Reproducible computational research

Reproducible computational research

- Motivation
- 10 simple rules

Based on:

- Peng, 2011: **Computational and Policy Tools for Reproducible Research**
- Sandve, 2013: **Ten Simple Rules for Reproducible Computational Research**. *PLOS Computational Biology*. http://dx.doi.org/10.1371/journal.pcbi.1003285

Replication

- Replication of findings and conducting studies with independent
  - Data (e.g. investigators, laboratories, instruments)
  - Code (e.g. implementation of analytical methods)
- Dimensions of reproducibility
  - Data
  - Code

10 simple rules

Sandve, 2013: **Ten Simple Rules for Reproducible Computational Research**. *PLOS Computational Biology*.

1. For Every Result, Keep Track of How It Was Produced
2. Avoid Manual Data Manipulation Steps
3. Archive the Exact Versions of All External Programs Used
4. Version Control All Custom Scripts
5. Record All Intermediate Results, When Possible in Standardized Formats
6. For Analyses That Include Randomness, Note Underlying Random Seeds
7. Always Store Raw Data behind Plots
8. Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected
9. Connect Textual Statements to Underlying Results
10. Provide Public Access to Scripts, Runs, and Results

1. For Every Result, Keep Track of How It Was Produced

- Deterministic re-run should be possible
- Code vs point-and-click(-hopefully-the-correct-checkboxes-are-checked)

2. Avoid Manual Data Manipulation Steps

- Do **not** change data files; correct them in the code
- New versions of data
- See Rule 1: ‘For Every Result, Keep Track of How It Was Produced’

```r
d <- read_delim(...)
d <- d %>%
  filter(!(ID %in% excluded_ids)) %>%
  mutate(Age = case_when(
    Id == 11 ~ 56, # Per mail X-Y-Z, ...
    TRUE ~ Age
  ))
```

Also: `assertr` package

```r
library(assertr)
mtcars %>%
  verify(mpg >= 0) # Error if not true
```

3. Archive the Exact Versions of All External Programs Used

- R package packrat
- R package checkpoint

```
sessionInfo() # or devtools::session_info()
```

```
##  [7] LC_PAPER=da_DK.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=da_DK.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  me
##
## other attached packages:
## [1] forcats_0.3.0   stringr_1.3.1   dplyr_0.8.0.1   purr
## [5] readr_1.3.0     tidyr_0.8.3     tibble_2.0.1    ggpl
## [9] tidyverse_1.2.1
##
## loaded via a namespace (and not attached):
```

4. Version Control All Custom Scripts

- git, subversion, . . .
- Manuel copy: `analysis_2016-10-13_1200_before_featureXYZ.Rmd`, `analysis_2016-11-01_1200_revision01.Rmd`
- Combination

5. Record All Intermediate Results, When Possible in Standardized Formats

- In principle not necessary; simply run analysis from beginning to end
- Faster development cycles (cache)
- Allow others to replicate without original data

```
save(...) # binary
write.table(...) # text
```

6. For Analyses That Include Randomness, Note Underlying Random Seeds

```
set.seed(1)
runif(3)
```

```
## [1] 0.2655087 0.3721239 0.5728534
```

```
runif(3)
```

```
## [1] 0.9082078 0.2016819 0.8983897
```

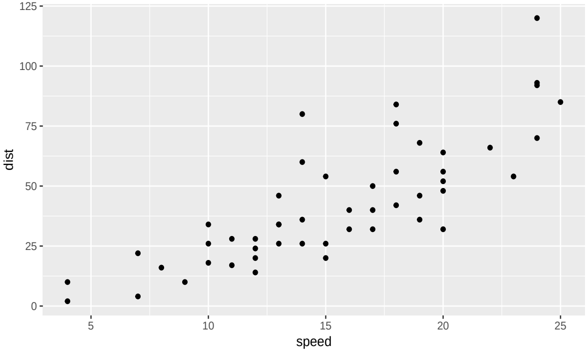
```
set.seed(1)
runif(3)
```

```
## [1] 0.2655087 0.3721239 0.5728534
```

7. Always Store Raw Data behind Plots

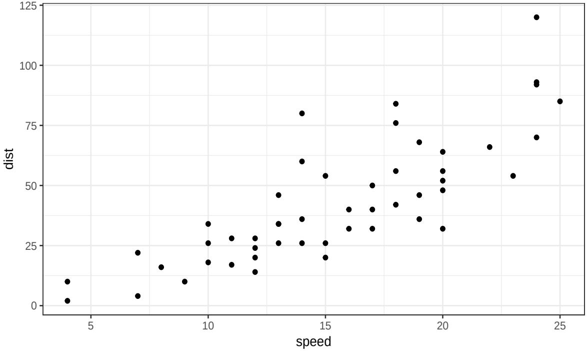
- In principle not necessary; simply run analysis from beginning to end
- Faster development cycles (cache)
- Modify plot programmatically

```
ggplot(cars, aes(speed, dist)) + geom_point()
```



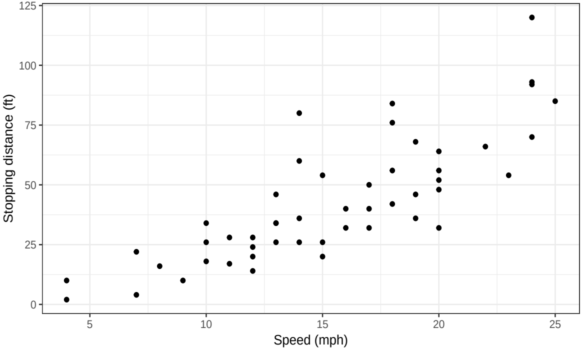
7. Always Store Raw Data behind Plots

```
ggplot(cars, aes(speed, dist)) +
  geom_point() +
  theme_bw()
```



7. Always Store Raw Data behind Plots

```r
ggplot(cars, aes(speed, dist)) +
  geom_point() +
  theme_bw() +
  labs(x = "Speed (mph)", y = "Stopping distance (ft)")
```



8. Generate Hierarchical Analysis Output, Allowing Layers of Increasing Detail to Be Inspected

- In principle not necessary; simply run analysis from beginning to end
- Faster development cycles (cache)
- Save raw data, not just summaries (e.g. for inspection, details in manuscript revisions, . . .)

9. Connect Textual Statements to Underlying Results

- Literate (statistical) programming
- Notes and analysis mixed

10. Provide Public Access to Scripts, Runs, and Results

- Publish a method as an R package alongside analyses scripts, runs and results

Literate (statistical) programming

Literate (statistical) programming, e.g. R Markdown.

My humble additions

- Data (transport/communication) is evil; be suspicious and challange it! [Decimal separators, ‘Dead’/‘dead’/‘DEAD’/’_dead’/. . .]
- Do not accept/create Excel spreadsheets where cell colouring has been used to encode information (that is not present elsewhere)!

Data Organization in Spreadsheets

- https://kbroman.org/dataorg/
- Broman and Woo. *Data Organization in Spreadsheets*. The American Statistician, 2018. https://doi.org/10.1080/00031305.2017.1375989