

Spatial Visualisations

Why R

Jan-Philipp Kolb

Monday, October 13, 2014



Outline

Why use R

- Interfaces

- Visualisations

Getting started

- How to get help

- Import and Export

- Sampling

- Basic graphics

- Literature

Exercises

- Exercise 1

- Exercise 2

Why use R?

- ▶ You can create your own graphics, objects, functions, and packages.
- ▶ Over 2,000 cutting-edge, user-contributed packages available on CRAN
- ▶ Allows for rapid development of new tools according to user demand

Why R? - its free!

As an open-source project, you can use R free of charge - no worries about:

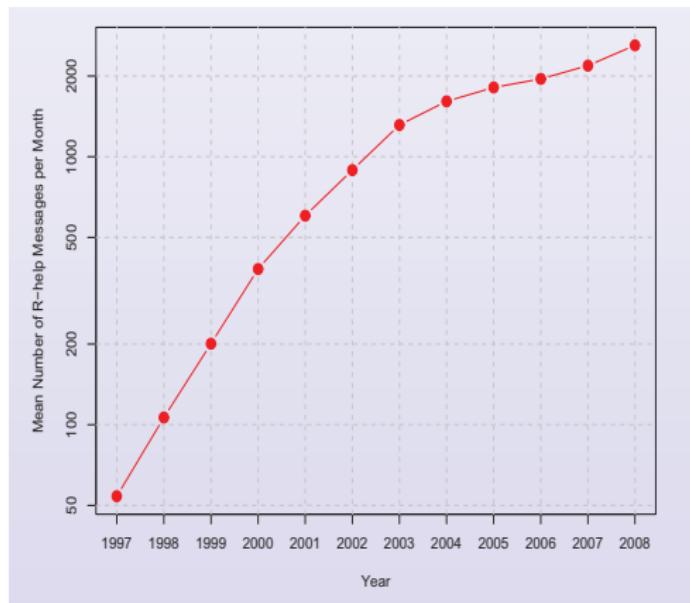
- ▶ subscription fees,
- ▶ license managers,
- ▶ or user limits.

But just as importantly, R is open:

- ▶ You can inspect the code and tinker with it as much as you like
- ▶ Thousands of experts have done just that, and their contributions benefit the millions of people who use R today.

<http://www.inside-r.org/why-use-r>

Number of R-questions in forums



Key features

- ▶ Its a mature, widely used (around 1-2 million users) and well-supported free and open source software project;
- ▶ its committed to an annual schedule for major updates (it runs on GNU/linux, unix, MacOS and Windows)
- ▶ Excellent graphics capabilities
- ▶ Vector arithmetic, plus many built-in basic & advanced statistical and numerical analysis tools
- ▶ Intelligent handling of missing data values (denoted by "NA")
- ▶ Highly extensible, with over 4300 user-contributed packages available
- ▶ Its easy to use and has excellent online help and associated documentation

http://www.sr.bham.ac.uk/~ajrs/R/why_R.html

The Way We Think

- ▶ The language should mirror the way that people think.
- ▶ A simple example: suppose we think that weight is a function of (dependent on) height and birth.

The R formula to express this is:

```
weight ~ height + birth
```

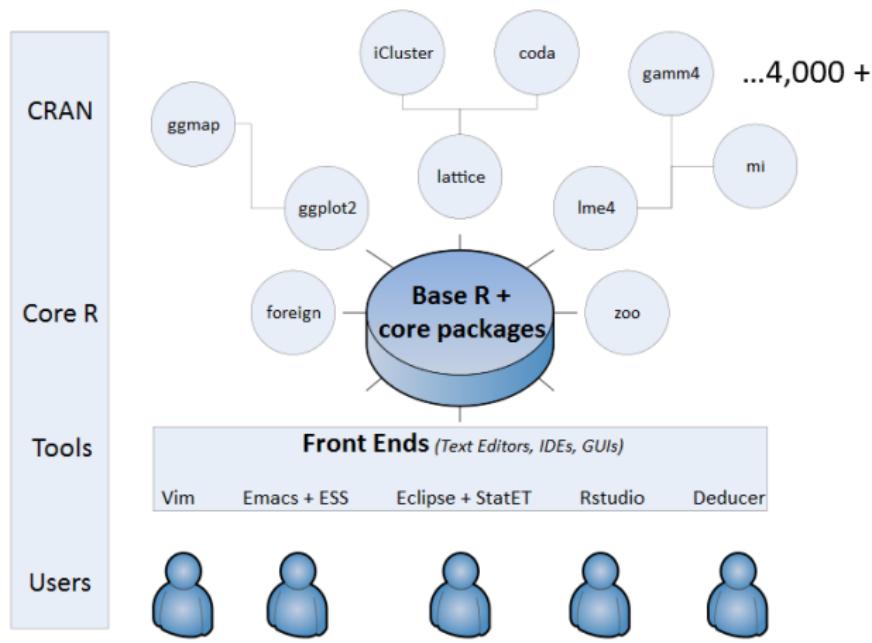
<http://www.burns-stat.com/documents/tutorials/why-use-the-r-language/>

Options for expansion

- ▶ R is a modular program
- ▶ Many functions are included in the basic R
- ▶ But more specific functions are embedded in libraries

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

R is modular



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

Jan-Philipp Kolb Spatial Visualisations

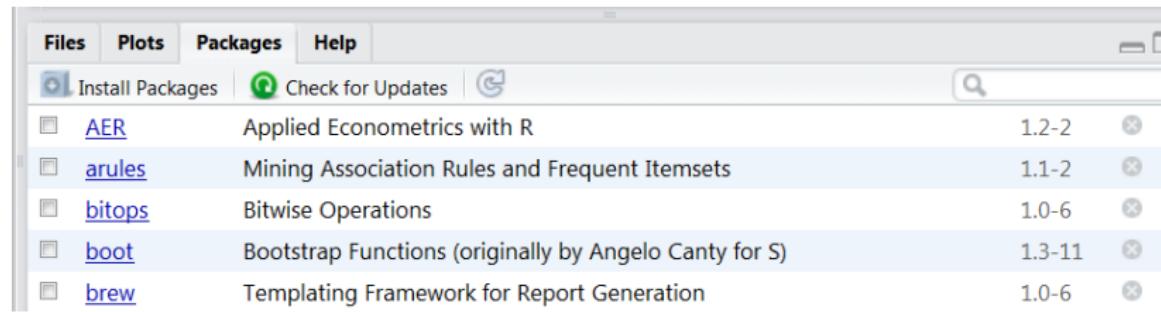
Installation of additional packages

The screenshot shows the RStudio interface with several windows open:

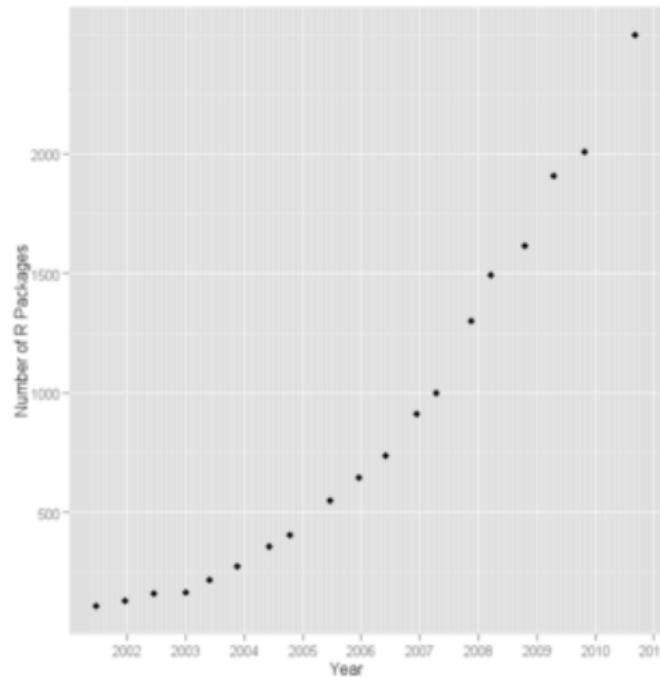
- Code Editor (top-left):** A script named "paths.R" containing the code `setwd("D:/Projekte/Rpackages/germanwebr/Rfunctions")`.
- Environment (top-right):** Shows the message "Environment is empty".
- Console (bottom-left):** Displays the message "R is a collaborative project with many contributors.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

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)	13-11
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 of additional packages



R is modular



<http://blog.revolutionanalytics.com/2010/09/what-can-other-languages-learn-from-r.html>

The most wanted packages



Important libraries

Library	Topic
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'

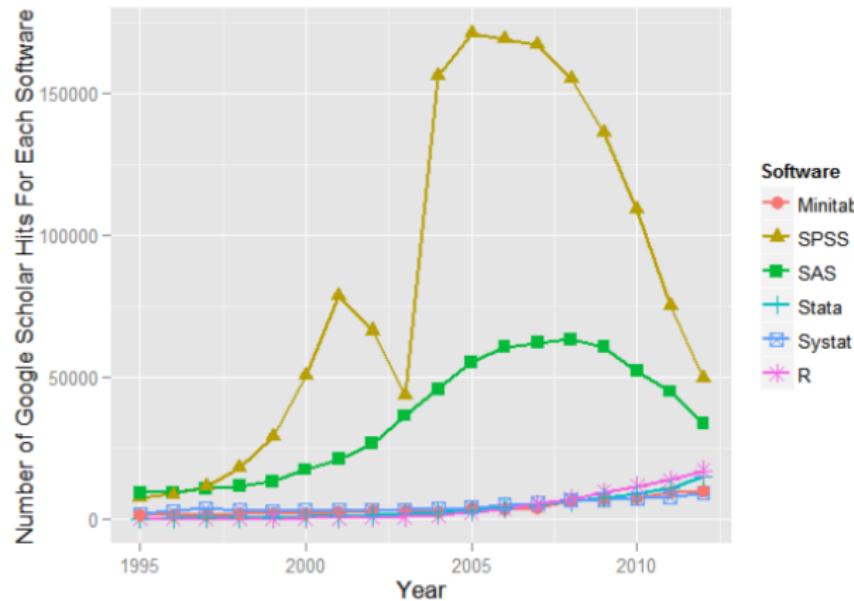
Important libraries for graphics

Library	Topic
lattice	Lattice is a powerful and elegant high-level data visualization system
vcd	Visualization techniques, data sets, summary and inference procedures aimed particularly at categorical data.
ggplot2	An implementation of the grammar of graphics in R.

Further useful libraries

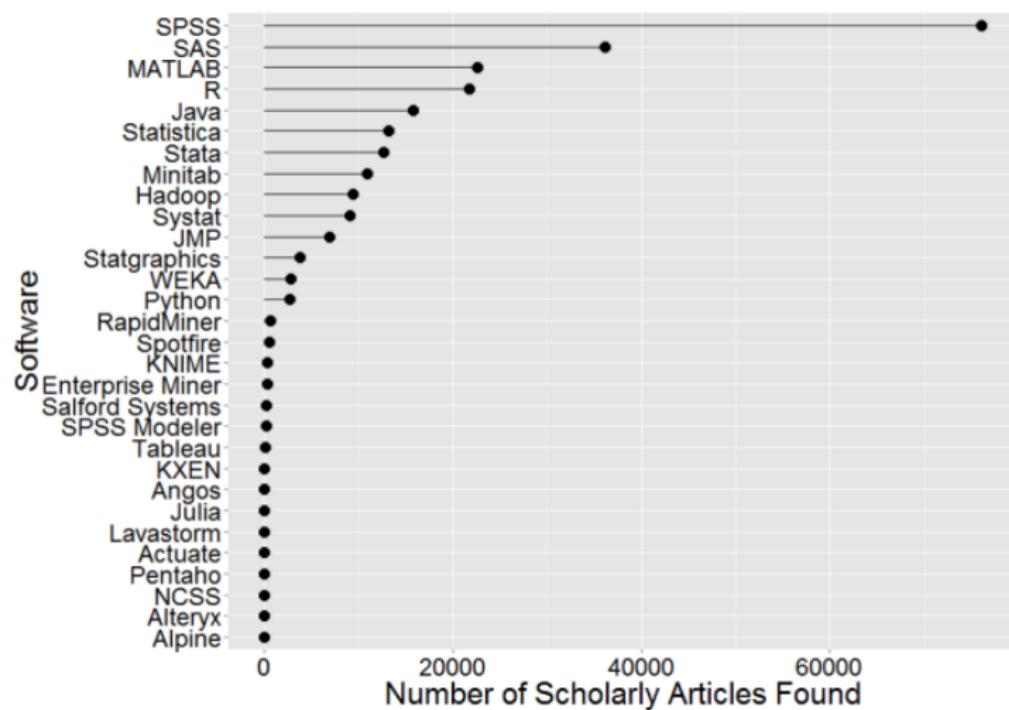
Library	Topic
xtable	Coerce data to LaTeX and HTML tables
dummies	Expands factors, characters and other eligible classes into dummy/indicator variables.
mvtnorm	Multivariate Normal and t Distributions
maptools	Tools for reading and handling spatial objects

Popularity of R



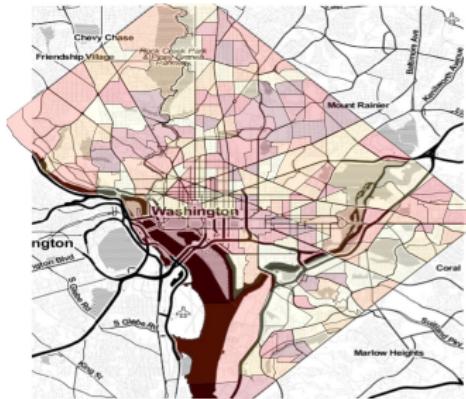
<http://r4stats.com/articles/popularity/>

Number of scholarly articles found for each software.



Number of scholarly articles found for each software.

- ▶ Number of articles found for software packages (all years Google Scholar can search)
- ▶ SPSS is by far the most dominant package,
- ▶ likely due to its balance between power and ease-of-use.
- ▶ SAS has around half as many, followed by MATLAB and R.



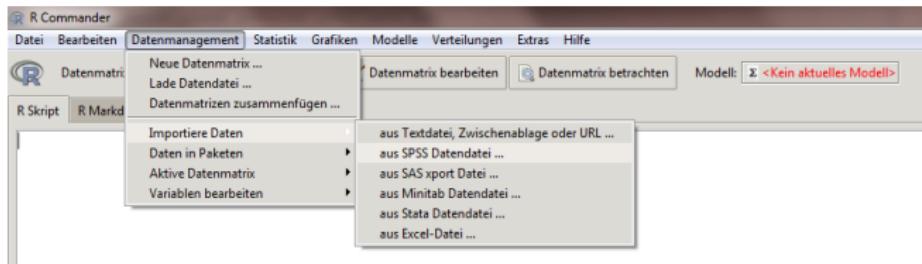
Possibility to plot nice maps

- ▶ <http://stathack.wordpress.com/>
- ▶ Find more examples under:
www.r-bloggers.com

R users over the world



Graphical User Interface Available



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

Interfaces with other software-packages

You can use it in combination with your favorite software package:

The image displays a collage of software interfaces and logos, illustrating various ways to integrate R with other statistical packages:

- Use R!**: A red header with the text "Use R!".
- Richard M. Heiberger
Erich Neuwirth**: Authors' names.
- R Through Excel**: A yellow header with the text "R Through Excel".
- IBM SPSS Statistics Essentials for R**: A screenshot of the IBM SPSS Statistics Essentials for R project page, showing a sidebar for "Open Source Software" and links for "Users", "Download IBM SPSS Statistics Essentials for R files", "Donate money", "Project detail and discuss", and "Get support".
- Avata**: A logo featuring a stylized "R" and "Avata" text.
- SASmixed**: The text "SASmixed" next to a small icon.
- R-Forge**: The text "R-Forge" next to a logo.
- rPython R package**: The text "rPython R package".
- Statistics and Computing**: A blue header with the text "Statistics and Computing".
- Robert A. Muenchen · Joseph M. Hilbe**: Authors' names.
- R for Stata Users**: A yellow header with the text "R for Stata Users".

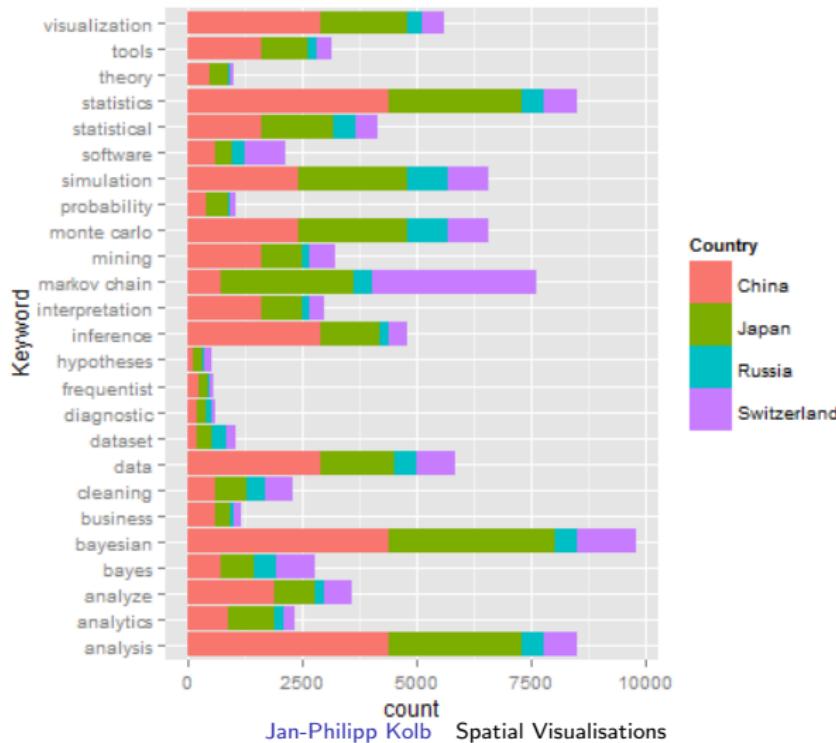
Interfaces with Python



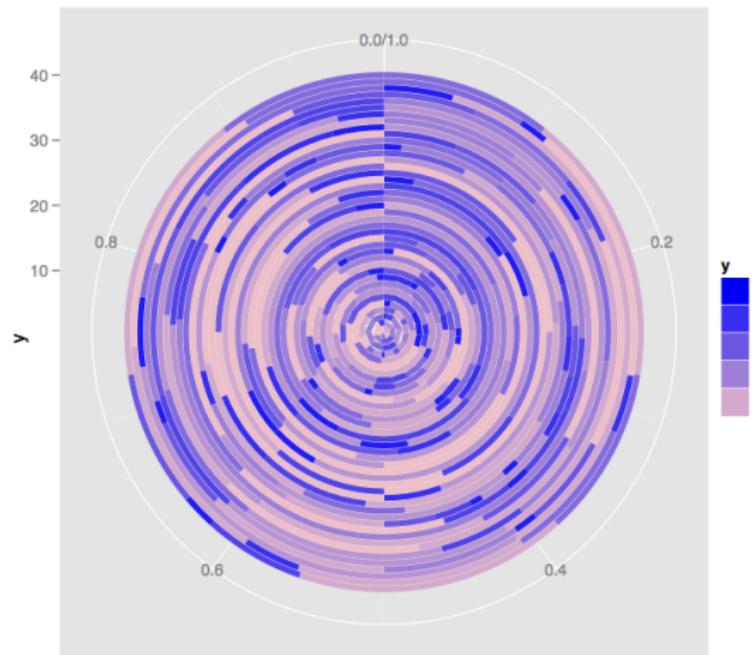
http:

[//rpy.sourceforge.net/rpy2/doc-2.2/html/graphics.html](http://rpy.sourceforge.net/rpy2/doc-2.2/html/graphics.html)

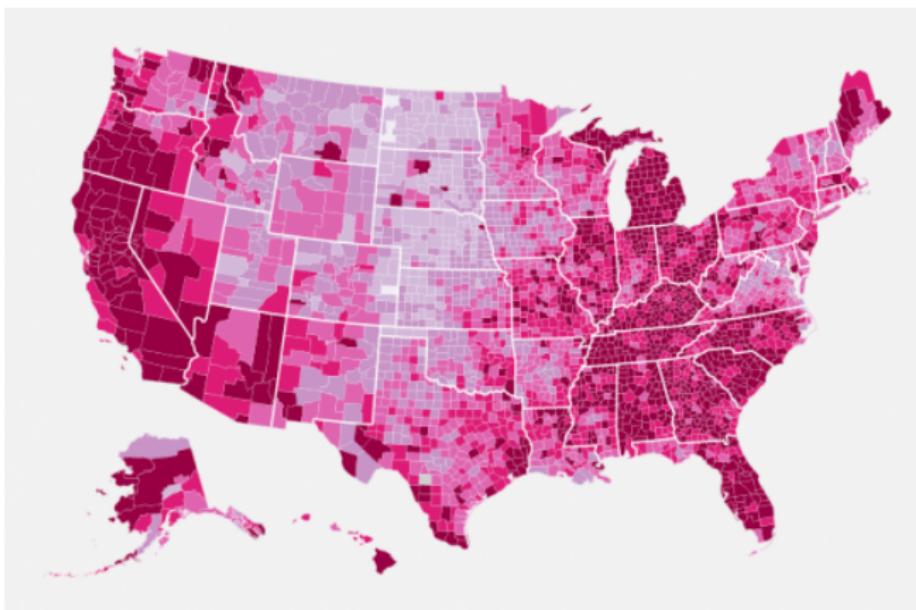
Visualisations with ggplot



Visualisations with ggplot



Visualisations with ggplot



<http://www.thisisthegreenroom.com/2009/choropleths-in-r/>

Outline

Why use R

Getting started

How to get help

Import and Export

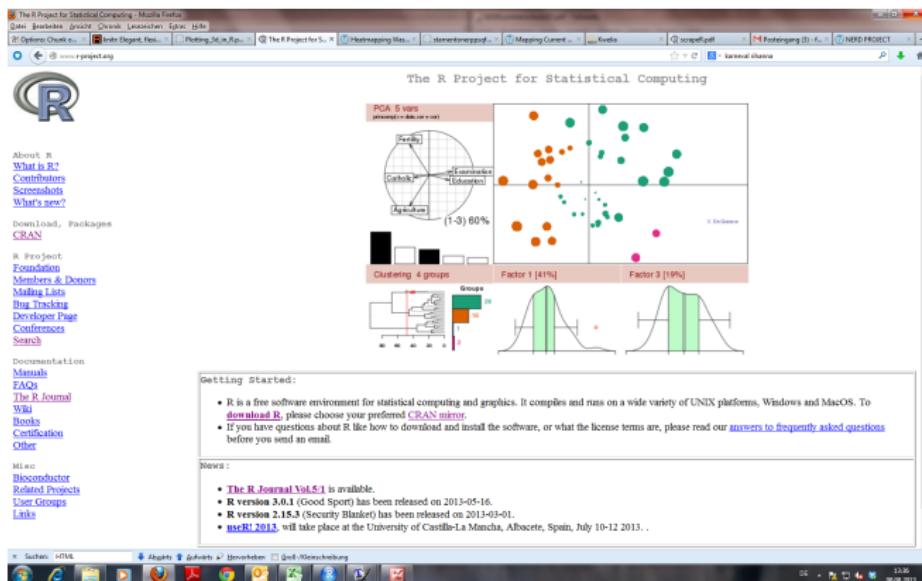
Sampling

Basic graphics

Literature

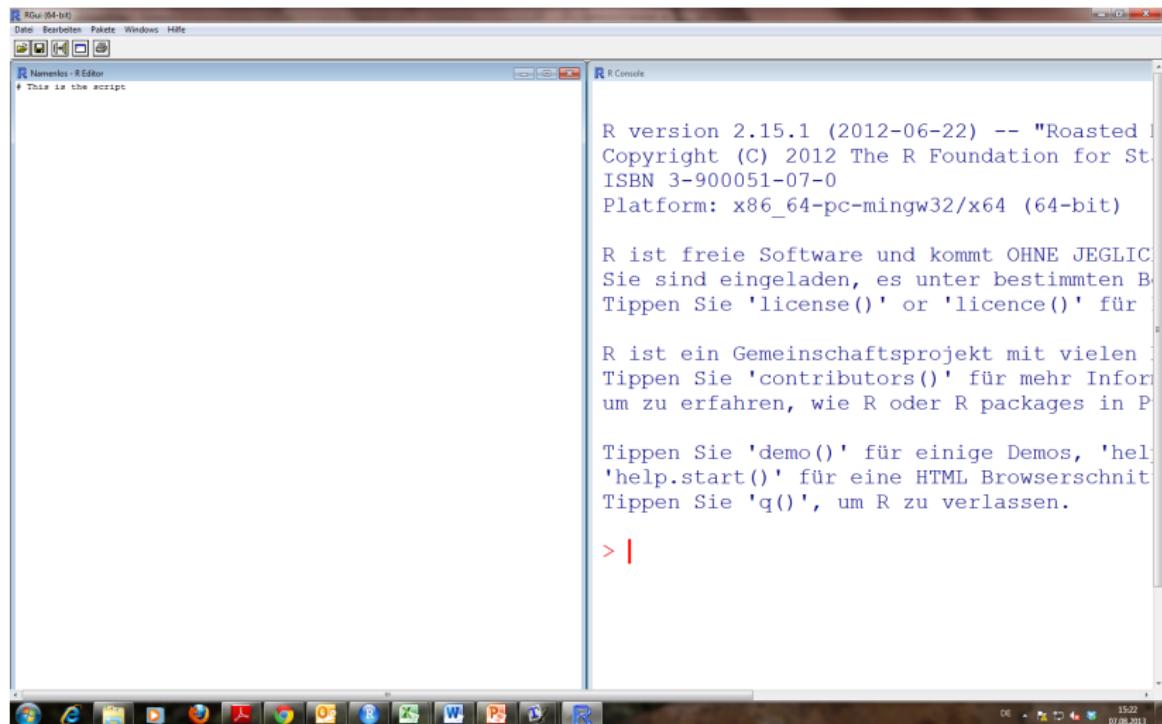
Exercises

Download R:



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

This is the basic R:



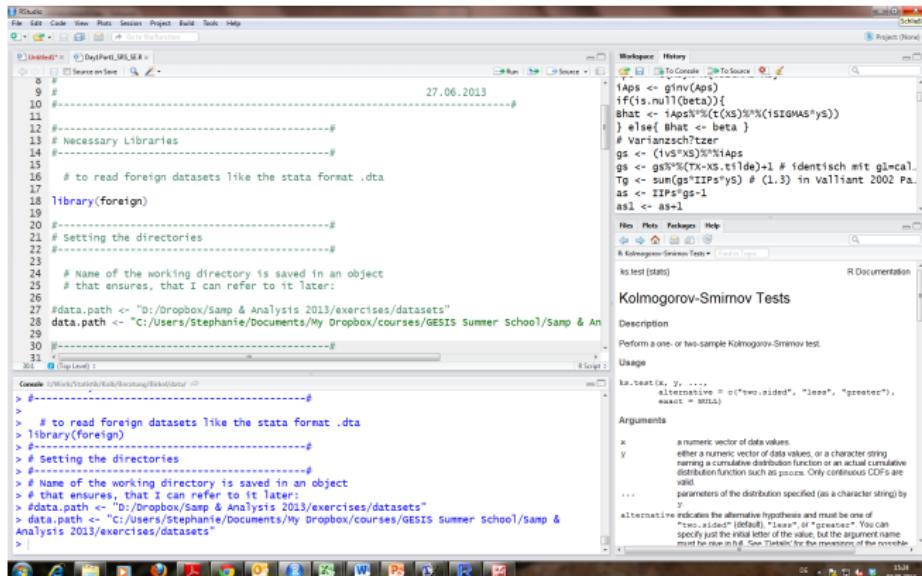
But most people use a special Editor or a *graphical user interface* (GUI) because:

- ▶ Syntax highlighting
- ▶ Auto-complete command line
- ▶ Better overview of plots, libraries

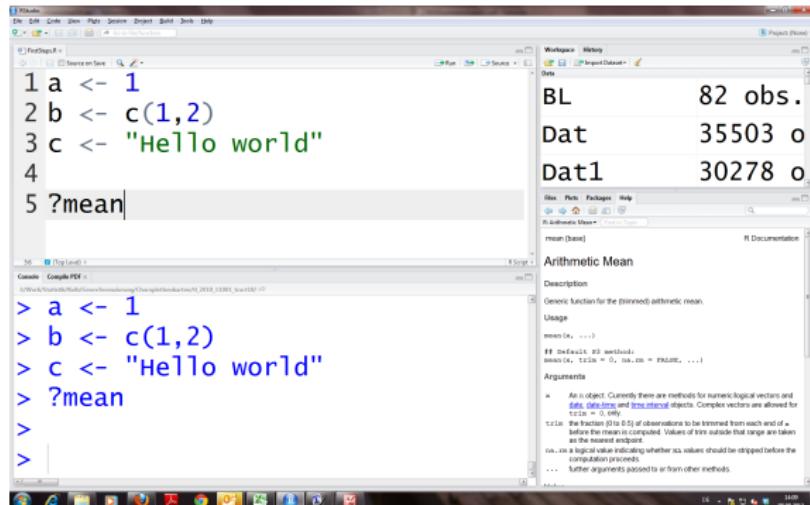
Graphical user interfaces

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

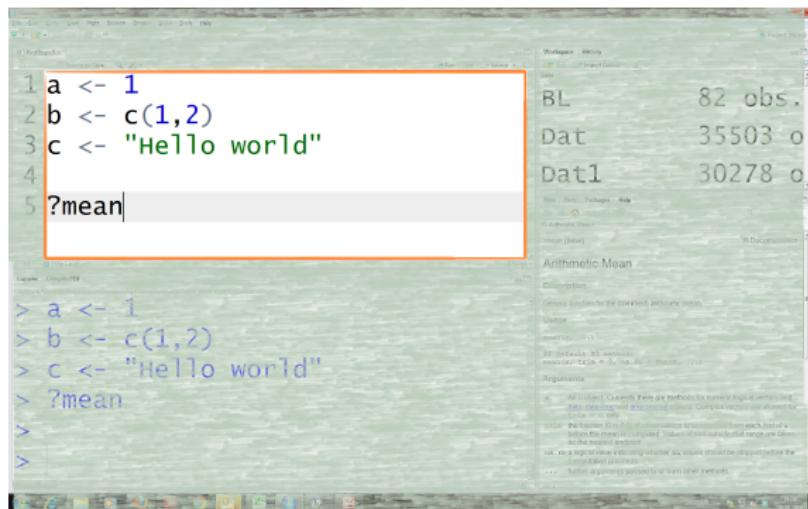
I'll use Rstudio!



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

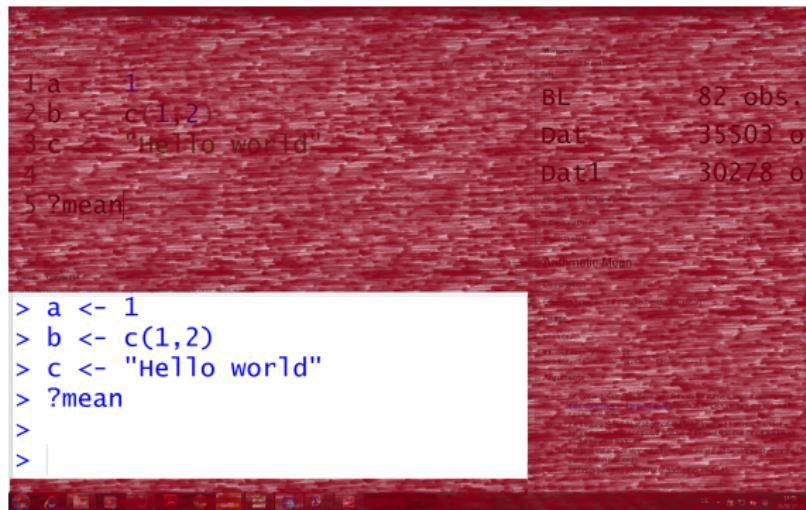


The different windows in Rstudio



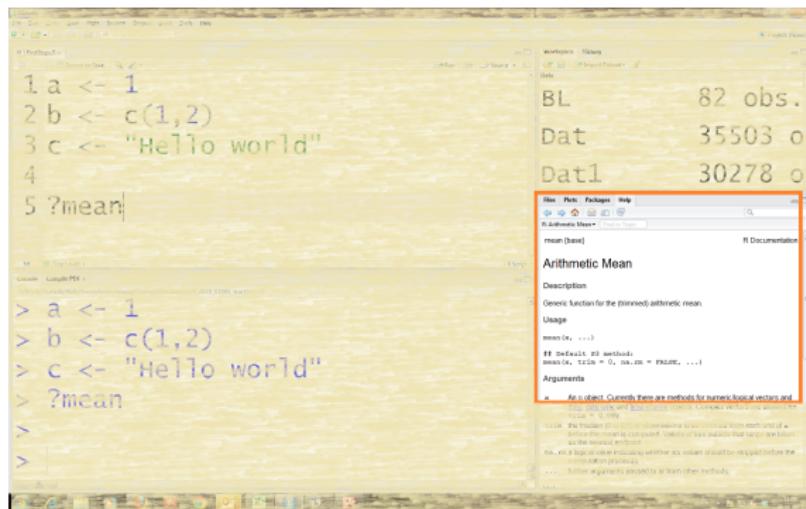
The different windows in Rstudio

The Script



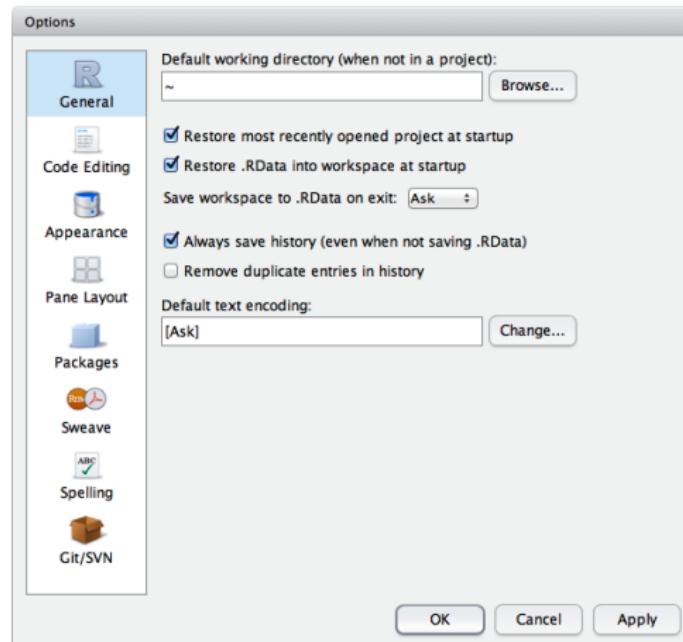
The different windows in Rstudio

Console



The different windows in Rstudio Help Files

Customizing RStudio



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

R as pocket calculator

```
# basic arithmetic operations
```

```
1+1
```

```
2-1
```

```
2*2
```

```
4/2
```

```
# Mathematical functions
```

```
log(4)
```

```
log(4, base=2)
```

```
exp(4)
```

```
sqrt(4)
```

```
2^4
```

Vectors and assignment

- ▶ `<-` is the assignment operator
- ▶ `b <- c(1,2)` creates an object with the numbers 1 and 2
- ▶ I can apply a function to that object:
- ▶ `mean(b)` gives me the mean

With the following functions we can learn about its characteristics:

- ▶ `length(b)` - b has the length 2
- ▶ `str(b)` - b is an numeric vector

If you are new to R have a look at:

<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

Different types of data

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

With `str()` you get the type.

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

Indexing

Indexing a vector:

```
> 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

Generate example data:

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

Combine the two vectors to get a data.frame

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

How many rows/columns do we have

```
nrow(Daten) # rows
ncol(Daten) # columns
```

Indexing

Indexing a 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
```

Matrices and arrays

- ▶ In matrices and arrays you often have only numericals
- ▶ That allows for matrix-manipulation
- ▶ In arrays more than two dimensions are possible.

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

Indexing

Indexing an 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
```

Lists

- ▶ A list in R corresponds to a nested array in other programming languages
- ▶ Everything can be ascribed to a list
- ▶ Lists can be nested
- ▶ Be careful with lists

Indexing

Indexing a list:

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

[[2]]
[1] 1

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

Sequences

```
> 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
```

The function 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"
```

Functions in the base package

Function	Meaning	Example
<code>length()</code>	Length	<code>length(b)</code>
<code>max()</code>	Maximum	<code>max(b)</code>
<code>min()</code>	Minimum	<code>min(b)</code>
<code>sd()</code>	Standard deviation	<code>sd(b)</code>
<code>var()</code>	Variance	<code>var(b)</code>
<code>mean()</code>	Mean	<code>mean(b)</code>
<code>median()</code>	Median	<code>median(b)</code>

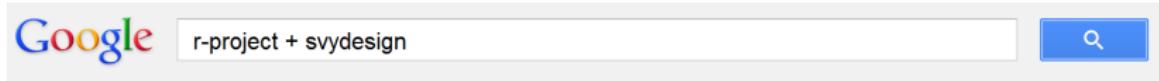
These functions do only need one argument.

There are others which need more:

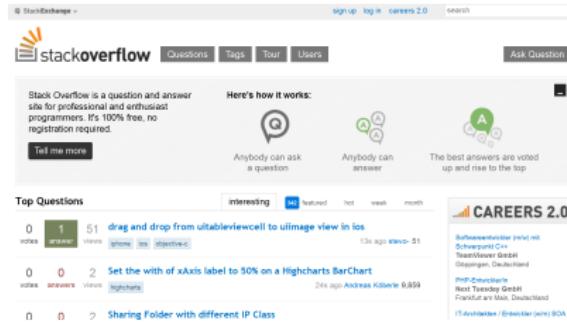
<code>quantile()</code>	90 % Quantile	<code>quantile(b,.9)</code>
<code>sample()</code>	Draw a sample	<code>sample(b,1)</code>

How to get help

- ▶ To get General help in R write the following command at R command prompt `help.start()`
- ▶ Online documentation for most of the functions and variables in R exists, and can be printed on-screen by typing `help(name)`
- ▶ Use of `?` for Help
Example: `?mean`
- ▶ `example(lm)` will provide an example of your required function such as `lm`



- ▶ Normally I use google if want to find something in R
- ▶ Type in:
R-project + What I always wanted to know
- ▶ That works of course with every search engine



- ▶ <http://stackoverflow.com/>
- ▶ For programming questions
- ▶ It is not focused on R
- ▶ Richly detailed discussions

CRAN - Task View - Spatial

CRAN Task View: Analysis of Spatial Data

Maintainer: Roger Bivand

Contact: Roger.Bivand at nhh.no

Version: 2014-09-19

Base R includes many functions that can be used for reading, visualising, and analysing spatial data. The focus in this view is on "geographical" spatial data, where observations can be identified with geographical locations, and where additional information about these locations may be retrieved if the location is recorded with care.

Base R functions are complemented by contributed packages, some of which are on CRAN, and others are still in development. One active location is [R-Forge](#), which lists "Spatial Data and Statistics" projects in its [project tree](#). Information on R-spatial packages, especially [sp](#) will be posted on the R-Forge r spatial project [website](#), including a visualisation gallery.

<http://cran.r-project.org/web/views/Spatial.html>

Outline

Why use R

Getting started

How to get help

Import and Export

Sampling

Basic graphics

Literature

Exercises

Data import



Datenzugang

Public-Use-File (PUF)

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)

Data sources:

- ▶ **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>

Formats - base package

- ▶ R can read many formats:
 - ▶ CSV (Comma Separated Values): `read.csv()`
 - ▶ FWF (Fixed With Format): `read.fwf()`
 - ▶ Tab-getrennte Werte: `read.delim()`

Import of Excel-data

- ▶ `library(foreign)` is necessary for import of foreign datasets.
- ▶ If have Excel data - save it as csv
- ▶ And than use `read.csv()` to import that data.
- ▶ If you have German data you might need `read.csv2()` because of comma-separation

Overview data 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>



- ▶ R project tutorial: reading Excel spreadsheets
<http://www.youtube.com/watch?v=Hq0JmSnBX8I>

- ▶ Loading Data Into R Software - (read.table, Data/CSV Import Tutorial)
<http://www.youtube.com/watch?v=VLtazaiYo-c>

R-library foreign

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

The function sample

[sample {base}](#)

R Documentation

Random Samples and Permutations

Description

`sample` takes a sample of the specified size from the elements of `x` using either with or without replacement.

Usage

```
sample(x, size, replace = FALSE, prob = NULL)
```

x: From what do we want
to sample ?

```
sample(x=1:10,n=1,replace=T)
```

The function sample

[sample {base}](#)

R Documentation

Random Samples and Permutations

Description

`sample` takes a sample of the specified size from the elements of `x` using either with or without replacement.

Usage

```
sample(x, size, replace = FALSE, prob = NULL)
```

n: How many elements
do we want to draw?

```
sample(x=1:10, n=1, replace=T)
```

The function sample

[sample {base}](#)

R Documentation

Random Samples and Permutations

Description

`sample` takes a sample of the specified size from the elements of `x` using either with or without replacement.

Usage

```
sample(x, size, replace = FALSE, prob = NULL)
```

Do we want to draw with
or without replacement?

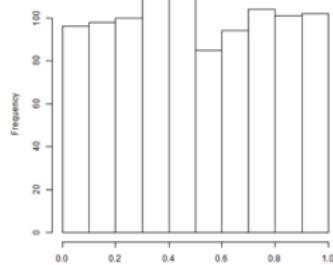
```
sample(x=1:10, n=1, replace=T)
```

Random numbers in R

Function	Distribution	Important parameter
runif	Uniform distribution	n, min, max
rnorm	Normal distribution	n, mean, sd
rpois	Poisson distribution	n, lambda
...

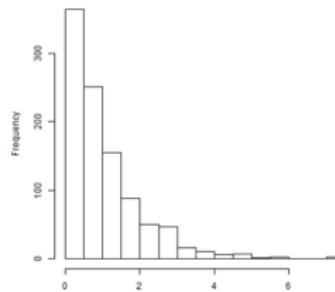
```
png("HistX2.png")
hist(x2,main="")
dev.off()
```

```
x3 <- rexp(1000)
```



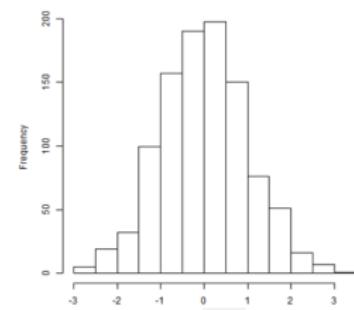
A

```
x1 <- runif(1000)
```



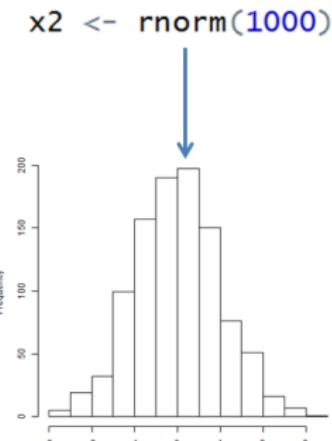
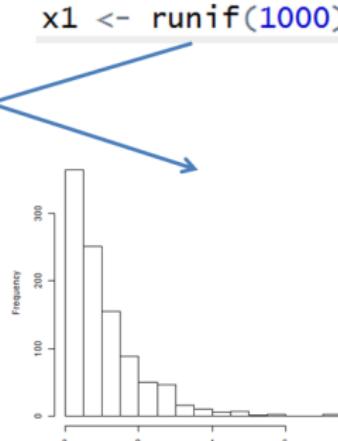
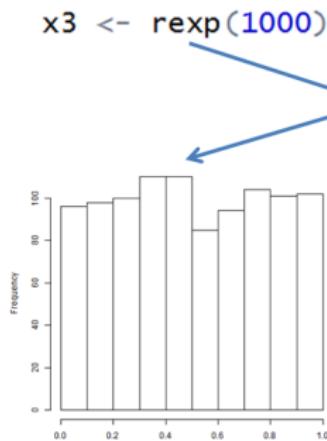
B

```
x2 <- rnorm(1000)
```



C

```
png("HistX2.png")
hist(x2,main="")
dev.off()
```

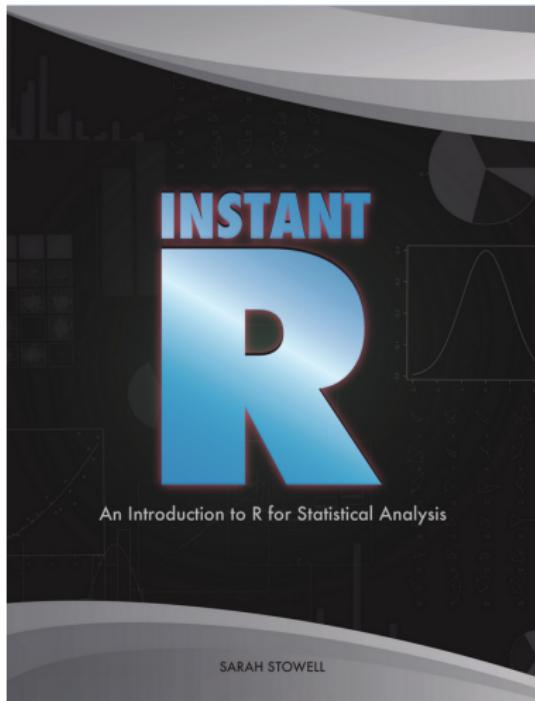


Literature



- ▶ Ligges, U. (2008):
Programmieren mit R.
Springer.
- ▶ Good book for beginners,
but unfortunately only in
German.

Literature



- ▶ Import and export of data
- ▶ Data manipulation
- ▶ Graphics

Exercise 1

Draw a simple random sample

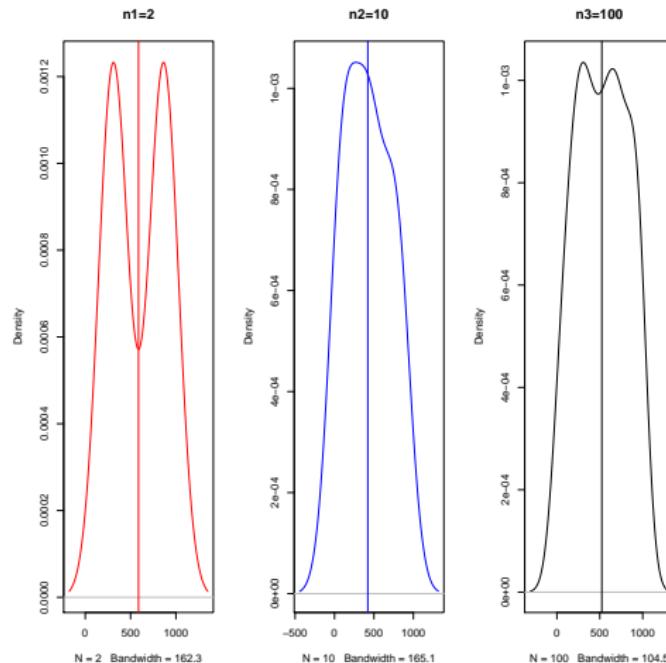
- ▶ Load the *Microcensus-PUF*.
forschungsdatenzentrum
- ▶ Draw a srs with replacement ($n=100$).
- ▶ Plot a barplot of the federal states.

Exercise 2

Sample sizes

- ▶ Generate 1000 numbers from a exponential distribution.
- ▶ Draw three samples($n_1=2, n_2=10, n_3=100$).
- ▶ Plot the density and add the means of the three samples as vertical lines (see the following picture).

Density



More help for beginners



stat > r > sk

Help the Stat Consulting Group by

[giving a gift](#)

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.

R-Literature

Books related to R

This page gives a partially annotated list of books that are related to S or R and may be useful to the R user community. See also the list of [other publications](#) related to R. An alternative searchable listing of both sets together is available [here](#).

- [1] Victor A. Bloomfield. *Using R for Numerical Analysis in Science and Engineering*. Chapman & Hall/CRC, 2014. ISBN 978-1439884485. [[bib](#) | <http://www.crcpress.com/product/isbn/9781439884485>]

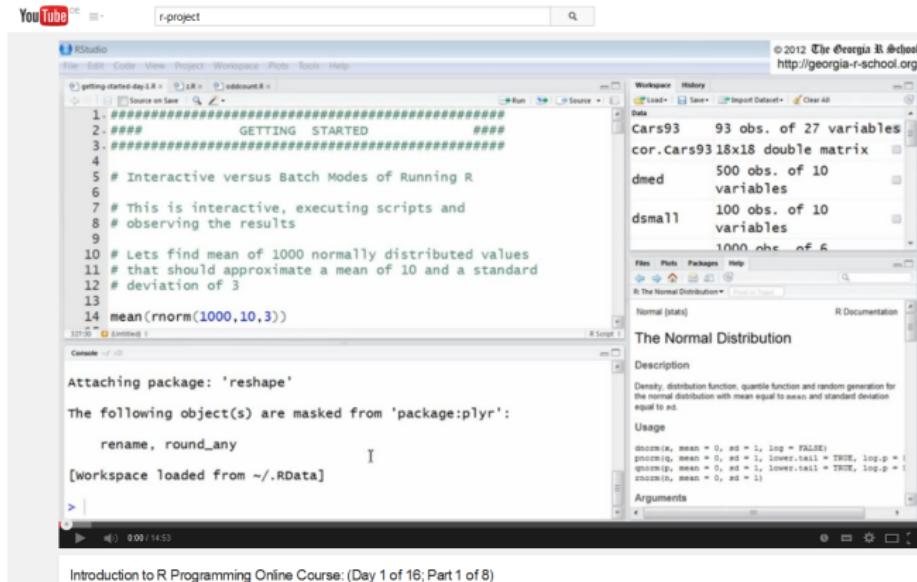
Instead of presenting the standard theoretical treatments that underlie the various numerical methods used by scientists and engineers, Using R for Numerical Analysis in Science and Engineering shows how to use R and its add-on packages to obtain numerical solutions to the complex mathematical problems commonly faced by scientists and engineers. This practical guide to the capabilities of R demonstrates Monte Carlo, stochastic, deterministic, and other numerical methods through an abundance of worked examples and code, covering the solution of systems of linear algebraic equations and nonlinear equations as well as ordinary differential equations and partial differential equations. It not only shows how to use R's powerful graphic tools to construct the types of plots most useful in scientific and engineering work, but also:

<http://www.r-project.org/doc/bib/R-books.html>

A (very) short introduction to R

[http://cran.r-project.org/doc/contrib/Torfs+
Brauer-Short-R-Intro.pdf](http://cran.r-project.org/doc/contrib/Torfs+
Brauer-Short-R-Intro.pdf)

Youtube-videos



<http://www.youtube.com/watch?v=qHfSTRNg6jE>

Youtube video - graphics

- ▶ R project tutorial: how to create and interpret a matrix scatter plot

<http://www.youtube.com/watch?v=kkhdB4dNg0>

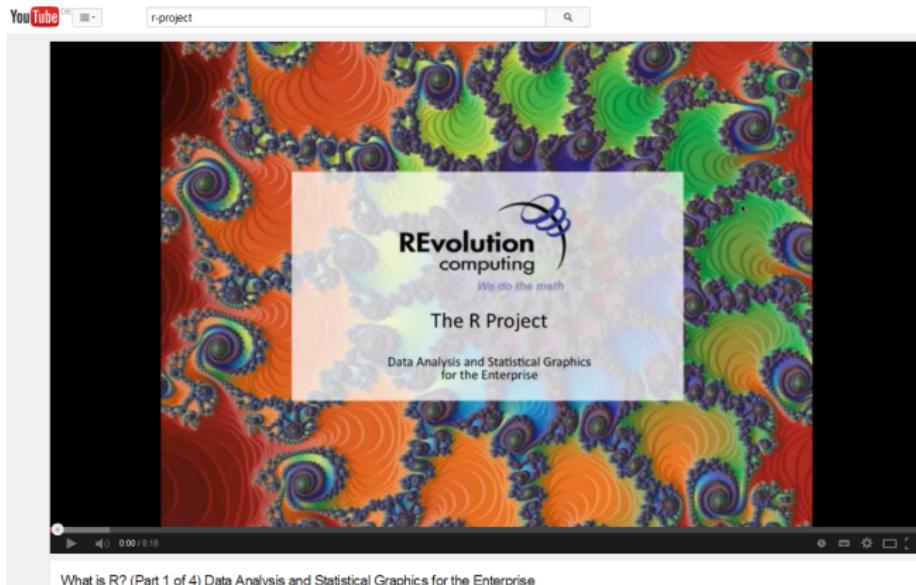
Youtube-videos

- ▶ Statistics with R (part 0: Download and Install R)
<http://www.youtube.com/watch?v=NfH5peM1RtI&list=PL9FE4E32D5F2AA0BA&index=3>
- ▶ R Tutorial 1 - Download, Installation, Setup - Statistical Programming Language R
<http://www.youtube.com/watch?v=ZoPJGmpYJzw>
- ▶ An Introduction to R - A Brief Tutorial for R Software for Statistical Analysis
<http://www.youtube.com/watch?v=LjuXiBjxryQ>
- ▶ R fr Anfnger (4): Daten einlesen, analysieren, plotten fr Eilige (German)
<http://www.youtube.com/watch?v=0ADUACud99k>
- ▶ R project tutorial: simple linear regression
<http://www.youtube.com/watch?v=ZGTB0hbahmY>

Youtube-videos

- ▶ R Tutorial 3 - More about Vectors - Statistical Programming Language R
<http://www.youtube.com/watch?v=hLozrMhpWuY>
- ▶ Twitter Sentiment Analysis
http://www.youtube.com/watch?v=adIvt_lu01o

Youtube-videos



What is R? (Part 1 of 4) Data Analysis and Statistical Graphics for the Enterprise

http://www.youtube.com/watch?v=M2u7kbcXI_k

Please contact me!

Jan-Philipp.Kolb@gesis.org

<https://github.com/Japhilko/GeoData>

The screenshot shows the GitHub repository page for 'Japhilko / GeoData'. The repository is described as 'Research on statistics and geodata'. It has 3 commits, 1 branch named 'master', 0 releases, and 1 contributor. The 'Code' sidebar is open, showing the repository structure. The 'Code' tab is selected, displaying files like 'rcode', 'gitattributes', 'gitignore', and 'README.md'. The 'README.md' file contains the text: 'GeoData' and 'This repo encloses my research with R on statistics and geodata'. The 'Issues' and 'Pull Requests' sections are empty. The 'Wiki' section is collapsed. The 'Code' sidebar also includes options for 'Clone In Desktop' and 'Download ZIP'.