# Reproducible Workflows
# for Better Science and Efficient Collaboration

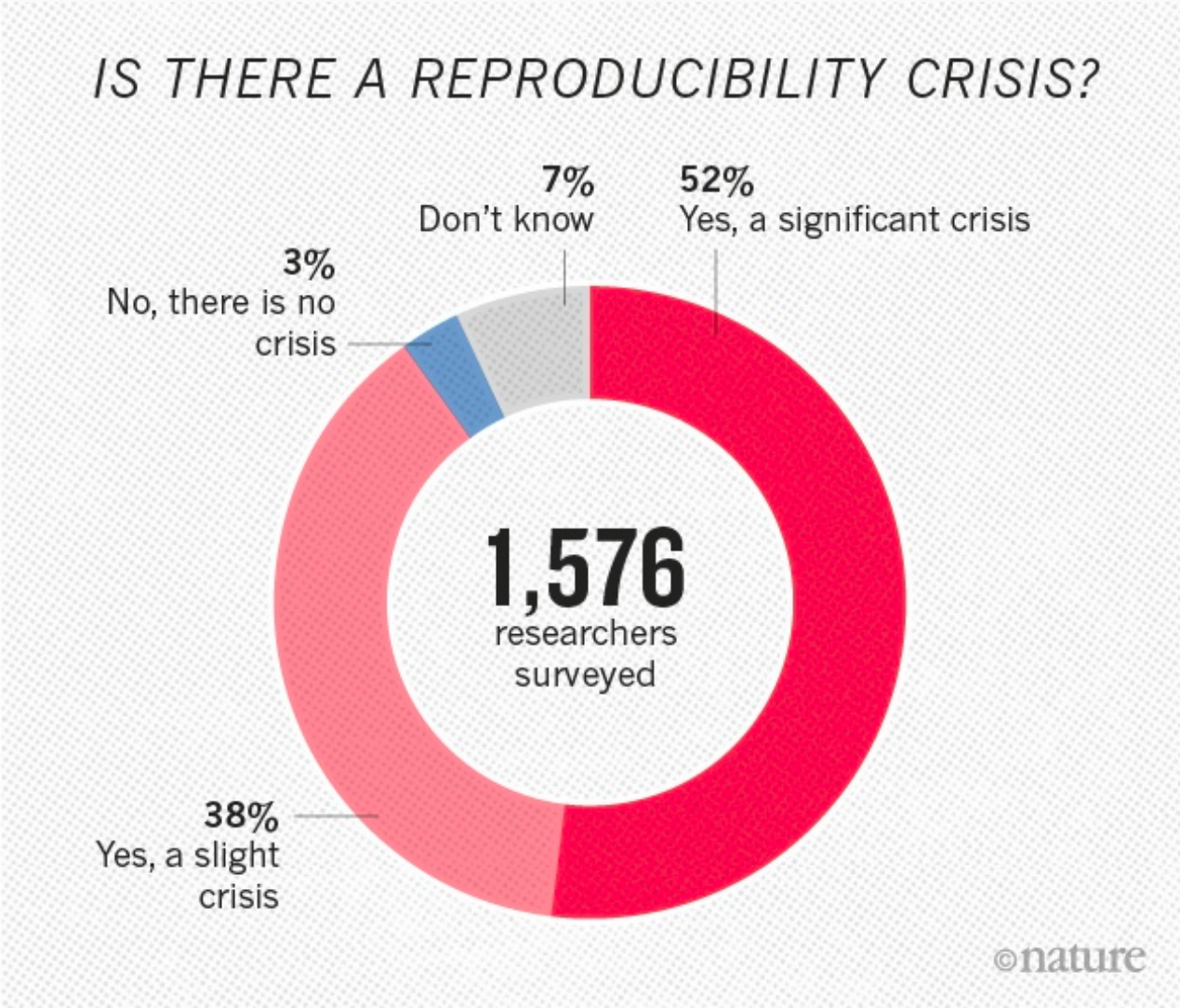
Francisco Rodriguez-Sanchez

https://frodriguezsanchez.net

# Outline

- WHAT is (computational) reproducibility?
- WHY is it important?
- HOW can we do reproducible research?

# The Reproducibility Crisis/Revolution


IS THERE A REPRODUCIBILITY CRISIS?
7%
Don't know
52%
Yes, a significant crisis
3%
No, there is no crisis
1,576
researchers surveyed
38%
Yes, a slight crisis
©nature

Baker 2016

# The reproducibility crisis in mass media

EL PAÍS

Materia

## La ciencia vive una epidemia de estudios inservibles

Científicos de EE UU, Reino Unido y Holanda denuncian que la investigación está perdiendo parte de su credibilidad

El País

# Reproducibility Project: Cancer Biology

NEWS | 09 December 2021

**Half of top cancer studies fail high-profile reproducibility effort**

- Goal: Replicate 193 experiments from 53 papers

Errington et al 2021

# Reproducibility Project: Cancer Biology

NEWS | 09 December 2021

**Half of top cancer studies fail high-profile reproducibility effort**

- Goal: Replicate 193 experiments from 53 papers
- Finally: 50 experiments from 23 papers

Errington et al 2021

# Reproducibility Project: Cancer Biology

NEWS | 09 December 2021

## Half of top cancer studies fail high-profile reproducibility effort

- Goal: Replicate 193 experiments from 53 papers
- Finally: 50 experiments from 23 papers
- ~Half **didn't replicate** (much smaller effect sizes)

Errington et al 2021

# Reproducibility Project: Cancer Biology

NEWS | 09 December 2021

## Half of top cancer studies fail high-profile reproducibility effort

- Goal: Replicate 193 experiments from 53 papers
- Finally: 50 experiments from 23 papers
- ~Half **didn't replicate** (much smaller effect sizes)
- **No paper reported** all required data

Errington et al 2021

# Reproducibility Project: Cancer Biology

NEWS | 09 December 2021

## Half of top cancer studies fail high-profile reproducibility effort

- Goal: Replicate 193 experiments from 53 papers
- Finally: 50 experiments from 23 papers
- ~Half **didn't replicate** (much smaller effect sizes)
- **No paper reported** all required data
- **Impossible to repeat** experiments w/o contacting authors

Errington et al 2021

# Reproducibility Project: Cancer Biology

NEWS | 09 December 2021

## Half of top cancer studies fail high-profile reproducibility effort

- Goal: Replicate 193 experiments from 53 papers
- Finally: 50 experiments from 23 papers
- ~Half **didn't replicate** (much smaller effect sizes)
- **No paper reported** all required data
- **Impossible to repeat** experiments w/o contacting authors
- 1/3 authors **didn't respond or help**

Errington et al 2021

**Sylvain Deville** ❄️👨‍🎓
@DevilleSy

Trying to reproduce the results of a paper using only what's in the Methods section

Most scientific articles

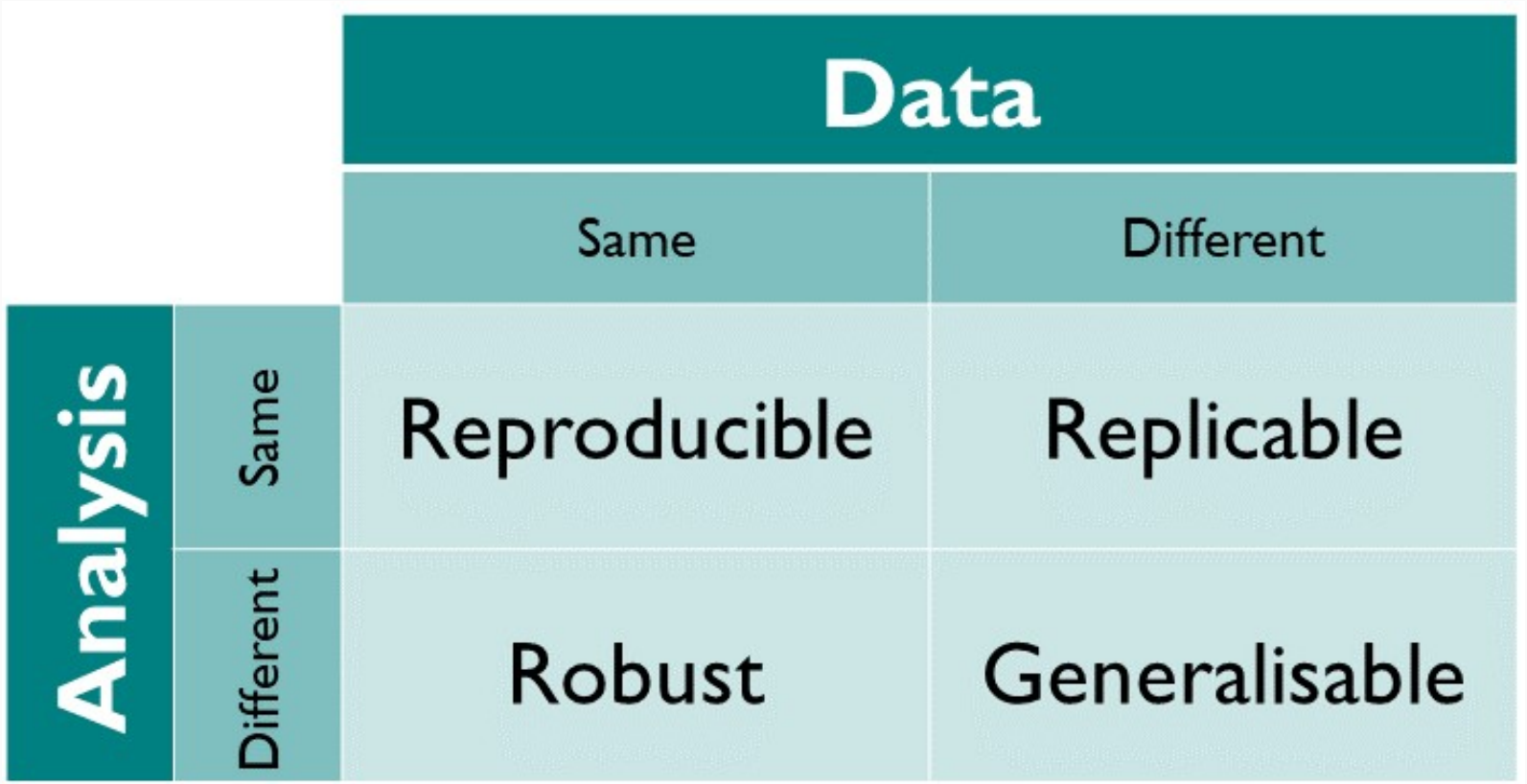
# are NOT reproducible

Reproducibility

*crisis* —> **REVOLUTION**

# What is reproducibility?

# Reproducibility vs Replicability




The Turing Way

We can’t guarantee that

our results are **REPLICABLE**.

But at least

they should be **REPRODUCIBLE**.

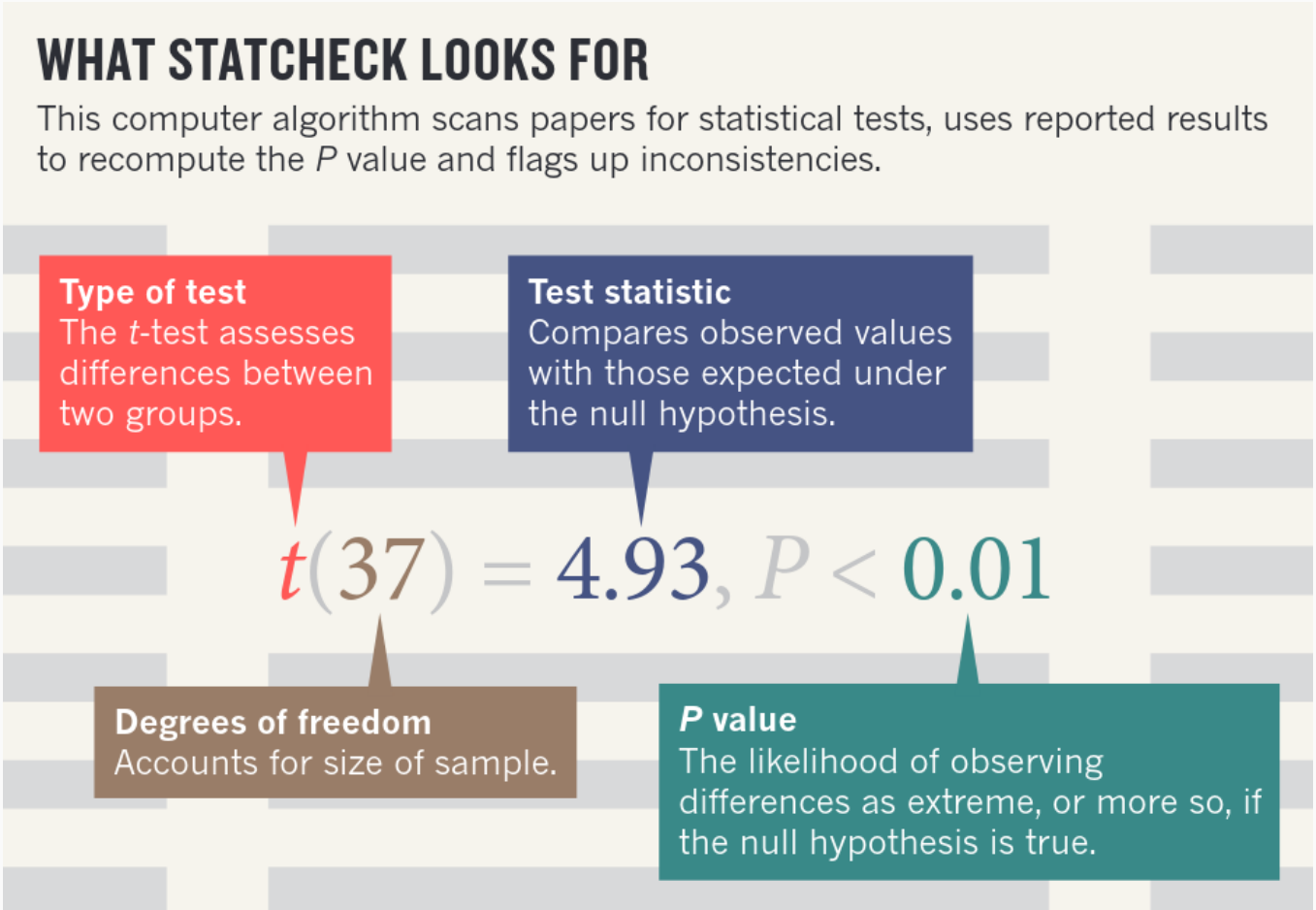
Most scientific articles

# are NOT reproducible

Behav Res (2016) 48:1205–1226
DOI 10.3758/s13428-015-0664-2

# The prevalence of statistical reporting errors in psychology (1985–2013)

Michèle B. Nuijten[1] · Chris H. J. Hartgerink[1] · Marcel A. L. M. van Assen[1] · Sacha Epskamp[2] · Jelte M. Wicherts[1]



Behav Res (2016) 48:1205–1226
DOI 10.3758/s13428-015-0664-2

**The prevalence of statistical reporting errors in psychology (1985–2013)**

**Michèle B. Nuijten[1] · Chris H. J. Hartgerink[1] · Marcel A. L. M. van Assen[1] · Sacha Epskamp[2] · Jelte M. Wicherts[1]**

1/2 articles: **inconsistencies** in p-values

1/8 articles: **grossly inconsistent** p-values

(affecting conclusions -> significance)

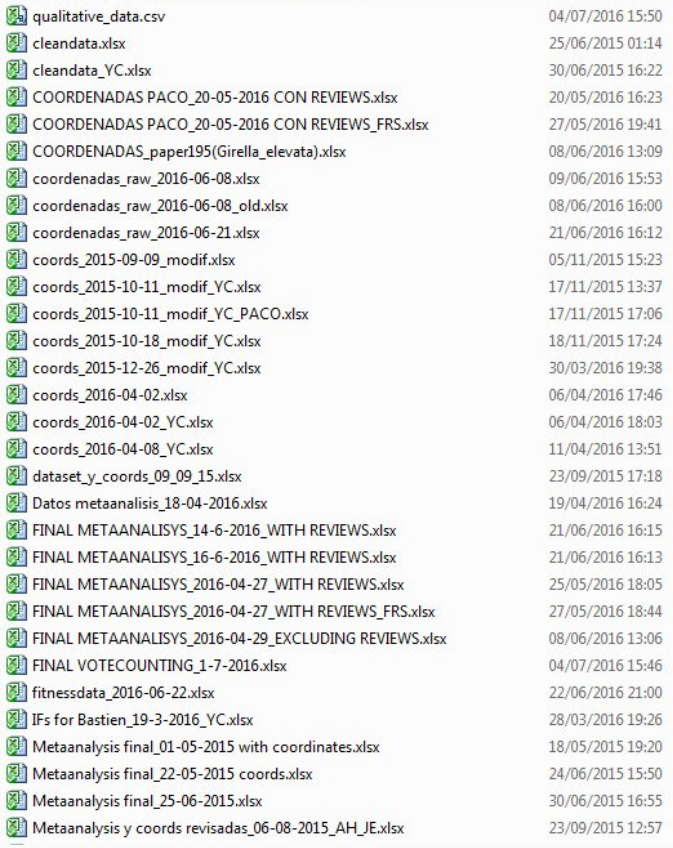
In ecology

# **< 20%** articles are **reproducible**

Culina et al 2020

# We can’t even reproduce our own work

## Data/Code lost or unusable



# What's a reproducible manuscript?


DATA + CODE

- analysis fully **traceable**
- results can be **regenerated**

A scientific article is **advertising**, not scholarship.

The actual scholarship is the **full software environment**,

**code and data**, that produced the result.

Claerbout & Karrenback 1992

# Reproducible article: text + data + code

# Are we sharing the data?

# Are we sharing data?

PERSPECTIVE

## Public Data Archiving in Ecology and Evolution: How Well Are We Doing?

**Dominique G. Roche[1,2]*, Loeske E. B. Kruuk[1,3], Robert Lanfear[1,4], Sandra A. Binning[1,2]**

**1** Division of Evolution, Ecology and Genetics, Research School of Biology, The Australian National University, Canberra, Australian Capital Territory, Australia, **2** Éco-Éthologie, Institut de Biologie, Université de Neuchâtel, Neuchâtel, Switzerland, **3** Institute of Evolutionary Biology, School of Biological Sciences, University of Edinburgh, Edinburgh, United Kingdom, **4** Department of Biological Sciences, Macquarie University, Sydney, Australia

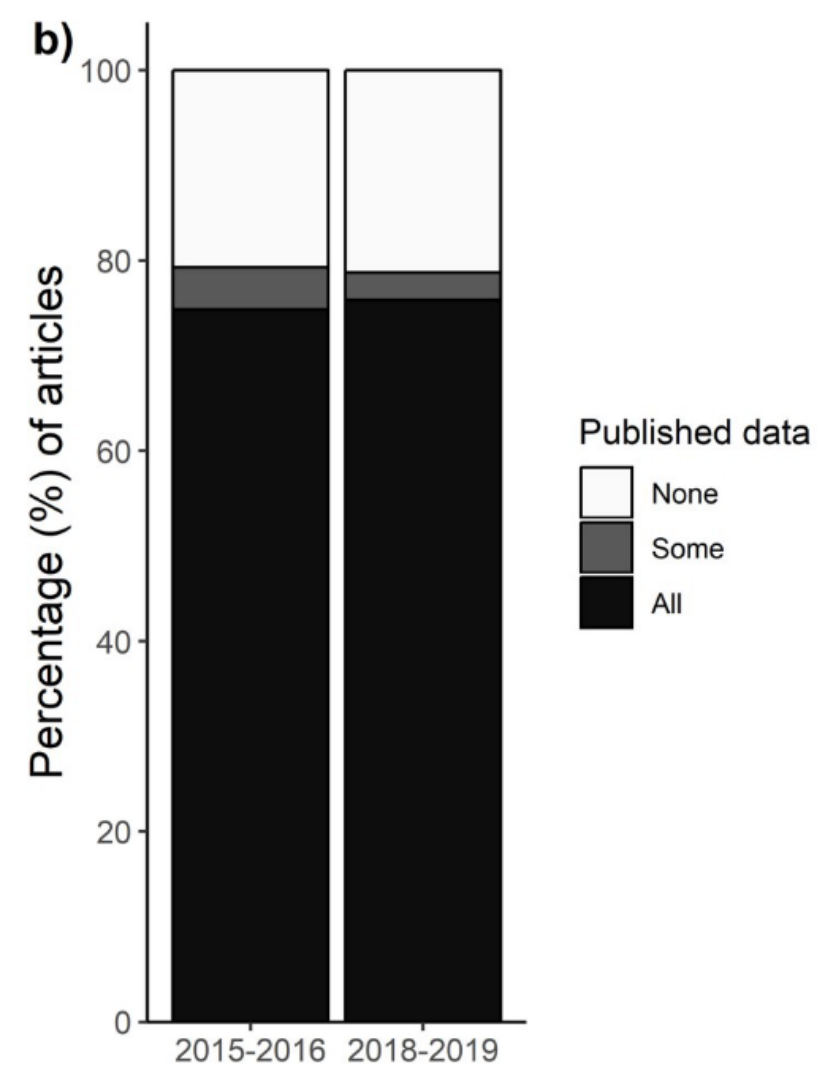
* dominique.roche@mail.mcgill.ca

### Abstract

Policies that mandate public data archiving (PDA) successfully increase accessibility to data underlying scientific publications. However, is the data quality sufficient to allow reuse and reanalysis? We surveyed 100 datasets associated with nonmolecular studies in journals that commonly publish ecological and evolutionary research and have a strong PDA policy. Out of these datasets, 56% were incomplete, and 64% were archived in a way that partially or entirely prevented reuse. We suggest that cultural shifts facilitating clearer benefits to authors are necessary to achieve high-quality PDA and highlight key guidelines to help authors increase their data's reuse potential and compliance with journal data policies.

Roche et al. 2015

# Are we sharing data?




Culina et al. 2020

## Quickly getting better

Scientific Life

Early Career Researchers Embrace Data Sharing

Hamish A. Campbell,[1,*] Mariana A. Micheli-Campbell,[1] and Vinay Udyawer[2]

Campbell et al. 2019

# Are we sharing the code?

# Code exists but rarely shared

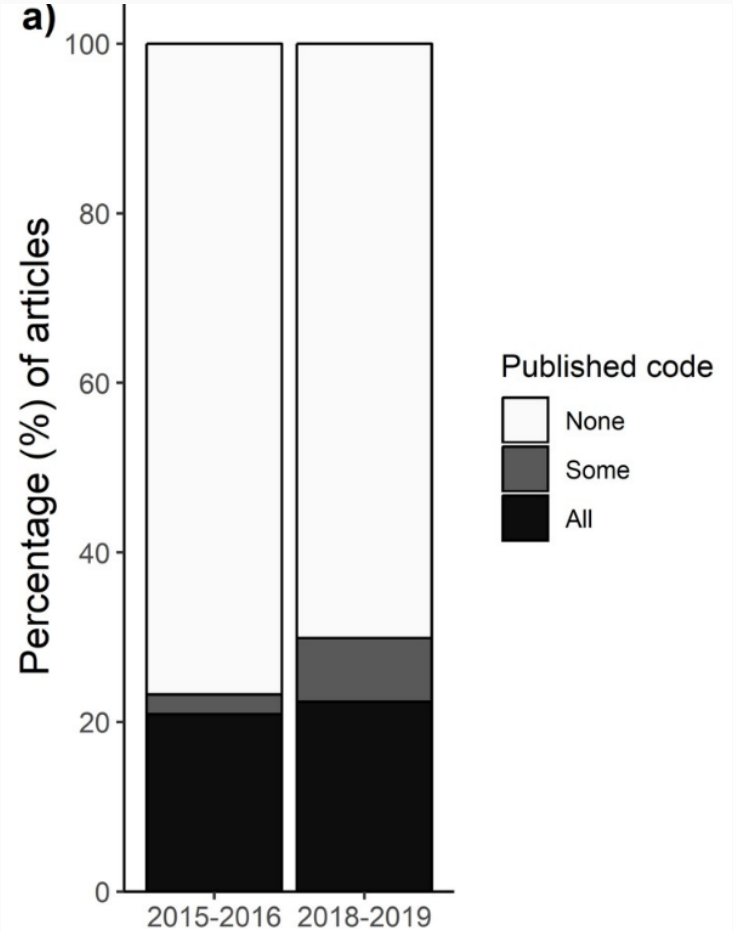
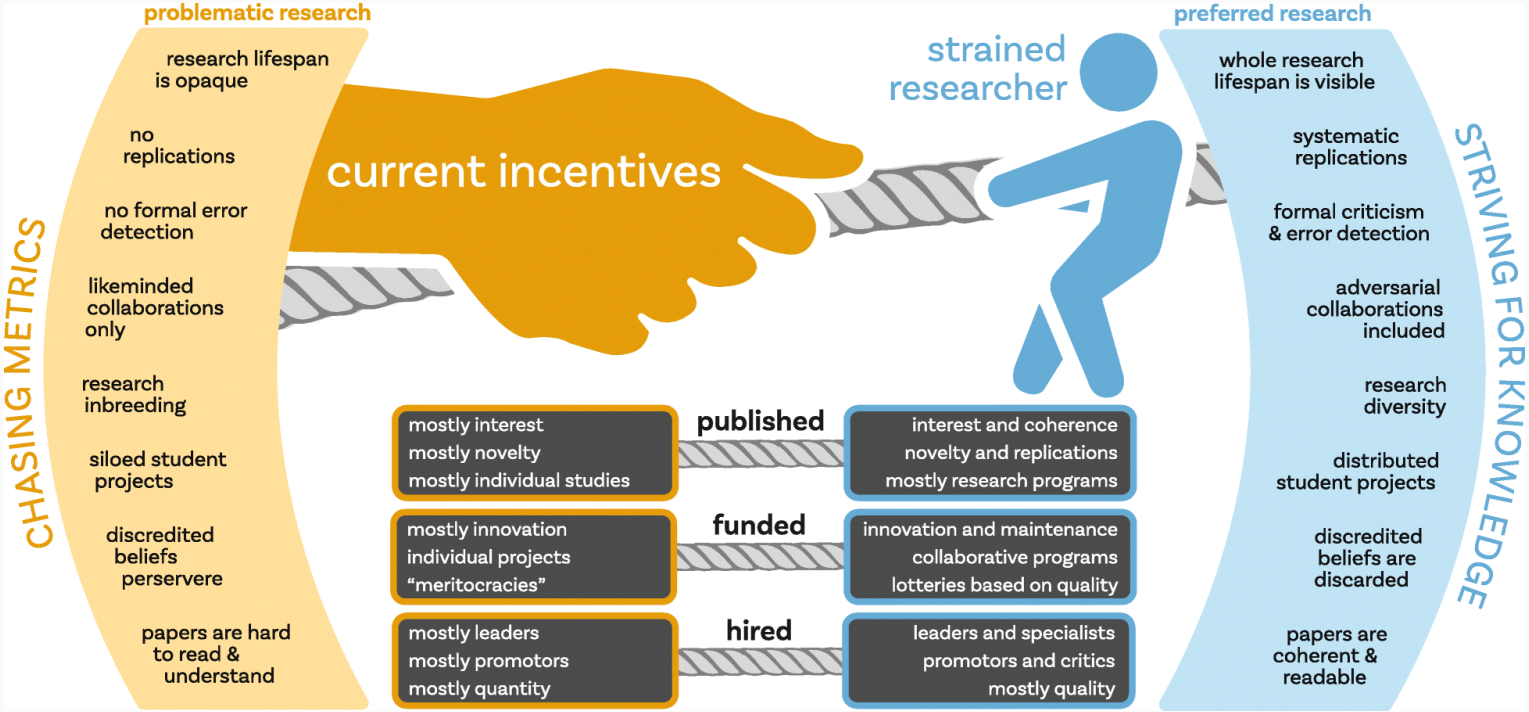



**Fig 1. Code-sharing is at its infancy in ecology, wher**

Culina et al. 2020

# WHY?

# Poor incentives




O'Dea et al 2021

# Doing reproducible research can be costly

## The Costs of Reproducibility

**Russell A. Poldrack**[1,*]
[1]Department of Psychology, Stanford University, Stanford, CA, USA
*Correspondence: poldrack@stanford.edu
https://doi.org/10.1016/j.neuron.2018.11.030

PERSPECTIVE

## Open science challenges, benefits and tips in early career and beyond

**Christopher Allen**[1]*, **David M. A. Mehler**[1,2]*

# Must value diverse contributions to reproducible research

## Credit data generators for data reuse

To promote effective sharing, we must create an enduring link between the people who generate data and its future uses, urge **Heather H. Pierce** and colleagues.

Pierce et al 2019

# Let’s knock down psychological barriers

## Publish your computer code: it is good enough

*Freely provided working code — whatever its quality — improves programming and enables others to engage with your research, says* **Nick Barnes**.

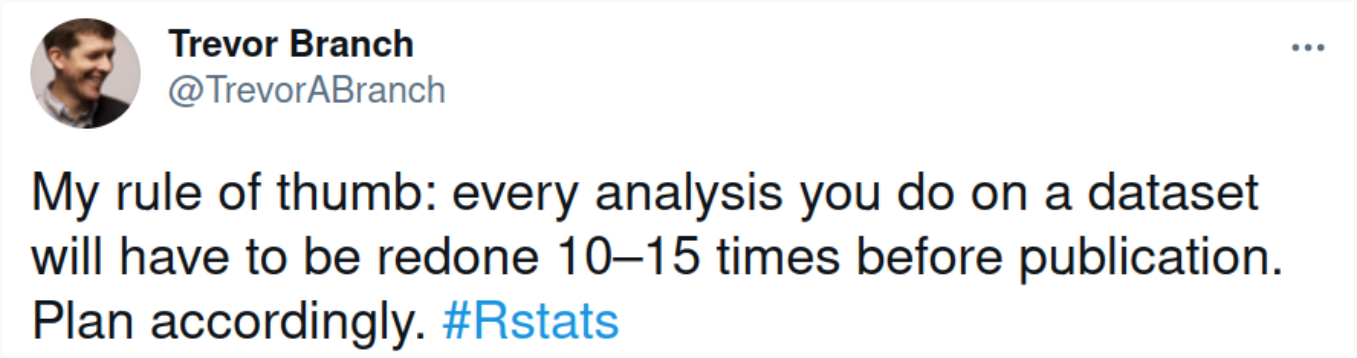
Barnes 2010

- Improve training
- Code review, preprints...
- Avoid shaming -> constructive critique
- Ugly code better than no code

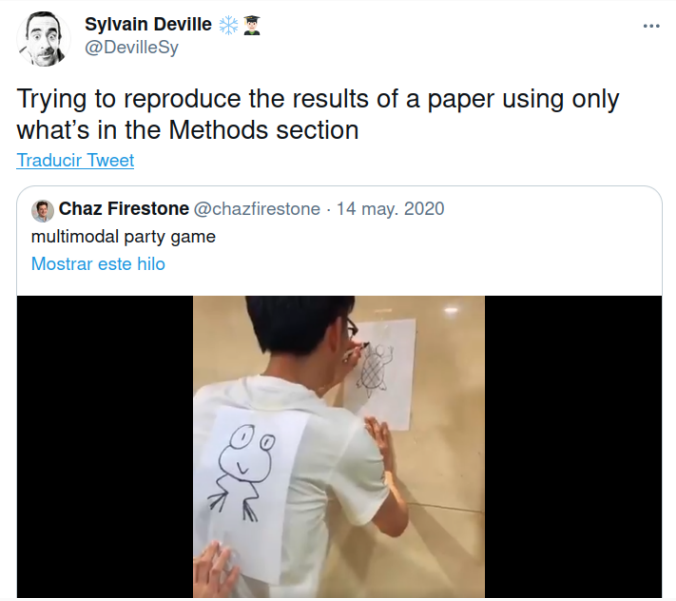
# Why doing reproducible research?

# Reproducibility: good for you, good for everyone

# Automation (good code) saves time



# Code = fully traceable, reproducible analysis




Code advantages:

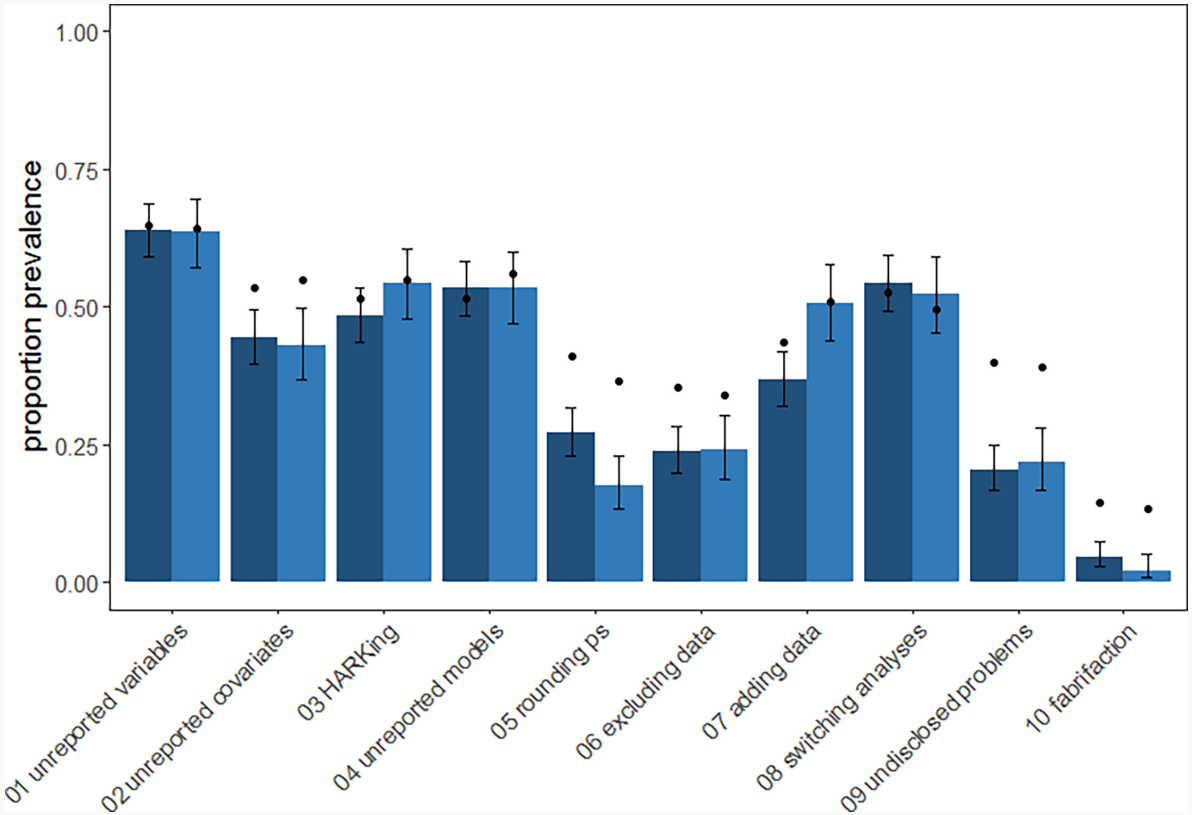
- Easier writing
- Easier, deeper review
- Reusable

# Transparency prevents bad practices

RESEARCH ARTICLE

## Questionable research practices in ecology and evolution

**Hannah Fraser[1]*, Tim Parker[2], Shinichi Nakagawa[3], Ashley Barnett[1], Fiona Fidler[1,4]**




p-hacking, HARKing, data fabrication...



Fraser et al 2018

# Transparency avoids unnecessary disputes

DOI:10.1063/PT.6.1.20180822a

22 Aug 2018 in Research & Technology

## The war over supercooled water

How a hidden coding error fueled a seven-year dispute between two of condensed matter's top theorists.

**Ashley G. Smart**

Over the next seven years, the perplexing discrepancy would ignite a bitter conflict, with junior scientists caught in the crossfire. At stake were not only the reputations of the two groups but also a peculiar theory that sought to explain some of water's deepest and most enduring mysteries. Earlier this year, the dispute was finally settled. And as it turns out, the entire ordeal was the result of botched code.

# Transparency brings better science

**Alexey Shiklomanov**
@ashiklom711

I'm co-author on a study currently published only as a publicly available discussion paper. My code was on GitHub.

A colleague read the paper, thought the results looked weird, checked my code, found a bug and emailed me about it.

This is how science should work. #openscience

# Many journals (and funders) value/require reproducibility

As a condition for publication in ESA journals, all underlying data and statistical code pertinent to the results presented in the publication must be made available in a permanent, publicly accessible data archive or repository, with rare exceptions (see

# Many journals value reproducibility

'Papers with exemplary **data and code archiving**

are **more valuable** for future research and [...]

will be given **higher priority** for publication'

(*Molecular Ecology*)

# Many journals require reproducibility


EDITORIAL

ECOLOGY LETTERS cnrs WILEY

**From raw data to publication: Introducing data editing at Ecology Letters**

'We require the **data and code** for reproducing statistical results and generating figures and tables'

'This material will need to be supplied at the **time of submission**'

# Higher impact: cites, reuse, reputation

RESEARCH ARTICLE

## The citation advantage of linking publications to research data

**Giovanni Colavizza[1,2], Iain Hrynaszkiewicz[3,4], Isla Staden[1,5], Kirstie Whitaker[1,6], Barbara McGillivray[1,6]***

Colavizza et al 2020

ACADEMIC PRACTICE IN ECOLOGY AND EVOLUTION

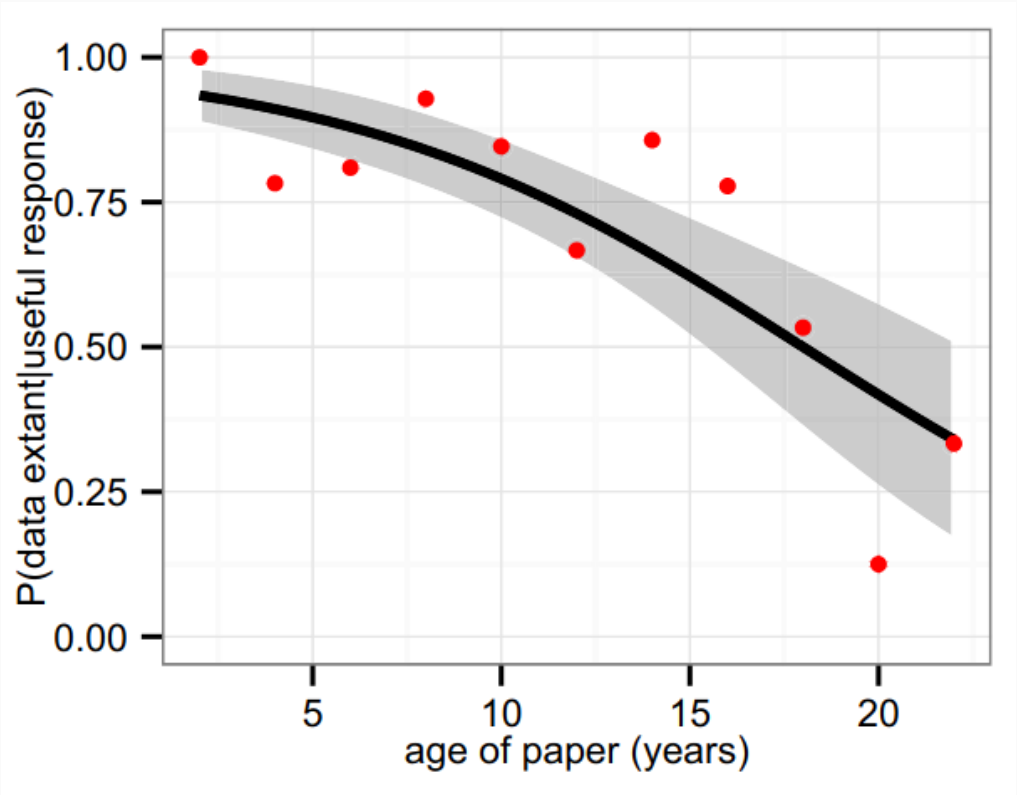
Ecology and Evolution Open Access WILEY

## Code sharing in ecology and evolution increases citation rates but remains uncommon

Maitner et al 2024

# Let’s stop losing data & code

**The Availability of Research Data Declines Rapidly with Article Age**



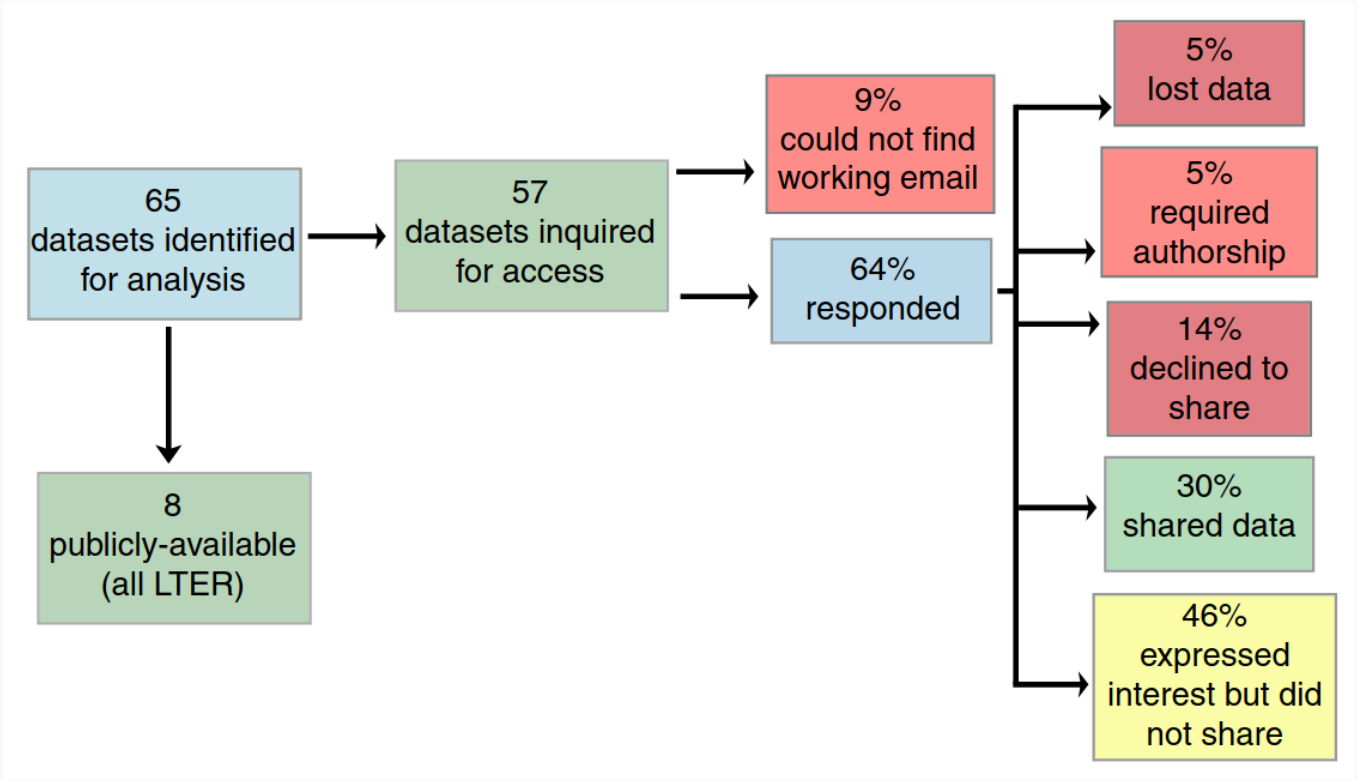

Vines et al 2014

# Open data & code enable synthesis

REVIEW

## Advances in global change research require open science by individual researchers

ELIZABETH M. WOLKOVICH*†, JAMES REGETZ‡ and MARY I. O'CONNOR†



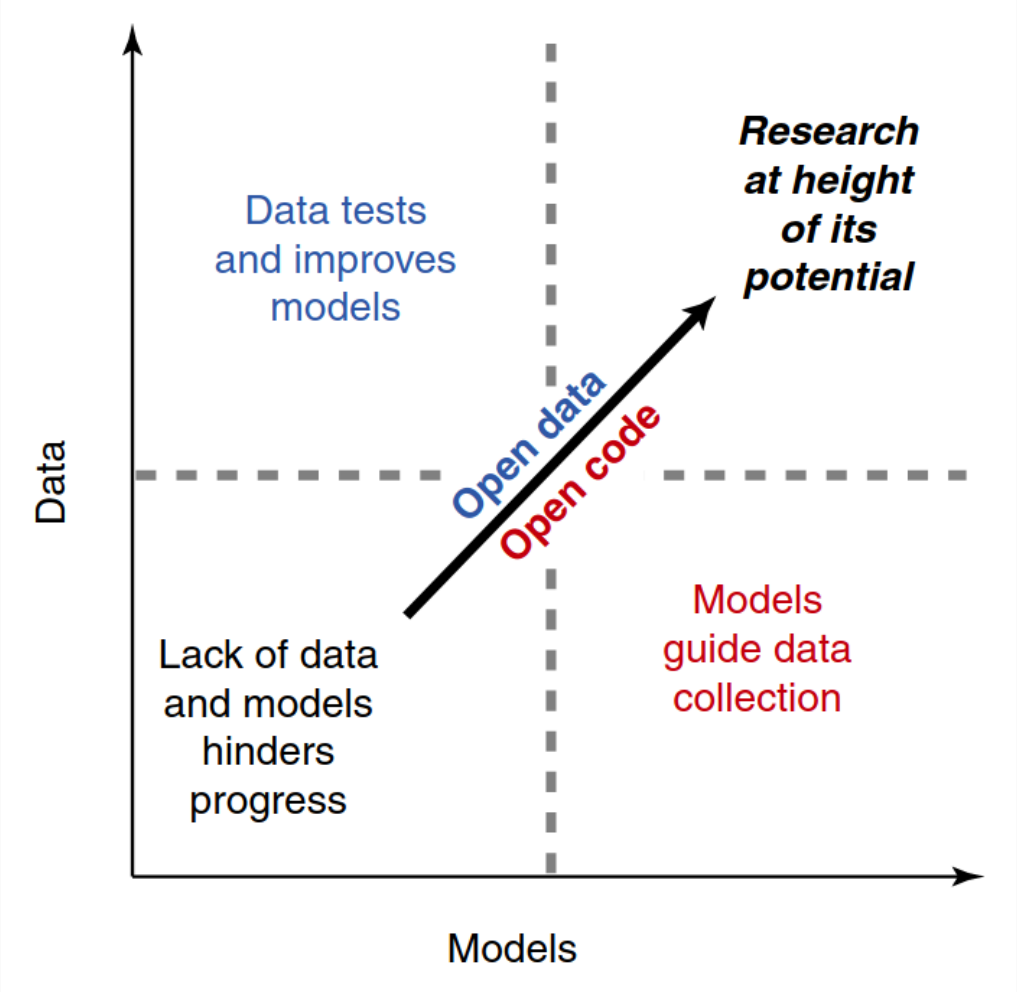

Wolkovich et al 2012

# Open data & code enable synthesis

REVIEW

**Advances in global change research require open science by individual researchers**

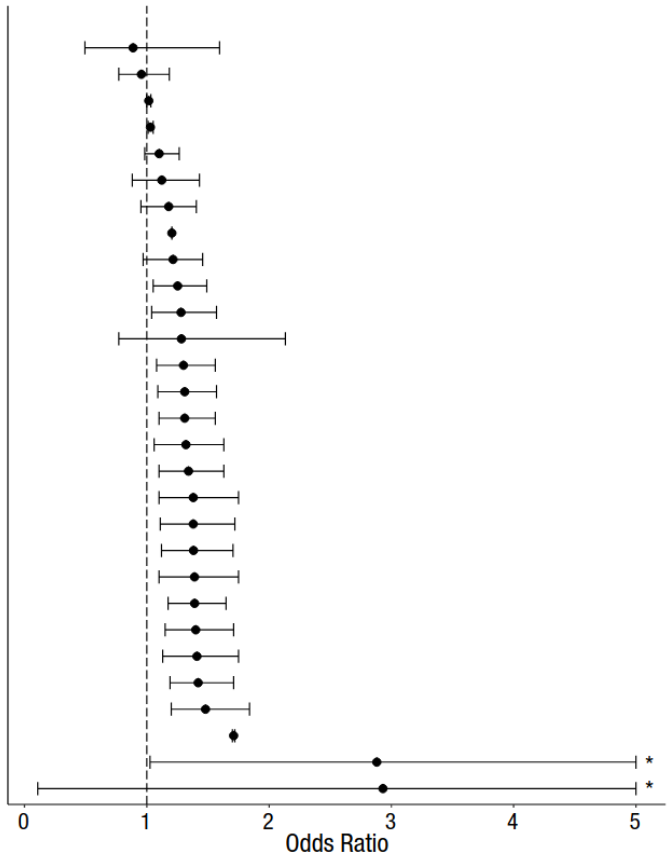
ELIZABETH M. WOLKOVICH*†, JAMES REGETZ‡ and MARY I. O'CONNOR†



# Same data -> different results

Do soccer referees give more red cards to dark-skin players?

| Team | Analytic Approach | Odds Ratio |
|---|---|---|
| 12 | Zero-Inflated Poisson Regression | 0.89 |
| 17 | Bayesian Logistic Regression | 0.96 |
| 15 | Hierarchical Log-Linear Modeling | 1.02 |
| 10 | Multilevel Regression and Logistic Regression | 1.03 |
| 18 | Hierarchical Bayes Model | 1.10 |
| 31 | Logistic Regression | 1.12 |
| 1 | OLS Regression With Robust Standard Errors, Logistic Regression | 1.18 |
| 4 | Spearman Correlation | 1.21 |
| 14 | WLS Regression With Clustered Standard Errors | 1.21 |
| 11 | Multiple Linear Regression | 1.25 |
| 30 | Clustered Robust Binomial Logistic Regression | 1.28 |
| 6 | Linear Probability Model | 1.28 |
| 26 | Hierarchical Generalized Linear Modeling With Poisson Sampling | 1.30 |
| 3 | Multilevel Logistic Regression Using Bayesian Inference | 1.31 |
| 23 | Mixed-Model Logistic Regression | 1.31 |
| 16 | Hierarchical Poisson Regression | 1.32 |
| 2 | Linear Probability Model, Logistic Regression | 1.34 |
| 5 | Generalized Linear Mixed Models | 1.38 |
| 24 | Multilevel Logistic Regression | 1.38 |
| 28 | Mixed-Effects Logistic Regression | 1.38 |
| 32 | Generalized Linear Models for Binary Data | 1.39 |
| 8 | Negative Binomial Regression With a Log Link | 1.39 |
| 20 | Cross-Classified Multilevel Negative Binomial Model | 1.40 |
| 13 | Poisson Multilevel Modeling | 1.41 |
| 25 | Multilevel Logistic Binomial Regression | 1.42 |
| 9 | Generalized Linear Mixed-Effects Models With a Logit Link | 1.48 |
| 7 | Dirichlet-Process Bayesian Clustering | 1.71 |
| 21 | Tobit Regression | 2.88 |
| 27 | Poisson Regression | 2.93 |



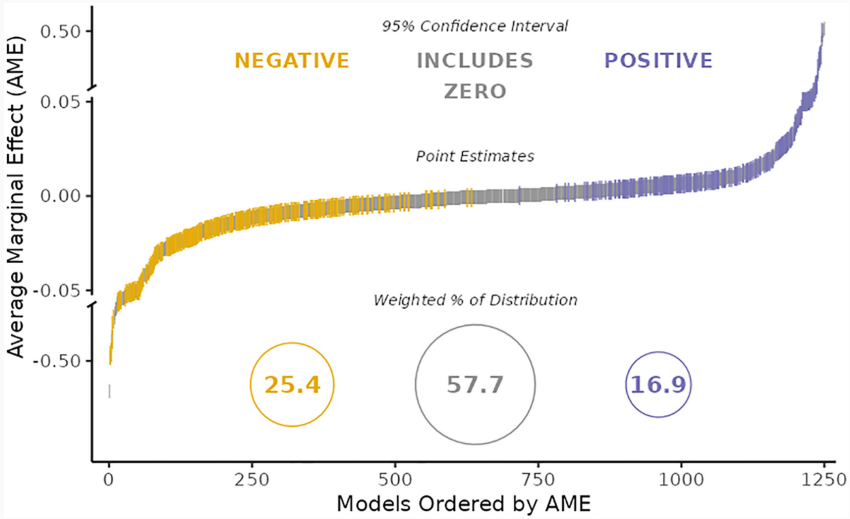

29 teams: 2/3 found significant effect

Silberzahn et al 2018

# 73 teams testing the same hypothesis with the same data

PNAS RESEARCH ARTICLE | SOCIAL SCIENCES OPEN ACCESS

**Observing many researchers using the same data and hypothesis reveals a hidden universe of uncertainty**



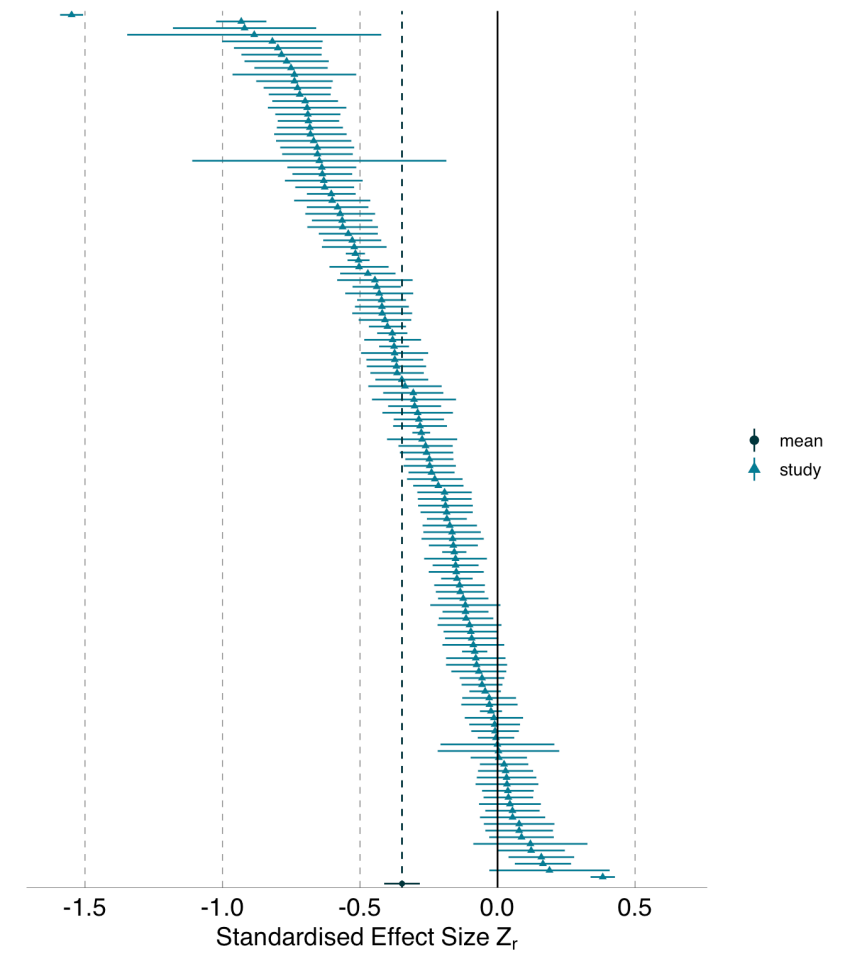

'This reveals a **universe of uncertainty** that remains hidden when considering a single study in isolation'

'These results call for greater **epistemic humility** and **clarity in reporting** scientific findings'

Breznau et al 2022

# 132 teams asking same question with same data

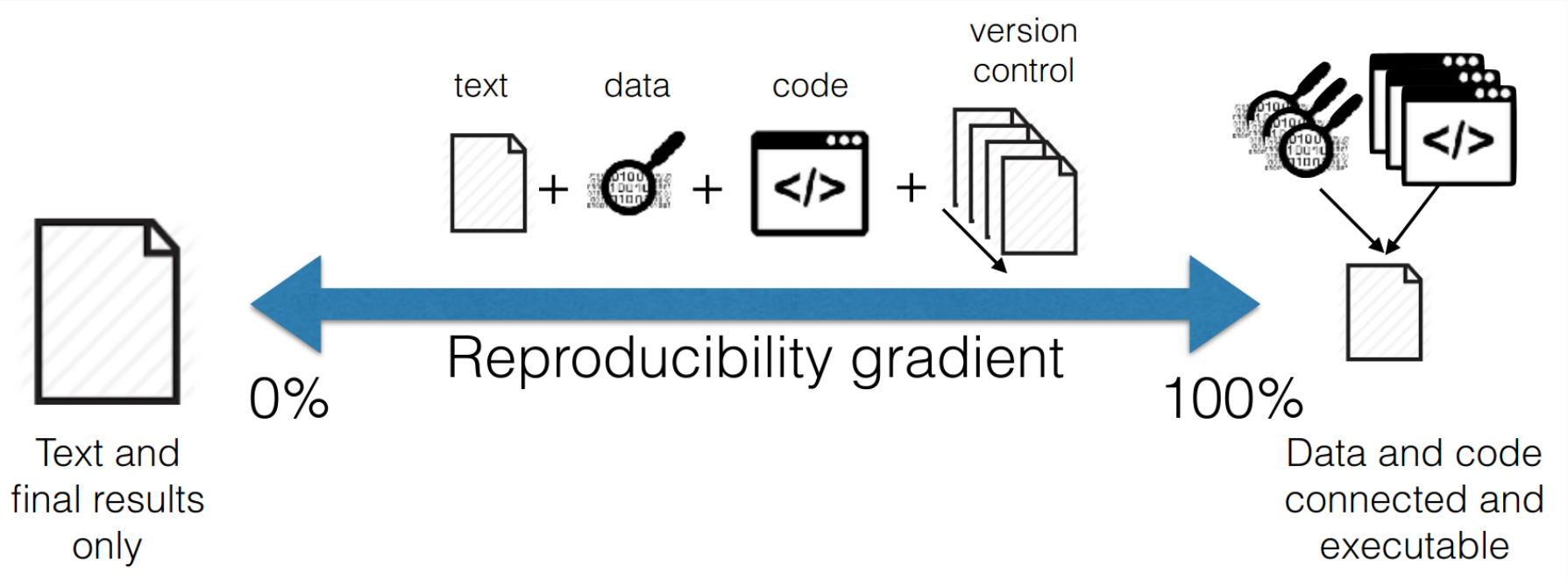
How does sibling competition affect nestling growth in blue tits?




Gould et al 2023

# How to do reproducible research

# Reproducibility is a gradient




Rodríguez-Sánchez et al. 2016 (modif. Peng 2011)

# Basic reproducibility

# Basic reproducibility

- **MANUSCRIPT** (Text + Tables + Figures)
- **DATA** in permanent archive (see Tierney & Ram 2021)
- **CODE** in permanent archive (see Eglen et al 2016)

*Permanent archive*:

- Zenodo, Dryad, OSF, Figshare, Data Paper...
- NOT GitHub, website...

# How to share data

- **Open** format (csv, txt...)
- **README** (who, what, when, where, why, how)
- **Describe variables**
- **Licence** (CC0, CC-BY, ODbL)
- **Citation** (DOI)
- **Metadata** standardised (JSON, XML)

Tierney & Ram 2021

# Document your data

```
library('dataspice')
create_spice()   # create CSV templates for metadata

edit_creators()  # open Shiny apps to edit the CSVs
prep_access()
edit_access()
prep_attributes()
edit_attributes()
edit_biblio()

write_spice()  # write machine-readable metadata

build_site()  # build human-readable metadata report
```

# How to share code

- Scripts: **plain text** (`.R`)
- **Permanent archive** (eg. Zenodo) with **DOI (citable)**
- Licence
- **README**
- Computational **environment** (session info)

Eglen et al 2016

# `sessionInfo` records OS & used packages

```
## R version 4.4.2 (2024-10-31)
## Platform: x86_64-pc-linux-gnu
## Running under: Ubuntu 20.04.6 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/liblapack.so.3;  LAPACK version 3.9.0
##
## locale:
##  [1] LC_CTYPE=en_GB.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=es_ES.UTF-8        LC_COLLATE=en_GB.UTF-8
##  [5] LC_MONETARY=es_ES.UTF-8    LC_MESSAGES=en_GB.UTF-8
##  [7] LC_PAPER=es_ES.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=es_ES.UTF-8 LC_IDENTIFICATION=C
##
## time zone: Europe/Madrid
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] knitr_1.49
##
## loaded via a namespace (and not attached):
##  [1] compiler_4.4.2    fastmap_1.2.0     cli_3.6.3         htmltools_0.5.8.1
##  [5] tools_4.4.2       rstudioapi_0.17.1 yaml_2.3.10       codetools_0.2-20
##  [9] rmarkdown_2.29    binb_0.0.7        xfun_0.50         digest_0.6.37
```

# `renv` also records packages used

```
renv::snapshot()
```

creates `renv.lock` file recording dependencies.

Can use `renv::restore()` to restore packages later or in different computer.

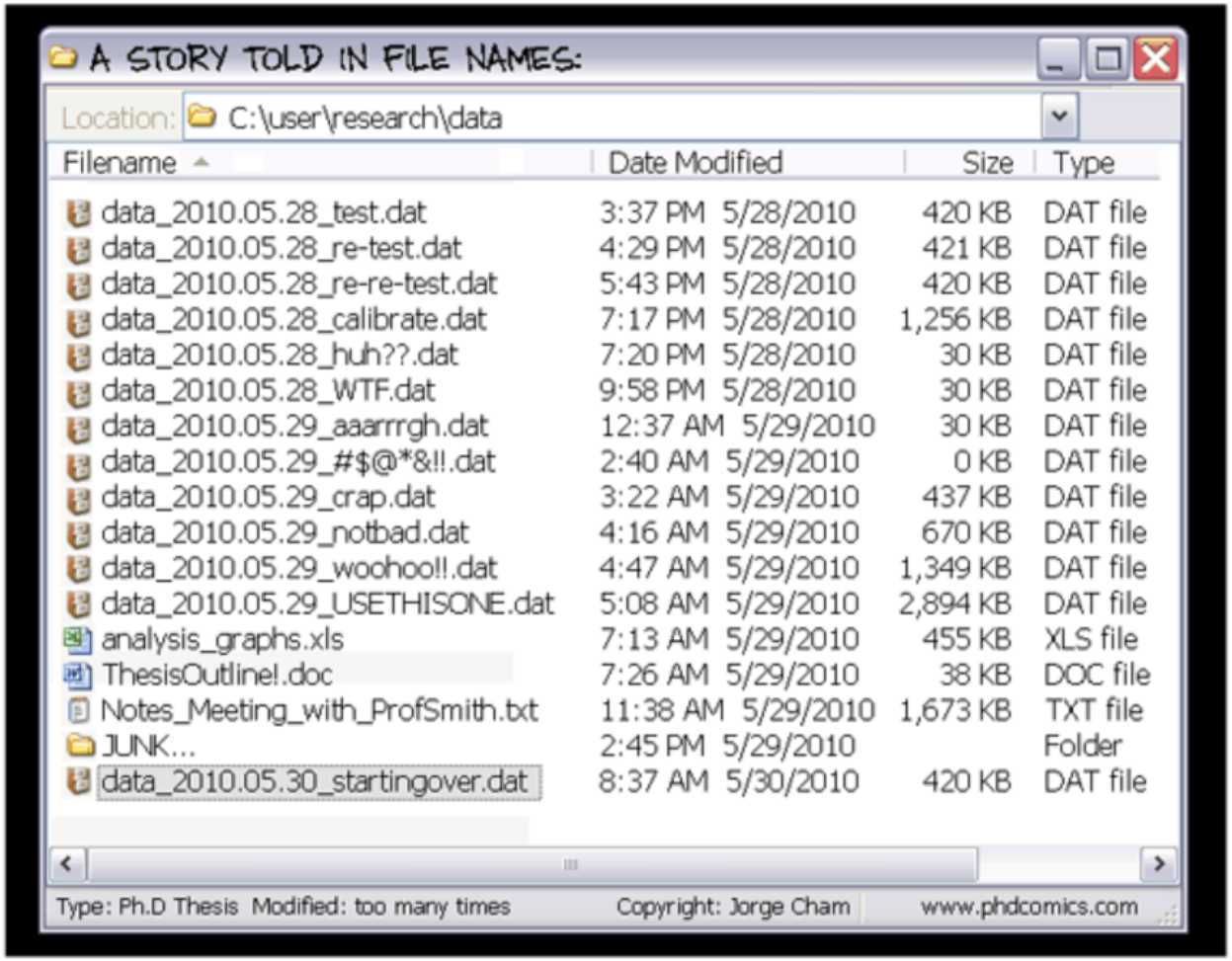
# Basic reproducibility


DATA + CODE

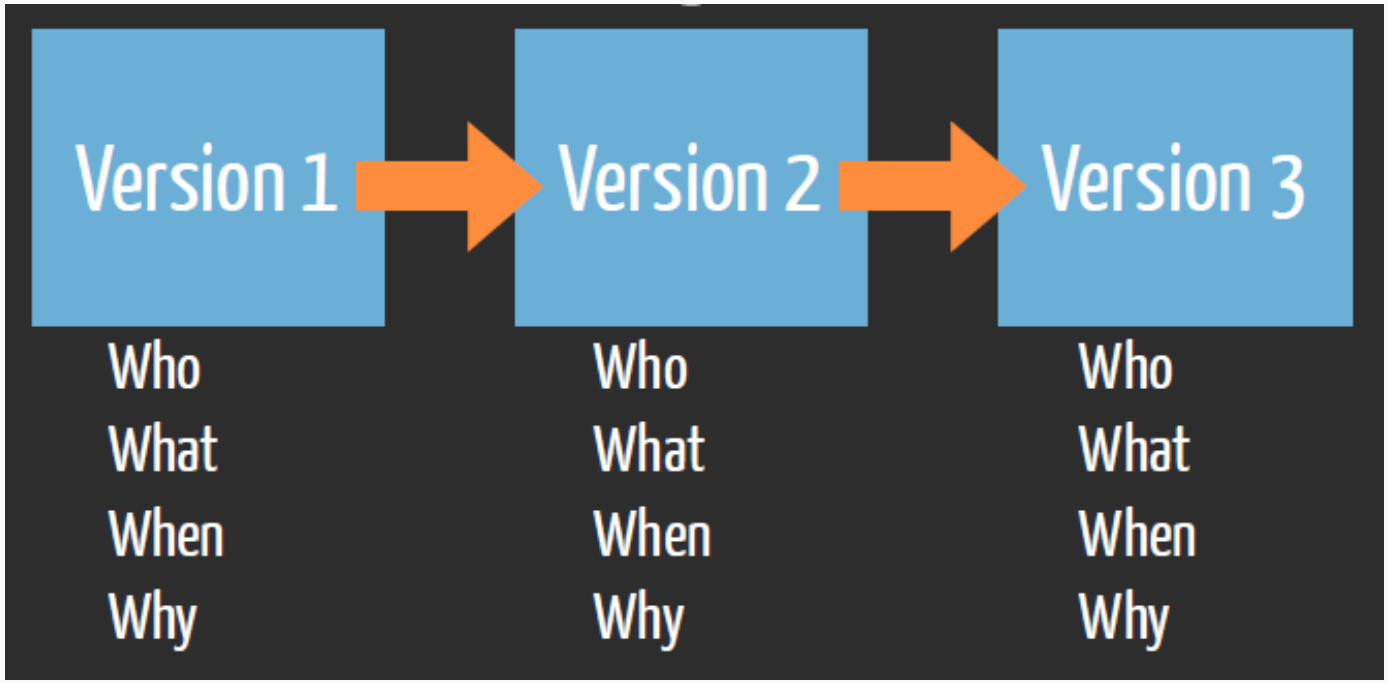
- analysis fully **traceable**
- results can be **regenerated**

# Version control

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



# Version control with `git`

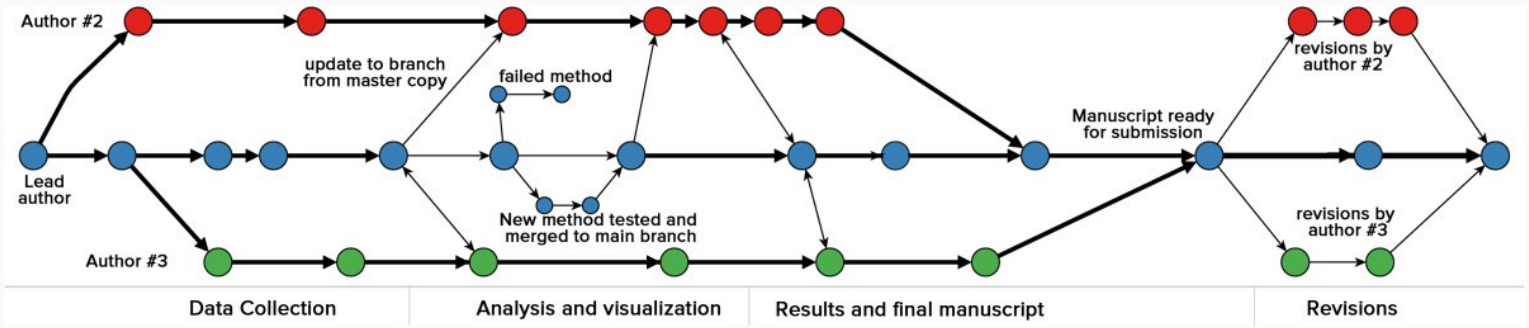



R. Fitzjohn

# Much to learn from software engineering

## Git can facilitate greater reproducibility and increased transparency in science

Karthik Ram

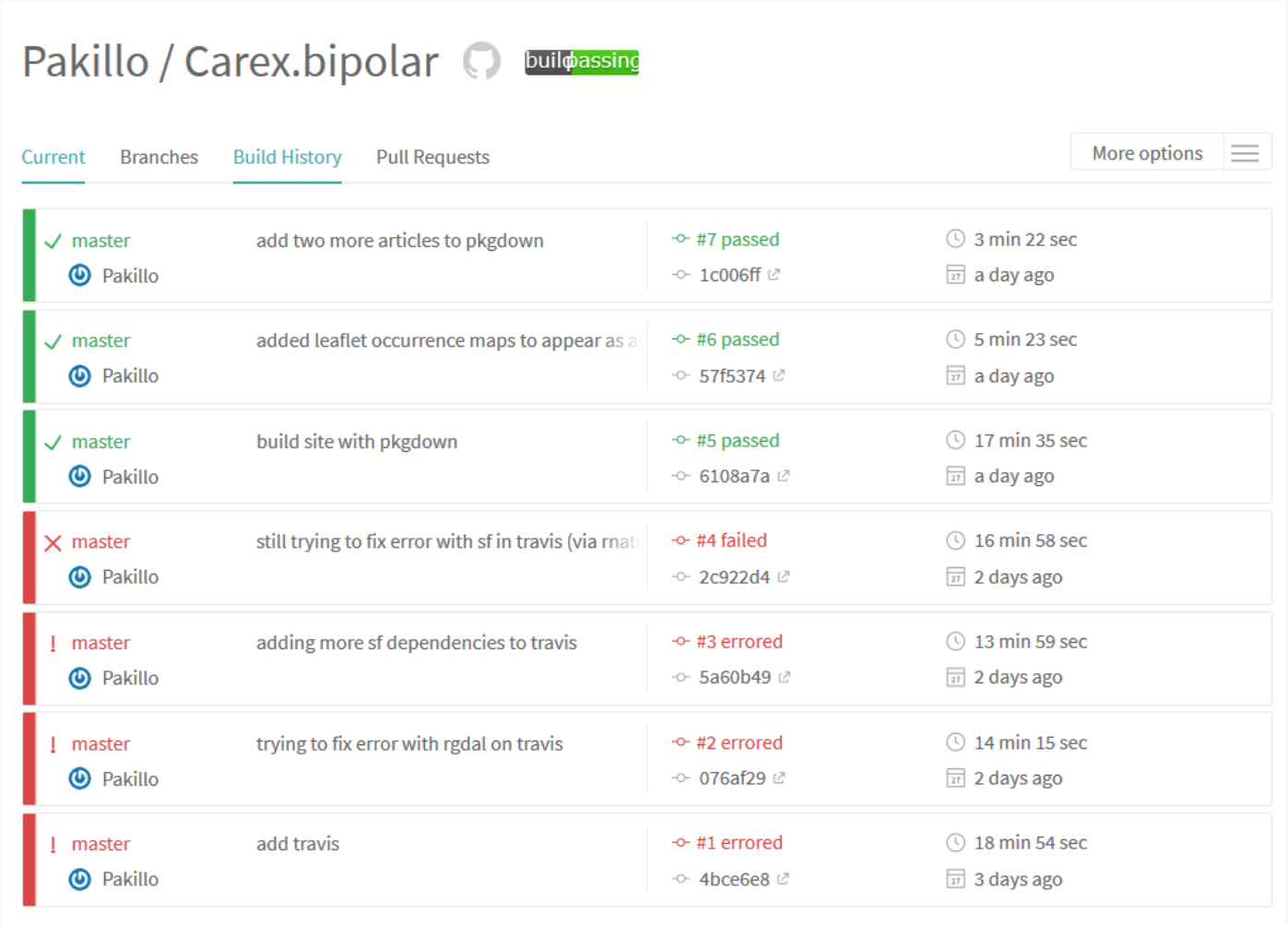



Ram 2013

# Automatic checks with Continuous Integration

Reproducibility of computational workflows is automated using continuous analysis

Brett K Beaulieu-Jones[1] & Casey S Greene[2]



# Structuring projects

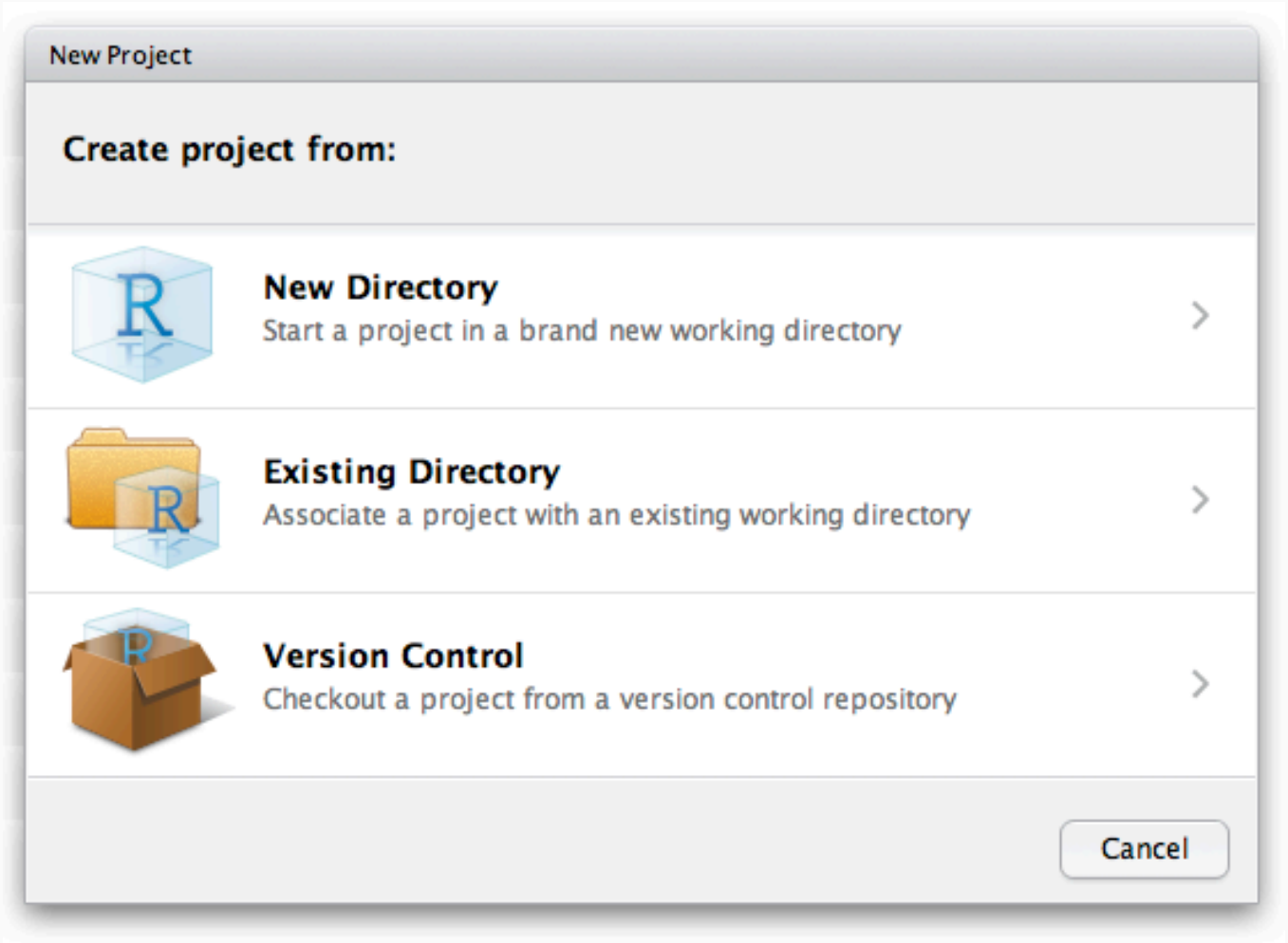
# One Project = One Folder

```
myproject
  |
  |- data
  |
  |- code
  |
  |- output (figures etc)
  |
  |- manuscript
```

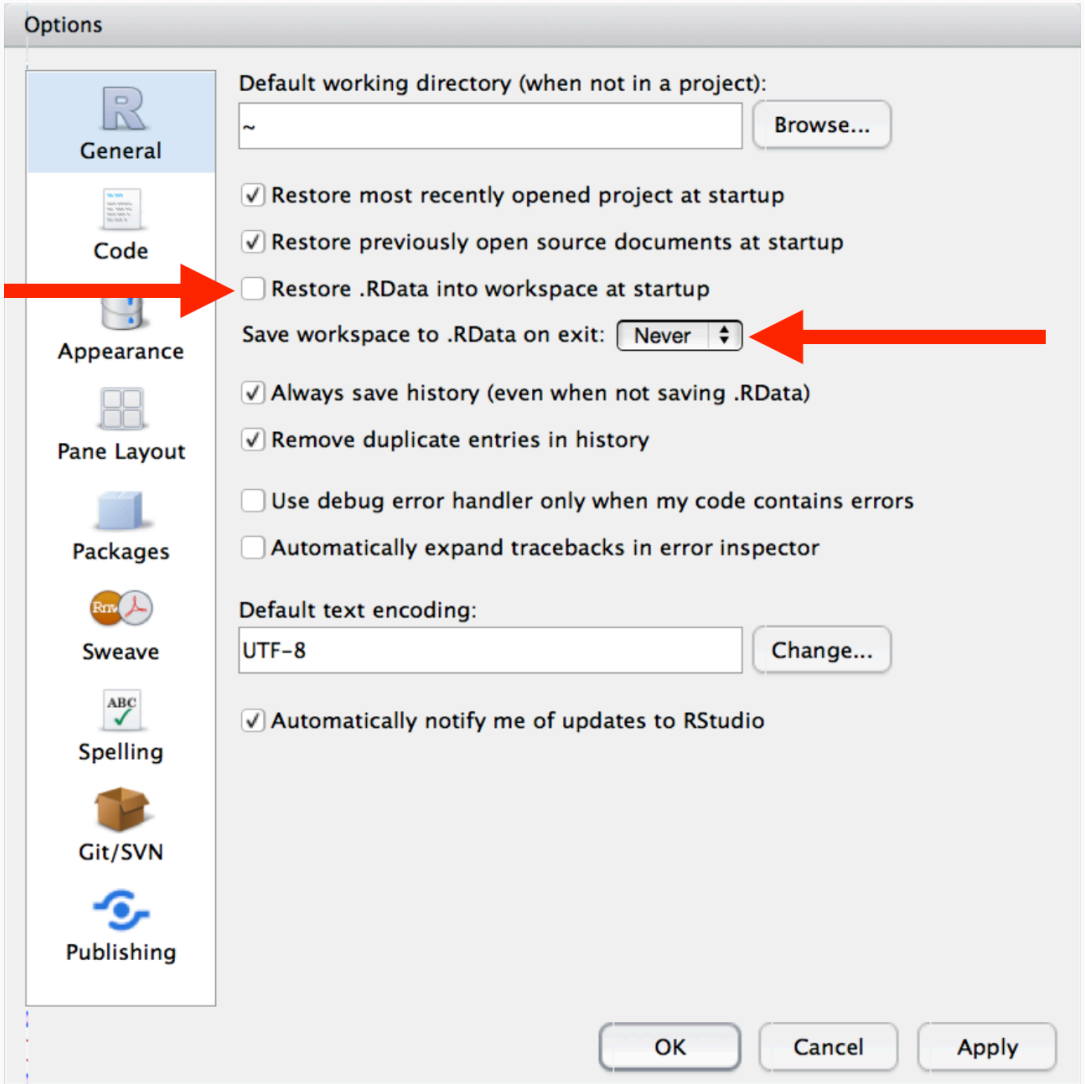
# Project-Oriented Workflow: advantages

- Self-contained
- Easy to navigate (file paths)
- Easy to share

# Rstudio projects

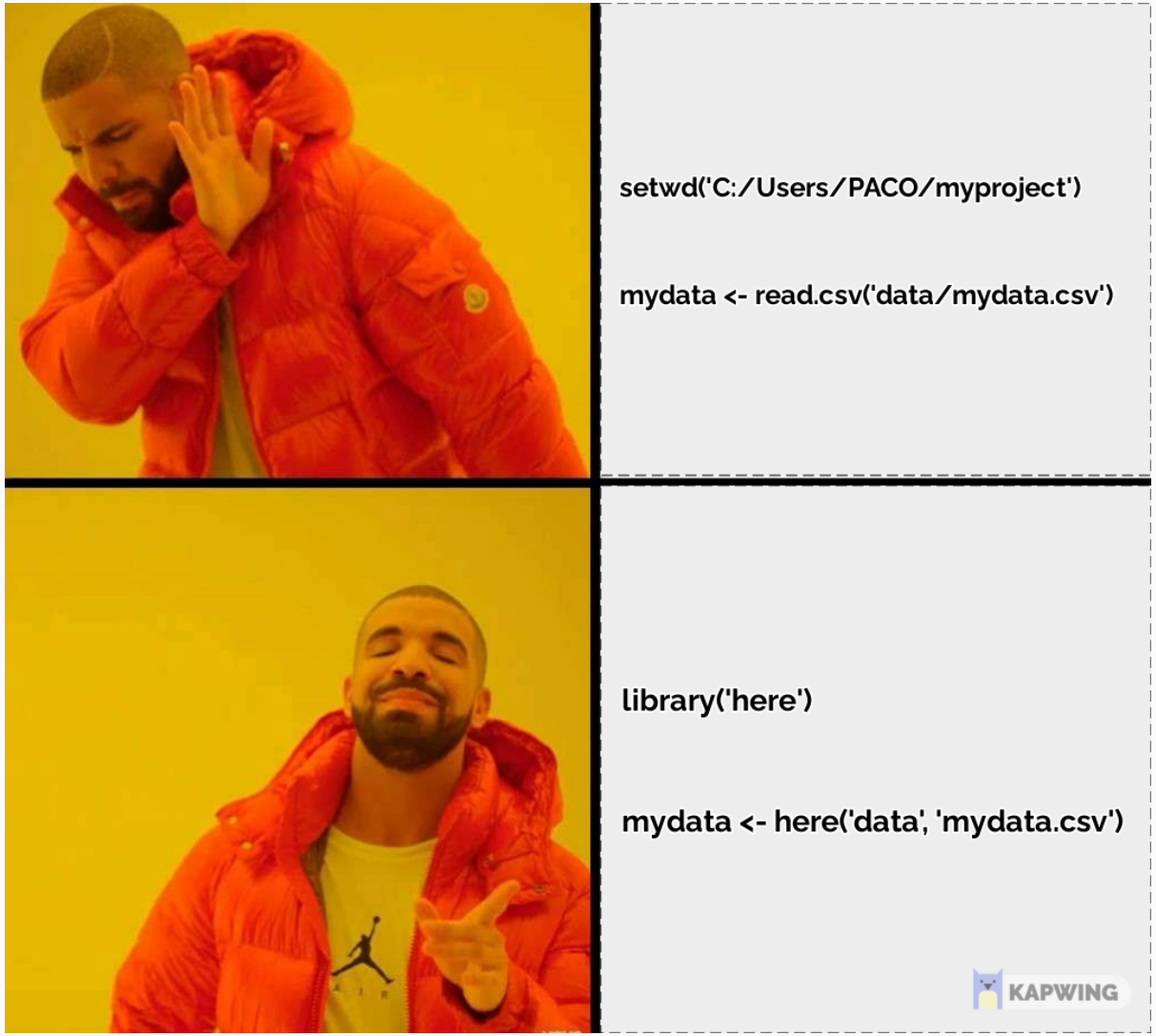


# Avoid saving workspace




https://rstats.wtf

# Use here for file paths



# `fertile` package: real-time feedback on reproducibility

```
library('fertile')

setwd("C:/Users/FRS")
```

*Error: setwd() is likely to break reproducibility. Use here::here() instead.*

https://github.com/baumer-lab/fertile

# Structuring projects: guidelines

# Guidelines for structuring projects

- All files in **same directory**

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**
- **Modular code** (functions)

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**
- **Modular code** (functions)
- **Output disposable & separate** from code

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**
- **Modular code** (functions)
- **Output disposable & separate** from code
- `makefile` runs analyses in **appropriate order**

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**
- **Modular code** (functions)
- **Output disposable & separate** from code
- `makefile` runs analyses in **appropriate order**
- **Software dependencies** under control

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**
- **Modular code** (functions)
- **Output disposable & separate** from code
- `makefile` runs analyses in **appropriate order**
- **Software dependencies** under control
- README

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Guidelines for structuring projects

- All files in **same directory**
- **Raw data** separate from **clean data**
- **Modular code** (functions)
- **Output disposable & separate** from code
- `makefile` runs analyses in **appropriate order**
- **Software dependencies** under control
- README
- License

Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

# Project organisation example

- data
  - data-raw
  - data-clean

- code

- output (figures etc)

- manuscript

- README
- License
- Makefile

# README file

- What
- Who
- How
- Licence
- Citation
- etc

**README.md**

## pandanusisotopes

launch binder

This repository contains the data and code for our paper:

> Florin, A. et al. (2020). *Palaeoprecipitation data from Madjedbebe, northern Australia: A novel proxy from ancient pandanus.*

### How to cite

Please cite this compendium as:

> Marwick, B., A. Florin et al., (2020). *Compendium of R code and data for Palaeoprecipitation data from Madjedbebe, northern Australia: A novel proxy from ancient pandanus.* Accessed 16 Oct 2020. Online at https://doi.org/xxx/xxx

### How to download

You can download the compendium as a zip from from this URL: https://github.com/benmarwick/pandanusisotopes/archive/master.zip

### Licenses

**Text and figures :** CC-BY-4.0

**Code :** See the DESCRIPTION file

**Data :** CC-0 attribution requested in reuse

# Document your data

```
library("dataspice")
create_spice()   # create CSV templates for metadata

edit_creators()  # open Shiny apps to edit the CSVs
prep_access()
edit_access()
prep_attributes()
edit_attributes()
edit_biblio()

write_spice()  # write machine-readable metadata

build_site()  # build human-readable metadata report
```

# Write modular code

Break up scripts

```
prepare_data.R

run_analysis.R

make_figures.R
```

(and `makefile` will run them in the right order)

# `makefile` runs code in appropriate order

`makefile.R`

```
source("prepare_data.R")

source("run_analysis.R")

source("make_figures.R")
```

# Don’t Repeat Yourself (DRY)

```
dataset |>
  filter(species == "Laurus nobilis") |>
  ggplot() +
  geom_point(aes(x, y))

dataset |>
  filter(species == "Laurus azorica") |>
  ggplot() +
  geom_point(aes(x, y))
```

# Don’t Repeat Yourself

**Write functions** (documented + tested)

```
plot_species <- function(sp, data) {
  data |>
    filter(species == sp) |>
    ggplot() +
    geom_point(aes(x, y))
}
```

# Don’t Repeat Yourself

Use functions

```
plot_species(sp = "Laurus nobilis", dataset)

plot_species(sp = "Laurus azorica", dataset)
```

# Don’t Repeat Yourself

Use for loops

```
for (i in species) {
  plot_species(sp = i, dataset)
}
```

# Don’t Repeat Yourself

Good ol’ `lapply`

```
lapply(species, plot_species, data = dataset)
```

# Don’t Repeat Yourself

```
library("purrr")


map(species, plot_species, data = dataset)
```

# Comment your code

**Why** rather than **What**

```
## Response is not linear, so fit gam rather than lm

model.height <- gam(height ~ s(diameter), data = trees)
```

# Use meaningful names for objects

```
m1 <- lm(height ~ diameter, data = trees)
m2 <- gam(height ~ s(diameter), data = trees)
```

# Use meaningful names for objects

```
m1 <- lm(height ~ diameter, data = trees)
m2 <- gam(height ~ s(diameter), data = trees)
```

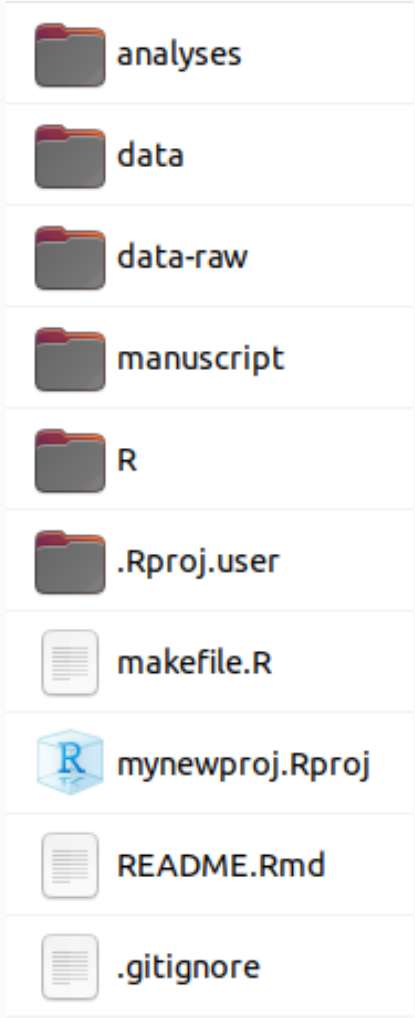
```
model.linear <- lm(height ~ diameter, data = trees)
model.gam <- gam(height ~ s(diameter), data = trees)
```

# Project templates

# Automatic project creation with `template`

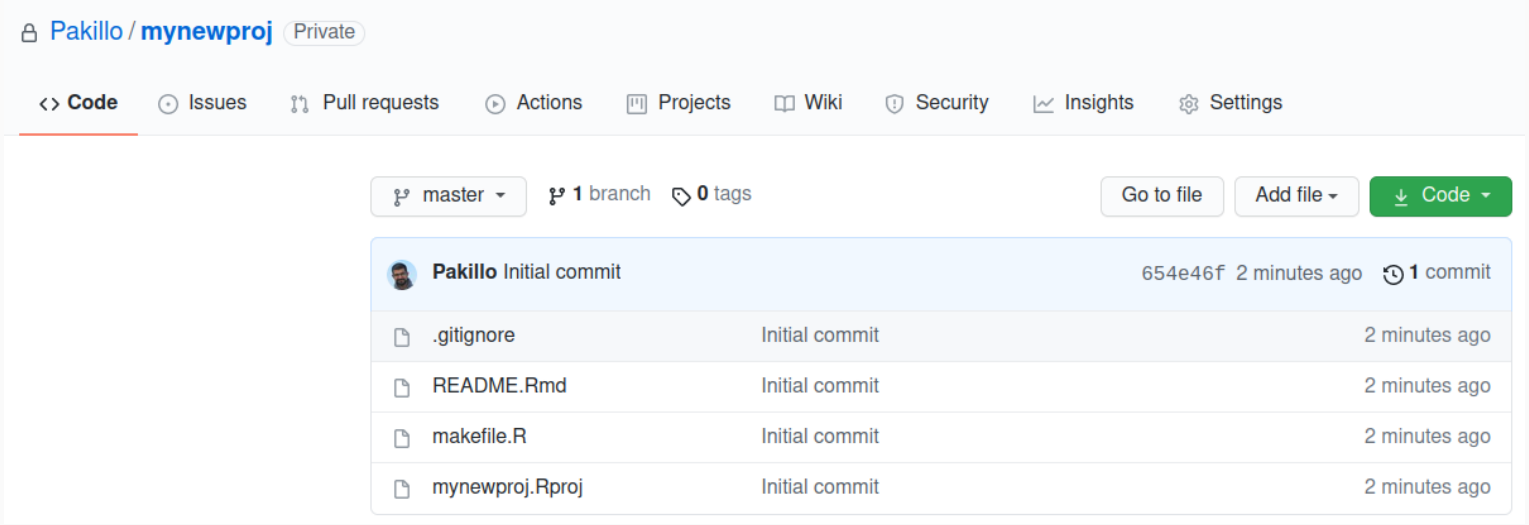
```
library('template')
```

```
new_project("mynewproj",
            package = FALSE)
```



# `template`: New projects also on GitHub

```
new_project("mynewproj",
            package = FALSE,
            github = TRUE)
```



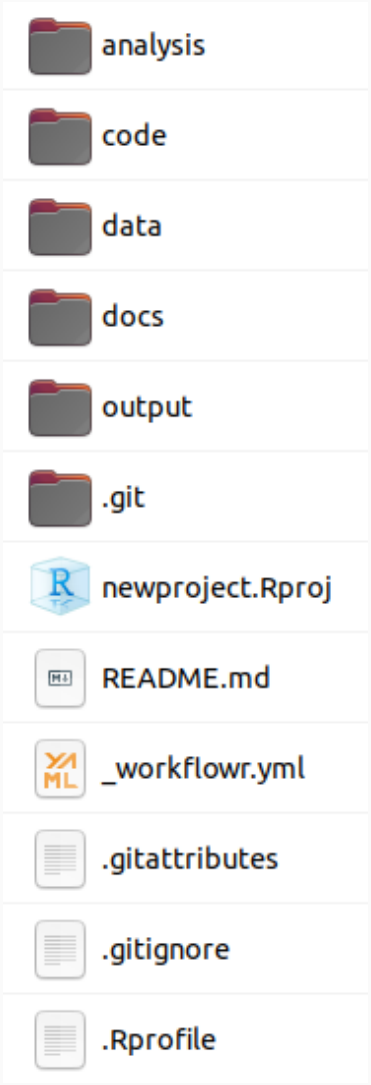

## `workflowr`:

## reproducible projects with website

# `wflow_start` creates project scaffolding

```
library('workflowr')

wflow_start("newproject")
```



# wflow_open starts new analysis

```
wflow_open("analysis/first-analysis.Rmd")
```

```
---
title: "first-analysis"
author: "Pakillo"
date: "2021-06-15"
output: workflowr::wflow_html
editor_options:
  chunk_output_type: console
---

## Introduction

```{r}
data(iris)
plot(iris)
```
```

# `wflow_build` runs analyses and generates website

```
wflow_build()
```

# `wflow_publish` commits changes & updates everything

```
wflow_publish(c("analysis/first-analysis.Rmd",
                "analysis/index.Rmd",
                "analysis/about.Rmd",
                "analysis/license.Rmd"),
              message = "Publish initial analyses")
```

# Connect with GitHub/GitLab and deploy website

```
wflow_use_github("Pakillo")


wflow_git_push()
```

# Research compendia: projects as packages

# Projects as packages

- Standard structure

Rodríguez-Sánchez et al. 2016, Marwick et al 2018, but see McBain 2020

# Projects as packages

- Standard structure
- Promotes modular code, documented and tested

Rodríguez-Sánchez et al. 2016, Marwick et al 2018, but see McBain 2020

# Projects as packages

- Standard structure
- Promotes modular code, documented and tested
- Easy to share and run

Rodríguez-Sánchez et al. 2016, Marwick et al 2018, but see McBain 2020

# Projects as packages

- Standard structure
- Promotes modular code, documented and tested
- Easy to share and run
- Automatic checks (Continuous Integration)

Rodríguez-Sánchez et al. 2016, Marwick et al 2018, but see McBain 2020

# Projects as packages

- Standard structure
- Promotes modular code, documented and tested
- Easy to share and run
- Automatic checks (Continuous Integration)
- Automatic code review (`goodpractice`)

Rodríguez-Sánchez et al. 2016, Marwick et al 2018, but see McBain 2020

# Projects as packages

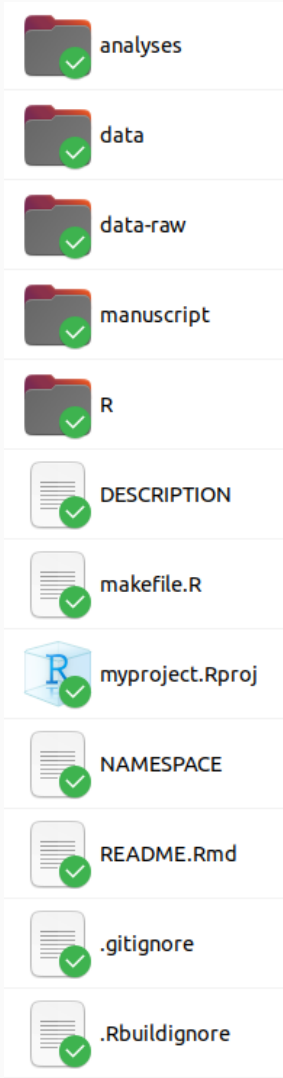
- Standard structure
- Promotes modular code, documented and tested
- Easy to share and run
- Automatic checks (Continuous Integration)
- Automatic code review (`goodpractice`)
- Easily create website with `pkgdown`

Rodríguez-Sánchez et al. 2016, Marwick et al 2018, but see McBain 2020

# Creating package structure with `template`

```
library('template')
```

```
new_project('myproject',
            package = TRUE)
```



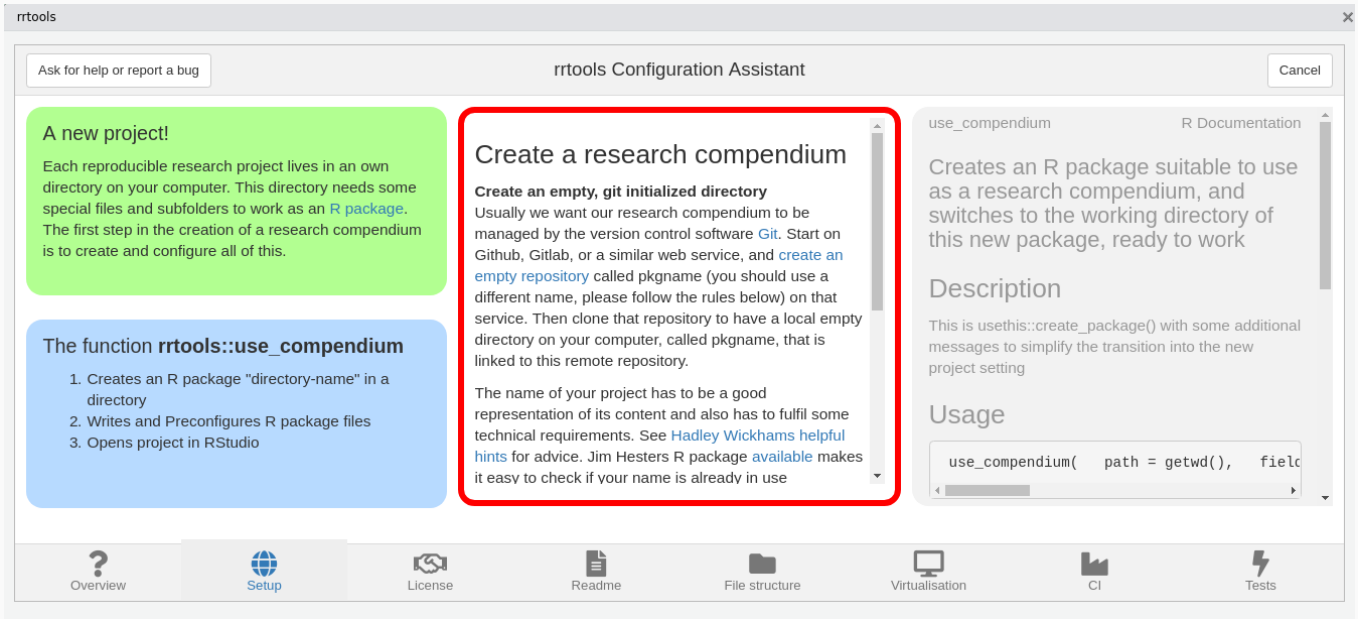
# rrtools

# `rrtools` creates research compendia

```
library('rrtools')

use_compendium('~/myproject/')
```

Rstudio addin: https://github.com/nevrome/rrtools.addin



# `rrtools`: project structure

```
- README
- LICENSE
- DESCRIPTION
- renv/
- Dockerfile
- analysis/
  |
  |- paper/
      |- paper.qmd
      |- references.bib
  |
  |- figures/
  |
  |- data/
      |- raw_data/
      |- derived_data/
```


# rcompendium

# `rcompendium` creates new project with all scaffolding

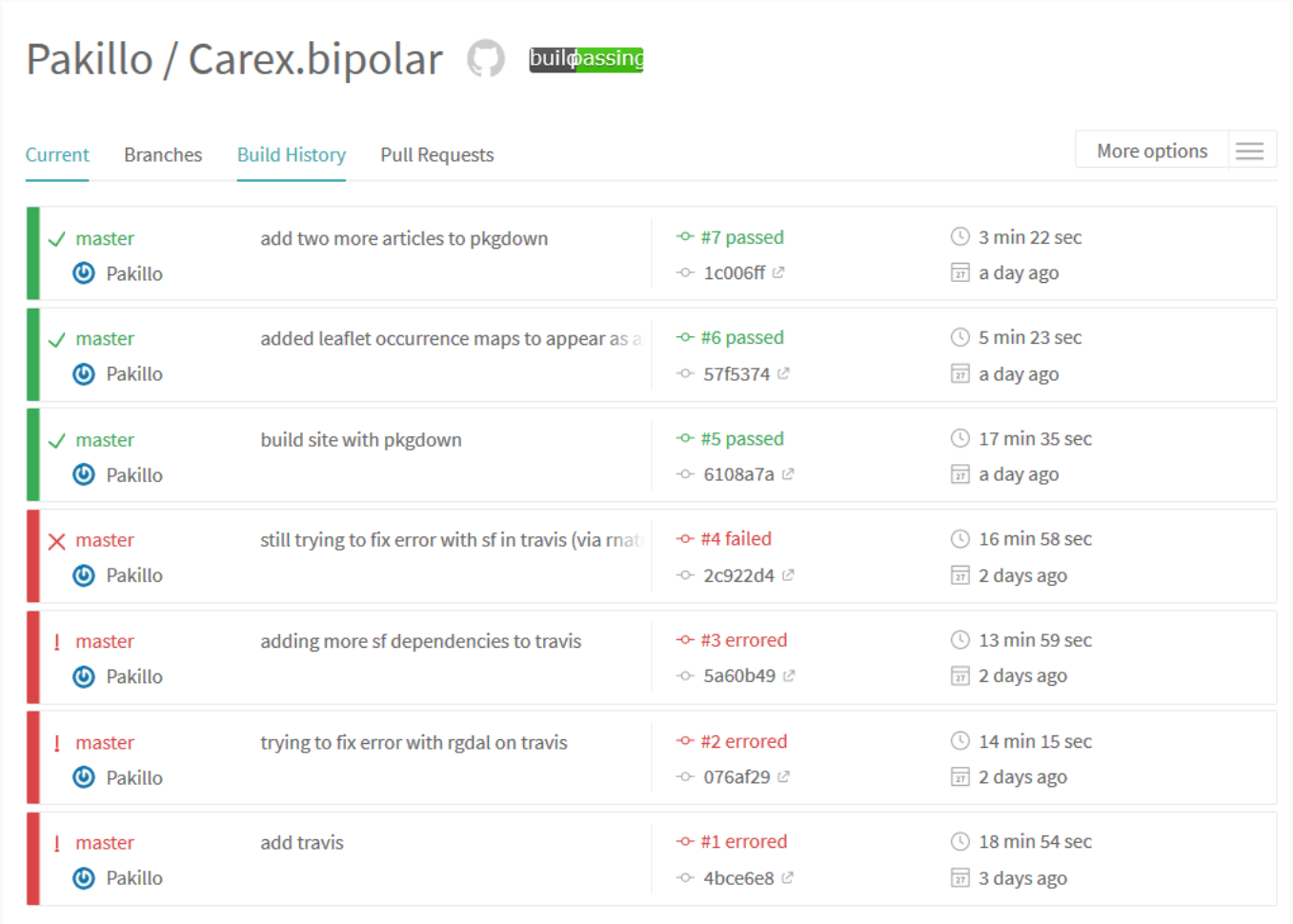
```
library('rcompendium')

new_compendium()
```

- R package structure
- GitHub repository
- Automatic testing & website update

# Continuous Integration (GitHub Actions, GitLab CI...)

Automatic testing with every commit!



# Minimalistic compendium

https://github.com/cboettig/compendium

- DESCRIPTION (dependencies)
- Manuscript (Rmd)
- GitHub Actions

# Data management

# Data management

See

https://dataoneorg.github.io/Education/bestpractices/

.

1. Planification (e.g. DMPTool)
2. Collection
3. Metadata description (dataspice, EML, Data Packages, DataPackageR)
4. Quality control (e.g. assertr, validate, pointblank)
5. Storage

# Document your data

```
library('dataspice')
create_spice()   # create CSV templates for metadata

edit_creators()  # open Shiny apps to edit the CSVs
prep_access()
edit_access()
prep_attributes()
edit_attributes()
edit_biblio()

write_spice()  # write machine-readable metadata

build_site()  # build human-readable metadata report
```

https://docs.ropensci.org/dataspice/

# Check data before analysis

```
library('assertr')

dataset |>
  assert(within_bounds(0, 0.20), fruit.weight) |>
  assert(in_set('black', 'red'), colour)
```

Check out also `pointblank`

# *Editorial expression of concern*

IN THE 3 June issue, *Science* published the Report "Environmentally relevant concentrations of microplastic particles influence larval fish ecology" by Oona M. Lönnstedt and Peter Eklöv (*1*). The authors have notified *Science* of the theft of the computer on which the raw data for the paper were stored. These data were not backed up on any other device nor deposited in an appropriate repository. *Science* is publishing this Editorial Expression of Concern to alert our readers to the fact that no further data can be made available, beyond those already presented in the paper and its supplement, to enable readers to understand, assess, reproduce, or extend the conclusions of the paper.

***Jeremy Berg***

Editor in Chief

http://science.sciencemag.org/content/354/6317/1242.1

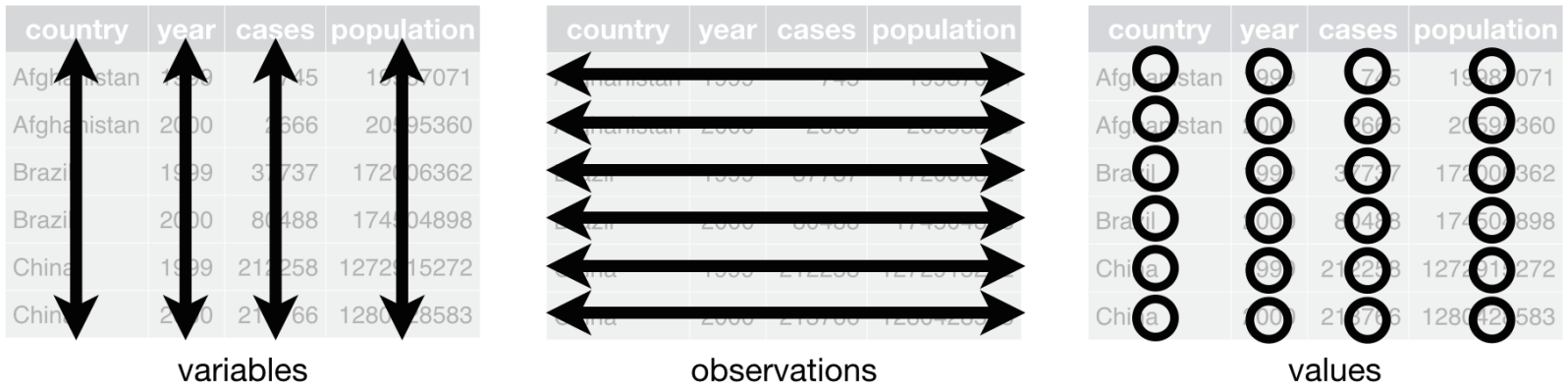
# Storage

Use the **cloud**: safe, persistent, easy to share

- Open Science Framework
- GitHub
- Dropbox
- Figshare, Zenodo, etc
- See all data repositories in www.re3data.org

# Tidy data

# Tidy data




| country | year | cases |
|---|---|---|
| Afghanistan | 1999 | 745 |
| Afghanistan | 2000 | 2666 |
| Brazil | 1999 | 37737 |
| Brazil | 2000 | 80488 |
| China | 1999 | 212258 |
| China | 2000 | 213766 |

| country | 1999 | 2000 |
|---|---|---|
| Afghanistan | 745 | 2666 |
| Brazil | 37737 | 80488 |
| China | 212258 | 213766 |

table4

https://r4ds.hadley.nz/data-tidy.html

# Be careful with data entry and management in Excel!

COMMENT Open Access

## Gene name errors are widespread in the scientific literature

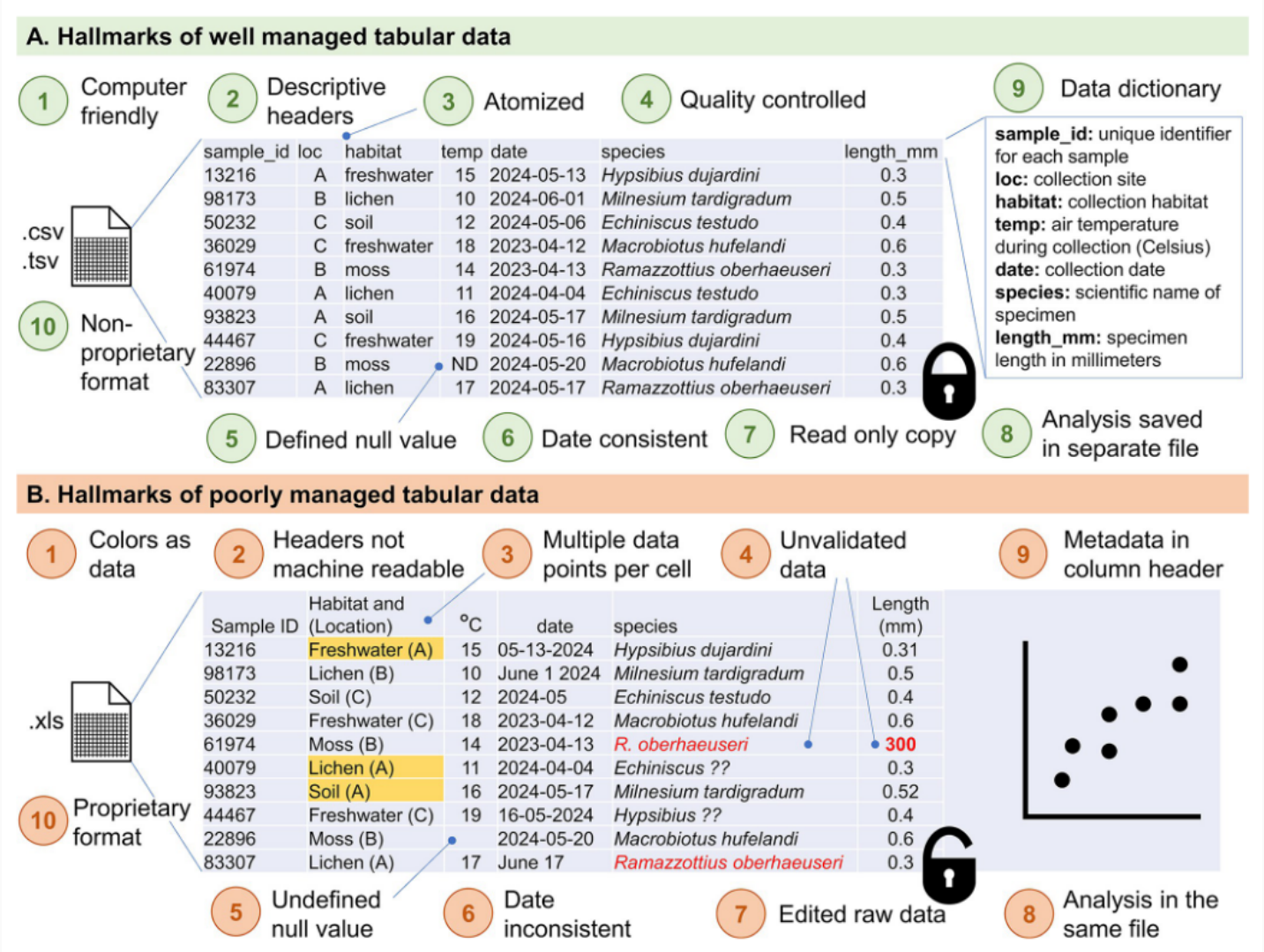
Mark Ziemann[1], Yotam Eren[1,2] and Assam El-Osta[1,3*]

**Abstract**

The spreadsheet software Microsoft Excel, when used with default settings, is known to convert gene names to dates and floating-point numbers. A programmatic scan of leading genomics journals reveals that approximately one-fifth of papers with supplementary Excel gene lists contain erroneous gene name conversions.

frequently reused. Our aim here is to raise awareness of the problem.

We downloaded and screened supplementary files from 18 journals published between 2005 and 2015 using a suite of shell scripts. Excel files (.xls and.xlsx suffixes) were converted to tabular separated files (tsv) with ssconvert (v1.12.9). Each sheet within the Excel file was converted to a separate tsv file. Each column of data in the tsv file was screened for the presence of gene sym-




| sample_id | loc | habitat | temp | date | species | length_mm |
|---|---|---|---|---|---|---|
| 13216 | A | freshwater | 15 | 2024-05-13 | *Hypsibius dujardini* | 0.3 |
| 98173 | B | lichen | 10 | 2024-06-01 | *Milnesium tardigradum* | 0.5 |
| 50232 | C | soil | 12 | 2024-05-06 | *Echiniscus testudo* | 0.4 |
| 36029 | C | freshwater | 18 | 2023-04-12 | *Macrobiotus hufelandi* | 0.6 |
| 61974 | B | moss | 14 | 2023-04-13 | *Ramazzottius oberhaeuseri* | 0.3 |
| 40079 | A | lichen | 11 | 2024-04-04 | *Echiniscus testudo* | 0.3 |
| 93823 | A | soil | 16 | 2024-05-17 | *Milnesium tardigradum* | 0.5 |
| 44467 | C | freshwater | 19 | 2024-05-16 | *Hypsibius dujardini* | 0.4 |
| 22896 | B | moss | ND | 2024-05-20 | *Macrobiotus hufelandi* | 0.6 |
| 83307 | A | lichen | 17 | 2024-05-17 | *Ramazzottius oberhaeuseri* | 0.3 |

| Sample ID | Habitat and (Location) | °C | date | species | Length (mm) |
|---|---|---|---|---|---|
| 13216 | Freshwater (A) | 15 | 05-13-2024 | *Hypsibius dujardini* | 0.31 |
| 98173 | Lichen (B) | 10 | June 1 2024 | *Milnesium tardigradum* | 0.5 |
| 50232 | Soil (C) | 12 | 2024-05 | *Echiniscus testudo* | 0.4 |
| 36029 | Freshwater (C) | 18 | 2023-04-12 | *Macrobiotus hufelandi* | 0.6 |
| 61974 | Moss (B) | 14 | 2023-04-13 | *R. oberhaeuseri* | 300 |
| 40079 | Lichen (A) | 11 | 2024-04-04 | *Echiniscus ??* | 0.3 |
| 93823 | Soil (A) | 16 | 2024-05-17 | *Milnesium tardigradum* | 0.52 |
| 44467 | Freshwater (C) | 19 | 16-05-2024 | *Hypsibius ??* | 0.4 |
| 22896 | Moss (B) | | 2024-05-20 | *Macrobiotus hufelandi* | 0.6 |
| 83307 | Lichen (A) | 17 | June 17 | *Ramazzottius oberhaeuseri* | 0.3 |

Hertz & McNeill 2024

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use Data validation in Excel (or GForms) to constrain data entry to accepted values.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use Data validation in Excel (or GForms) to constrain data entry to accepted values.
- Don't combine multiple pieces of information in one cell.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use Data validation in Excel (or GForms) to constrain data entry to accepted values.
- Don't combine multiple pieces of information in one cell.
- **Don't touch raw data**. Do all data manipulation through code.

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use Data validation in Excel (or GForms) to constrain data entry to accepted values.
- Don't combine multiple pieces of information in one cell.
- **Don't touch raw data**. Do all data manipulation through code.
- Export data as plain text (txt, csv).

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use **Data validation** in Excel (or GForms) to constrain data entry to accepted values.
- Don't combine multiple pieces of information in one cell.
- **Don't touch raw data**. Do all data manipulation through code.
- Export data as plain text (txt, csv).
- `http://www.datacarpentry.org/spreadsheet-ecology-lesson/`

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use **Data validation** in Excel (or GForms) to constrain data entry to accepted values.
- Don't combine multiple pieces of information in one cell.
- **Don't touch raw data**. Do all data manipulation through code.
- Export data as plain text (txt, csv).
- `http://www.datacarpentry.org/spreadsheet-ecology-lesson/`
- `http://kbroman.org/dataorg/`

# Spreadsheet good practices

- Put **variables** in **columns** (things you are measuring: height, weight, sex)
- Each **observation** in one **row** (e.g. individuals).
- **Avoid** spaces, numbers, and **special characters** in column names.
- Always **write zero values**, to distinguish from blank/missing data.
- Use blank/empty cells, or NA, for missing data.
- Input dates as **year, month, day** in separate columns. Or **YYYY-MM-DD** as text.
- Use Data validation in Excel (or GForms) to constrain data entry to accepted values.
- Don't combine multiple pieces of information in one cell.
- **Don't touch raw data**. Do all data manipulation through code.
- Export data as plain text (txt, csv).
- `http://www.datacarpentry.org/spreadsheet-ecology-lesson/`
- `http://kbroman.org/dataorg/`
- Broman & Woo: Data organization in spreadsheets

# Common spreadsheet errors

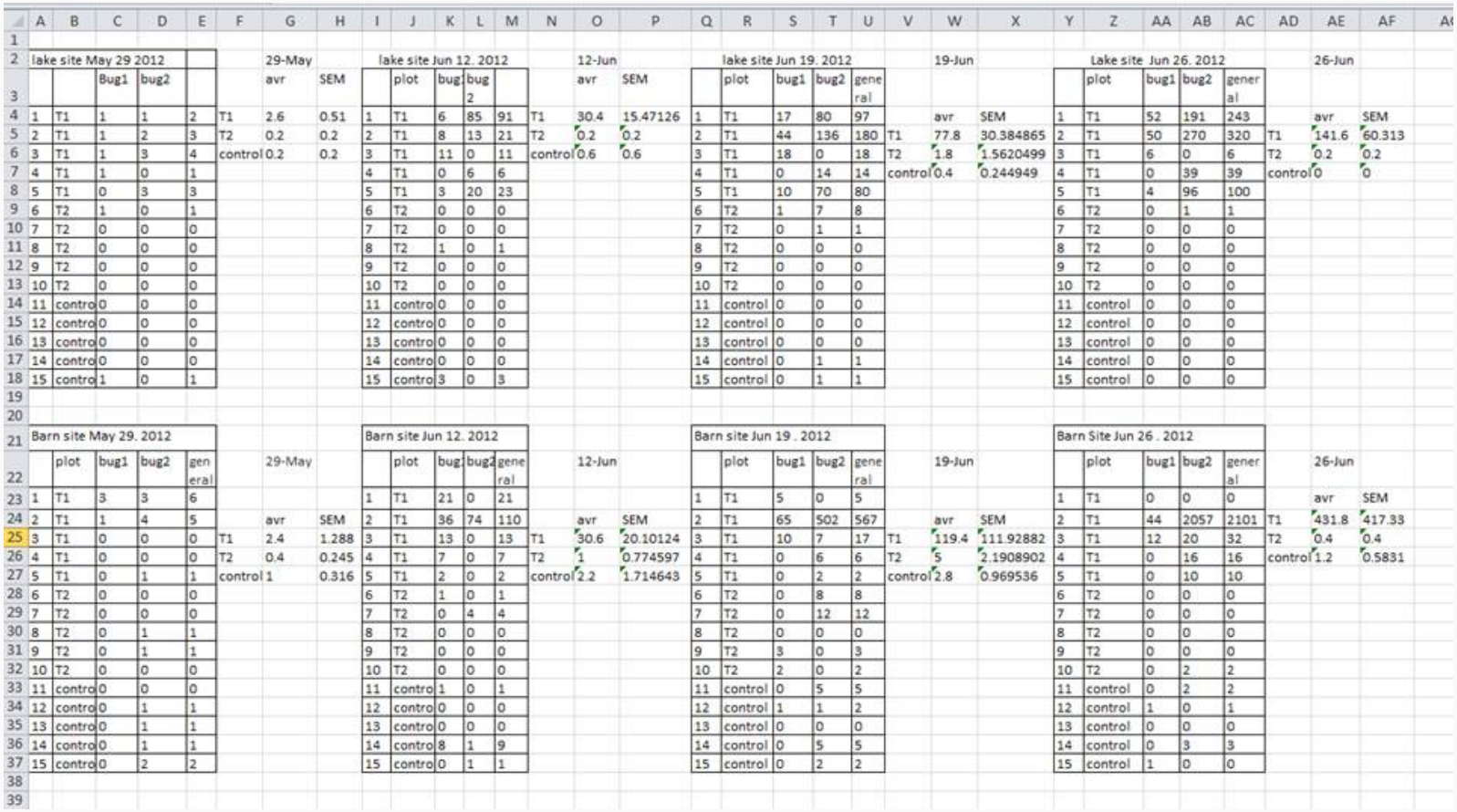
# More than one variable per column

| Date collected | Plot | Species-Sex | Weight |
|---|---|---|---|
| 1/9/78 | 1 | DM-M | 40 |
| 1/9/78 | 1 | DM-F | 36 |
| 1/9/78 | 1 | DS-F | 135 |
| 1/20/78 | 1 | DM-F | 39 |
| 1/20/78 | 2 | DM-M | 43 |
| 1/20/78 | 2 | DS-F | 144 |
| 3/13/78 | 2 | DM-F | 51 |
| 3/13/78 | 2 | DM-F | 44 |
| 3/13/78 | 2 | DS-F | 146 |

| Date collected | Plot | Species | Sex | Weight |
|---|---|---|---|---|
| 1/9/78 | 1 | DM | M | 40 |
| 1/9/78 | 1 | DM | F | 36 |
| 1/9/78 | 1 | DS | F | 135 |
| 1/20/78 | 1 | DM | F | 39 |
| 1/20/78 | 2 | DM | M | 43 |
| 1/20/78 | 2 | DS | F | 144 |
| 3/13/78 | 2 | DM | F | 51 |
| 3/13/78 | 2 | DM | F | 44 |
| 3/13/78 | 2 | DS | F | 146 |

Source: Data Carpentry

# Multiple tables



| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | AA | AB | AC | AD | AE | AF |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | lake site May 29 2012 | | | | | | 29-May | | lake site Jun 12. 2012 | | | | | | 12-Jun | | lake site Jun 19. 2012 | | | | | | 19-Jun | | Lake site Jun 26. 2012 | | | | | | 26-Jun | |
| 3 | | | Bug1 | bug2 | | | avr | SEM | | plot | bug1 | bug2 | | | avr | SEM | | plot | bug1 | bug2 | general | | | | | plot | bug1 | bug2 | general | | | |
| 4 | 1 | T1 | 1 | 1 | 2 | T1 | 2.6 | 0.51 | 1 | T1 | 6 | 85 | 91 | T1 | 30.4 | 15.47126 | 1 | T1 | 17 | 80 | 97 | | avr | SEM | 1 | T1 | 52 | 191 | 243 | | avr | SEM |
| 5 | 2 | T1 | 1 | 2 | 3 | T2 | 0.2 | 0.2 | 2 | T1 | 8 | 13 | 21 | T2 | 0.2 | 0.2 | 2 | T1 | 44 | 136 | 180 | T1 | 77.8 | 30.384865 | 2 | T1 | 50 | 270 | 320 | T1 | 141.6 | 60.313 |
| 6 | 3 | T1 | 1 | 3 | 4 | control | 0.2 | 0.2 | 3 | T1 | 11 | 0 | 11 | control | 0.6 | 0.6 | 3 | T1 | 18 | 0 | 18 | T2 | 1.8 | 1.5620499 | 3 | T1 | 6 | 0 | 6 | T2 | 0.2 | 0.2 |
| 7 | 4 | T1 | 1 | 0 | 1 | | | | 4 | T1 | 0 | 6 | 6 | | | | 4 | T1 | 0 | 14 | 14 | control | 0.4 | 0.244949 | 4 | T1 | 0 | 39 | 39 | control | 0 | 0 |
| 8 | 5 | T1 | 0 | 3 | 3 | | | | 5 | T1 | 3 | 20 | 23 | | | | 5 | T1 | 10 | 70 | 80 | | | | 5 | T1 | 4 | 96 | 100 | | | |
| 9 | 6 | T2 | 1 | 0 | 1 | | | | 6 | T2 | 0 | 0 | 0 | | | | 6 | T2 | 1 | 7 | 8 | | | | 6 | T2 | 0 | 1 | 1 | | | |
| 10 | 7 | T2 | 0 | 0 | 0 | | | | 7 | T2 | 0 | 0 | 0 | | | | 7 | T2 | 0 | 1 | 1 | | | | 7 | T2 | 0 | 0 | 0 | | | |
| 11 | 8 | T2 | 0 | 0 | 0 | | | | 8 | T2 | 1 | 0 | 1 | | | | 8 | T2 | 0 | 0 | 0 | | | | 8 | T2 | 0 | 0 | 0 | | | |
| 12 | 9 | T2 | 0 | 0 | 0 | | | | 9 | T2 | 0 | 0 | 0 | | | | 9 | T2 | 0 | 0 | 0 | | | | 9 | T2 | 0 | 0 | 0 | | | |
| 13 | 10 | T2 | 0 | 0 | 0 | | | | 10 | T2 | 0 | 0 | 0 | | | | 10 | T2 | 0 | 0 | 0 | | | | 10 | T2 | 0 | 0 | 0 | | | |
| 14 | 11 | contro | 0 | 0 | 0 | | | | 11 | contro | 0 | 0 | 0 | | | | 11 | control | 0 | 0 | 0 | | | | 11 | control | 0 | 0 | 0 | | | |
| 15 | 12 | contro | 0 | 0 | 0 | | | | 12 | contro | 0 | 0 | 0 | | | | 12 | control | 0 | 0 | 0 | | | | 12 | control | 0 | 0 | 0 | | | |
| 16 | 13 | contro | 0 | 0 | 0 | | | | 13 | contro | 0 | 0 | 0 | | | | 13 | control | 0 | 0 | 0 | | | | 13 | control | 0 | 0 | 0 | | | |
| 17 | 14 | contro | 0 | 0 | 0 | | | | 14 | contro | 0 | 0 | 0 | | | | 14 | control | 0 | 1 | 1 | | | | 14 | control | 0 | 0 | 0 | | | |
| 18 | 15 | contro | 1 | 0 | 1 | | | | 15 | contro | 3 | 0 | 3 | | | | 15 | control | 0 | 1 | 1 | | | | 15 | control | 0 | 0 | 0 | | | |
| 19 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 20 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 21 | Barn site May 29. 2012 | | | | | | | | Barn site Jun 12. 2012 | | | | | | | | Barn site Jun 19 . 2012 | | | | | | | | Barn Site Jun 26 . 2012 | | | | | | | |
| 22 | | plot | bug1 | bug2 | general | | 29-May | | | plot | bug1 | bug2 | general | | 12-Jun | | | plot | bug1 | bug2 | general | | 19-Jun | | | plot | bug1 | bug2 | general | | 26-Jun | |
| 23 | 1 | T1 | 3 | 3 | 6 | | | | 1 | T1 | 21 | 0 | 21 | | | | 1 | T1 | 5 | 0 | 5 | | | | 1 | T1 | 0 | 0 | 0 | | avr | SEM |
| 24 | 2 | T1 | 1 | 4 | 5 | | avr | SEM | 2 | T1 | 36 | 74 | 110 | | avr | SEM | 2 | T1 | 65 | 502 | 567 | | avr | SEM | 2 | T1 | 44 | 2057 | 2101 | T1 | 431.8 | 417.33 |
| 25 | 3 | T1 | 0 | 0 | 0 | T1 | 2.4 | 1.288 | 3 | T1 | 13 | 0 | 13 | T1 | 30.6 | 20.10124 | 3 | T1 | 10 | 7 | 17 | T1 | 119.4 | 111.92882 | 3 | T1 | 12 | 20 | 32 | T2 | 0.4 | 0.4 |
| 26 | 4 | T1 | 0 | 0 | 0 | T2 | 0.4 | 0.245 | 4 | T1 | 7 | 0 | 7 | T2 | 1 | 0.774597 | 4 | T1 | 0 | 6 | 6 | T2 | 5 | 2.1908902 | 4 | T1 | 0 | 16 | 16 | control | 1.2 | 0.5831 |
| 27 | 5 | T1 | 0 | 1 | 1 | control | 1 | 0.316 | 5 | T1 | 2 | 0 | 2 | control | 2.2 | 1.714643 | 5 | T1 | 0 | 2 | 2 | control | 2.8 | 0.969536 | 5 | T1 | 0 | 10 | 10 | | | |
| 28 | 6 | T2 | 0 | 0 | 0 | | | | 6 | T2 | 1 | 0 | 1 | | | | 6 | T2 | 0 | 8 | 8 | | | | 6 | T2 | 0 | 0 | 0 | | | |
| 29 | 7 | T2 | 0 | 0 | 0 | | | | 7 | T2 | 0 | 4 | 4 | | | | 7 | T2 | 0 | 12 | 12 | | | | 7 | T2 | 0 | 0 | 0 | | | |
| 30 | 8 | T2 | 0 | 1 | 1 | | | | 8 | T2 | 0 | 0 | 0 | | | | 8 | T2 | 0 | 0 | 0 | | | | 8 | T2 | 0 | 0 | 0 | | | |
| 31 | 9 | T2 | 0 | 1 | 1 | | | | 9 | T2 | 0 | 0 | 0 | | | | 9 | T2 | 3 | 0 | 3 | | | | 9 | T2 | 0 | 0 | 0 | | | |
| 32 | 10 | T2 | 0 | 0 | 0 | | | | 10 | T2 | 0 | 0 | 0 | | | | 10 | T2 | 2 | 0 | 2 | | | | 10 | T2 | 0 | 2 | 2 | | | |
| 33 | 11 | contro | 0 | 0 | 0 | | | | 11 | contro | 1 | 0 | 1 | | | | 11 | control | 0 | 5 | 5 | | | | 11 | control | 0 | 2 | 2 | | | |
| 34 | 12 | contro | 0 | 1 | 1 | | | | 12 | contro | 0 | 0 | 0 | | | | 12 | control | 1 | 1 | 2 | | | | 12 | control | 1 | 0 | 1 | | | |
| 35 | 13 | contro | 0 | 1 | 1 | | | | 13 | contro | 0 | 0 | 0 | | | | 13 | control | 0 | 0 | 0 | | | | 13 | control | 0 | 0 | 0 | | | |
| 36 | 14 | contro | 0 | 1 | 1 | | | | 14 | contro | 8 | 1 | 9 | | | | 14 | control | 0 | 5 | 5 | | | | 14 | control | 0 | 3 | 3 | | | |
| 37 | 15 | contro | 0 | 2 | 2 | | | | 15 | contro | 0 | 1 | 1 | | | | 15 | control | 0 | 2 | 2 | | | | 15 | control | 1 | 0 | 0 | | | |
| 38 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 39 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Could you avoid new tab by adding a column to original spreadsheet?

# Using formatting, comments, etc to convey information

| Plot: 2 | | | |
|---|---|---|---|
| Date collecte | Species | Sex | Weight |
| 1/8/14 | NA | | |
| 1/8/14 | DM | M | 44 |
| 1/8/14 | DM | M | 38 |
| 1/8/14 | OL | | |
| 1/8/14 | PE | M | 22 |
| 1/8/14 | DM | M | 38 |
| 1/8/14 | DM | M | 48 |
| 1/8/14 | DM | M | 43 |
| 1/8/14 | DM | F | 35 |
| 1/8/14 | DM | M | 43 |
| 1/8/14 | DM | F | 37 |
| 1/8/14 | PF | F | 7 |
| 1/8/14 | DM | M | 45 |
| 1/8/14 | OT | | |
| 1/8/14 | DS | M | 157 |
| 1/8/14 | OX | | |
| | | | |
| 2/18/14 | NA | M | 218 |
| 2/18/14 | PF | F | 7 |
| 2/18/14 | DM | M | 52 |
| | | | |
| | measurement device not calibrated | | |

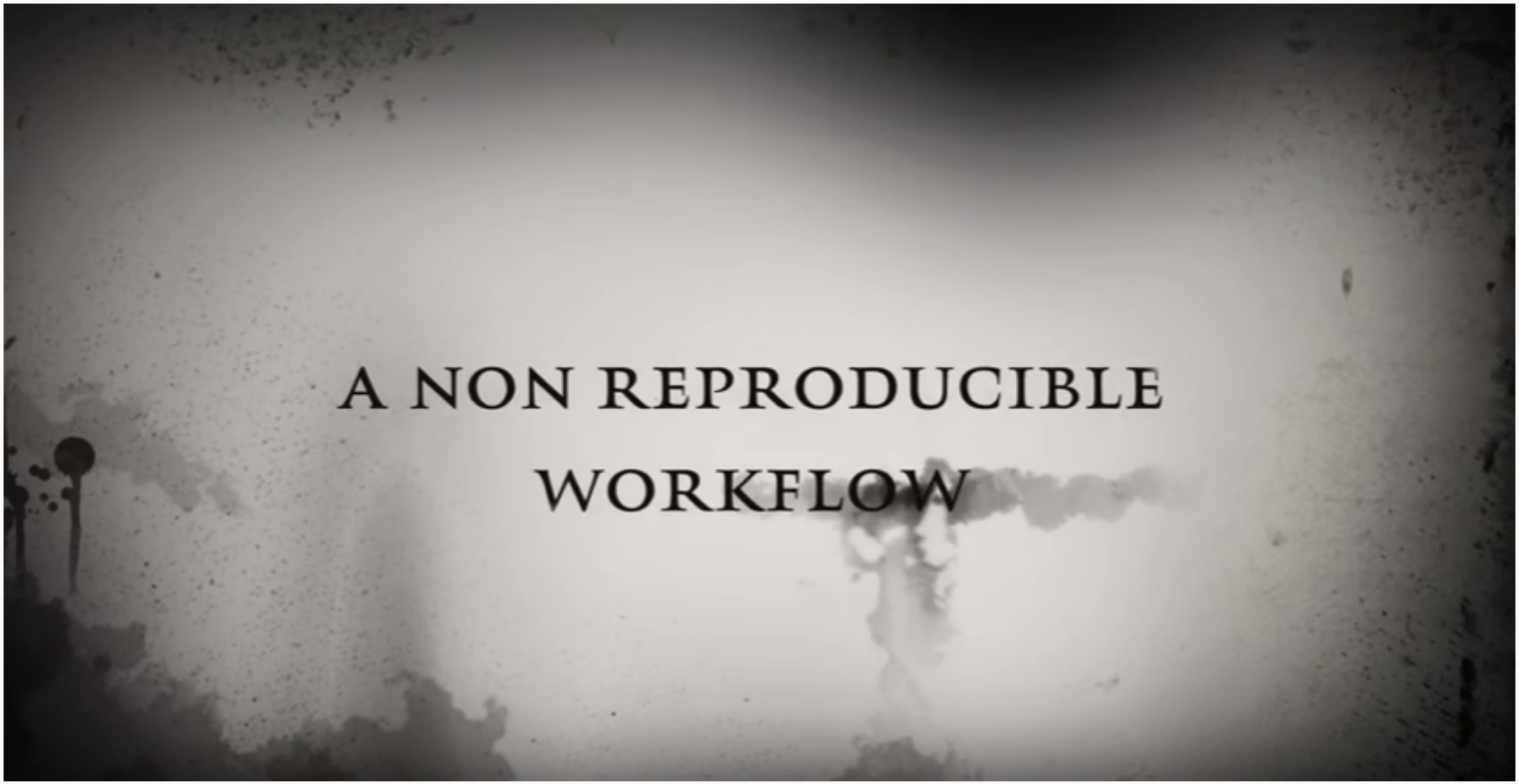
| Date collecte | Species | Sex | Weight | Calibrated |
|---|---|---|---|---|
| 1/8/14 | NA | | | |
| 1/8/14 | DM | M | 44 | Y |
| 1/8/14 | DM | M | 38 | Y |
| 1/8/14 | OL | | | |
| 1/8/14 | PE | M | 22 | Y |
| 1/8/14 | DM | M | 38 | Y |
| 1/8/14 | DM | M | 48 | Y |
| 1/8/14 | DM | M | 43 | Y |
| 1/8/14 | DM | F | 35 | Y |
| 1/8/14 | DM | M | 43 | Y |
| 1/8/14 | DM | F | 37 | Y |
| 1/8/14 | PF | F | 7 | Y |
| 1/8/14 | DM | M | 45 | Y |
| 1/8/14 | OT | | | |
| 1/8/14 | DS | M | 157 | N |
| 1/8/14 | OX | | | |
| 2/18/14 | NA | M | 218 | N |
| 2/18/14 | PF | F | 7 | Y |
| 2/18/14 | DM | M | 52 | Y |

Your turn: tidy up this messy dataset

https://ndownloader.figshare.com/files/2252083

# Reproducible dynamic documents with Rmarkdown

# A scary movie... with happy ending

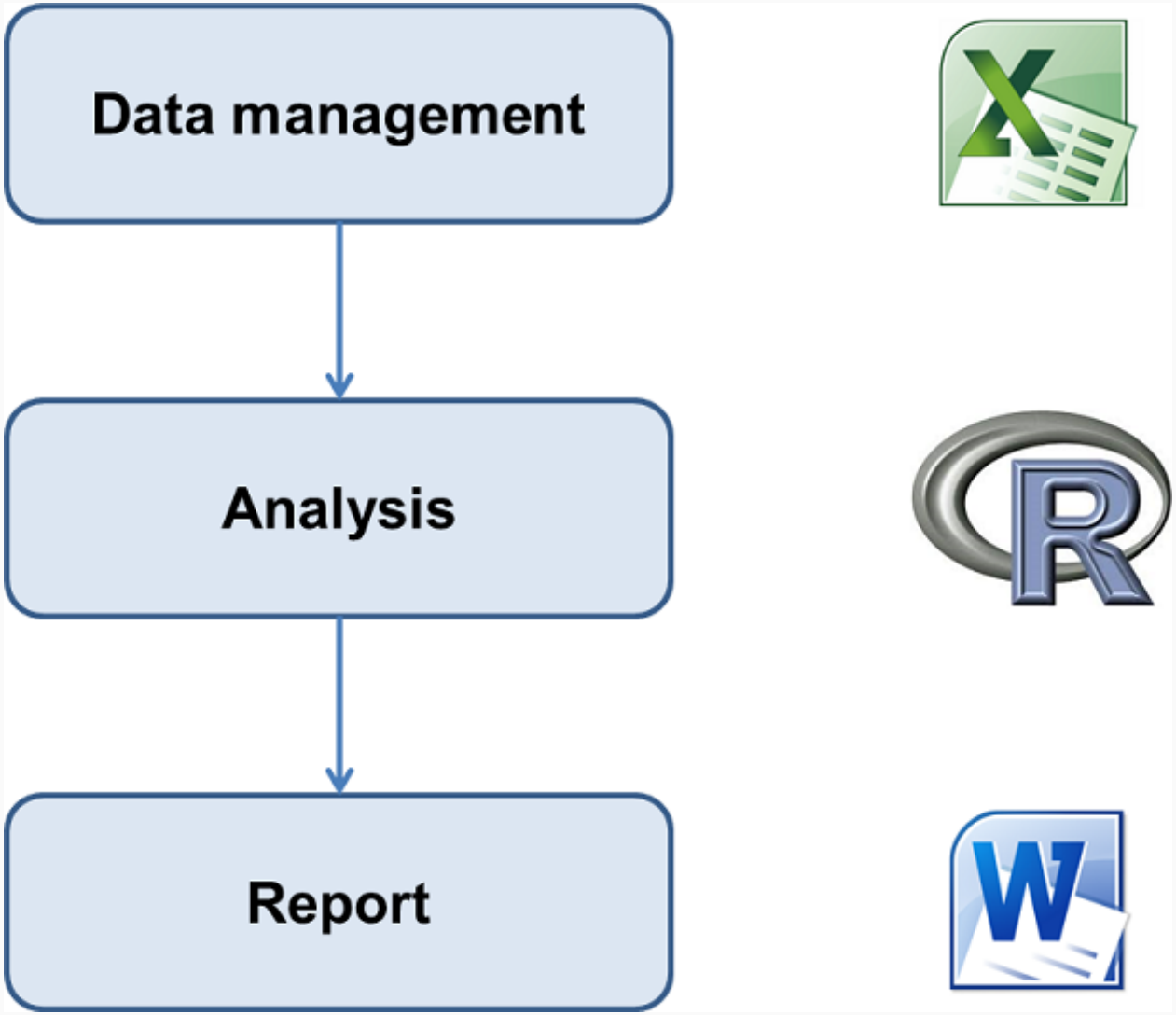



https://youtu.be/s3JldKoA0zw

# A typical research workflow

1. Prepare data (**spreadsheet**)
2. Analyse data (**R**)
3. Write report/paper (**Word**)
4. Start the email attachments nightmare...

# This workflow is broken



# Problems of a broken workflow

- **How did you do this?** What analysis is behind this figure? Did you account for ...?
- **What dataset was used?** Which individuals were left out? Where is the clean dataset?
- Oops, there is an error in the data. **Can you repeat the analysis?** And update figures/tables in Word!

# Manual copy-paste is tedious & problematic

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0651657  0.4264970  -0.153    0.879
sunshine     0.0100228  0.0004232  23.683   <2e-16
```

'Transcribing numbers **from stats software by hand** was the largest source of errors'

(Eubank 2016)

**Trevor A. Branch**
@TrevorABranch

Follow

My rule of thumb: every analysis you do on a dataset will have to be redone 10–15 times before publication. Plan accordingly. #Rstats

Your **closest collaborator** is you 6 months ago,

and you don’t respond to emails.

(P. Wilson)

Even **you** will struggle to reproduce

**your own results** from a few weeks/months ago.

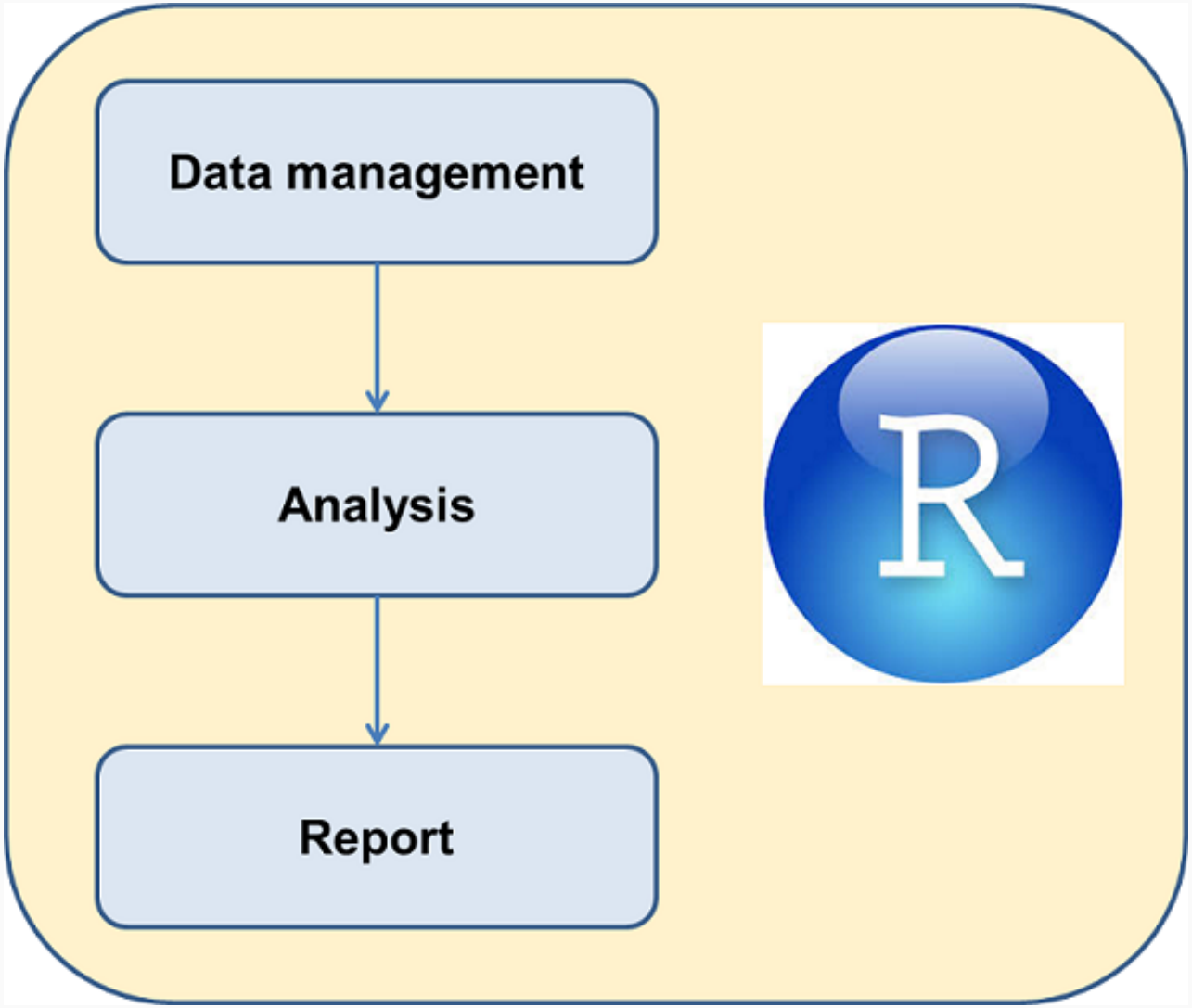
Writing reproducible manuscripts is hard

Revising non-reproducible manuscripts is even harder

.

Also, please note that because rev#1 asked to re-calculate effect sizes (...) we need to change every single number in the main text.

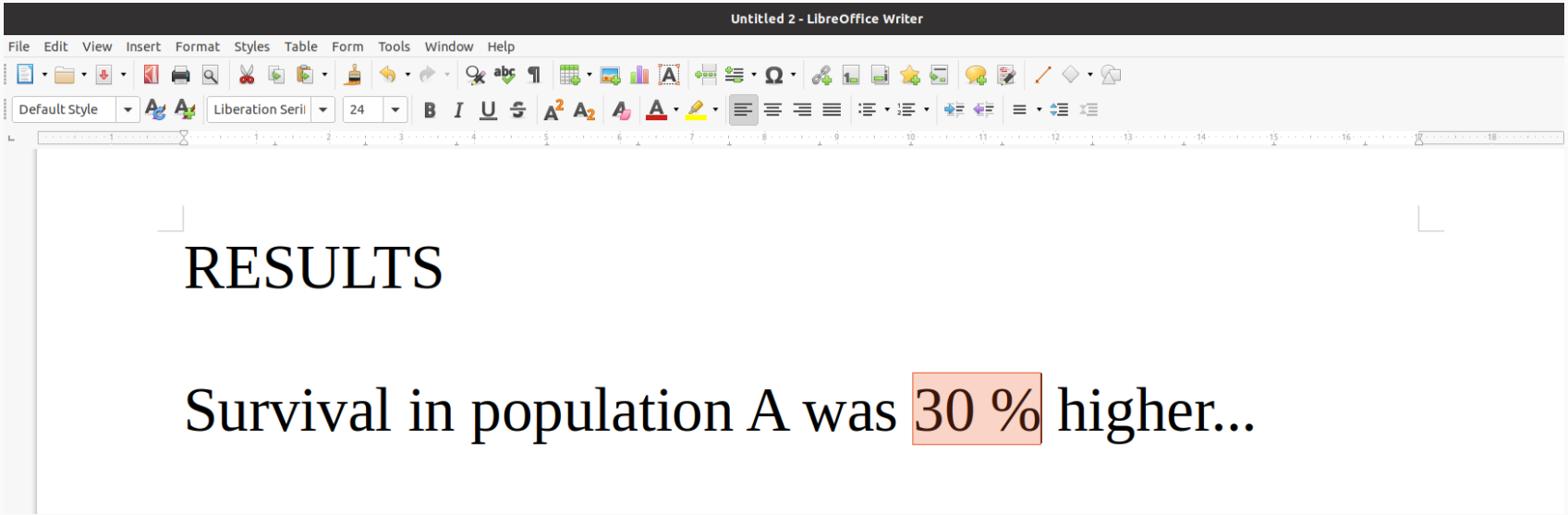
# Dynamic reports



# Rmarkdown documents

- **Fully reproducible** (trace all results inc. tables and plots)
- **Dynamic** (regenerate with 1 click)
- **Multiple outputs**:
  - documents (HTML, Word, PDF)
  - presentations (HTML, PDF, PowerPoint)
  - books
  - websites...

# Where does this value come from?



*Rmarkdown*:

> Survival in population A was `` `r surv.diff` `` % higher

*Output*:

> Survival in population A was **30** % higher

```
mydata <- read.csv('data.txt')
```

*Rmarkdown:*

> We measured `` `r nrow(mydata)` `` individuals

*Output:*

> We measured **100** individuals

---

*Much better than copy-paste!*

# Rmarkdown: code (R, Python, etc) + text (Markdown)

````
---
title: "Does sunshine make people happy?"
author: "FRS"
output: word_document
---

## Introduction

It is well known that individual well-being can be influenced by climatic conditions.


## Methods

```{r echo=FALSE}
## Read data
data <- read.table("data.txt", header = TRUE)

# Fit linear model
model <- lm(happiness ~ sunshine, data = data)
```

We collected data on `r nrow(data)` individuals and fitted a linear model.
````

Metadata
(YAML)

Text
(Markdown)

Code
(R, Python...)

# Code chunk options

```
```{r echo=FALSE, eval=TRUE, fig.height=3}
plot(iris)
```
```

https://yihui.org/knitr/options/

# Code chunk options

````
```{r}
#| echo = FALSE
#| eval = TRUE
#| fig.cap = 'My figure caption'

plot(iris)
```
````

# Naming chunks helps debugging

```
processing file: test.Rmd
  |.........                                                       |  14%
  ordinary text without R code

  |.................                                               |  29%
label: setup (with options)
List of 1
 $ include: logi FALSE

  |.............................                                   |  43%
  ordinary text without R code

  |.....................................                           |  57%
label: read.data
  |.............................................                   |  71%
  ordinary text without R code

  |.........................................................       |  86%
label: plot (with options)
List of 1
 $ echo: logi FALSE


Quitting from lines 28-29 (test.Rmd)
Error in eval(predvars, data, env) : object 'specie' not found
Calls: <Anonymous> ... plot.formula -> eval -> eval -> <Anonymous> -> eval -> eval
Execution halted
```

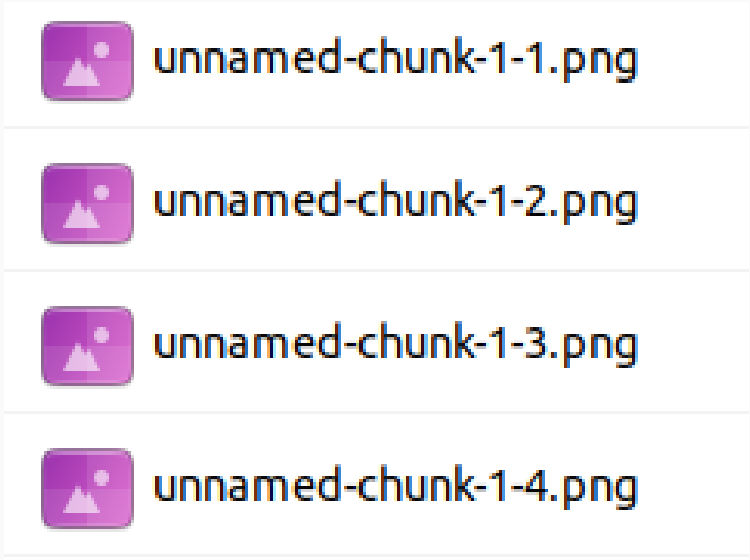
# Naming chunks helps navigating long docs

```
1  ---
2  title: "My Analysis"
3  author: "FRS"
4  output: html_document
5  ---
6
7  ```{r setup, include=FALSE}
8  knitr::opts_chunk$set(echo = TRUE)
9  ```
10
11 This is an R Markdown document. Markdown is a simple
                  for authoring HTML, PDF, and MS Word
                re details on using R Markdown see
                .rstudio.com>.
12
```

**My Analysis**
Chunk 1: setup
Chunk 2: read.data
Chunk 3: plot

11:60 (Top Level) R Markdown

# Naming chunks: figure files take chunk name



# ¡Not only R! Python, Julia, C++, SQL, Stan, etc

`knitr` engines:

```
 [1] "asis"       "asy"        "awk"        "bash"       "block"      "block2"
 [7] "bslib"      "c"          "cat"        "cc"         "coffee"     "comment"
[13] "css"        "ditaa"      "dot"        "embed"      "eviews"     "exec"
[19] "fortran"    "fortran95"  "gawk"       "go"         "groovy"     "haskell"
[25] "highlight"  "js"         "julia"      "lein"       "mysql"      "node"
[31] "octave"     "perl"       "php"        "psql"       "python"     "R"
[37] "Rcpp"       "Rscript"    "ruby"       "sas"        "sass"       "scala"
[43] "scss"       "sed"        "sh"         "sql"        "stan"       "stata"
[49] "targets"    "tikz"       "verbatim"   "zsh"
```

# Markdown: easy text formatting

```
# Header

## Subheader

*italic*

**bold**

[a link](https://example.com)
```

.

Handy: https://thinkr-open.github.io/remedy/

Or use Visual Markdown Editor

# Regenerate Word/PDF/HTML with one click

**a**

```
---
title: "Does sunshine make people happy?"
output: pdf_document
bibliography: refs.bib
---

# Introduction

Climate influences individual well-being [@Rehdanz_2005].
However, ...

# Methods

```{r echo=FALSE}
# read data
data <- read.table("data.txt", header=T)
data[10,1] <- 11   # correct error

# fit linear model
model <- lm(happiness ~ sunshine, data=data)
```

We collected data on `r nrow(data)` individuals and fitted a
linear model.

# Results

We found that...

```{r echo=FALSE, results='asis'}
# make table with model output
print(xtable::xtable(model), comment = FALSE)
```

```{r echo=FALSE, fig.height=3, fig.width=3, fig.align='center'}
visreg::visreg(model) # plot
```

# Discussion

Our results confirm that happiness is related to
sunshine (slope = `r coef(model)[2]`).

# References
```

**b**

## Does sunshine make people happy?

### Introduction

Climate influences individual well-being (Rehdanz and Maddison 2005). However, ...

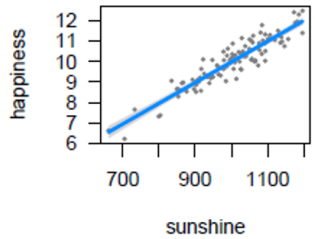
### Methods

We collected data on 100 individuals and fitted a linear model.

### Results

We found that...

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -0.0986 | 0.4271 | -0.23 | 0.8180 |
| sunshine | 0.0101 | 0.0004 | 23.75 | 0.0000 |




### Discussion

Our results confirm that happiness is related to sunshine (slope = 0.0100652).

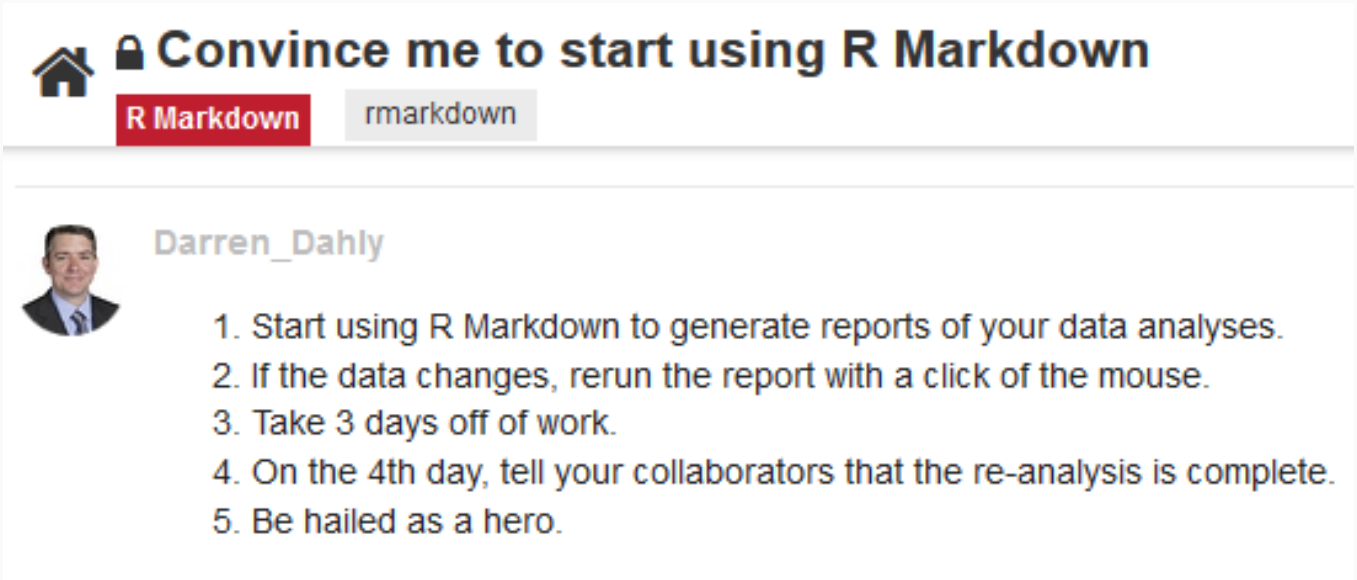
### References

Rehdanz, Katrin, and David Maddison. 2005. "Climate and Happiness." *Ecological Economics* 52 (1). Elsevier BV: 111–25. doi:10.1016/j.ecolecon.2004.06.015.

# Spotted error in the data? No problem!

- Make changes in Rmarkdown document
- Click `Knit` in Rstudio
- Report will **update automatically!**

# Why Rmarkdown?



**Convince me to start using R Markdown**

R Markdown rmarkdown

Darren_Dahly

1. Start using R Markdown to generate reports of your data analyses.
2. If the data changes, rerun the report with a click of the mouse.
3. Take 3 days off of work.
4. On the 4th day, tell your collaborators that the re-analysis is complete.
5. Be hailed as a hero.

https://community.rstudio.com/t/convince-me-to-start-using-r-markdown/1636/12

# Your turn

# Create, edit and share Rmarkdown document

File > New File > Rmarkdown

Write text

Insert code chunks

Change chunk options (echo, eval, etc)

HTML/Word/PDF output

**PDF generation requires LaTeX**

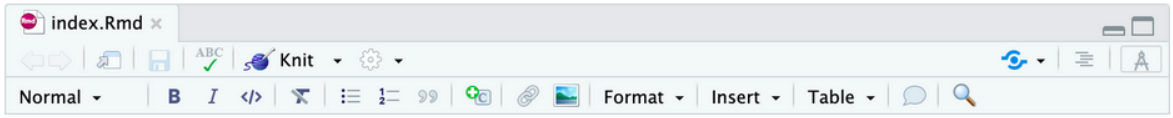
```
library('tinytex')

install_tinytex()
```

# Rmarkdown bells and whistles

# ‘Visual Rmarkdown’: Rmd as in word processor

The editor toolbar includes buttons for the most commonly used formatting commands:




Additional commands are available on the **Format**, **Insert**, and **Table** menus:

**Format**

- Bold ⌘B
- Italic ⌘I
- Code ⌘D
- Text ▸
- Bullets & Numbering ▸
- Blockquote
- Line Block
- Div Block...
- Code Block...
- Raw ▸
- Clear Formatting ⌘\
- Edit Attributes... F4

**Insert**

- Rmd Chunk ⌥⌘I
- Image... ⇧⌘I
- Link... ⌘K
- Horizontal Rule ⌘_
- Definition ▸
- Inline Math
- Display Math
- Footnote ⇧⌘F7
- Citation...
- Div Block...
- Code Block...
- YAML Block
- Comment ⇧⌘C

**Table**

- Insert Table... ⌥⌘T
- ✓ Table Header
- Table Caption
- Align Column ▸
- Insert Row Above
- Insert Row Below
- Insert Column Left
- Insert Column Right
- Delete Row
- Delete Column
- Delete Table

https://rstudio.github.io/visual-markdown-editing

# Automatic table generation

```
model <- lm(happiness ~ sunshine, data = mydata)
xtable(model)
```

| | Estimate | Std. Error | t value | Pr(>\|t\|) |
|---|---|---|---|---|
| (Intercept) | -0.0652 | 0.4265 | -0.15 | 0.8789 |
| sunshine | 0.0100 | 0.0004 | 23.68 | 0.0000 |

Many alternatives: `gtsummary`, `modelsummary`, `huxtable`, etc

# `equatiomatic` describes model structure

We fitted a linear model:

```
library('equatiomatic')
model <- lm(happiness ~ sunshine, data = mydata)
extract_eq(model)
```

$$\text{happiness} = \alpha + \beta_1(\text{sunshine}) + \epsilon \tag{1}$$

# Models that describe themselves!

```
library('report')
model <- lm(happiness ~ sunshine, data = mydata)
report(model)
```

We fitted a linear model (estimated using OLS) to predict happiness with sunshine (formula: happiness ~ sunshine). The model explains a statistically significant and substantial proportion of variance (R2 = 0.85, F(1, 98) = 560.90, p < .001, adj. R2 = 0.85). The model's intercept, corresponding to sunshine = 0, is at -0.07 (95% CI [-0.91, 0.78], t(98) = -0.15, p = 0.879). Within this model:

- The effect of sunshine is statistically significant and positive (beta = 0.01, 95% CI [9.18e-03, 0.01], t(98) = 23.68, p < .001; Std. beta = 0.92, 95% CI [0.85, 1.00])

Standardized parameters were obtained by fitting the model on a standardized version of the dataset.

95% Confidence Intervals (CIs) and p-values were computed using a Wald t-distribution approximation.

# Insert equations with LaTeX

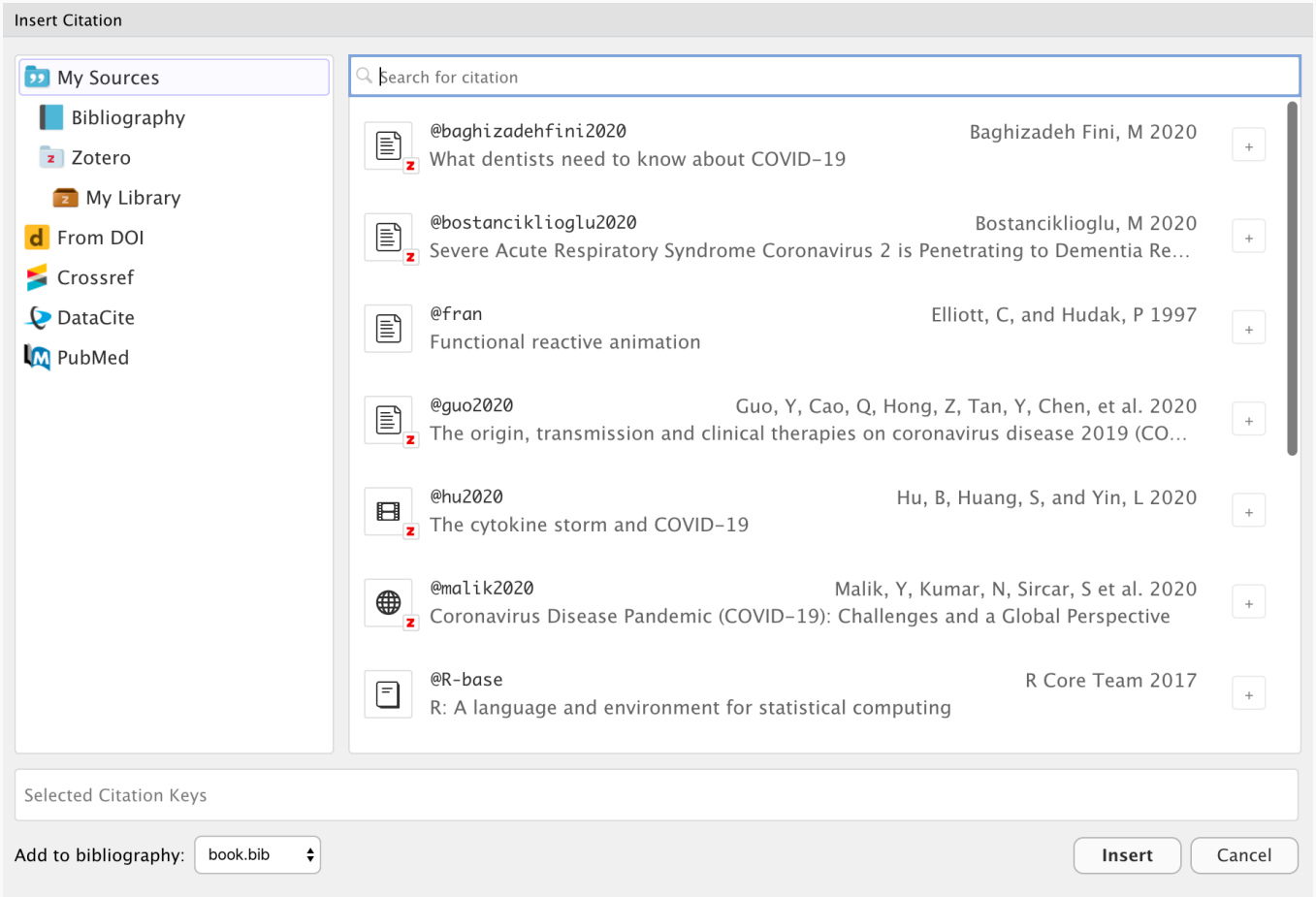
Using LaTeX:

```
$$
y \sim N(\mu, \sigma^2)
$$
```

$$y \sim N(\mu, \sigma^2)$$

- Mathpix: https://github.com/jonocarroll/mathpix

# Citing bibliography




https://rstudio.github.io/visual-markdown-editing/#/citations

# Using BibTeX file with references

```
---
title: "My awesome Rmd"
output: html_document
bibliography: references.bib
---
```

# Format bibliography for any journal

```
---
title: "Does sunshine make people happy?"
author: "FRS"
output: word_document
bibliography: myrefs.bib
csl: ecology-letters.csl
---
```

Thousands of Citation Styles:

https://www.zotero.org/styles

https://github.com/citation-style-language/styles

# Rmarkdown templates

- rticles
- papaja
- rrtools
- pinp
- rmdTemplates
- pagedreport
- GitHub!

My cool paper written in Rmarkdown

F. Rodriguez-Sanchez[a,1,2] and And Friends[a,b]

[a]Some Institute of Technology, Department, Street, City, State, Zip; [b]Another University Department, Street, City, State, Zip

This manuscript was compiled on September 18, 2018

**Please provide an abstract of no more than 250 words in a single paragraph. Abstracts should explain to the general reader the major contributions of the article. References in the abstract must be cited in full within the abstract itself and cited in the text.**

one | two | optional | optional | optional

This PNAS journal template is provided to help you write your work in the correct journal format. Instructions for use are provided below.

Note: please start your introduction without including the word "Introduction" as a section heading (except for math articles in the Physical Sciences section); this heading is implied in the first paragraphs.

**Guide to using this template**

Please note that whilst this template provides a preview of the typeset manuscript for submission, to help in this preparation, it will not necessarily be the final publication layout. For more detailed information please see the PNAS Information for Authors.

**Author Affiliations.** Include department, institution, and complete address, with the ZIP/postal code, for each author. Use lower case letters to match authors with institutions, as shown in the example. Authors with an ORCID ID may supply this information at submission.

**Submitting Manuscripts.** All authors must submit their articles at PNAScentral. If you are using Overleaf to write your article, you can use the "Submit to PNAS" option in the top bar of the editor window.

**Format.** Many authors find it useful to organize their manuscripts with the following order of sections; Title, Author Affiliation, Keywords, Abstract, Significance Statement, Results, Discussion, Materials and methods, Acknowledgments, and References. Other orders and headings are permitted.

**Manuscript Length.** PNAS generally uses a two-column format averaging 67 characters, including spaces, per line. The maximum length of a Direct Submission research article is six pages and a PNAS PLUS research article is ten pages including all text, spaces, and the number of characters displaced by figures, tables, and equations. When submitting tables, figures, and/or equations in addition to text, keep the text for your manuscript under 39,000 characters (including spaces) for Direct Submissions and 72,000 characters (including spaces) for PNAS PLUS.

**References.** References should be cited in numerical order as they appear in text; this will be done automatically via bibtex, e.g. (1) and (2, 3). All references, including for the SI, should be included in the main manuscript file. References appearing in both sections should not be duplicated. SI references

Fig. 1. Placeholder image of a frog with a longer example caption to show justification setting.

included in tables should be included with the main reference section.

**Data Archival.** PNAS must be able to archive the data essential to a published article. Where such archiving is not possible, deposition of data in public databases, such as GenBank, ArrayExpress, Protein Data Bank, UniData, and others outlined in the Information for Authors, is acceptable.

**Language-Editing Services.** Prior to submission, authors who believe their manuscripts would benefit from professional editing are encouraged to use a language-editing service (see list at www.pnas.org/site/authors/language-editing.xhtml). PNAS does not take responsibility for or endorse these services, and their use has no bearing on acceptance of a manuscript for publication.

**Significance Statement**

Authors must submit a 120-word maximum statement about the significance of their research paper written at a level understandable to an undergraduate educated scientist outside their field of speciality. The primary goal of the Significance Statement is to explain the relevance of the work in broad context to a broad readership. The Significance Statement appears in the paper itself and is required for all research papers.

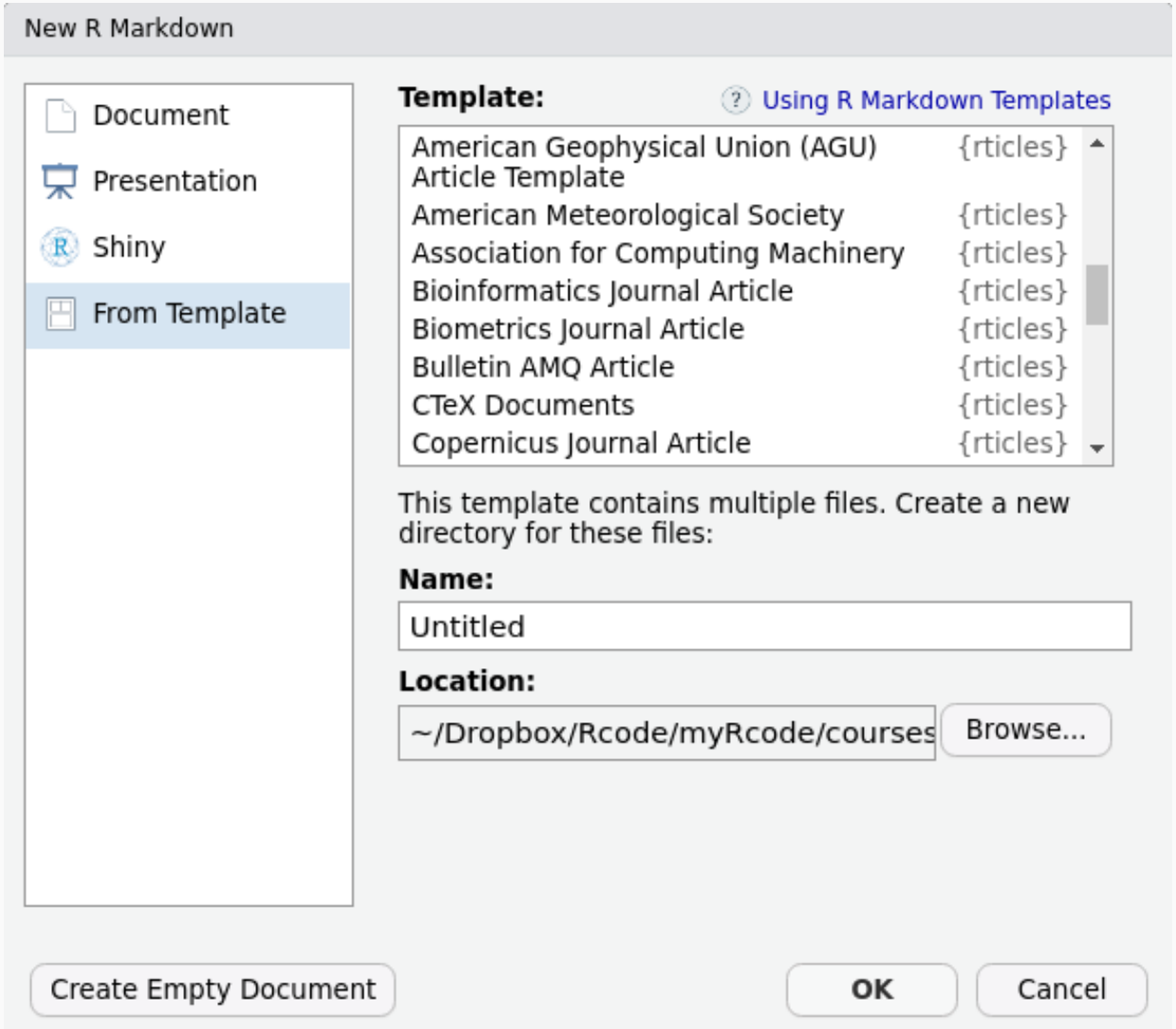
Please provide details of author contributions here.

Please declare any conflict of interest here.

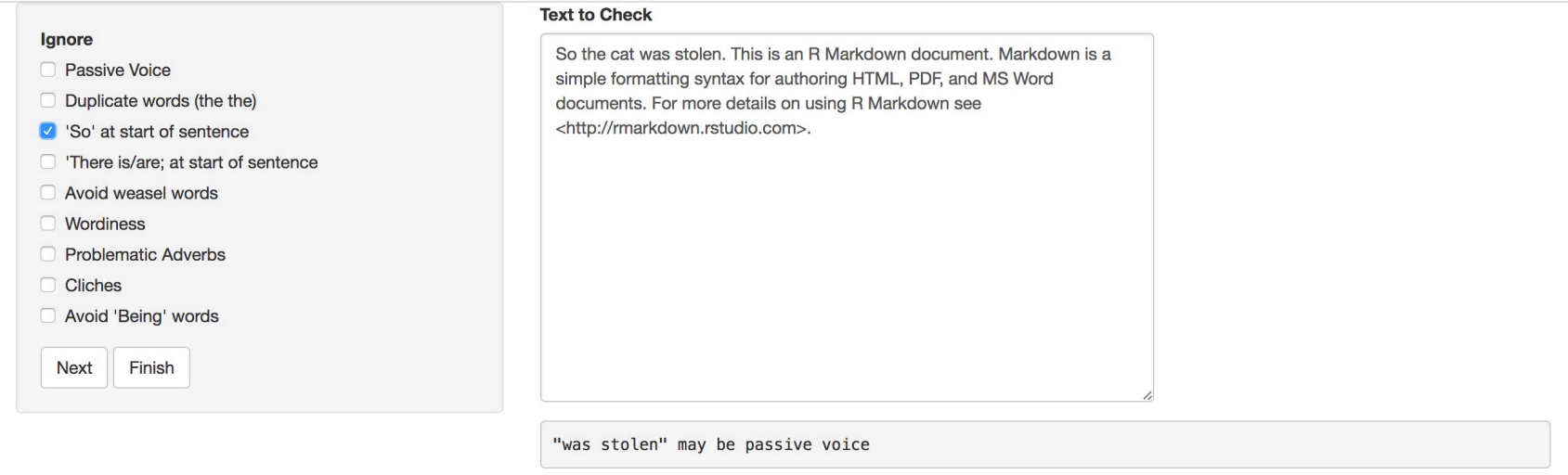
www.pnas.org/cgi/doi/10.1073/pnas.XXXXXXXXXX

PNAS | September 18, 2018 | vol. XXX | no. XX | 1–2

# Accessing Rmd templates



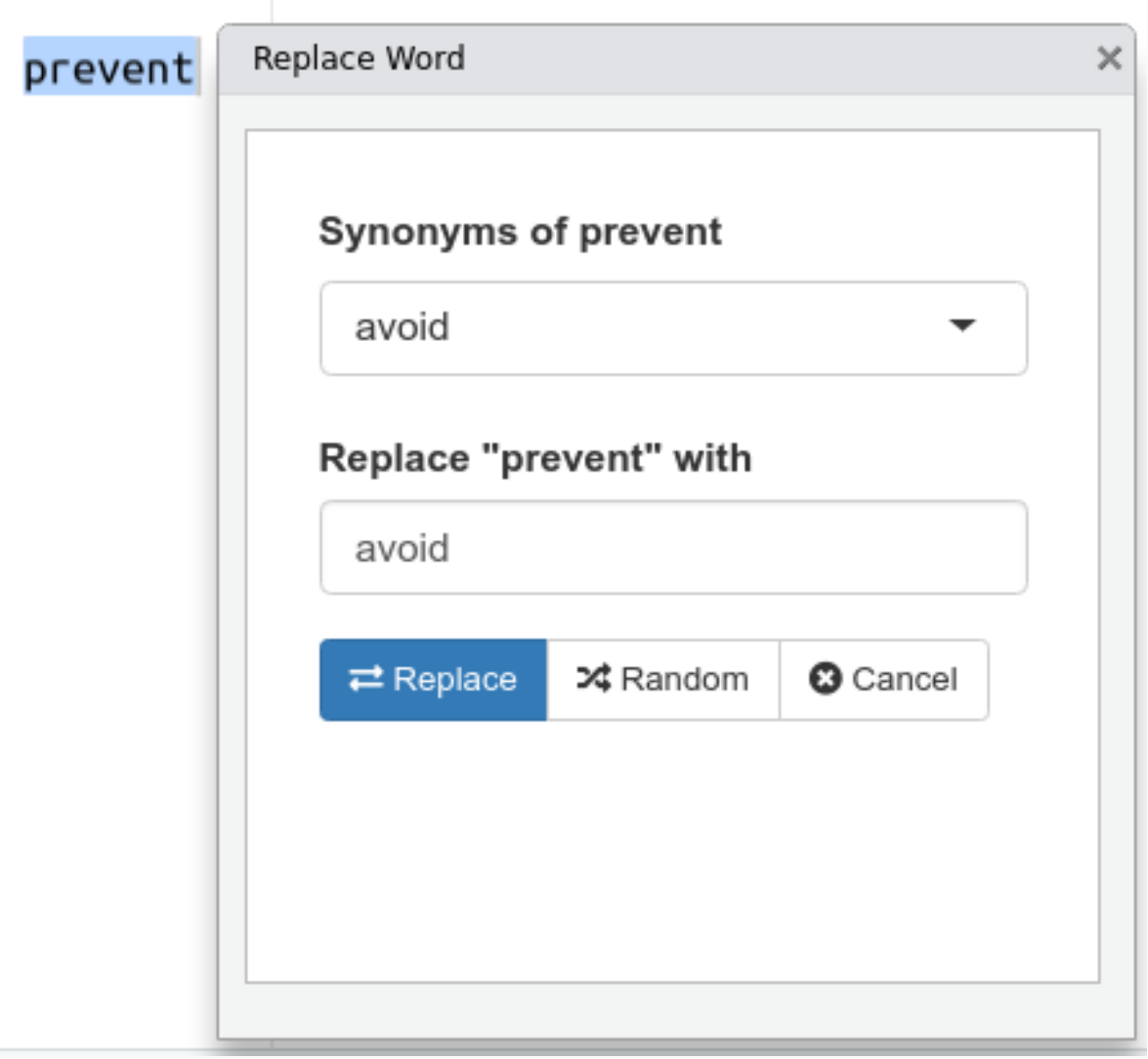
# Revise writing style: gramr




https://github.com/ropenscilabs/gramr

https://github.com/nevrome/wellspell.addin

# Find synonyms




https://github.com/gadenbuie/synamyn

# Word count and readability

| Method | koRpus | stringi |
|:---|:---|:---|
| Word count | 107 | 104 |
| Character count | 604 | 603 |
| Sentence count | 10 | Not available |
| Reading time | 0.5 minutes | 0.5 minutes |

https://github.com/benmarwick/wordcountaddin

# Automated reproducibility checks

https://github.com/brandmaier/reproducibleRchunks

```
15 ## Some Computation
16
17 Here is a computation:
18
19 ```{reproducibleR addition}
20 my_sum <- x + 1
21 ```
22
```

Here is a computation:

```
my_sum <- x + 1
```

Code Chunk Reproduction Report

- ✅ my_sum: REPRODUCTION SUCCESSFUL

Here is a computation:

```
my_sum <- x + 1
```

Code Chunk Reproduction Report

- ❌ my_sum: **REPRODUCTION FAILED** Fingerprints are not identical.

# Write books, theses, with bookdown

## BOOKDOWN

### Write HTML, PDF, ePub, and Kindle books with R Markdown

The **bookdown** package is an open-source R package that facilitates writing books and long-form articles/reports with R Markdown. Features include:

- Generate printer-ready books and ebooks from R Markdown documents.
- A markup language easier to learn than LaTeX, and to write elements such as section headers, lists, quotes, figures, tables, and citations.
- Multiple choices of output formats: PDF, LaTeX, HTML, EPUB, and Word.
- Possibility of including dynamic graphics and interactive applications (HTML widgets and Shiny apps).
- Support a wide range of languages: R, C/C++, Python, Fortran, Julia, Shell scripts, and SQL, etc.
- LaTeX equations, theorems, and proofs work for all output formats.
- Can be published to GitHub, bookdown.org, and any web servers.
- Integrated with the RStudio IDE.
- One-click publishing to https://bookdown.org.


https://bookdown.org/


https://slides.yihui.org/xaringan/

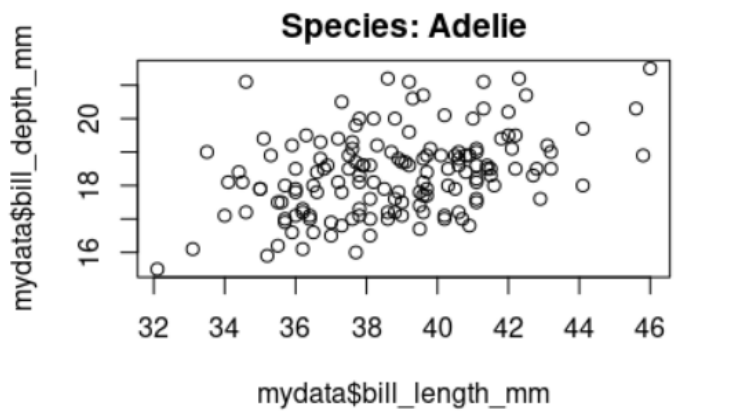
# Parameterised reports

```
---
title: "My template report"
output: html_document
params:
  sp: Adelie
---
```

```{r}
library(palmerpenguins)
data("penguins")

mydata <- subset(penguins, species == params$sp)

plot(mydata$bill_length_mm, mydata$bill_depth_mm,
     main = paste0("Species: ", params$sp))
```




https://bookdown.org/yihui/rmarkdown/parameterized-reports.html

# Render thousands of individual reports from Rmd template

```
library('rmarkdown')

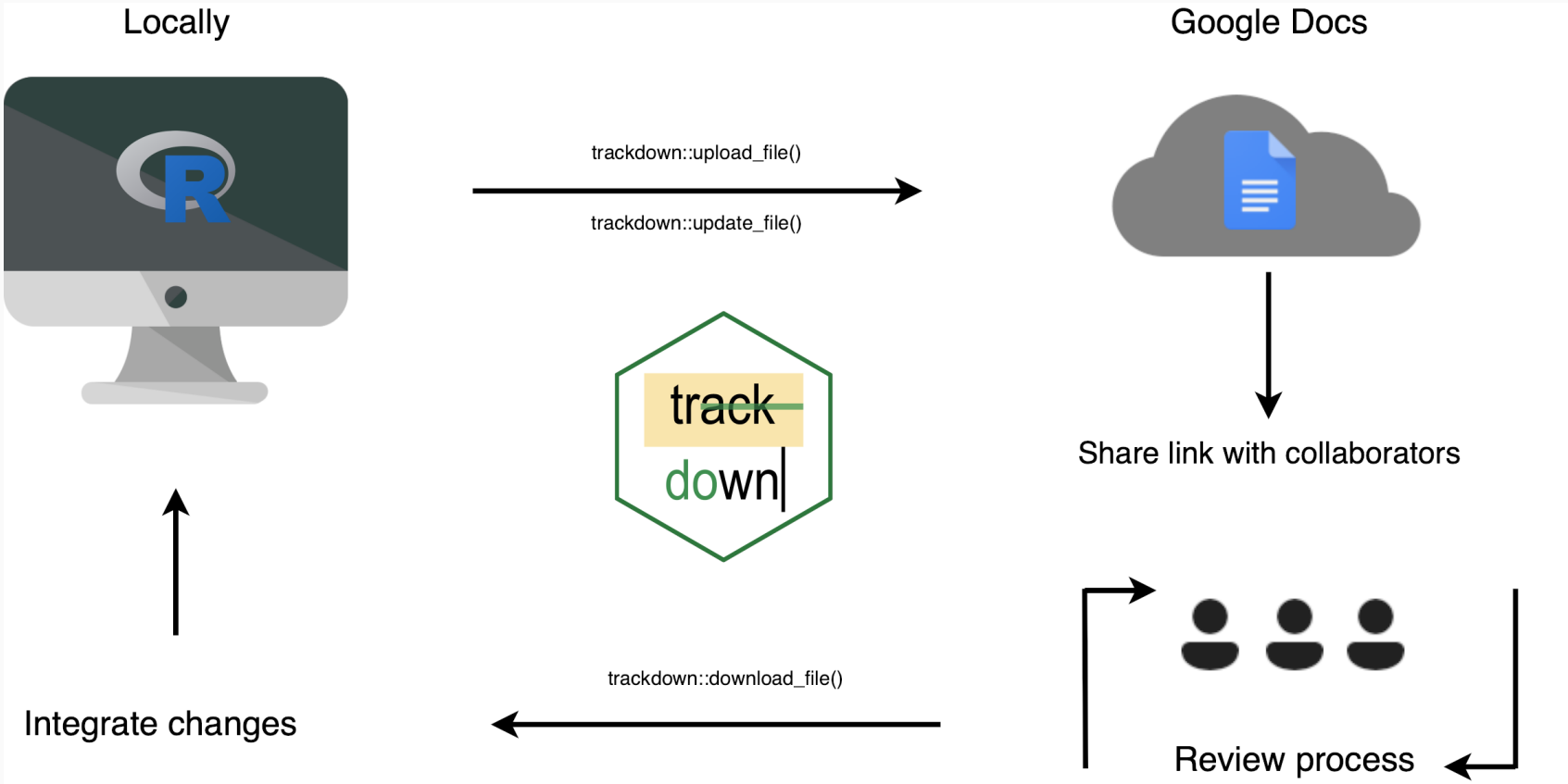
for (i in unique(penguins$species)) {

  render('template_report.Rmd',
         params = list(sp = i))

}
```

# Collaborative writing

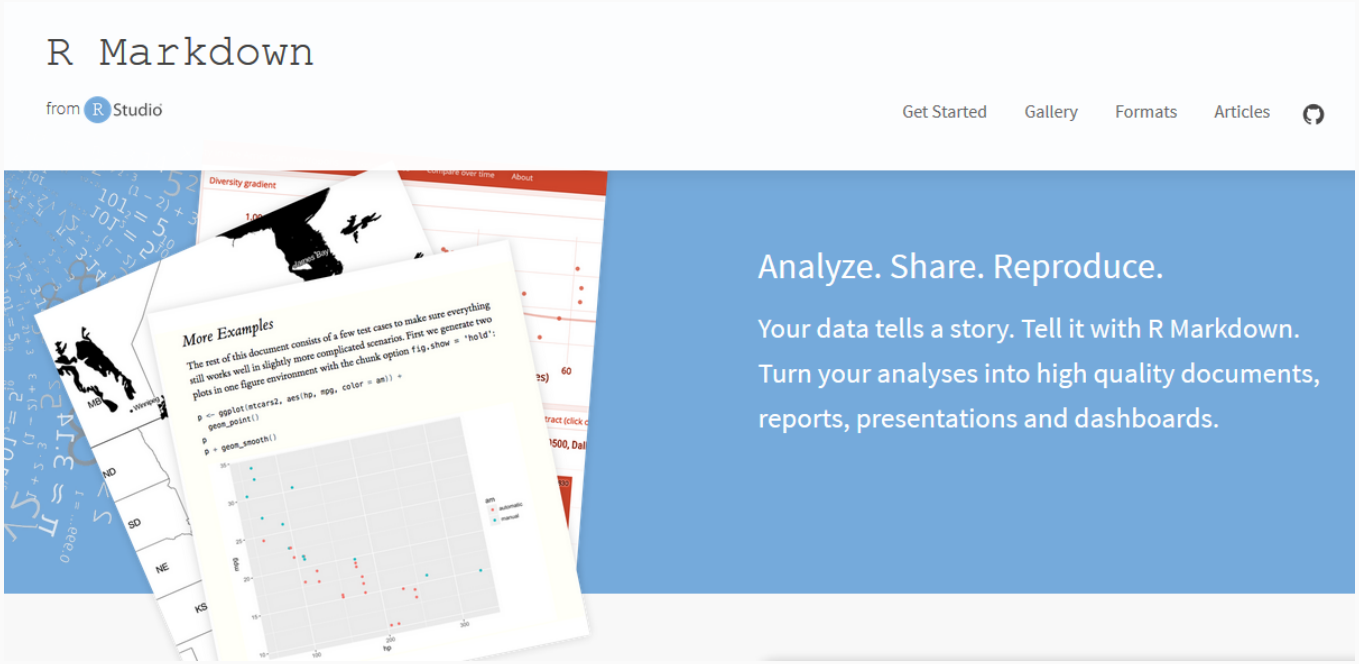
- GitHub, GitLab, etc
- Google Docs (trackdown)
- redoc



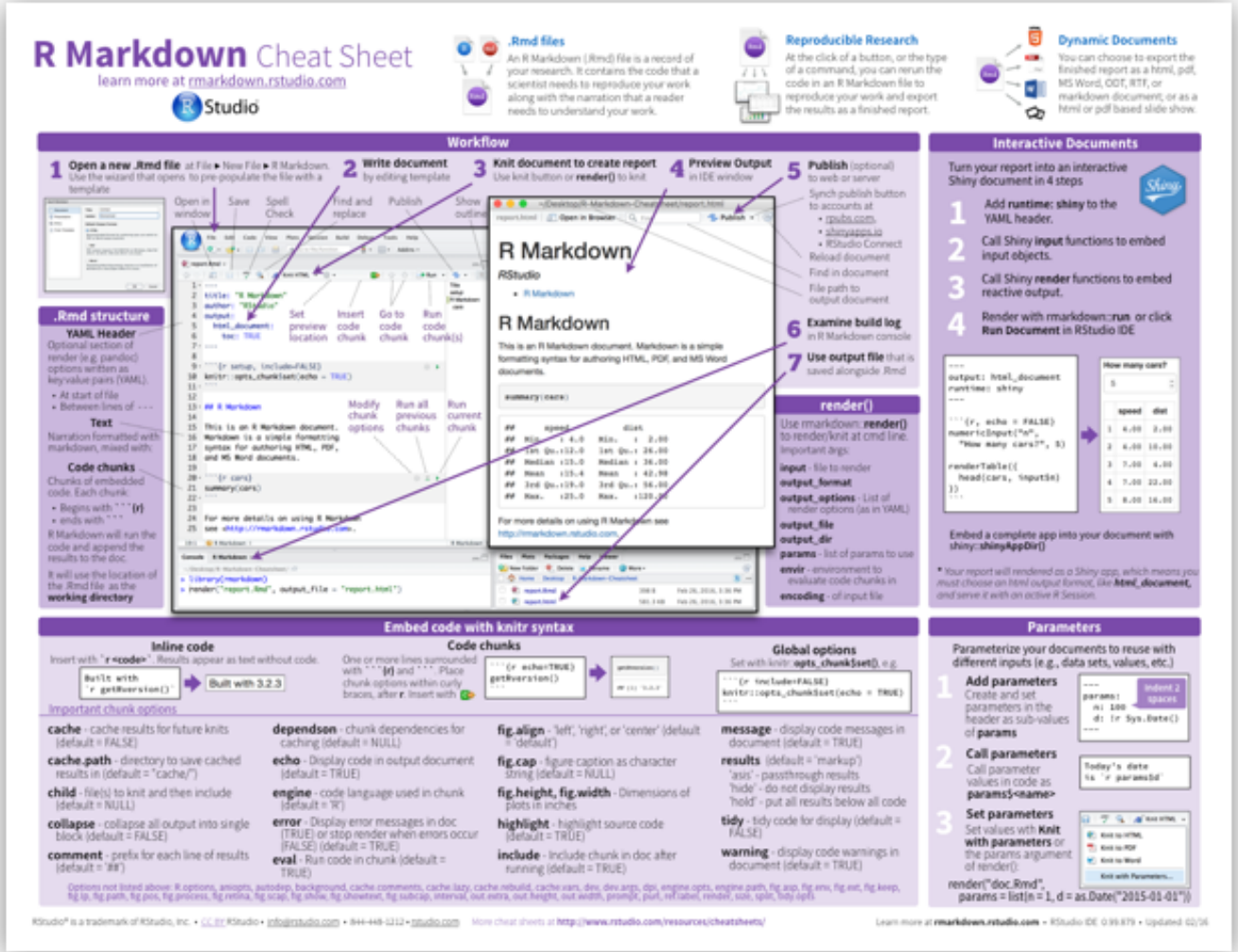
# Rmarkdown resources

# Rmarkdown website

http://rmarkdown.rstudio.com/

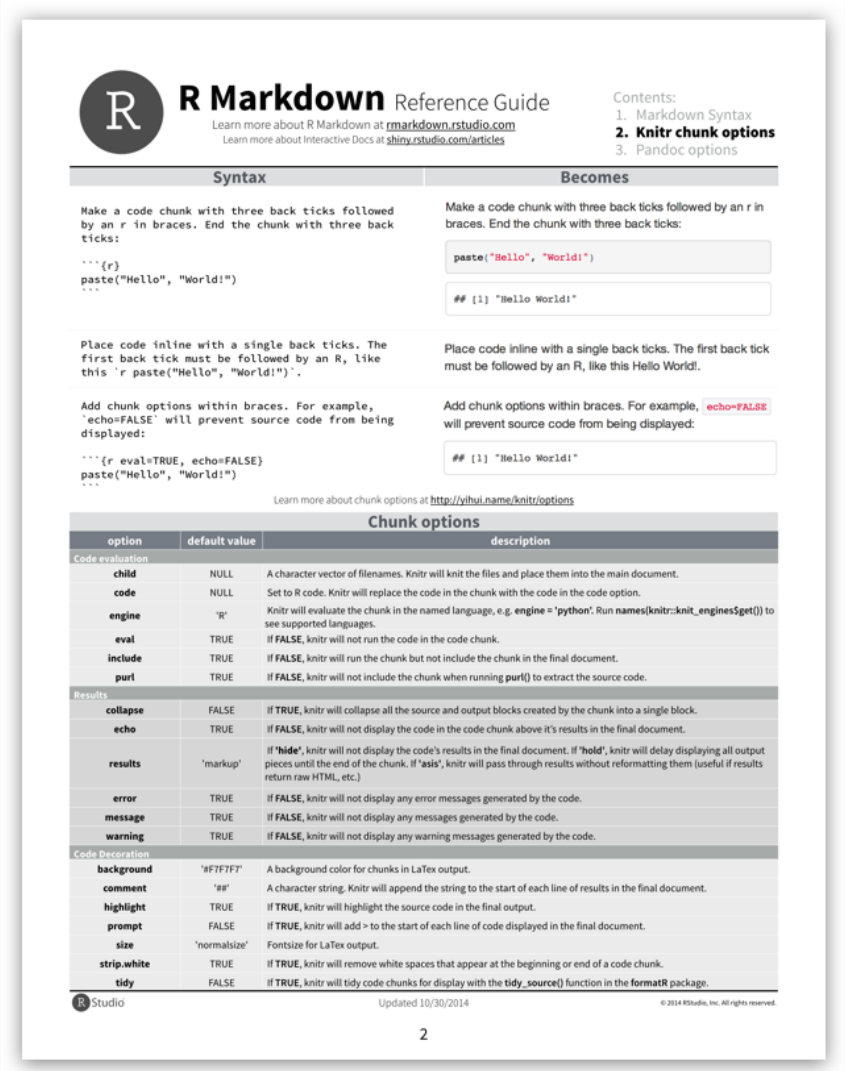


# Rmarkdown cheat sheet




https://www.rstudio.org/links/r_markdown_cheat_sheet

# Rmarkdown reference guide



## R Markdown Reference Guide

Learn more about R Markdown at rmarkdown.rstudio.com
Learn more about Interactive Docs at shiny.rstudio.com/articles

Contents:
1. Markdown Syntax
2. **Knitr chunk options**
3. Pandoc options

| Syntax | Becomes |
|---|---|
| Make a code chunk with three back ticks followed by an r in braces. End the chunk with three back ticks:<br>```` ```{r} paste("Hello", "World!") ``` ```` | Make a code chunk with three back ticks followed by an r in braces. End the chunk with three back ticks:<br>`paste("Hello", "World!")`<br>`## [1] "Hello World!"` |
| Place code inline with a single back ticks. The first back tick must be followed by an R, like this `` `r paste("Hello", "World!")` ``. | Place code inline with a single back ticks. The first back tick must be followed by an R, like this Hello World!. |
| Add chunk options within braces. For example, `` `echo=FALSE` `` will prevent source code from being displayed:<br>```` ```{r eval=TRUE, echo=FALSE} paste("Hello", "World!") ``` ```` | Add chunk options within braces. For example, `echo=FALSE` will prevent source code from being displayed:<br>`## [1] "Hello World!"` |

Learn more about chunk options at http://yihui.name/knitr/options

### Chunk options

| option | default value | description |
|---|---|---|
| **Code evaluation** | | |
| **child** | NULL | A character vector of filenames. Knitr will knit the files and place them into the main document. |
| **code** | NULL | Set to R code. Knitr will replace the code in the chunk with the code in the code option. |
| **engine** | 'R' | Knitr will evaluate the chunk in the named language, e.g. **engine = 'python'**. Run **names(knitr::knit_engines$get())** to see supported languages. |
| **eval** | TRUE | If **FALSE**, knitr will not run the code in the code chunk. |
| **include** | TRUE | If **FALSE**, knitr will run the chunk but not include the chunk in the final document. |
| **purl** | TRUE | If **FALSE**, knitr will not include the chunk when running **purl()** to extract the source code. |
| **Results** | | |
| **collapse** | FALSE | If **TRUE**, knitr will collapse all the source and output blocks created by the chunk into a single block. |
| **echo** | TRUE | If **FALSE**, knitr will not display the code in the code chunk above it's results in the final document. |
| **results** | 'markup' | If **'hide'**, knitr will not display the code's results in the final document. If **'hold'**, knitr will delay displaying all output pieces until the end of the chunk. If **'asis'**, knitr will pass through results without reformatting them (useful if results return raw HTML, etc.) |
| **error** | TRUE | If **FALSE**, knitr will not display any error messages generated by the code. |
| **message** | TRUE | If **FALSE**, knitr will not display any messages generated by the code. |
| **warning** | TRUE | If **FALSE**, knitr will not display any warning messages generated by the code. |
| **Code Decoration** | | |
| **background** | '#F7F7F7' | A background color for chunks in LaTex output. |
| **comment** | '##' | A character string. Knitr will append the string to the start of each line of results in the final document. |
| **highlight** | TRUE | If **TRUE**, knitr will highlight the source code in the final output. |
| **prompt** | FALSE | If **TRUE**, knitr will add > to the start of each line of code displayed in the final document. |
| **size** | 'normalsize' | Fontsize for LaTex output. |
| **strip.white** | TRUE | If **TRUE**, knitr will remove white spaces that appear at the beginning or end of a code chunk. |
| **tidy** | FALSE | If **TRUE**, knitr will tidy code chunks for display with the **tidy_source()** function in the **formatR** package. |

RStudio Updated 10/30/2014 

2

https://github.com/rstudio/cheatsheets/blob/main/old/pdfs/rmarkdown-reference.pdf

# Rmarkdown books

## R Markdown: The Definitive Guide

**by Yihui Xie, J. J. Allaire, Garrett Grolemund**

2018-09-11

Star 239


The first official book authored by the core R Markdown developers that provides a comprehensive and accurate reference to the R Markdown ecosystem. With R Markdown, you can easily create reproducible data analysis reports, presentations, dashboards, interactive applications, books, dissertations, websites, and journal articles, while enjoying the simplicity of Markdown and the great power of R and other languages. *Read more* →

https://bookdown.org/yihui/rmarkdown/

https://bookdown.org/yihui/rmarkdown-cookbook/

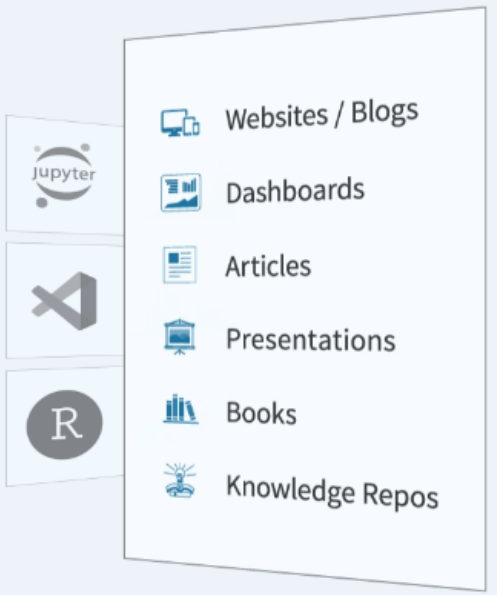
# Quarto

# Welcome to Quarto®

## An open-source scientific and technical publishing system

- Author using Jupyter notebooks or with plain text markdown in your favorite editor.
- Create dynamic content with Python, R, Julia, and Observable.
- Publish reproducible, production quality articles, presentations, dashboards, websites, blogs, and books in HTML, PDF, MS Word, ePub, and more.
- Share knowledge and insights organization-wide by publishing to Posit Connect, Confluence, or other publishing systems.
- Write using Pandoc markdown, including equations, citations, crossrefs, figure panels, callouts, advanced layout, and more.

**Analyze. Share. Reproduce. You have a story to tell with data—tell it with Quarto.**




https://quarto.org/

# Quarto manuscripts

## La Palma Earthquakes


AUTHORS
Steve Purves
Rowan Cockett

AFFILIATION
Curvenote
Curvenote


OTHER FORMATS
MS Word
PDF (agu)
MECA Bundle


PUBLISHED
February 23, 2024



ABSTRACT
In September 2021, a significant jump in seismic activity on the island of La Palma (Canary Islands, Spain) signaled the start of a volcanic crisis that still continues at the time of writing. Earthquake data is continually collected and published by the Instituto Geográphico Nacional (IGN). …



KEYWORDS
La Palma, Earthquakes


Table of contents
- 1 Introduction
- 2 Data & Methods
- 3 Conclusion
- References

Notebooks
- Article Notebook
- Data Screening

## 1 Introduction

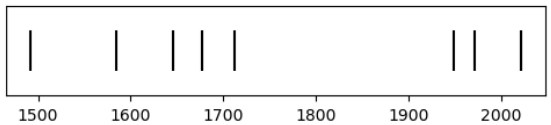



Figure 1: Timeline of recent earthquakes on La Palma

Source: Article Notebook

Based on data up to and including 1971, eruptions on La Palma happen every 79.8 years on average.

Studies of the magma systems feeding the volcano, such as Marrero et al. (2019), have proposed that there are two main magma reservoirs feeding the Cumbre Vieja volcano; one in the mantle (30-40km depth) which charges and in turn feeds a shallower crustal reservoir (10-20km depth).

Eight eruptions have been recorded since the late 1400s (Figure 1).

Data and methods are discussed in Section 2.

https://quarto-ext.github.io/manuscript-template-jupyter/

https://quarto.org/docs/manuscripts/

# Hundreds of Quarto extensions

https://m.canouil.dev/quarto-extensions/

Journal templates:

https://quarto.org/docs/extensions/listing-journals.html

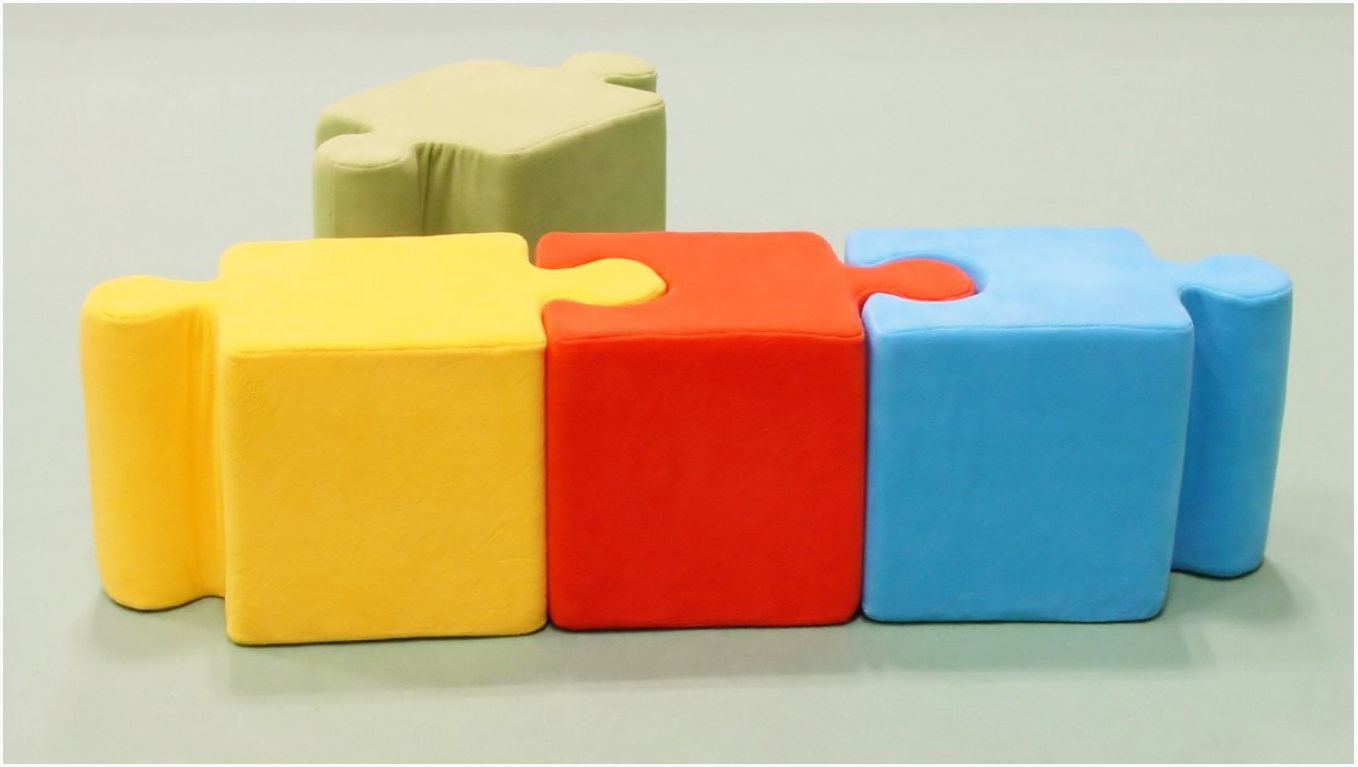
# Your turn

# Your turn

- Try visual markdown editor
- Add bibliography
- Try templates (rticles, rmdTemplates)
- Parameterised reports (e.g. different iris or penguin species)
- Quarto manuscript

# Workflow management

In complex projects we must **keep pieces organised**



# `makefile` runs all code in right order

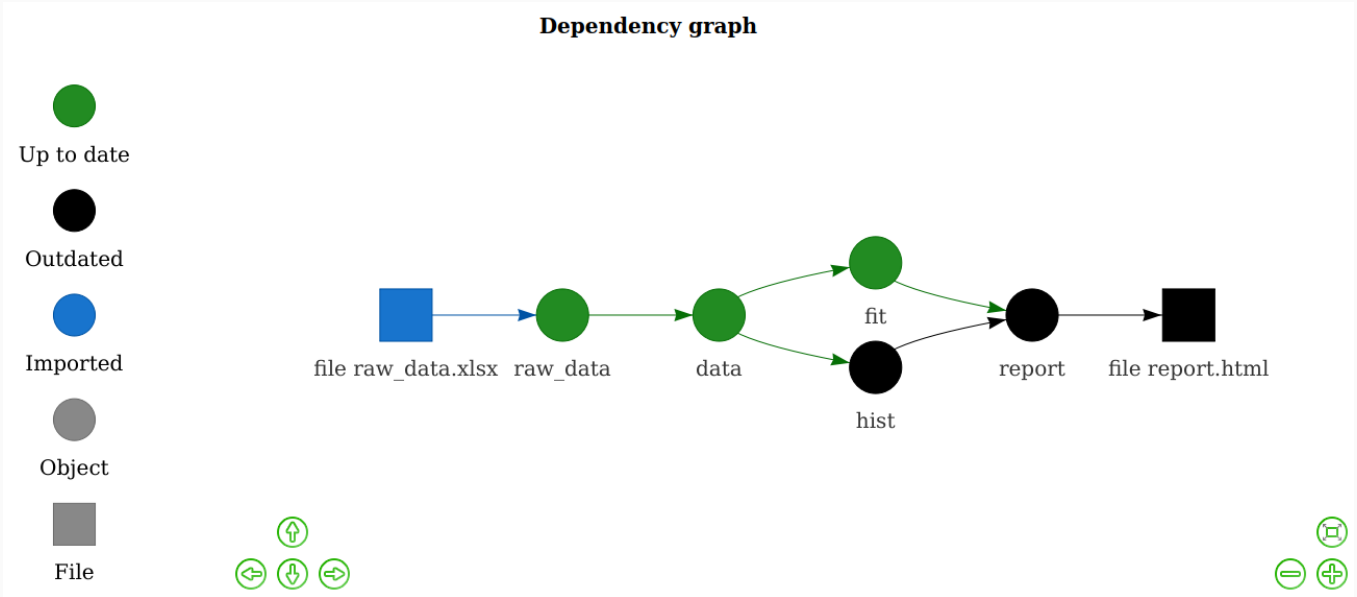
`makefile.R`

```
source('clean_data.R')

source('fit_model.R')

render('report.Rmd')
```

# targets: advanced workflow management

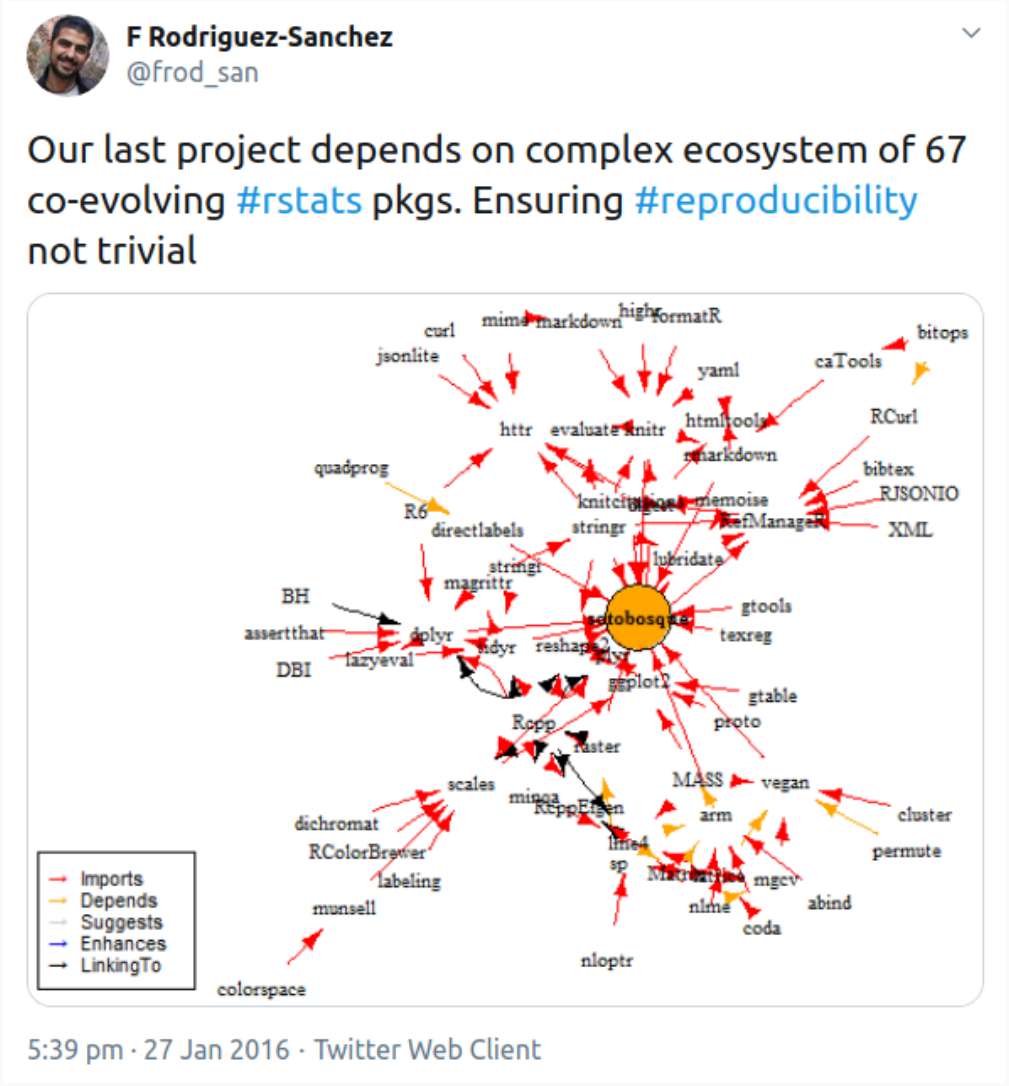



https://docs.ropensci.org/targets/

# Your turn

- Write makefile.R for your project
- Try `targets` minimal example
- https://github.com/wlandau/targets-minimal

# Controlling software dependencies


F Rodriguez-Sanchez
@frod_san
Our last project depends on complex ecosystem of 67 co-evolving #rstats pkgs. Ensuring #reproducibility not trivial
curl
mime
markdown
highr
formatR
bitops
jsonlite
caTools
yaml
RCurl
httr
evaluate
knitr
htmltools
rmarkdown
bibtex
quadprog
RJSONIO
R6
memoise
XML
directlabels
stringr
lubridate
magrittr
BH
gtools
assertthat
dplyr
texreg
DBI
lazyeval
gplot2
gtable
Rcpp
proto
raster
MASS
vegan
scales
arm
cluster
dichromat
permute
RColorBrewer
sp
labeling
mgcv
munsell
nlme
abind
coda
nloptr
colorspace
Imports
Depends
Suggests
Enhances
LinkingTo
5:39 pm · 27 Jan 2016 · Twitter Web Client

Updating R packages broke your script?

Need to run an old script from you, or someone else?

How to reproduce your analysis in a year,
or different computer?

# `sessionInfo` records OS & used packages

```
sessionInfo()
```

```
R version 4.4.2 (2024-10-31)
Platform: x86_64-pc-linux-gnu
Running under: Ubuntu 20.04.6 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/liblapack.so.3;  LAPACK version 3.9.0

locale:
 [1] LC_CTYPE=en_GB.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=es_ES.UTF-8        LC_COLLATE=en_GB.UTF-8
 [5] LC_MONETARY=es_ES.UTF-8    LC_MESSAGES=en_GB.UTF-8
 [7] LC_PAPER=es_ES.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=es_ES.UTF-8 LC_IDENTIFICATION=C

time zone: Europe/Madrid
tzcode source: system (glibc)

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] report_0.5.9       equatiomatic_0.3.3 xtable_1.8-4       knitr_1.49

loaded via a namespace (and not attached):
 [1] sandwich_3.1-1     generics_0.1.3     tidyr_1.3.1        lattice_0.22-6
```

# `checkpoint` recreates R packages in given date

```
library('checkpoint')

options(checkpoint.mranUrl="https://packagemanager.posit.co/")

checkpoint('2024-10-08')

source('analysis.R')
```

1. Detects packages used
2. Installs version from given date (only CRAN)
3. Independent install (not messing w/ main library)

# `automagic` records & install packages (CRAN + GitHub)

```
automagic::make_deps_file()
```

File `deps.yaml` records dependencies:

```
- Package: equatiomatic
  Repository: CRAN
  Version: 0.1.0

- Package: report
  GithubUsername: easystats
  GithubRepo: report
  GithubRef: HEAD
  GithubSHA1: c48a4bb0a40df7116bc502aa3ce2cbbc9d70b7e2
```

To install all those dependencies:

```
automagic()
```

# renv: recommended way to control dependencies

```
renv::init()
# Create private package library for project

renv::snapshot()
# Capture dependencies in lockfile

renv::restore()
# Regenerate dependencies from lockfile
```

https://rstudio.github.io/renv/

To ensure reproducibility,

besides R packages

we also need to control

**computational environment**

`Docker` recreates virtual systems

from a `Dockerfile`

# rang recreates environment (pkgs + external software)

https://gesistsa.github.io/rang/

**GA1:** Get the dependency graph of several R packages on CRAN or Github at a specific snapshot date(time)

```
graph ← resolve(c("crsh/papaja", "rio"), snapshot_date = "2019-07-21")
```

Dockerize the dependency graph to a directory

```
dockerize(graph, output_dir = "rangtest")
```

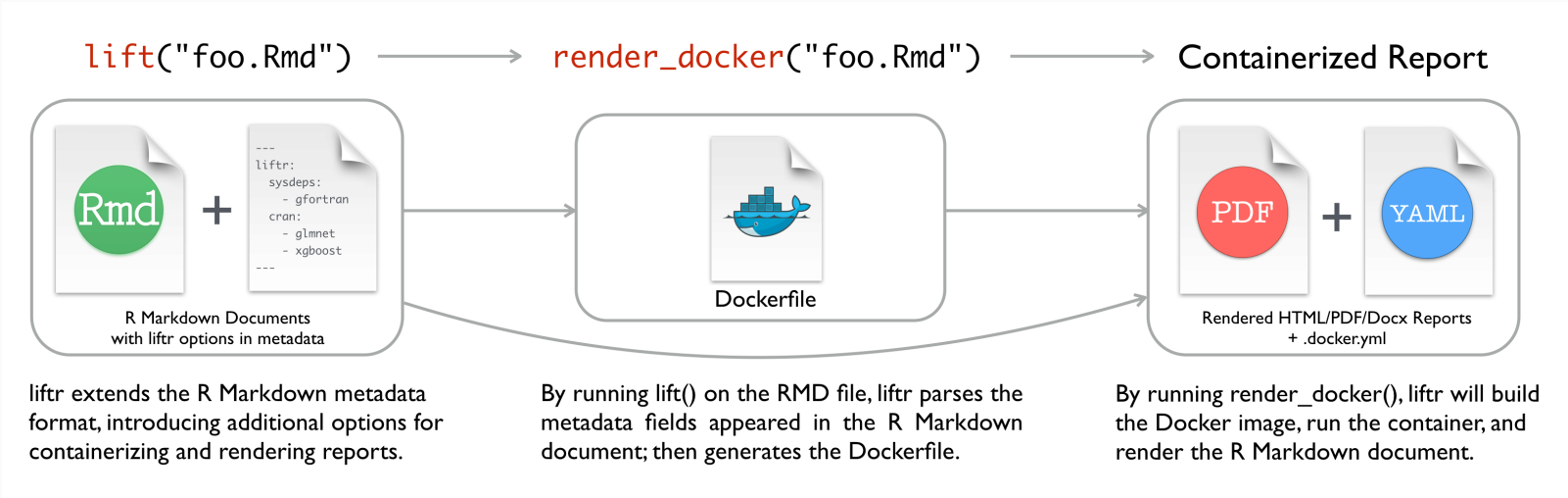
You can build the Docker image either by the R package `stevedore` or Docker CLI client. We use the CLI client.

```
docker build -t rangimg ./rangtest ## might need sudo
```

Launch the container with the built image

```
docker run --rm --name "rangcontainer" -ti rangimg
```

# liftr: process Rmd in Docker container




https://liftr.me/

# containerit creates Dockerfile

```
library('containerit')

dockfile <- dockerfile(from = 'mypaper.Rmd')
```

https://o2r.info/containerit

# `tugboat` creates Dockerfile w/ entire software environment

## tugboat


A simple R package to generate a Dockerfile and corresponding Docker image from an analysis directory. tugboat uses the renv package to automatically detect all the packages necessary to replicate your analysis and will generate a Dockerfile that contains an exact copy of your entire directory with all the packages installed.

tugboat transforms an unstructured analysis folder into a renv.lock file and constructs a Docker image that includes all your essential R packages based on this lockfile.

tugboat may be of use, for example, when preparing a replication package for research. With tugboat, you can take a directory on your local computer and quickly generate a Dockerfile and Docker image that contains all the code and the necessary software to reproduce your findings.

```
library(tugboat)
create()
build()
```

https://www.dmolitor.com/tugboat/

# `rix`: reproducible environments with Nix

https://docs.ropensci.org/rix/

> rix is an R package that leverages Nix, a package manager focused on reproducible builds. With Nix, you can create project-specific environments with a custom version of R, its packages, and all system dependencies (e.g., `GDAL`). Nix ensures full reproducibility, which is crucial for research and development projects.

# Remember to cite software used!

https://pakillo.github.io/grateful/

```
library('grateful')
cite_packages()
```

## `grateful` citation report

### R packages used

| Package | Version | Citation |
|---|---|---|
| base | 4.2.3 | R Core Team (2023) |
| lme4 | 1.1.32 | Bates et al. (2015) |
| tidyverse | 2.0.0 | Wickham et al. (2019) |
| vegan | 2.6.4 | Oksanen et al. (2022) |

**You can paste this paragraph directly in your report:**

We used R version 4.2.3 (R Core Team 2023) and the following R packages: lme4 v. 1.1.32 (Bates et al. 2015), tidyverse v. 2.0.0 (Wickham et al. 2019), vegan v. 2.6.4 (Oksanen et al. 2022).

### Package citations


Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker. 2015. "Fitting Linear Mixed-Effects Models Using lme4." *Journal of Statistical Software* 67 (1): 1–48. https://doi.org/10.18637/jss.v067.i01.

Oksanen, Jari, Gavin L. Simpson, F. Guillaume Blanchet, Roeland Kindt, Pierre Legendre, Peter R. Minchin, R. B. O'Hara, et al. 2022. *vegan: Community Ecology Package*. https://github.com/vegandevs/vegan.

R Core Team. 2023. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

# Your turn

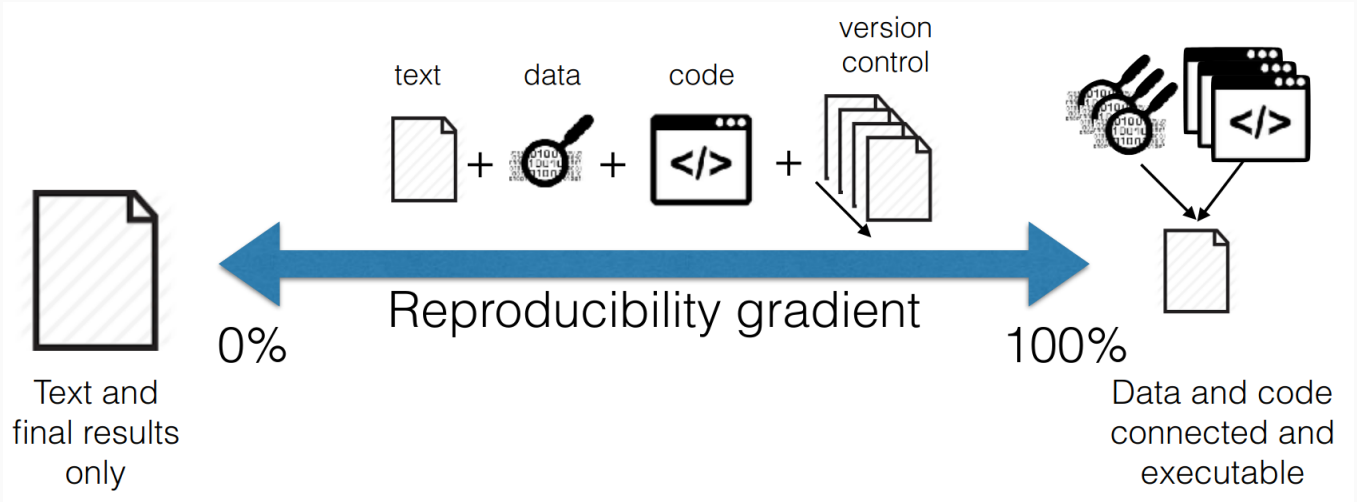
- Create script/Rmd using different packages
- Call `checkpoint` on former date
- Record dependencies:
  - `renv::snapshot`
- Recreate packages
  - `restore()`

# How to write more reproducible code

- Building reproducible analytical pipelines with R
- BES guide to reproducible code
- Turing Way
- Good enough practices in scientific computing
- Ciencia reproducible: qué, por qué, cómo
- `https://rstats.wtf`
- `fertile` package
- CodeCheck

# Reproducibility

- Good for you, good for science
- Requires systemic changes
- Reproducibility gradient: step by step



# Happy collaboration!