` lesen
- ▶ Es müssen bei `read.spss()` allerdings einige Argumente gesetzt werden
- ▶ `to.data.frame=TRUE` sorgt dafür, dass ein Data Frame entsteht (default ist `FALSE`)
- ▶ `use.value.labels=FALSE` verwendet numerische Werte anstelle von Labels (default ist `TRUE`)

SPSS mit `read.spss()`

```
library(foreign)
?read.spss
MZ02 <- read.spss(file="MZ02.sav",
                    to.data.frame=TRUE,
                    use.value.labels=FALSE)
```

STATA Dateien einlesen

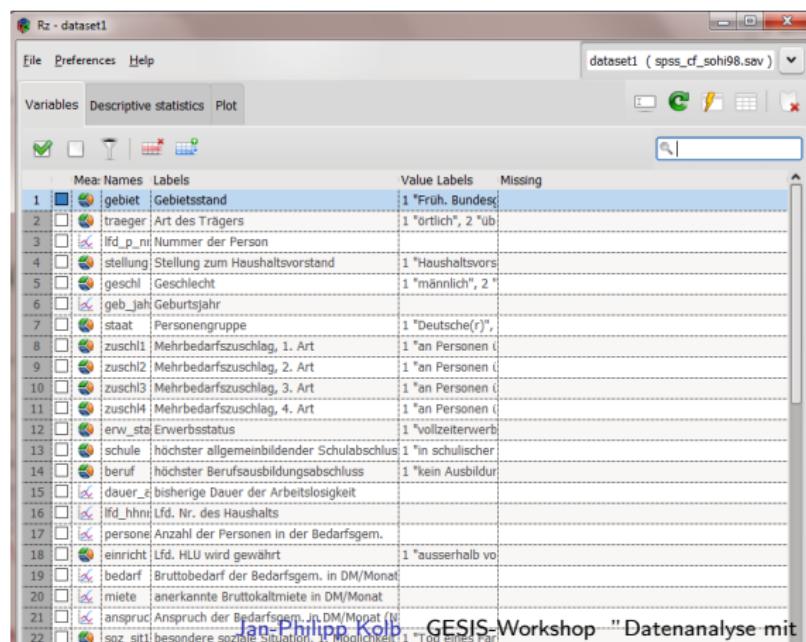
Mit dem `foreign` paket können auch STATA (.dta) Dateien eingelesen werden:

STATA mit `read.dta()`

```
?read.dta  
MZ02 <- read.dta(file="MZ02.dta")
```

Datenmanagement wie in SPSS oder Stata

```
install.packages("Rz")
library(Rz)
```



The screenshot shows the Rz software interface with the title bar "Rz · dataset1". The window contains a dataset viewer with the following columns:

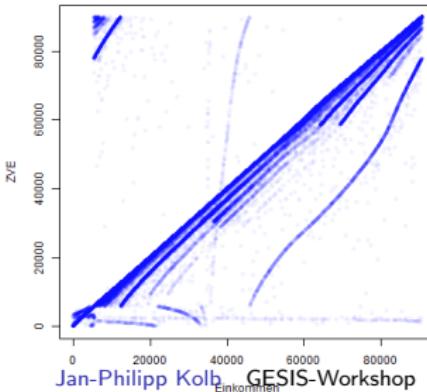
	Mea: Names	Labels	Value Labels	Missing
1	[checkbox] gebiet	Gebietsstand	1 "Früh, Bundes"	
2	[checkbox]	traeger	1 "örtlich", 2 "üb."	
3	[checkbox]	lfd_p_nr	Nummer der Person	
4	[checkbox]	stellung	Stellung zum Haushaltsvorstand	1 "Haushaltsvorstand"
5	[checkbox]	geschi	Geschlecht	1 "männlich", 2 "
6	[checkbox]	geb_jahr	Geburtsjahr	
7	[checkbox]	staat	Personengruppe	1 "Deutsche(r)",
8	[checkbox]	zuschl1	Mehrbedarfszuschlag, 1. Art	1 "an Personen",
9	[checkbox]	zuschl2	Mehrbedarfszuschlag, 2. Art	1 "an Personen",
10	[checkbox]	zuschl3	Mehrbedarfszuschlag, 3. Art	1 "an Personen",
11	[checkbox]	zuschl4	Mehrbedarfszuschlag, 4. Art	1 "an Personen",
12	[checkbox]	erv_sta	Erwerbsstatus	1 "vollzeiterwerb",
13	[checkbox]	schule	höchster allgemeinbildender Schulabschluss	1 "in schulischer",
14	[checkbox]	beruf	höchster Berufsausbildungsabschluss	1 "kein Ausbildung",
15	[checkbox]	dauer_e	bisherige Dauer der Arbeitslosigkeit	
16	[checkbox]	lfd_hhnr	Lfd. Nr. des Haushalts	
17	[checkbox]	personen	Anzahl der Personen in der Bedarfsgem.	
18	[checkbox]	einricht	Lfd. HLU wird gewährt	1 "ausserhalb vo",
19	[checkbox]	bedarf	Bruttobedarf der Bedarfsgem. in DM/Monat	
20	[checkbox]	miete	anerkannte Bruttokaltmiete in DM/Monat	
21	[checkbox]	anspruc	Anspruch der Bedarfsgem. Jp. DM/Monat (W)	
22	[checkbox]	soz_situ	besondere soziale Situation, 1 Möglichkeit, 1 Tod eines Par	

Aufgabe 4 - Datenimport

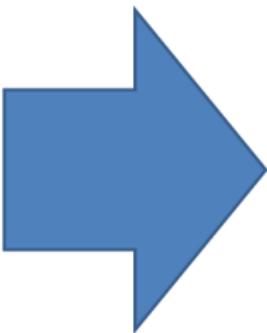
- ▶ Gehen Sie auf die Seite des Forschungsdatenzentrums und laden Sie den Campusfile des Mikrozensus 2002 herunter
- ▶ Laden Sie den Datensatz mit einer geeigneten Funktion in Ihren Workspace.
- ▶ Finden Sie heraus, wieviele Beobachtungen und Variablen der Datensatz umfasst.

Warum nicht ein Datensatz mit dem die ganze Zeit gearbeitet wird?

- ▶ Viele Datensätze um zu zeigen, was mit R alles möglich ist
- ▶ Jeder Datensatz ist anders und es erfordert Zeit sich einzuarbeiten
- ▶ Für Scientific-Use-Files muss extra ein Antrag gestellt werden
- ▶ Bei Public-Use-Files gehen oft sinnvolle Zusammenhänge verloren



Datenexport



R's Exportformate

- ▶ In R werden offene Dateiformate bevorzugt
- ▶ Als Äquivalenz zu den `read.X()` Funktionen stehen viele `write.X()` Funktionen zur Verfügung
- ▶ Das eigene Format von R sind sog. Workspaces (`.RData`)

Einfache Textdateien schreiben

- ▶ Mit Hilfe von `write.table()` lässt sich ein Data Frame sehr einfach in eine Textdatei schreiben
- ▶ Einige Argumente können jedoch angepasst werden
- ▶ Als Trenner sollte ein Tabulator dienen (default ist Leerzeichen)
- ▶ Zeilennamen sollten nicht geschrieben werden (default ist TRUE)

`write.table()`

```
?write.table
write.table(MZ02,file="MZ02.txt",
            sep="\t", na="", dec=",", row.names=FALSE)
?write.csv
```

Objekt(e) in R-Workspaces sichern

- ▶ In einem Workspace kann ein oder mehrere Objekte gespeichert werden
- ▶ Workspaces sind komprimierte Binärdateien (sehr klein)
- ▶ Der Befehl `save.image()` speichert alle Objekte in einem Workspace
- ▶ Dem Befehl `save()` kann eine Liste von Objekten (auch nur eines) übergeben werden

`save.image()` und `save()`

```
?save.image
save.image(file = ".../data/Day01.RData")
```

```
?save
save(MZ02, file = "MZ02.RData")
```

Überblick Daten Import/Export

R Data Import/Export

Version 3.1.0 (2014-04-10)

<http://cran.r-project.org/doc/manuals/r-release/R-data.pdf>

Gliederung

R kam, sah und blieb

Dein Freund das GUI

Grundlagen im Umgang mit der Sprache R

Rein und raus – Datenimport und -export

Ein erster Eindruck – Was uns die Daten sagen

Häufigkeiten und gruppierte Kennwerte

Die apply-Familie

Streuungsmaße

Im base Package sind die wichtigsten Streuungsmaße enthalten:

- ▶ Varianz: `var()`
- ▶ Standardabweichung: `sd()`
- ▶ Minimum und Maximum: `min()` und `max()`
- ▶ Range: `range()`

Fehlende Werte

- Sind NAs vorhanden muss dies der Funktion mitgeteilt werden

```
?var  
var(x)  
var(xNA)  
var(xNA, na.rm=TRUE)
```

Häufigkeitstabellen

- ▶ Eine Auszählung der Häufigkeiten der Merkmale einer Variable liefert `table()`
- ▶ Mit `table()` sind auch Kreuztabellierungen möglich indem zwei Variablen durch Komma getrennt werden: `table(x,y)` liefert Häufigkeiten von `y` für gegebene Ausprägungen von `x`

Die Funktion `table()`

```
?table  
table(x)  
table(x, musician)  
data(esoph)  
table(esoph$agegp)
```

Häufigkeitstabellen

- ▶ `prop.table()` liefert die relativen Häufigkeiten
- ▶ Wird die Funktion außerhalb einer `table()` Funktion geschrieben erhält man die relativen Häufigkeiten bezogen auf alle Zellen

Die Funktion `prop.table()`

```
table(esoph$agegp, esoph$alcgp)
?prop.table
prop.table(table(esoph$agegp,
esoph$alcgp), 1)
```

Die aggregate() Funktion

- ▶ Mit der aggregate() Funktion können Kennwerte für Untergruppen erstellt werden
- ▶ `aggregate(x, by, FUN)` müssen mindestens drei Argumente übergeben werden:
 - x:** ein oder mehrere Beobachtungsvektor(en) für den der Kennwert berechnet werden soll
 - by:** eine oder mehrere bedingende Variable(n)
 - FUN:** die Funktion welche den Kennwert berechnet (z.B. `mean` oder `sd`)
- ▶ Die Ausgabe kann mit Hilfe von `xtabs()` in eine schöne zweidimensionale Tabelle überführt werden

Die Funktion apply

apply {base}

R Documentation

Apply Functions Over Array Margins

Description

Returns a vector or array or list of values obtained by applying a function to margins of an array or matrix.

Usage

```
apply(X, MARGIN, FUN, ...)
```

Arguments

X an array, including a matrix.

MARGIN a vector giving the subscripts which the function will be applied over. E.g., for a matrix 1 indicates rows, 2 indicates columns, `c(1, 2)` indicates rows and columns. Where **x** has named dimnames, it can be a character vector selecting dimension names.

FUN the function to be applied: see 'Details'. In the case of functions like `+`, `%*%`, etc., the function name must be backquoted or quoted.

... optional arguments to **FUN**.

Die Funktion apply

Für `margin=1` die Funktion `mean` auf die Reihen angewendet,

Für `margin=2` die Funktion `mean` auf die Spalten angewendet,

```
> ApplyDat <- cbind(1:4,runif(4),rnorm(4))
> apply(ApplyDat,1,mean)
[1] 0.4798562 0.9655396 1.0619841 1.8760724
> apply(ApplyDat,2,mean)
[1] 2.5000000 0.4251074 0.3624818
```

Anstatt `mean` können auch andere Funktionen wie `var`, `sd` oder `length` verwendet werden.

Die Funktion tapply

tapply {base}

R Documentation

Apply a Function Over a Ragged Array

Description

Apply a function to each cell of a ragged array, that is to each (non-empty) group of values given by a unique combination of the levels of certain factors.

Usage

```
tapply(X, INDEX, FUN = NULL, ..., simplify = TRUE)
```

Arguments

- X** an atomic object, typically a vector.
- INDEX** list of one or more factors, each of same length as **x**. The elements are coerced to factors by [as.factor](#).
- FUN** the function to be applied, or **NULL**. In the case of functions like **+**, **%*%**, etc., the function name must be backquoted or quoted. If **FUN** is **NULL**, **tapply** returns a vector which can be used to subscript the multi-way array **tapply** normally produces.
- ...** optional arguments to **FUN**: the Note section.

Die Funktion tapply

```
> ApplyDat <- data.frame(Income=rnorm(5,1400,200),  
+                           Sex=sample(c(1,2),5,replace=T))  
> ApplyDat  
   Income Sex  
1 1230.7846  1  
2  871.6304  1  
3 1454.7069  2  
4 1495.1007  2  
5 1348.5055  1  
> tapply(ApplyDat$Income,ApplyDat$Sex,mean)  
      1       2  
1150.307 1474.904
```

Die Funktion tapply

Auch andere Funktionen können eingesetzt werden....
Auch selbst programmierte Funktionen

```
> ApplyDat
  Income Sex
1 1230.7846  1
2  871.6304  1
3 1454.7069  2
4 1495.1007  2
5 1348.5055  1
> tapply(ApplyDat$Income,ApplyDat$Sex,function(x)x)
$`1`
[1] 1230.7846 871.6304 1348.5055

$`2`
[1] 1454.707 1495.101
```

Im Beispiel wird die einfachste eigene Funktion angewendet.

Aufgabe 5 -apply

1 Aufgabe: Apply-Funktion verwenden

- Erstellen Sie eine Matrix A mit 4 Zeilen und 25 Spalten, die die Werte 1 bis 100 enthält. Analog dazu erstellen Sie eine Matrix B mit 25 Zeilen und 4 Spalten, die die Werte 1 bis 100 enthält.
- Berechnen Sie mittels dem `apply()`-Befehl den Mittelwert und die Varianz für jede Zeile von A bzw. B.
- Berechnen Sie mittels dem `apply()`-Befehl den Mittelwert und die Varianz für jede Spalte von A bzw. B.
- Standardisieren ist eine häufige Transformation von Daten; dafür wird der Mittelwert von der entsprechenden Zeile oder Spalte abgezogen und durch die entsprechende Standardabweichung geteilt. Somit besitzen die Daten einen Mittelwert von 0 und eine Standardabweichung von 1.

Standardisieren Sie die Spalten der Matrix A. Abschließend überprüfen Sie, ob die Spalten richtig standardisiert wurden.

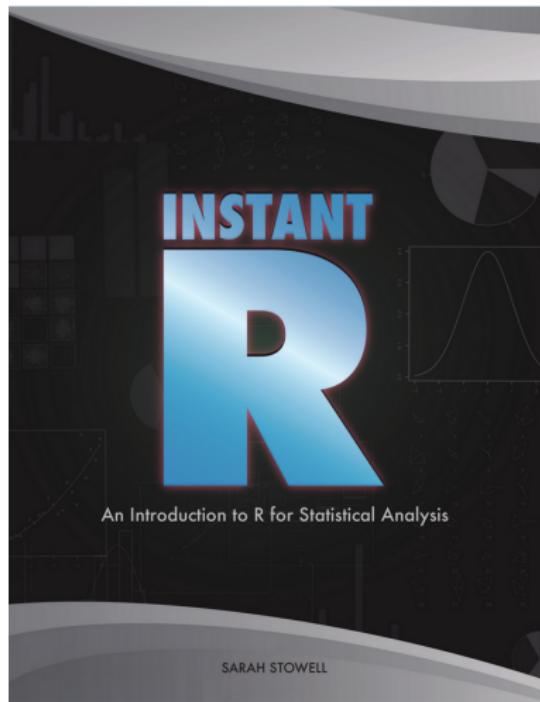
Quelle: <http://www.uni-leipzig.de/~zuber/teaching/ws12/r-kurs/praxis/U2.pdf>

Literatur



- ▶ Ligges, U. (2008):
Programmieren mit R.
Springer.
- ▶ Gut für Anfänger

Literatur



- ▶ Import und Export von Daten
- ▶ Daten editieren
- ▶ Graphiken

Mehr Hilfe für Anfänger



Help the Stat Consulting Group by

[Google™ Custom Search](#)

[giving a gift](#)



stat > r > sk

R Starter Kit

This page is intended for people who:

Are just starting	Have a question or two about	Want a quick refresher
<ul style="list-style-type: none">• to learn R• to utilize basic statistical procedures	<ul style="list-style-type: none">• how to do a simple task in R• how to interpret the output from commonly used procedures	<ul style="list-style-type: none">• on how to do basic tasks in R• on frequently used statistical procedures and the interpretation of their output.

These materials have been collected from various places on our website and have been ordered so that you can, in step-by-step fashion, develop the skills needed to conduct common analyses in R.

Getting familiar with R

- [Class notes](#): There is no point in waiting to take an introductory class on how to use R. Instead, we have notes of our introductory class that you can download and view.
- [Learning modules](#): We have developed a set of web pages called learning modules which show you how to accomplish basic data management tasks in R, including how to get data into R, how to recode variable and how to subset data. The R code and the output produced are shown, as well as tips on things to look out for.

<http://www.ats.ucla.edu/stat/r/sk/>

Hilfe und erste Schritte:

- ▶ Skript für die ersten Schritte:
www.stamats.de/InstallationUndErsteSchritteMitR.pdf

- ▶ Einführung direkt auf CRAN:
cran.r-project.org/doc/contrib/Sawitzki-Einfuehrung.pdf

Vielen Dank für die
Aufmerksamkeit

