

Programming and Analyzing in R 1

Wouter van Atteveldt <http://vanatteveldt.com>

Spring 2016

Course overview

① Lab 2

- Getting started
- Your data in R

② Lab 2

- Merging and transforming data
- Classical statistics and visualization

③ Lab 3

- Advanced statistics and programming

④ Lab 4

- Analysing texts and networks

What is R?

- Open source
- Multi-platform
- Text/Console (programming)
- Community driven
 - Packages

Cathedral vs. Bazaar



R vs SPSS

- Syntax mode only
- Full featured programming language
- Tons of extensions

Interactive sessions

- Slides: <http://vanatteveldt.com/>
- Handouts and data:
<http://github.com/vanatteveldt/learningr>
- Please interrupt me!

Introduction

- Who are you? :-)
- What are you working on?
- What do you think R will do for you?
- What experience do you have with
 - R
 - Other statistical packages
 - Programming (inc. SPSS syntax, Stata .do)

Using R

- R as a calculator
- Reading in data
- Data frames vs. vectors
- Simple plots

Using RStudio

- Scripts, results, environment, help/output
- Running scripts
- Projects

Packages and imports

- Most functionality is in **packages**
- Packages need to be installed once
 - `install.packages("reshape2")`
- Packages need to be activated every time
 - `library(reshape2)`

Errors

- You will encounter error messages :-)
 - Some of them will be cryptic
 - Debugging as a valuable skill
- Run scripts line by line
- Look at output (console):
 - Is there an error message?
 - Is there a normal prompt `>` or a continuation prompt `+`?
 - "Escape" to stop continuation
- Inspect variables in between steps

First hands-on session

- `1_r_calculator.md`
- <http://www.openintro.org/stat/labs.php>, "Intro to R and RStudio"

Reading/writing data

- R: `save`, `load(file)`
- CSV: `read.csv`, `read.csv2`, `write.csv`
- SPSS: + `read.spss(to.data.frame=T)`
 - `library(foreign)`

Inspecting data

- Inspecting a data frame
 - `names`
 - `summary`
 - `head` and `tail`
 - RStudio: `View`
- Inspecting a column (vector)
 - `summary`
 - `length`
 - `mean`, `sum`, etc

Data manipulation: subsets

- `data[rows, columns]`
- Leave out either argument:
 - `data[rows,]`
 - `data[, columns]`
- Use index, names, or calculation
 - `data[5,]`
 - `data[, c(1,2,3)]`
 - `data[, c("x", "y")]`
 - `data[, -1]`
 - `data[data$x > 3,]`

Data manipulation: calculated columns

- Assignment to part of a data frame
- `mydata$c = mydata$a + mydata$b`
- `mydata$c[mydata$a > 3] =`
`mydata$c[mydata$a > 3] + 1`

Good coding practice

- Code is written once, read many times
- Some good practices:
 - ① Descriptive variable names
 - ② Use comments to explain decisions
 - ③ Clean up code when you're done
 - ④ Write self-contained scripts
 - Start with reading data, end with analysis
 - Test by clearing all variables, running script

Hands on 1b

- `2_playing.md`
- `http://www.openintro.org/stat/labs.php,
"Introduction to data"`