

HelmholtzZentrum münchen

German Research Center for Environmental Health

Core Facility

Statistical Consulting

How to make statistical analysis code reproducible and reusable

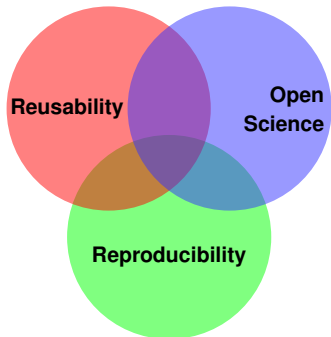
Dr. Elmar Spiegel

elmar.spiegel@helmholtz-muenchen.de

HELMHOLTZ
Open Science

HELMHOLTZ RESEARCH FOR
GRAND CHALLENGES

Definitions



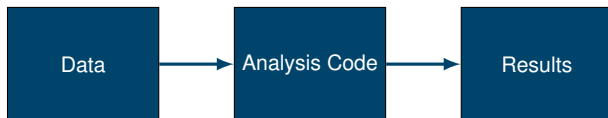
- **Reproducibility:** Possibility to replicate the findings from the original study.
 - *Direct reproducibility*
 - ***Computational reproducibility***
- **Reusability:** Possibility to use material / knowledge / code from a previous study in a new one.
- **Open Science:** Attempt to make all parts of a study available free of charge: paper, data, code, ...

Basic idea of reproducible code



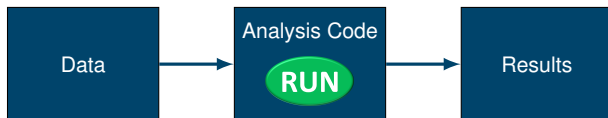
Basic idea of reproducible code

Use the original data and the original code and regain the same plots and tables.



Basic idea of reproducible code

Use the original data and the original code and regain the same plots and tables.

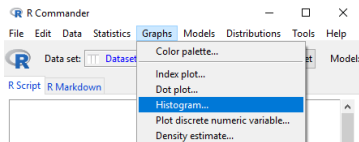


- Ideally: "Press one button"

Basic principles of computational reproducibility

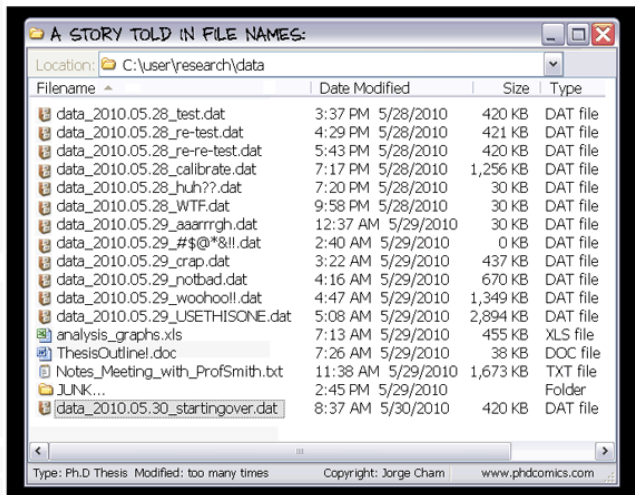
- Use a **script-based** analysis program, e.g. R or SAS and **no GUI**

```
1 # R Script
2
3 hist (Dataset$Age,
4       col = "red",
5       xlab = "Age")
```



- Don't save **different** versions of your data set as physical files
- Write a **Readme** file to explain your storage structure
 - Readme.pdf or Readme.txt
 - Provides all information needed to run the code
 - State dependencies and prerequisites
 - Determine the configuration of the programs used including version number of program and package numbers.
- **Comment** your code and use **meaningful** variable names
- Don't keep code which produces **errors**: Fix or delete it

Take care...



"Piled Higher and Deeper" by Jorge Cham www.phdcomics.com

https://phdcomics.com/comics/archive_print.php?comicid=1323

Problems often occurring

The *Biometrical Journal* asks for code and data during the submission of an article. Hofner et al. (2016) evaluated 56 articles accepted for publication:

- On the first try, only 5 submissions reproduced the whole article.
- Up to 4 revisions were necessary to get reproducible files.
- **Typical issues:**

Missing data or code	44.6%
Code produced errors	37.5%
Code ran but did not reproduce all tables/figures	51.8%
Missing README	80.4%

Technical assistance

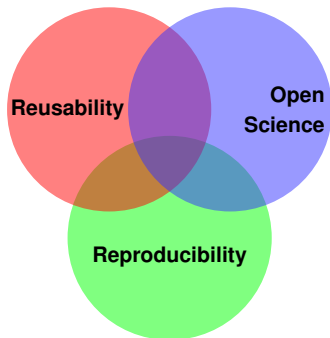
- **Rerun** your analysis before submission
- Use **literate programming**
 - Combine coding and manuscript writing in one document
 - Examples: RMarkdown, Jupyter Notebook
- Apply version control software:
 - Keep **track** of different versions
 - **Share** code with collaborators
 - Open code to **public**
 - Examples: Github, Gitlab

Technical assistance

- **Rerun** your analysis before submission
- Use **literate programming**
 - Combine coding and manuscript writing in one document
 - Examples: RMarkdown, Jupyter Notebook
- Apply version control software:
 - Keep **track** of different versions
 - **Share** code with collaborators
 - Open code to **public**
 - Examples: Github, Gitlab

Reproducible → **Reusable code**

Outline



Styling of code

The code should

- use good and **consistent code styling** to be easy to read.
- be **well documented**.
- be written in plain English.
- be **portable** (no absolute paths like `D:/Users/Elmar/Documents/`).
- avoid **user input**.

Styling of code

Bad Example

```
1 arb_data<-read.table("D:/Users/elmar.spiegel/Documents/Lehre/Reproduc
2 dec=".",header=TRUE,stringsAsFactors=TRUE)
3 arb_data<-arb_data[-c(3,4),]
4 treatment<-arb_data$outcome_var[grouping_var=="treat"]
5 control<-arb_data$outcome_var[grouping_var=="contr"]
6 wilcox.test(treatment,control,"two.sided",paired=FALSE,conf.level=0.9
7 ZusätzlicherCode<-1
8 BöBeZeichen<-ZusätzlicherCode-5
9 FALSE<-1
```

Styling of code

Good Example

```
1 # set seed
2 set.seed(2)
3
4 # Read arbitrary data
5 arb_data <- read.table(file = "Arbitrary_data.txt",
6                         sep = ";", dec = ".", header = TRUE,
7                         stringsAsFactors = TRUE)
8
9 # Remove rows with false data
10 arb_data <- arb_data[arb_data$outcome_var > 0, ]
11
12 # Apply Wilcoxon test for differences between the groups
13 wilcox.test(outcome_var ~ grouping_var, data = arb_data
14             alternative = "two.sided", paired = FALSE,
15             conf.level = 0.95)
```

Structure of projects

Repeated usage:

- Repeated calls of some structure \Rightarrow Write a function
- Apply functions in multiple projects \Rightarrow Write a library/package
- Share ideas with unknown others \Rightarrow Upload on software specific repository

Efforts for ...

- ... using proper styling
- ... making code reproducible including documentation
- ... simple reusability by yourself
- ... writing a public available package

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation
- ... simple reusability by yourself
- ... writing a public available package

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation \Rightarrow Days
- ... simple reusability by yourself
- ... writing a public available package

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation \Rightarrow Days
- ... simple reusability by yourself \Rightarrow Days/Weeks
- ... writing a public available package

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation \Rightarrow Days
- ... simple reusability by yourself \Rightarrow Days/Weeks
- ... writing a public available package \Leftrightarrow Paper / Book \Rightarrow Months

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation \Rightarrow Days
- ... simple reusability by yourself \Rightarrow Days/Weeks
- ... writing a public available package \Leftrightarrow Paper / Book \Rightarrow Months

Rewarding?

A little, only paper-publications count

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation \Rightarrow Days
- ... simple reusability by yourself \Rightarrow Days/Weeks
- ... writing a public available package \Leftrightarrow Paper / Book \Rightarrow Months

Rewarding?

A little, only paper-publications count

\Rightarrow Lots of badly written non-reproducible code and not maintained packages

Efforts for ...

- ... using proper styling \Rightarrow Training
- ... making code reproducible including documentation \Rightarrow Days
- ... simple reusability by yourself \Rightarrow Days/Weeks
- ... writing a public available package \Leftrightarrow Paper / Book \Rightarrow Months

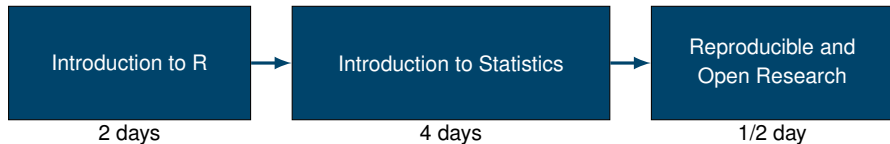
Rewarding?

A little, only paper-publications count

\Rightarrow Lots of badly written non-reproducible code and not maintained packages

\Rightarrow Research Software Engineers?

Training program



- Number of courses from January 2019 to September 2019: 16
- Number of participants from January 2019 to September 2019: 200+

References I

- Gandrud, C. (2016). *Reproducible research with R and R studio*. Chapman and Hall/CRC.
- Hofner, B., Irish, W., Turner, D., & Jacoby, W. G. (2015). Reporting for reproducibility. http://benjaminhofner.de/downloads/2015/talks/reporting-for-reproducibility-webinar_final-slides_125771.pdf.
- Hofner, B. & Scheipl, F. (2016). Guidelines for code and data submission, specific guidance on reproducible research. https://onlinelibrary.wiley.com/pb-assets/assets/15214036/RR_Guideline.pdf.
- Hofner, B., Schmid, M., & Edler, L. (2016). Reproducible research in statistics: A review and guidelines for the biometrical journal. *Biometrical Journal*, 58(2), 416–427.

BACKUP

README Example

README Analysis of mortality rates in Sao Paulo

Elmar Spiegel

25th April 2019

This README document specifies how to rebuild one example (mortality rate in Sao Paulo) of the paper Spiegel et al. (2019) <https://doi.org/10.1007/s11222-017-9799-6>

• Analyst:

Dr. Elmar Spiegel
Institute of Computational Biology
Helmholtz Zentrum München
elsar.spiegel@helmholtz-muenchen.de

• Analysis Software:

- R version 3.5.2
- Attached base packages:
stats graphics grDevices utils datasets methods base
- Other attached packages:
FlexGAM_0.7.0 mgcv_1.8-26 nlme_3.1-137
- Packages loaded via a namespace (and not attached):
MASS_7.3-51.1 compiler_3.5.2 Matrix_1.2-15 tools_3.5.2 yaml_2.2.0 splines_3.5.2 grid_3.5.2
lattice_0.20-38 scam_1.2-3

• Preprint.pdf

- Pre-print of the article Spiegel et al. (2019) <https://doi.org/10.1007/s11222-017-9799-6>

• Data:

- master.xls
 - Original data downloaded via www.ime.usp.br/~jmsinger/Polata9497.zip.
- Final_Data.csv
 - Main data used in the analysis.
 - Extracted from master.xls via Build_data.R
- master.doc
 - Original data description downloaded via www.ime.usp.br/~jmsinger/Polata9497.zip.
 - In Portuguese (Brazil)
- Results.RData
 - Saved model outcome of Main_model.R
- Variable_description.txt
 - Description of the variables used in the analysis, i.e. Final_Data.csv and Main_model.R
 - In English

• Code:

- Build_data.R
 - 1. to be executed
 - Input: master.xls
 - Output: Final_Data.csv
 - Description: In this file the original data (master.xls) are loaded and transformed to fit to the analysis.
- Build_figures.R
 - 3. to be executed
 - Input: Results.RData
 - Output: Fig5.pdf, Fig6.pdf
 - Description: This file uses the model output and provides the graphics for the paper.
- Main_model.R
 - 2. to be executed
 - Input: Final_Data.csv
 - Output: Results.RData
 - Description:
 - This file runs all models.
 - Also the bootstrap confidence intervals are estimated in this file.
 - The code depends on the package FlexGAM, which can be downloaded from <https://cran.r-project.org/>

