Reproducible Workflows for Better Science and Efficient Collaboration

---

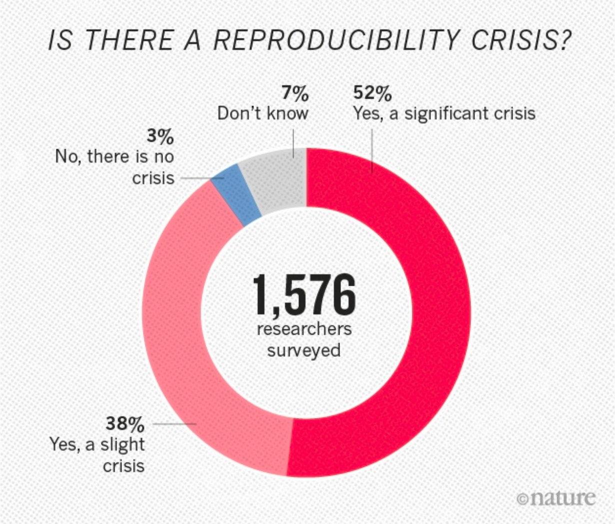
Francisco Rodriguez-Sanchez

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

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

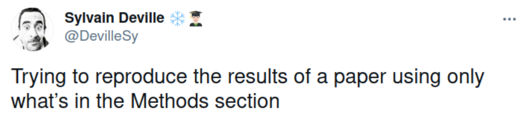
Reproducibility Project: Cancer Biology

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

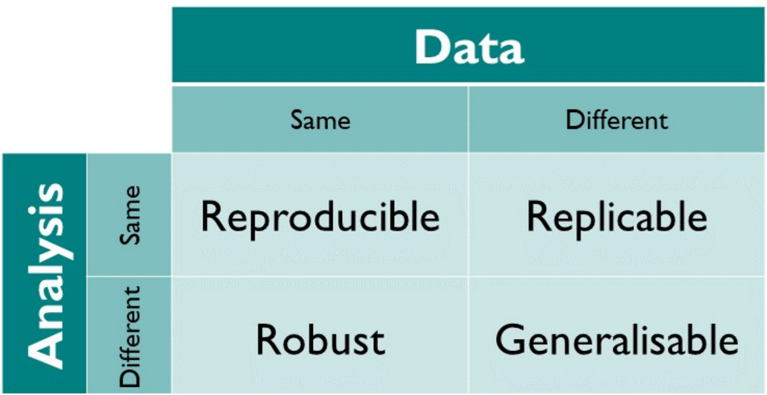
are NOT reproducible

Reproducibility

crisis —> REVOLUTION

What is reproducibility?

Reproducibility vs Replicability



		Data	
		Same	Different
Analysis	Same	Reproducible	Replicable
	Different	Robust	Generalisable

The Turing Way

We can’t guarantee that

our results are REPLICABLE.

But at least

they should be REPRODUCIBLE.

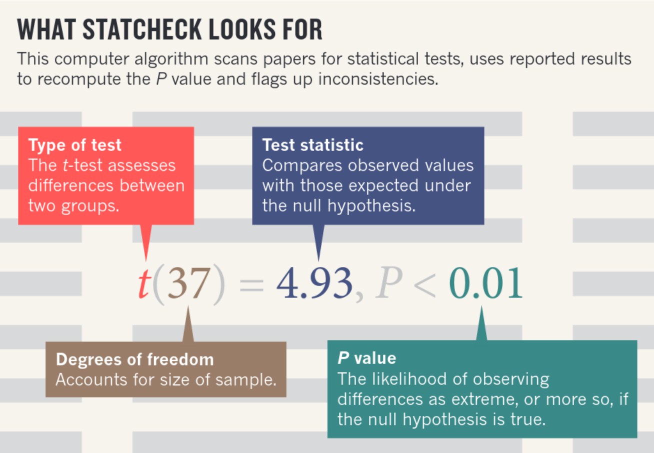
Most scientific articles

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]



WHAT STATCHECK LOOKS FOR

This computer algorithm scans papers for statistical tests, uses reported results to recompute the *P* value and flags up inconsistencies.

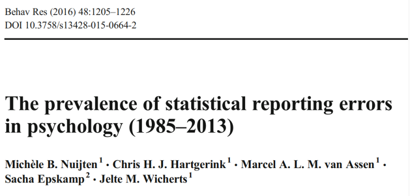
Type of test
The *t*-test assesses differences between two groups.

Test statistic
Compares observed values with those expected under the null hypothesis.

$$t(37) = 4.93, P < 0.01$$

Degrees of freedom
Accounts for size of sample.

***P* value**
The likelihood of observing differences as extreme, or more so, if the null hypothesis is true.



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)

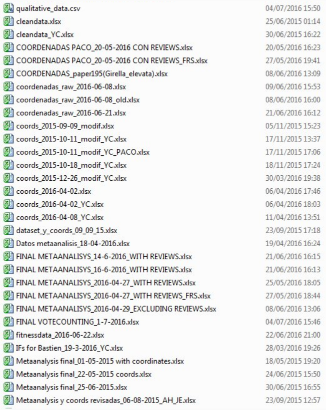
In ecology

< 20% articles are **reproducible**

Culina et al 2020

We can't even reproduce our own work

Data/Code lost or unusable



qualitative_data.csv	04/07/2016 15:50
cleandata.xlsx	25/06/2015 01:14
cleandata_YC.xlsx	30/06/2015 16:22
COORDENADAS PACO_20-05-2016 CON REVIEWS.xlsx	20/05/2016 16:23
COORDENADAS PACO_20-05-2016 CON REVIEWS_FRS.xlsx	27/05/2016 19:41
COORDENADAS_paper195(Girella_elevata).xlsx	08/06/2016 13:09
coordenadas_raw_2016-06-08.xlsx	09/06/2016 15:53
coordenadas_raw_2016-06-08_old.xlsx	08/06/2016 16:00
coordenadas_raw_2016-06-21.xlsx	21/06/2016 16:12
coords_2015-09-09_modif.xlsx	05/11/2015 15:23
coords_2015-10-11_modif_YC.xlsx	17/11/2015 13:37
coords_2015-10-11_modif_YC_PACO.xlsx	17/11/2015 17:06
coords_2015-10-18_modif_YC.xlsx	18/11/2015 17:24
coords_2015-12-26_modif_YC.xlsx	30/03/2016 19:38
coords_2016-04-02.xlsx	06/04/2016 17:46
coords_2016-04-02_YC.xlsx	06/04/2016 18:03
coords_2016-04-08_YC.xlsx	11/04/2016 13:51
dataset_y_coords_09_09_15.xlsx	23/09/2015 17:18
Datos metaanalisis_18-04-2016.xlsx	19/04/2016 16:24
FINAL METAANALISYS_14-6-2016_WITH REVIEWS.xlsx	21/06/2016 16:15
FINAL METAANALISYS_16-6-2016_WITH REVIEWS.xlsx	21/06/2016 16:13
FINAL METAANALISYS_2016-04-27_WITH REVIEWS.xlsx	25/05/2016 18:05
FINAL METAANALISYS_2016-04-27_WITH REVIEWS_FRS.xlsx	27/05/2016 18:44
FINAL METAANALISYS_2016-04-29_EXCLUDING REVIEWS.xlsx	08/06/2016 13:06
FINAL VOTECOUNTING_1-7-2016.xlsx	04/07/2016 15:46
fitnessdata_2016-06-22.xlsx	22/06/2016 21:00
IFs for Bastien_19-3-2016_YC.xlsx	28/03/2016 19:26
Metaanalysis final_01-05-2015 with coordinates.xlsx	18/05/2015 19:20
Metaanalysis final_22-05-2015 coords.xlsx	24/06/2015 15:50
Metaanalysis final_25-06-2015.xlsx	30/06/2015 16:55
Metaanalysis y coords revisadas_06-08-2015_AH_JE.xlsx	23/09/2015 12:57

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**,

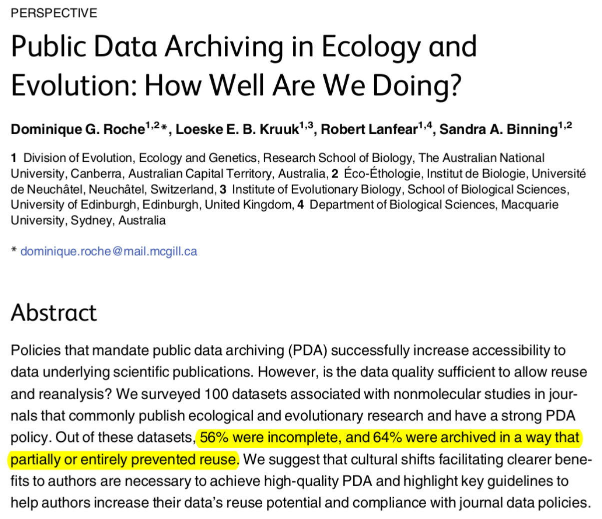
code and data, that produced the result.

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

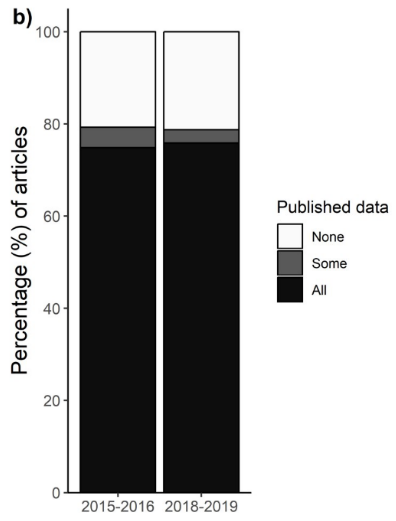
* dominique.roche@mail.mcgill.ca

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

Are we sharing data?

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

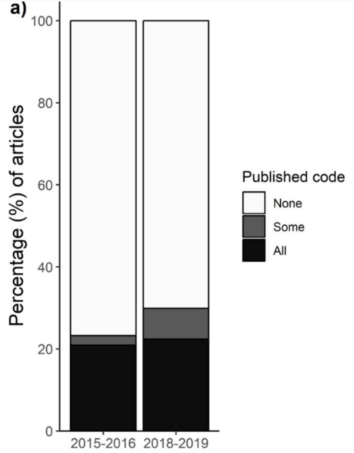
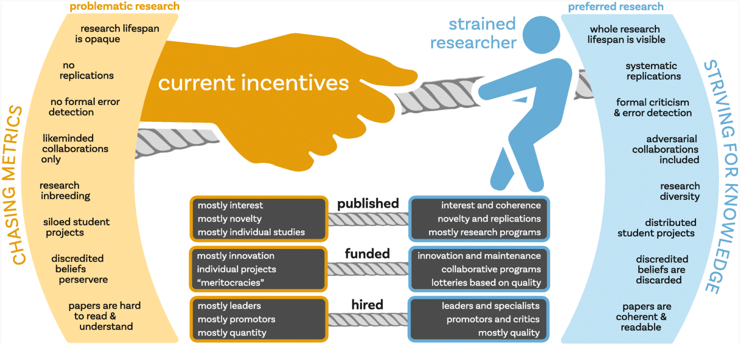


Fig 1. Code-sharing is at its infancy in ecology, wher

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**.

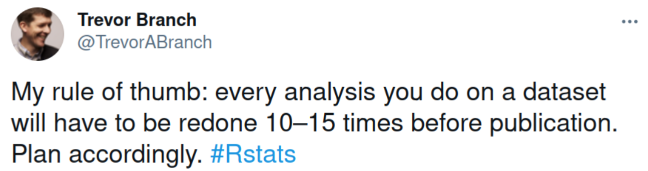
Barnes 2010

- 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

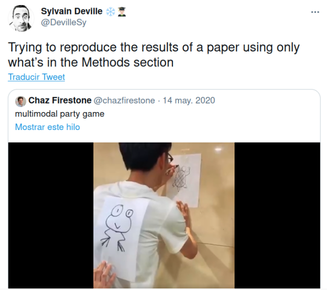
Automation (good code) saves time



Trevor Branch
@TrevorABranch

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

Code = fully traceable, reproducible analysis



Sylvain Deville ❄
@DevilleSy

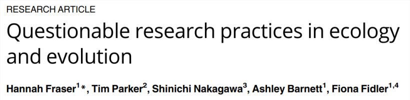
Trying to reproduce the results of a paper using only what's in the Methods section
Traducir Tweet

Chaz Firestone @chazfirestone · 14 may. 2020
multimodal party game
Mostrar este hilo

Code advantages:

- Easier writing
- Easier, deeper review
- Reusable

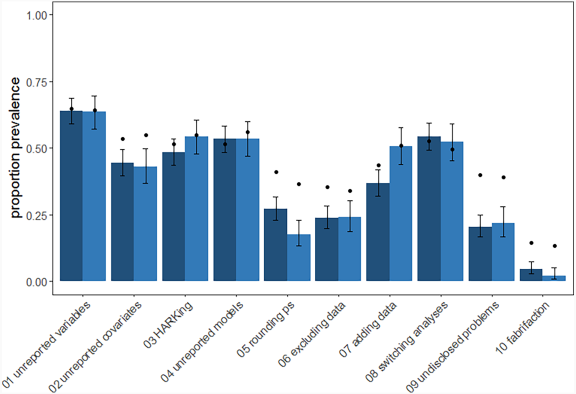
Transparency prevents bad practices



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

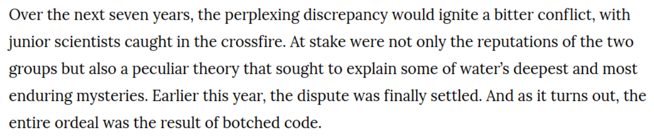
DOI:10.1063/PT.6.1.20180822a

22 Aug 2018 in Research & Technology

The war over supercooled water

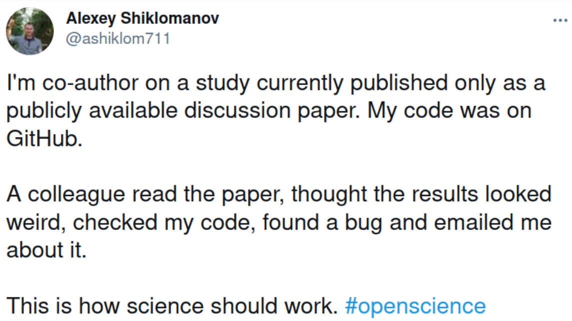
How a hidden coding error fueled a seven-year dispute between two of condensed matter's top theorists.

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



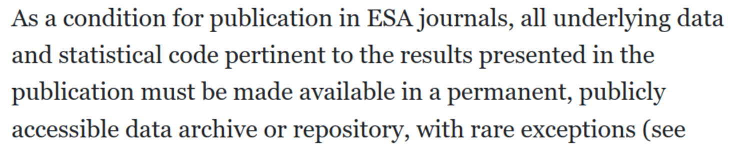
Alexey Shiklomanov
@ashiklom711

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

ECOLOGICAL SOCIETY OF AMERICA
esa

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

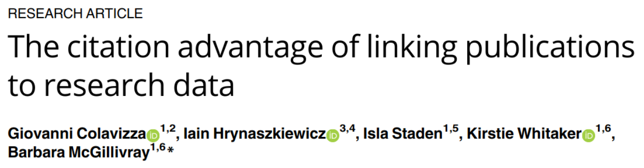
ECOLOGY LETTERS cnrs WILEY

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

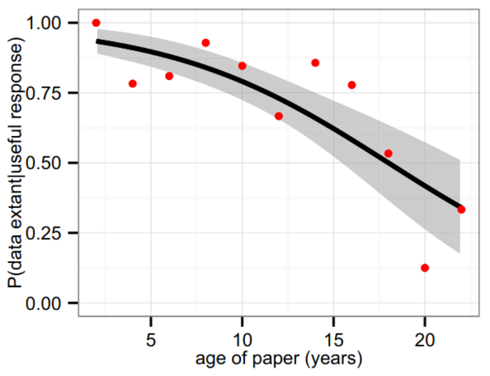
WILEY

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



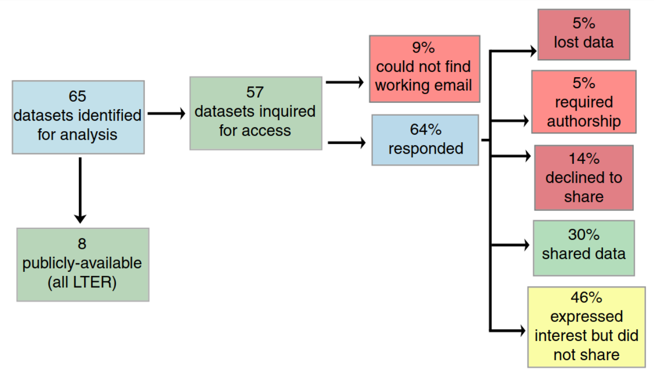
Vines et al 2014

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[†]



65 datasets identified for analysis

8 publicly-available (all LTER)

57 datasets inquired for access

9% could not find working email

64% responded

5% lost data

5% required authorship

14% declined to share

30% shared data

46% expressed interest but did not share

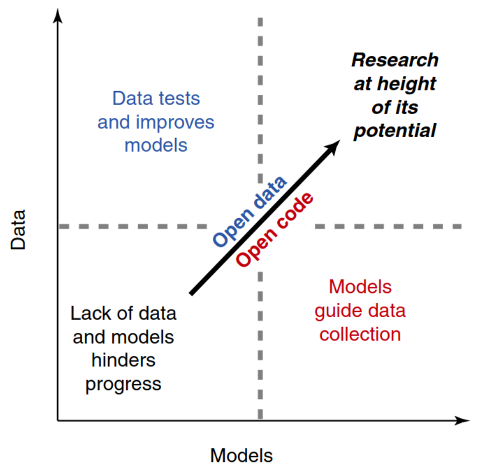
Wolkovich et al 2012

Open data & code enable synthesis

REVIEW

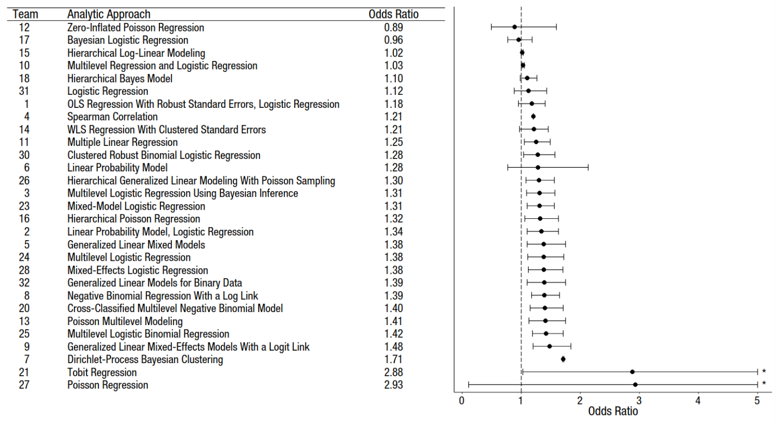
Advances in global change research require open science by individual researchers

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

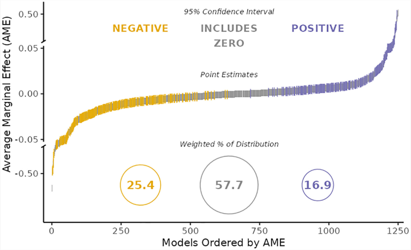
29 teams: 2/3 found significant effect

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



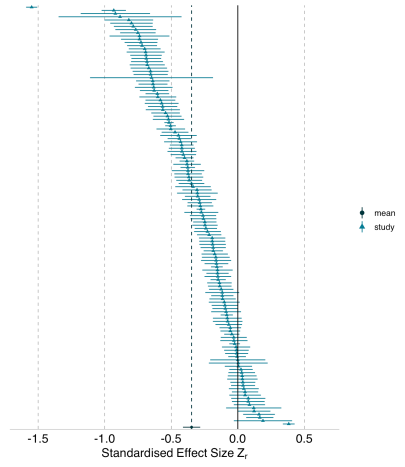
‘This reveals a **universe of uncertainty** that remains hidden when considering a single study in isolation’

‘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

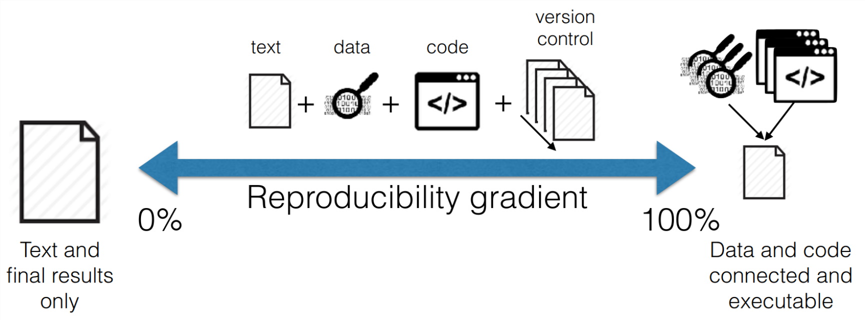
How does sibling competition affect nestling growth in blue tits?



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

```r
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.

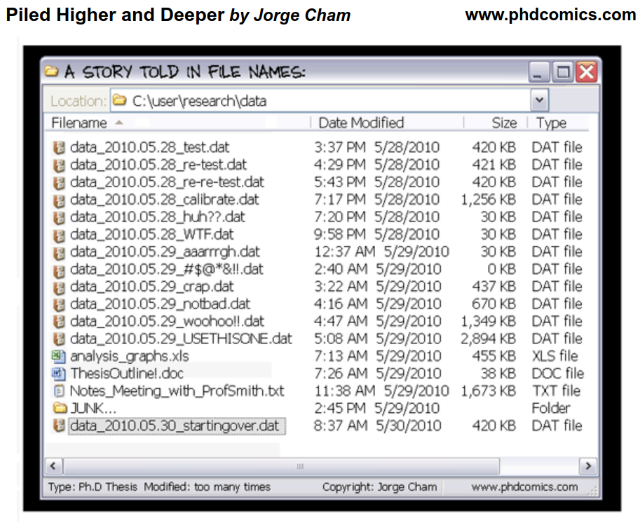
Can use `renv::restore()` to restore packages later or in different computer.

Basic reproducibility

DATA + CODE

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

Version control



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

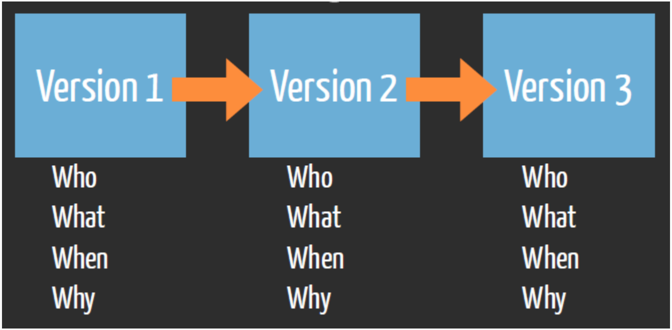
A STORY TOLD IN FILE NAMES:

Location: C:\user\research\data

Filename	Date Modified	Size	Type
data_2010.05.28_test.dat	3:37 PM 5/28/2010	420 KB	DAT file
data_2010.05.28_re-test.dat	4:29 PM 5/28/2010	421 KB	DAT file
data_2010.05.28_re-re-test.dat	5:43 PM 5/28/2010	420 KB	DAT file
data_2010.05.28_calibrate.dat	7:17 PM 5/28/2010	1,256 KB	DAT file
data_2010.05.28_huh??.dat	7:20 PM 5/28/2010	30 KB	DAT file
data_2010.05.28_WTF.dat	9:58 PM 5/28/2010	30 KB	DAT file
data_2010.05.29_aaarrrgh.dat	12:37 AM 5/29/2010	30 KB	DAT file
data_2010.05.29_#$@*&!!.dat	2:40 AM 5/29/2010	0 KB	DAT file
data_2010.05.29_crap.dat	3:22 AM 5/29/2010	437 KB	DAT file
data_2010.05.29_notbad.dat	4:16 AM 5/29/2010	670 KB	DAT file
data_2010.05.29_woohoo!!.dat	4:47 AM 5/29/2010	1,349 KB	DAT file
data_2010.05.29_USETHISONE.dat	5:08 AM 5/29/2010	2,894 KB	DAT file
analysis_graphs.xls	7:13 AM 5/29/2010	455 KB	XLS file
ThesisOutline!.doc	7:26 AM 5/29/2010	38 KB	DOC file
Notes_Meeting_with_ProfSmith.txt	11:38 AM 5/29/2010	1,673 KB	TXT file
JUNK...	2:45 PM 5/29/2010		Folder
data_2010.05.30_startingover.dat	8:37 AM 5/30/2010	420 KB	DAT file

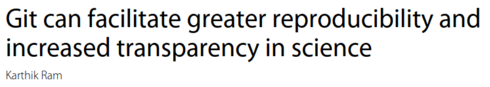
Type: Ph.D Thesis Modified: too many times Copyright: Jorge Cham www.phdcomics.com

Version control with `git`



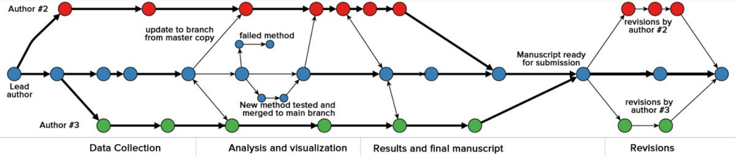
R. Fitzjohn

Much to learn from software engineering



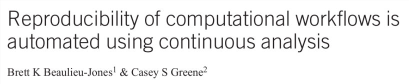
Git can facilitate greater reproducibility and increased transparency in science

Karthik Ram



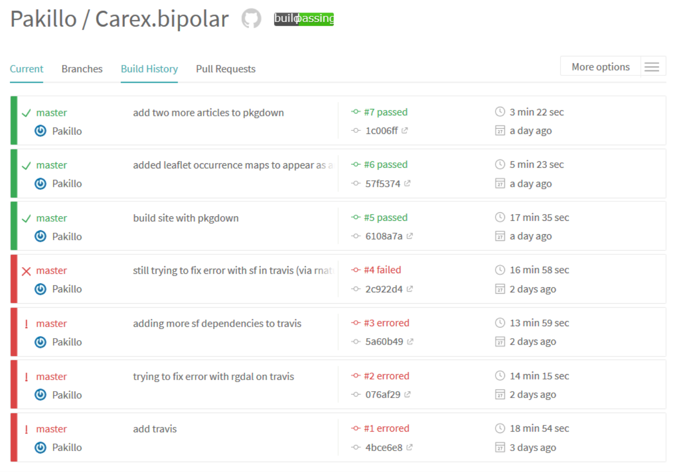
Ram 2013

Automatic checks with Continuous Integration



Reproducibility of computational workflows is automated using continuous analysis

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



Pakillo / Carex.bipolar build passing

✓ master Pakillo	add two more articles to pkgdown	#7 passed 1c006ff	3 min 22 sec a day ago
✓ master Pakillo	added leaflet occurrence maps to appear as a	#6 passed 57f5374	5 min 23 sec a day ago
✓ master Pakillo	build site with pkgdown	#5 passed 6108a7a	17 min 35 sec a day ago
✕ master Pakillo	still trying to fix error with sf in travis (via rnat	#4 failed 2c922d4	16 min 58 sec 2 days ago
! master Pakillo	adding more sf dependencies to travis	#3 errored 5a60b49	13 min 59 sec 2 days ago
! master Pakillo	trying to fix error with rgdal on travis	#2 errored 076af29	14 min 15 sec 2 days ago
! master Pakillo	add travis	#1 errored 4bce6e8	18 min 54 sec 3 days ago

Structuring projects

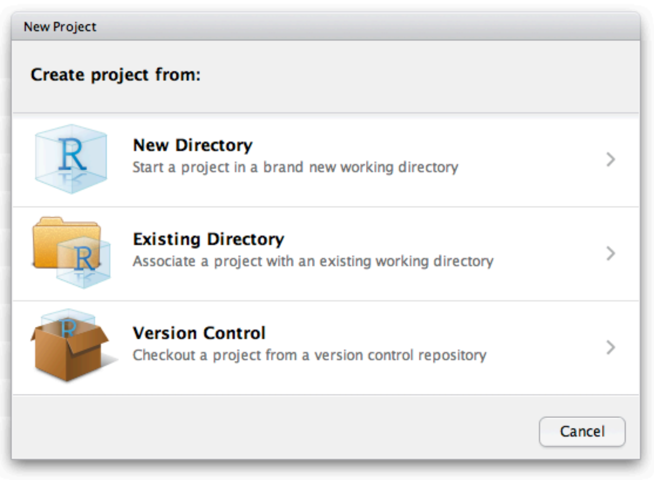
One Project = One Folder

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

Project-Oriented Workflow: advantages

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

Rstudio projects



New Project

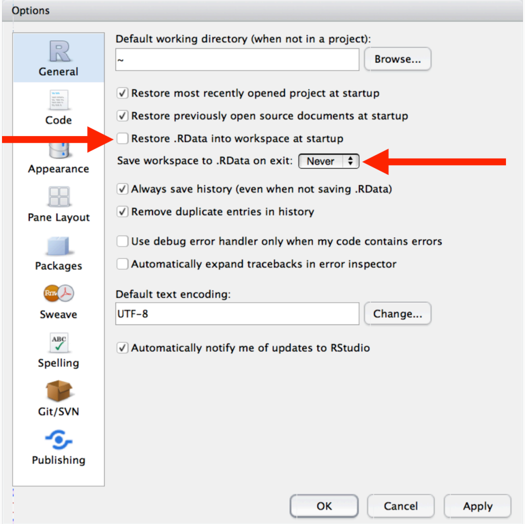
Create project from:

New Directory
Start a project in a brand new working directory

Existing Directory
Associate a project with an existing working directory

Version Control
Checkout a project from a version control repository

Avoid saving workspace



Default working directory (when not in a project): ~

☑ Restore most recently opened project at startup

☑ Restore previously open source documents at startup

☐ Restore .RData into workspace at startup

Save workspace to .RData on exit: Never

☑ Always save history (even when not saving .RData)

☑ Remove duplicate entries in history

☐ Use debug error handler only when my code contains errors

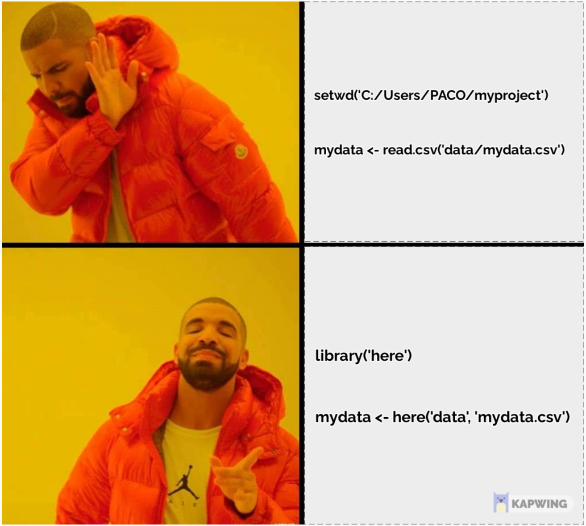
☐ Automatically expand tracebacks in error inspector

Default text encoding: UTF-8

☑ Automatically notify me of updates to RStudio

`https://rstats.wtf`

Use `here` for file paths



```
setwd('C:/Users/PACO/myproject')

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

```
library('here')

mydata <- here('data', 'mydata.csv')
```

`fertile` package: real-time feedback on reproducibility

```r
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

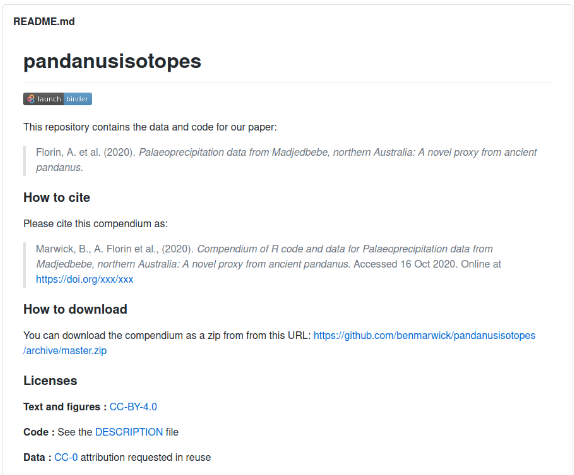
Noble 2009, Rodriguez-Sanchez et al 2016, Wilson et al 2017

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

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`

```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

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

Use meaningful names for objects

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

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

Project templates

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

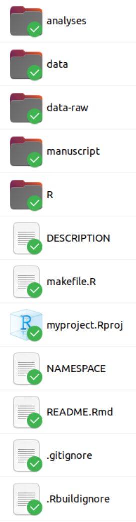
- Standard structure
- Promotes modular code, documented and tested
- Easy to share and run
- Automatic checks (Continuous Integration)
- Automatic code review (`goodpractice`)
- Easily create website with `pkgdown`

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

Creating package structure with `template`

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

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



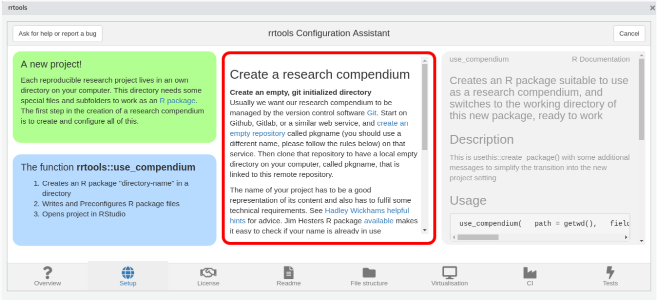
analyses
data
data-raw
manuscript
R
DESCRIPTION
makefile.R
myproject.Rproj
NAMESPACE
README.Rmd
.gitignore
.Rbuildignore

rrtools

`rrtools` creates research compendia

```r
library('rrtools')

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

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



A new project!

Each reproducible research project lives in an own directory on your computer. This directory needs some special files and subfolders to work as an R package. The first step in the creation of a research compendium is to create and configure all of this.

The function **rrtools::use_compendium**

1. Creates an R package "directory-name" in a directory
2. Writes and Preconfigures R package files
3. Opens project in RStudio

Create a research compendium

Create an empty, git initialized directory
Usually we want our research compendium to be managed by the version control software Git. Start on Github, Gitlab, or a similar web service, and create an empty repository called pkgname (you should use a different name, please follow the rules below) on that service. Then clone that repository to have a local empty directory on your computer, called pkgname, that is linked to this remote repository.

The name of your project has to be a good representation of its content and also has to fulfil some technical requirements. See Hadley Wickhams helpful hints for advice. Jim Hesters R package available makes it easy to check if your name is already in use

use_compendium R Documentation

Creates an R package suitable to use as a research compendium, and switches to the working directory of this new package, ready to work

Description

This is usethis::create_package() with some additional messages to simplify the transition into the new project setting

Usage

```
use_compendium(   path = getwd(),   fiel
```

`rrtools`: project structure

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

rcompendium

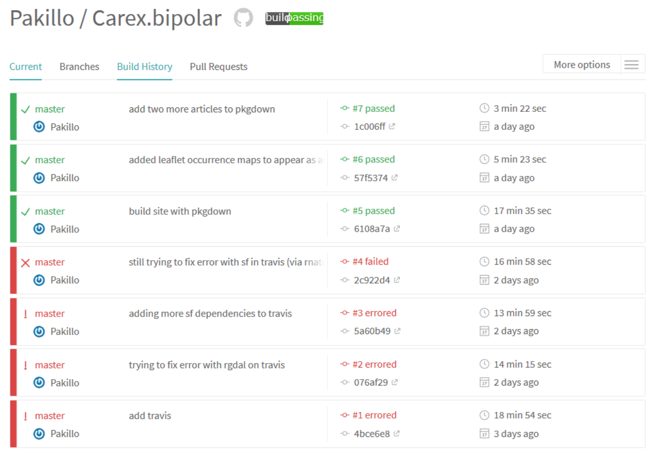
`rcompendium` creates new project with all scaffolding

```r
library('rcompendium')


new_compendium()
```

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

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

Automatic testing with every commit!



Pakillo / Carex.bipolar

✓ master Pakillo	add two more articles to pkgdown	#7 passed 1c006ff	3 min 22 sec a day ago
✓ master Pakillo	added leaflet occurrence maps to appear as a	#6 passed 57f5374	5 min 23 sec a day ago
✓ master Pakillo	build site with pkgdown	#5 passed 6108a7a	17 min 35 sec a day ago
✕ master Pakillo	still trying to fix error with sf in travis (via rnat	#4 failed 2c922d4	16 min 58 sec 2 days ago
! master Pakillo	adding more sf dependencies to travis	#3 errored 5a60b49	13 min 59 sec 2 days ago
! master Pakillo	trying to fix error with rgdal on travis	#2 errored 076af29	14 min 15 sec 2 days ago
! master Pakillo	add travis	#1 errored 4bce6e8	18 min 54 sec 3 days ago

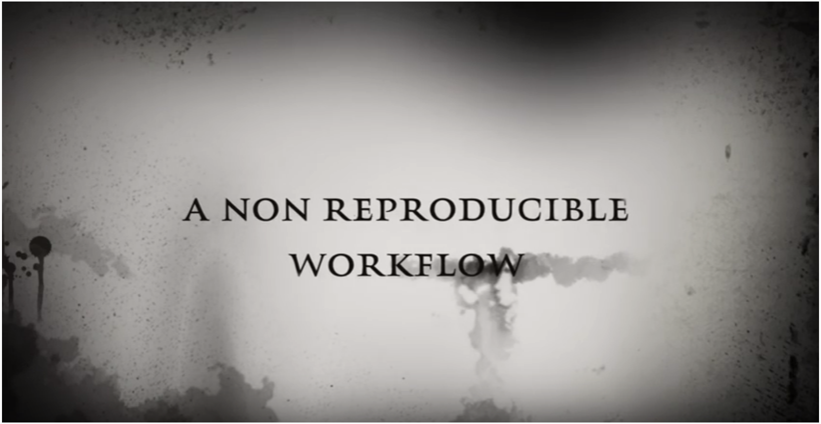
Minimalistic compendium

https://github.com/cboettig/compendium

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

Reproducible dynamic documents with Rmarkdown

A scary movie... with happy ending

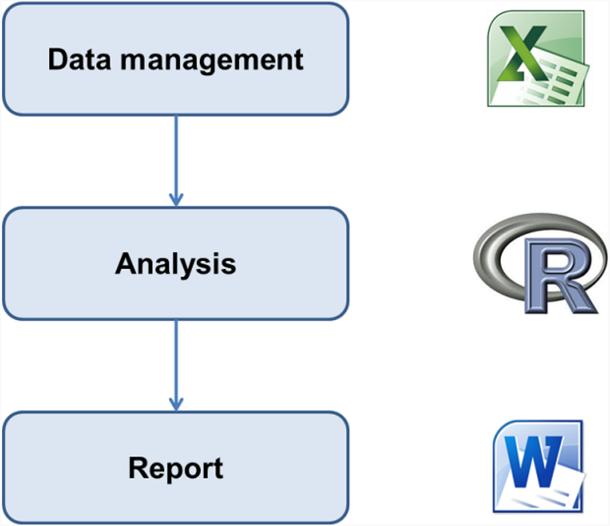


https://youtu.be/s3JldKoA0zw

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



Data management

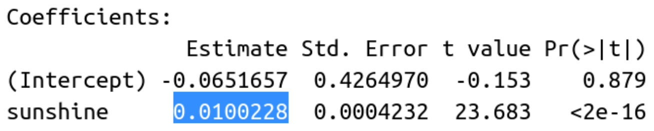
Analysis

Report

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!

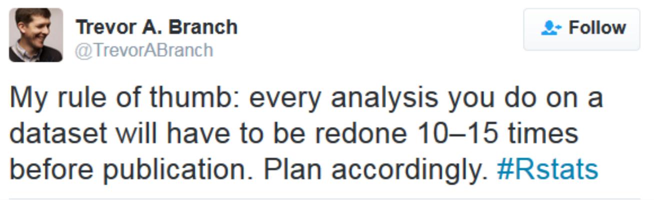
Manual copy-paste is tedious & problematic



```
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

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)

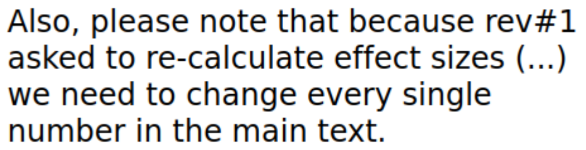
Even **you** will struggle to reproduce

your own results from a few weeks/months ago.

Writing reproducible manuscripts is hard

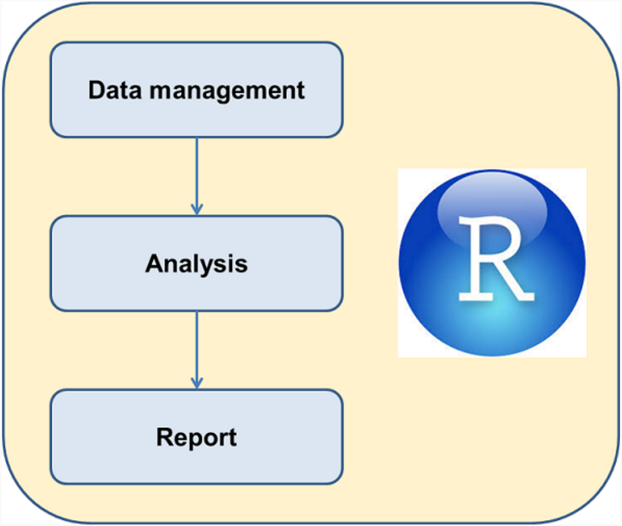
Revising non-reproducible manuscripts is even harder

.



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



Data management

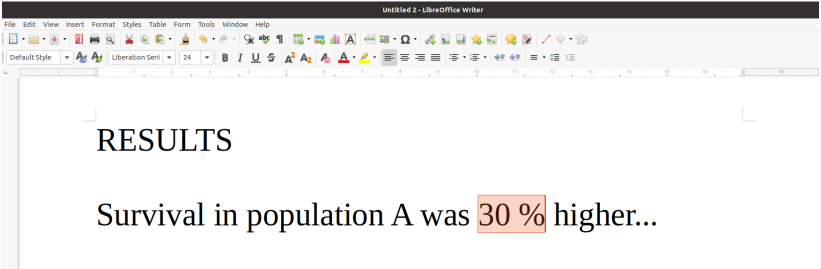
Analysis

Report

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?



RESULTS

Survival in population A was 30 % higher...

Dynamic documents with Rmarkdown

Rmarkdown:

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

Output:

> Survival in population A was 30 % higher

Dynamic documents with Rmarkdown

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

Rmarkdown:

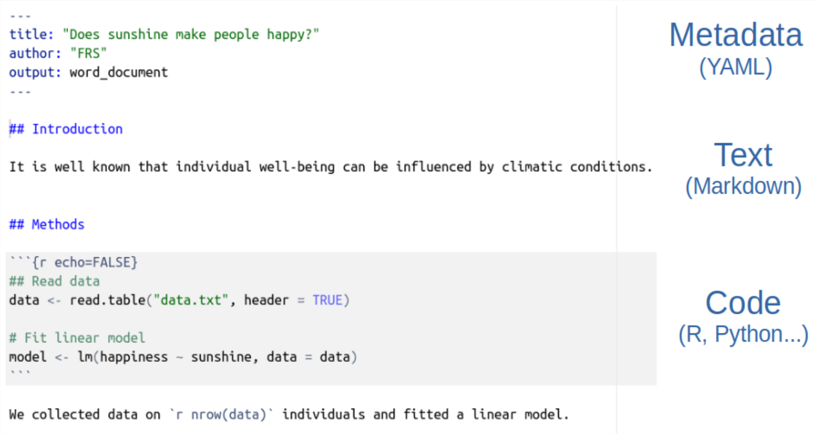
> We measured `` `r nrow(mydata)` `` individuals

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.
~~~

Code chunk options

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

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

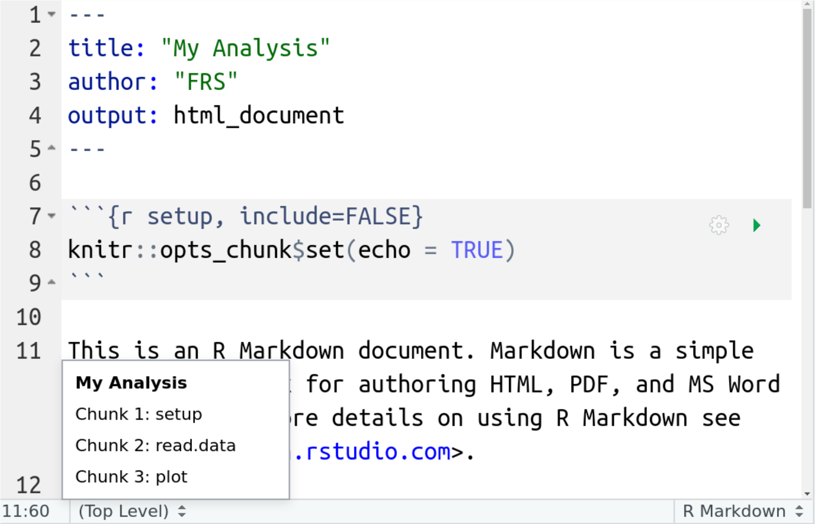
Code chunk options

```
‘‘‘{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
```

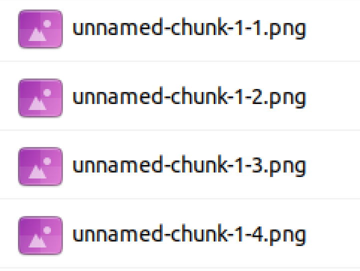
Naming chunks helps navigating long docs



~~~
---
title: "My Analysis"
author: "FRS"
output: html_document
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
~~~

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>.
```
```

Naming chunks: figure files take chunk name



unnamed-chunk-1-1.png
unnamed-chunk-1-2.png
unnamed-chunk-1-3.png
unnamed-chunk-1-4.png

¡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

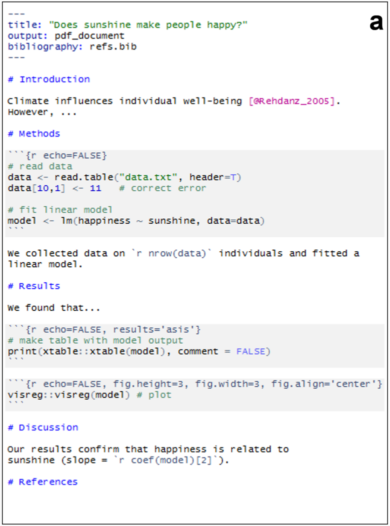
```
# Header

## 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



~~~
---
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

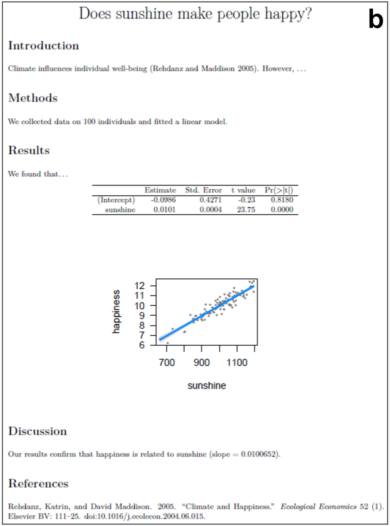
We found that...

```{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

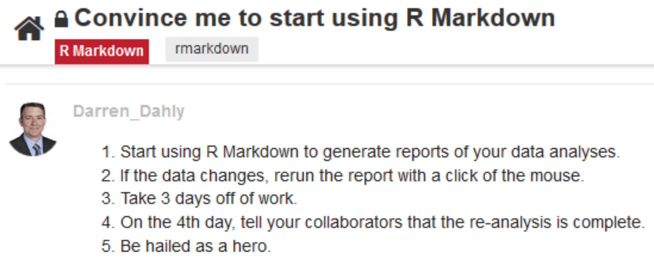
Our results confirm that happiness is related to
sunshine (slope = `r coef(model)[2]`).

References
```



## 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

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)

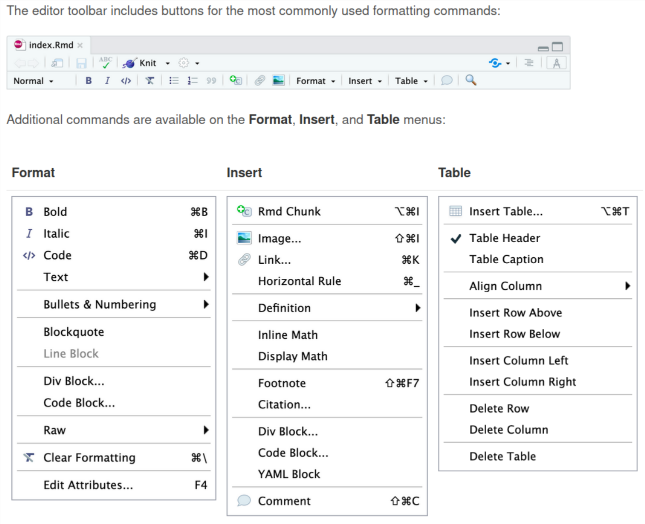
HTML/Word/PDF output

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:

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!

```r
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

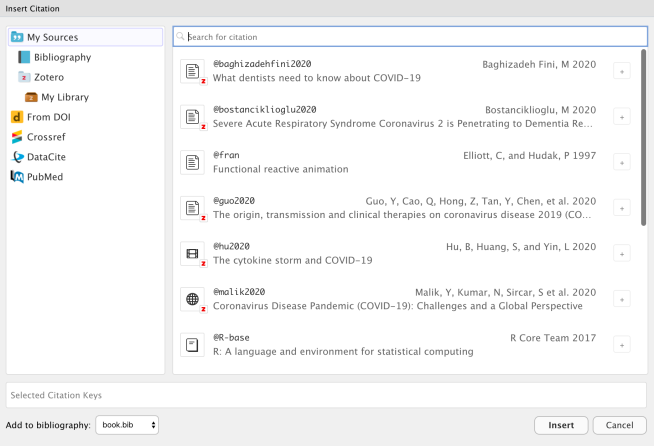
Using LaTeX:

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

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

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

Citing bibliography



@baghizadehfini2020 Baghizadeh Fini, M 2020
What dentists need to know about COVID-19

@bostancioglu2020 Bostancioglu, M 2020
Severe Acute Respiratory Syndrome Coronavirus 2 is Penetrating to Dementia Re...

@fran Elliott, C, and Hudak, P 1997
Functional reactive animation

@guo2020 Guo, Y, Cao, Q, Hong, Z, Tan, Y, Chen, et al. 2020
The origin, transmission and clinical therapies on coronavirus disease 2019 (CO...

@hu2020 Hu, B, Huang, S, and Yin, L 2020
The cytokine storm and COVID-19

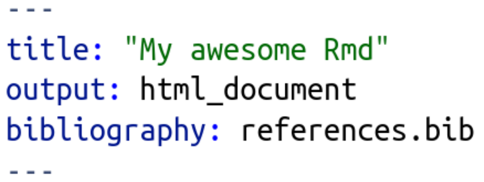
@malik2020 Malik, Y, Kumar, N, Sircar, S et al. 2020
Coronavirus Disease Pandemic (COVID-19): Challenges and a Global Perspective

@R-base R Core Team 2017
R: A language and environment for statistical computing

Add to bibliography: book.bib

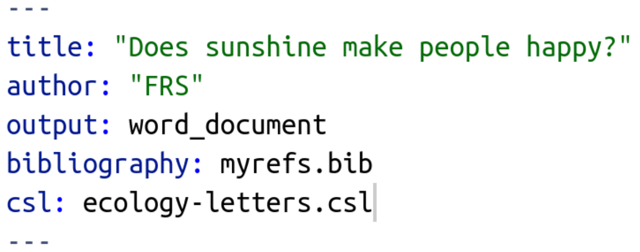
https://rstudio.github.io/visual-markdown-editing/#/citations

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

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

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.

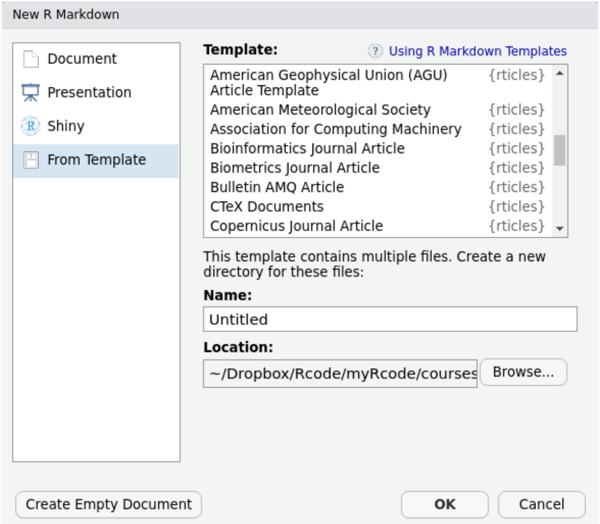
Please provide details of author contributions here.

Please declare any conflict of interest here.

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

PNAS | September 19, 2018 | vol. XXX | no. XX | 1–2

Accessing Rmd templates



Template:

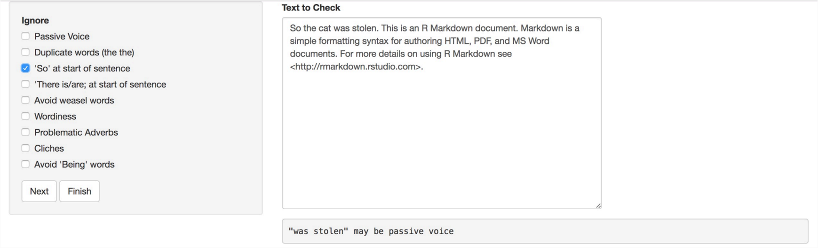
American Geophysical Union (AGU) Article Template	{rticles}
American Meteorological Society	{rticles}
Association for Computing Machinery	{rticles}
Bioinformatics Journal Article	{rticles}
Biometrics Journal Article	{rticles}
Bulletin AMQ Article	{rticles}
CTeX Documents	{rticles}
Copernicus Journal Article	{rticles}

This template contains multiple files. Create a new directory for these files:

Name: Untitled

Location: ~/Dropbox/Rcode/myRcode/courses

Revise writing style: gramr



Ignore

- ☐ Passive Voice
- ☐ Duplicate words (the the)
- ☑ 'So' at start of sentence
- ☐ 'There is/are; at start of sentence
- ☐ Avoid weasel words
- ☐ Wordiness
- ☐ Problematic Adverbs
- ☐ Cliches
- ☐ Avoid 'Being' words

Text to Check

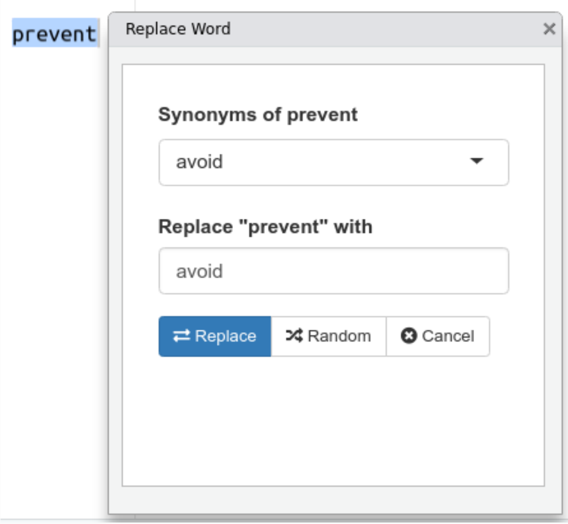
So the cat was stolen. This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

```
"was stolen" may be passive voice
```

`https://github.com/ropenscilabs/gramr`

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

Find synonyms



prevent

Replace Word

Synonyms of prevent

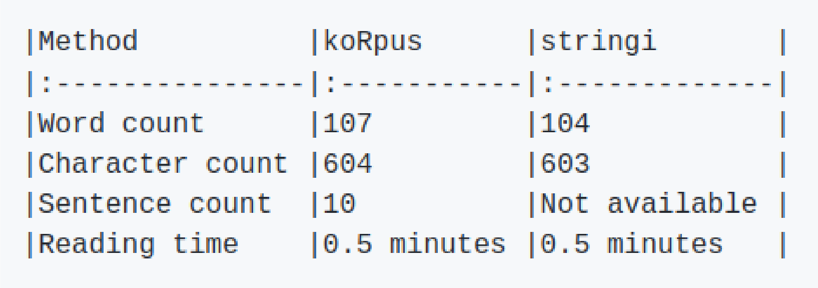
avoid

Replace "prevent" with

avoid

https://github.com/gadenbuie/synamyn

Word count and readability

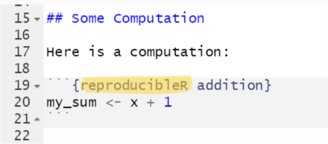


```
|Method          |koRpus      |stringi       |
|:---------------|:-----------|:-------------|
|Word count      |107         |104           |
|Character count |604         |603           |
|Sentence count  |10          |Not available |
|Reading time    |0.5 minutes |0.5 minutes   |
```

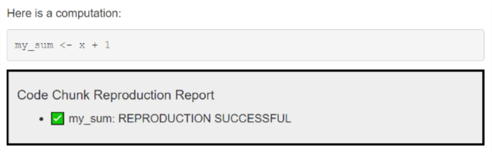
https://github.com/benmarwick/wordcountaddin

Automated reproducibility checks

https://github.com/brandmaier/reproducibleRchunks



~~~
## Some Computation

Here is a computation:

```{reproducibleR addition}
my_sum <- x + 1
```

~~~

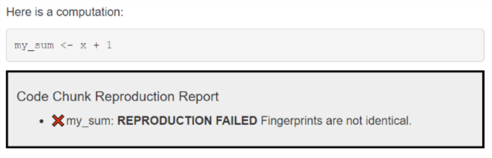


Here is a computation:

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

Code Chunk Reproduction Report

- ✅ my_sum: REPRODUCTION SUCCESSFUL



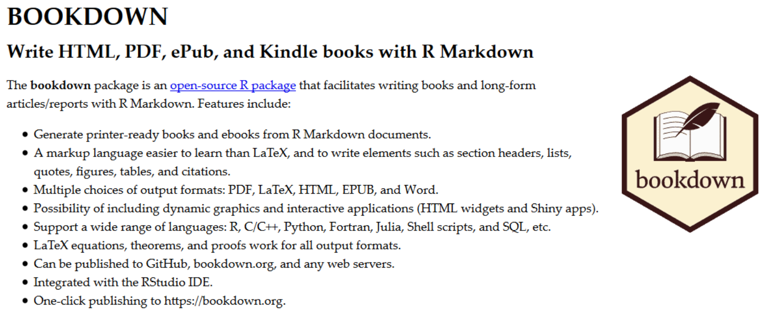
Here is a computation:

```
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/`

Slide presentations with `xaringan`

Presentation Ninja

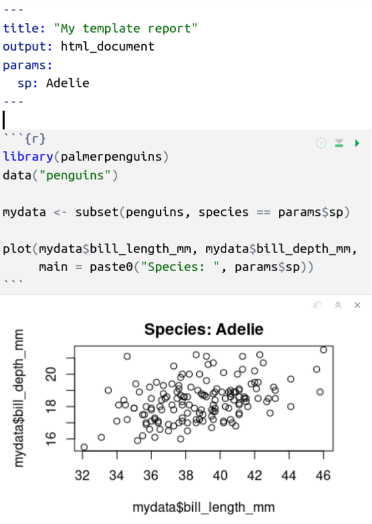
with xaringan

Yihui Xie

RStudio, PBC

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))
```
~~~

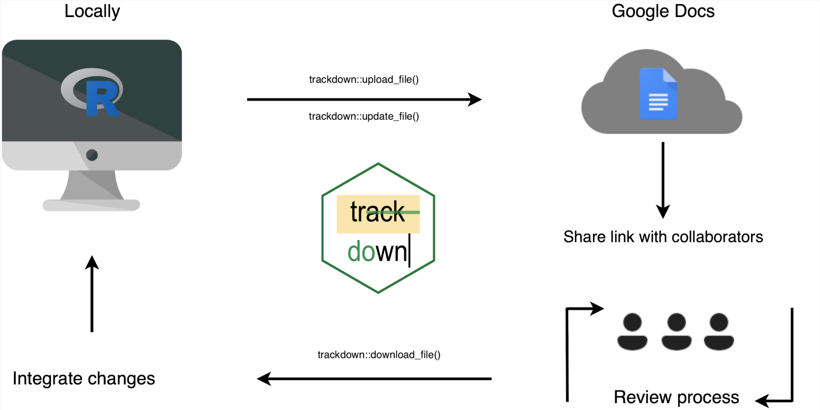
https://bookdown.org/yihui/rmarkdown/parameterized-reports.html

Render thousands of individual reports from Rmd template

```r
library('rmarkdown')

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

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

}
```

Collaborative writing

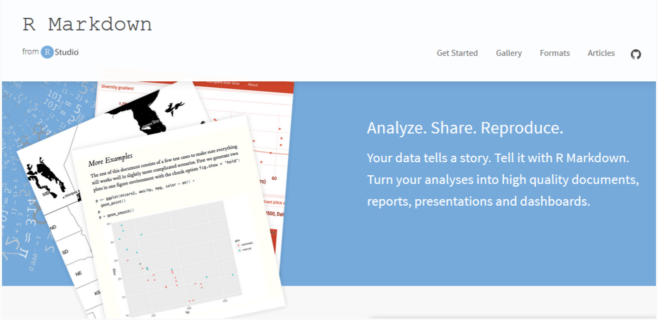
- GitHub, GitLab, etc
- Google Docs (trackdown)
- redoc



Rmarkdown resources

Rmarkdown website

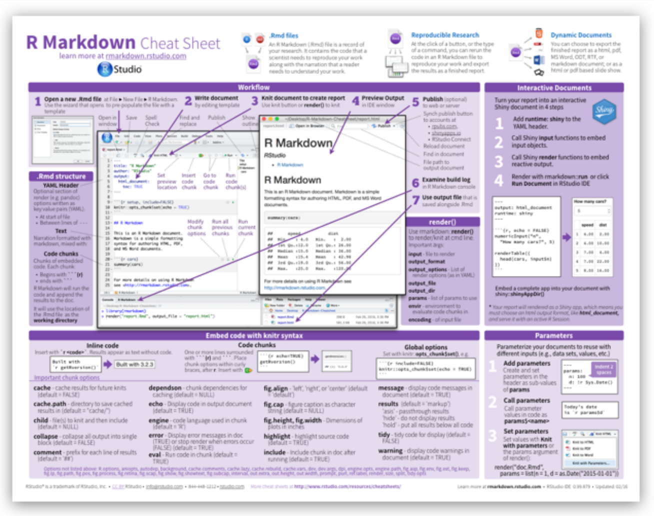
http://rmarkdown.rstudio.com/



Analyze. Share. Reproduce.

Your data tells a story. Tell it with R Markdown. Turn your analyses into high quality documents, reports, presentations and dashboards.

Rmarkdown cheat sheet



R Markdown Cheat Sheet
learn more at rmarkdown.rstudio.com
RStudio

.Rmd files
An R Markdown (.Rmd) file is a record of your research. It contains the code that a scientist needs to reproduce your work along with the narration that a reader needs to understand your work.

Reproducible Research
At the click of a button, or the type of a command, you can rerun the code in an R Markdown file to reproduce your work and export the results as a finished report.

Dynamic Documents
You can choose to export the finished report as a html, pdf, MS Word, ODT, RTF, or markdown document; or as a html or pdf based slide show.

Workflow

1 **Open a new .Rmd file** at File ▸ New File ▸ R Markdown. Use the wizard that opens to pre-populate the file with a template

2 **Write document** by editing template

3 **Knit document to create report** Use knit button or **render()** to knit

4 **Preview Output** in IDE window

5 **Publish** (optional) to web or server

~~~
---
title: "R Markdown"
author: "RStudio"
output:
  html_document:
    toc: TRUE
---

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
~~~

R Markdown

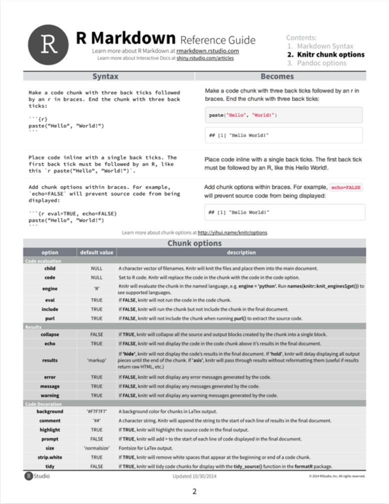
This is an R Markdown document.
Markdown is a simple formatting
syntax for authoring HTML, PDF,
and MS Word documents.

```{r cars}
summary(cars)
```

For more details on using R Markdown
see <http://rmarkdown.rstudio.com>.
```

```
> library(rmarkdown)
> render("report.Rmd", output_file = "report.html")
```

R Markdown

RStudio

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.

For more details on using R Markdown see http://rmarkdown.rstudio.com.

- Synch publish button to accounts at
  - rpubs.com,
  - shinyapps.io
  - RStudio Connect
- Reload document
- Find in document
- File path to output document

6 **Examine build log** in R Markdown console

7 **Use output file** that is saved alongside .Rmd

.Rmd structure

YAML Header
Optional section of render (e.g. pandoc) options written as key:value pairs (YAML).
- At start of file
- Between lines of - - -

Text
Narration formatted with markdown, mixed with:

Code chunks
Chunks of embedded code. Each chunk:
- Begins with ```{r}
- ends with ```

R Markdown will run the code and append the results to the doc.

It will use the location of the .Rmd file as the **working directory**

render()
Use rmarkdown::**render()** to render/knit at cmd line. Important args:
- **input** - file to render
- **output_format**
- **output_options** - List of render options (as in YAML)
- **output_file**
- **output_dir**
- **params** - list of params to use
- **envir** - environment to evaluate code chunks in
- **encoding** - of input file

Interactive Documents

Turn your report into an interactive Shiny document in 4 steps

1 Add **runtime: shiny** to the YAML header.

2 Call Shiny **input** functions to embed input objects.

3 Call Shiny **render** functions to embed reactive output.

4 Render with rmarkdown::**run** or click **Run Document** in RStudio IDE

```
---
output: html_document
runtime: shiny
---

```{r, echo = FALSE}
numericInput("n",
  "How many cars?", 5)

renderTable({
  head(cars, input$n)
})
```
```

How many cars?

	speed	dist
1	4.00	2.00
2	4.00	10.00
3	7.00	4.00
4	7.00	22.00
5	8.00	16.00

Embed a complete app into your document with shiny::**shinyAppDir()**

* *Your report will rendered as a Shiny app, which means you must choose an html output format, like* ***html_document****, and serve it with an active R Session.*

Embed code with knitr syntax

Inline code
Insert with `r <code>`. Results appear as text without code.

```
Built with
`r getRversion()`
```

Built with 3.2.3

Code chunks
One or more lines surrounded with ```{r} and ```. Place chunk options within curly braces, after **r**. Insert with

```
```{r echo=TRUE}
getRversion()
```
```

Global options
Set with knitr::**opts_chunk$set()**, e.g.

```
```{r include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```

Important chunk options

- **cache** - cache results for future knits (default = FALSE)
- **cache.path** - directory to save cached results in (default = "cache/")
- **child** - file(s) to knit and then include (default = NULL)
- **collapse** - collapse all output into single block (default = FALSE)
- **comment** - prefix for each line of results (default = '##')
- **dependson** - chunk dependencies for caching (default = NULL)
- **echo** - Display code in output document (default = TRUE)
- **engine** - code language used in chunk (default = 'R')
- **error** - Display error messages in doc (TRUE) or stop render when errors occur (FALSE) (default = TRUE)
- **eval** - Run code in chunk (default = TRUE)
- **fig.align** - 'left', 'right', or 'center' (default = 'default')
- **fig.cap** - figure caption as character string (default = NULL)
- **fig.height, fig.width** - Dimensions of plots in inches
- **highlight** - highlight source code (default = TRUE)
- **include** - Include chunk in doc after running (default = TRUE)
- **message** - display code messages in document (default = TRUE)
- **results** (default = 'markup')
  - 'asis' - passthrough results
  - 'hide' - do not display results
  - 'hold' - put all results below all code
- **tidy** - tidy code for display (default = FALSE)
- **warning** - display code warnings in document (default = TRUE)

Options not listed above: R.options, aniopts, autodep, background, cache.comments, cache.lazy, cache.rebuild, cache.vars, dev, dev.args, dpi, engine.opts, engine.path, fig.asp, fig.env, fig.ext, fig.keep, fig.lp, fig.path, fig.pos, fig.process, fig.retina, fig.scap, fig.show, fig.showtext, fig.subcap, interval, out.extra, out.height, out.width, prompt, purl, ref.label, render, size, split, tidy.opts

Parameters

Parameterize your documents to reuse with different inputs (e.g., data sets, values, etc.)

1 **Add parameters**
Create and set parameters in the header as sub-values of **params**

```
---
params:
  n: 100
  d: !r Sys.Date()
---
```


Indent 2 spaces

2 **Call parameters**
Call parameter values in code as **params$<name>**

```
Today's date
is `r params$d`
```

3 **Set parameters**
Set values with **Knit with parameters** or the params argument of render():

Knit to HTML
Knit to PDF
Knit to Word
Knit with Parameters...


```
render("doc.Rmd",
  params = list(n = 1, d = as.Date("2015-01-01"))
```

RStudio® is a trademark of RStudio, Inc. • CC BY RStudio • info@rstudio.com • 844-448-1212 • rstudio.com   More cheat sheets at http://www.rstudio.com/resources/cheatsheets/   Learn more at **rmarkdown.rstudio.com** • RStudio IDE 0.99.879 • Updated: 02/16

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

# Rmarkdown reference guide



R

**R Markdown** Reference Guide

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

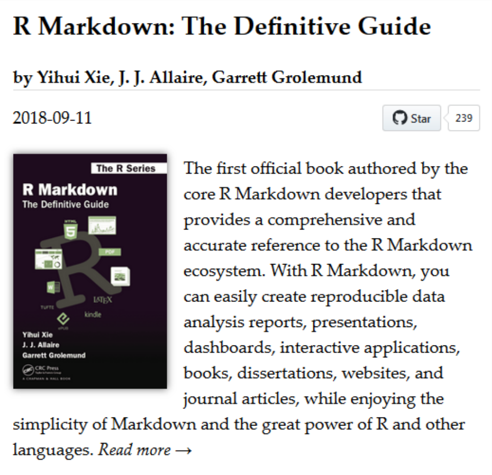
Contents:
1. Markdown Syntax
2. **Knitr chunk options**
3. Pandoc options

**Syntax** | **Becomes**

~~~
Make a code chunk with three back ticks followed
by an r in braces. End the chunk with three back
ticks:

```{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:

```
paste("Hello", "World!")
```

```
## [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:

```{r eval=TRUE, echo=FALSE}
paste("Hello", "World!")
```
```

Add chunk options within braces. For example, `echo=FALSE` will prevent source code from being displayed:

```
## [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

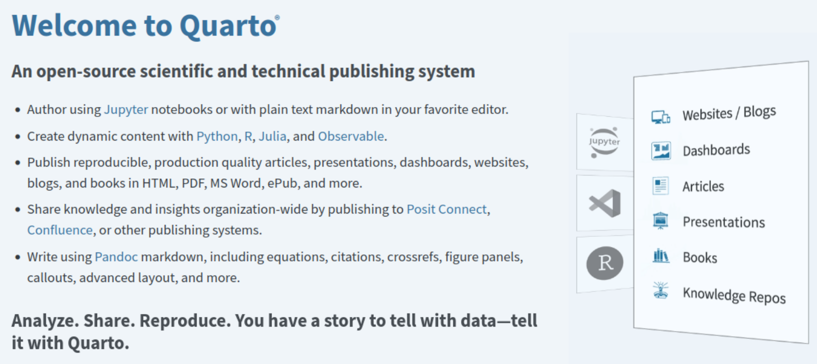
The first official book authored by the core R Markdown developers that provides a comprehensive and accurate reference to the R Markdown ecosystem. With R Markdown, you can easily create reproducible data analysis reports, presentations, dashboards, interactive applications, books, dissertations, websites, 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

Quarto: 2nd generation Rmarkdown



Welcome to Quarto®

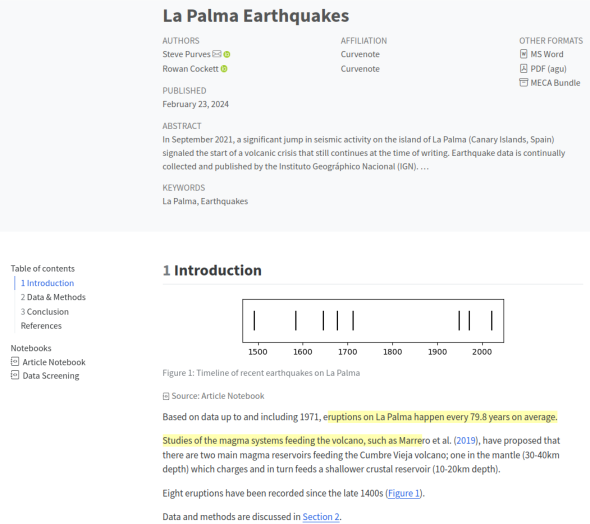
An open-source scientific and technical publishing system

- 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`

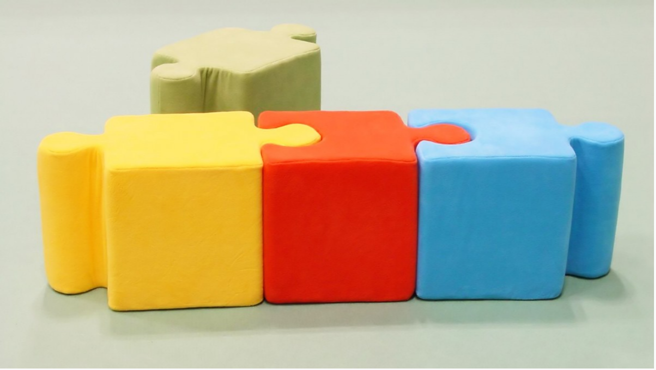
Your turn

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**

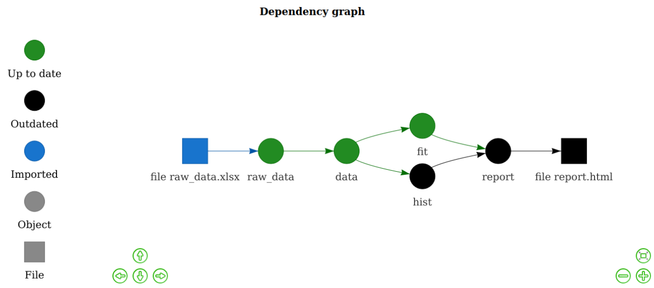


`makefile` runs all code in right order

makefile.R

```r
source('clean_data.R')

source('fit_model.R')

render('report.Rmd')
```

`targets`: advanced workflow management

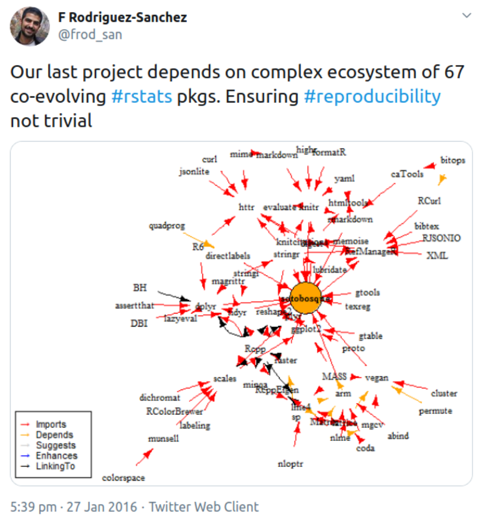


https://docs.ropensci.org/targets/

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

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

```r
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

```r
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()
```

`groundhog` loads packages as available on given date

`https://groundhogr.com/`

```
library('groundhog')

groundhog.library(pkg = c('dplyr', 'ggplot2'),
                  date = ”2024-09-01”)
```

`renv`: recommended way to control dependencies

```r
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

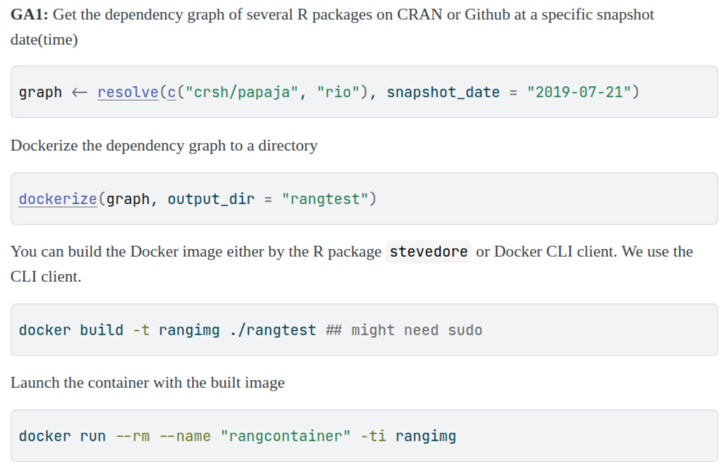
computational environment

`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")
```

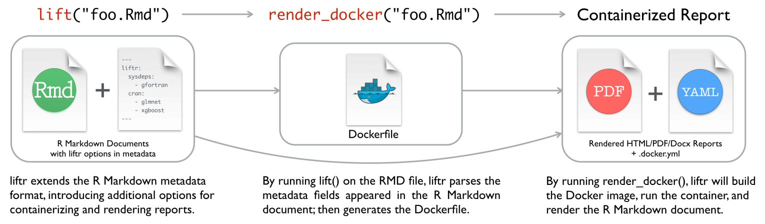
You can build the Docker image either by the R package `stevedore` or Docker CLI client. We use the CLI client.

```
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



liftr extends the R Markdown metadata format, introducing additional options for containerizing and rendering reports.

By running lift() on the RMD file, liftr parses the metadata fields appeared in the R Markdown document; then generates the Dockerfile.

By running render_docker(), liftr will build the Docker image, run the container, and render the R Markdown document.

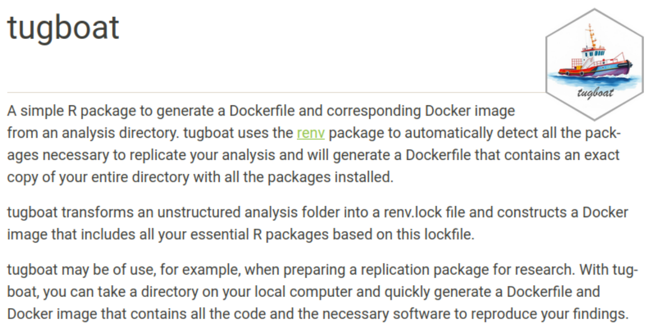
https://liftr.me/

`containerit` creates Dockerfile

```r
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.

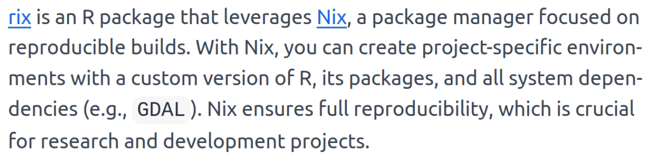
tugboat may be of use, for example, when preparing a replication package for research. With tugboat, you can take a directory on your local computer and quickly generate a Dockerfile and Docker image that contains all the code and the necessary software to reproduce your findings.

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

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

`rix`: reproducible environments with Nix

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

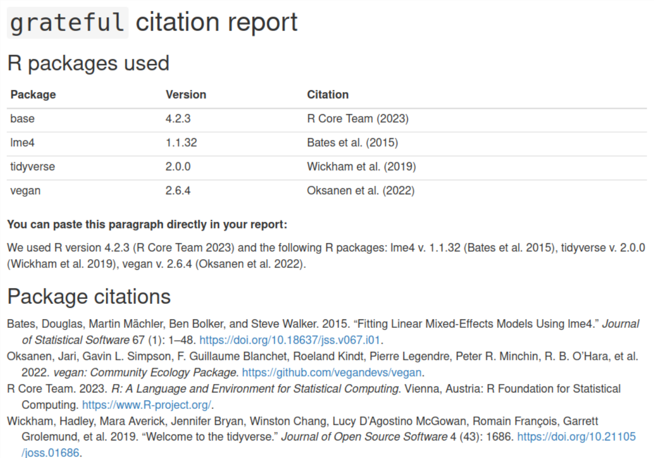


rix is an R package that leverages Nix, a package manager focused on reproducible builds. With Nix, you can create project-specific environments with a custom version of R, its packages, and all system dependencies (e.g., `GDAL`). Nix ensures full reproducibility, which is crucial for research and development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

Remember to cite software used!

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

```r
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

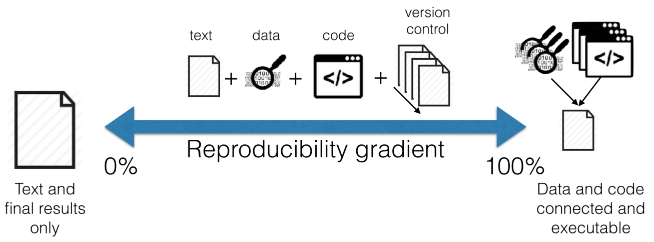
- Create script/Rmd using different packages
- Call `checkpoint` on former date
- Record dependencies:
    - `renv::snapshot`
- Recreate packages
    - `restore()`

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!