CODECHECK

An open-science initiative to facilitate sharing of computer programs and results presented in scientific publications

Stephen J Eglen, Cambridge Computational Biology Institute, University of Cambridge

Daniel Nüst, Institute for Geoinformatics, University of Münster

https://codecheck.org.uk/

Springer Nature / BMC, September 14, 2020

Challenge

> *The problem is that most modern science is so complicated, and most journal articles so brief, it's impossible for the article to include details of many important methods and decisions made by the researcher as he analyzed his data on his computer.*

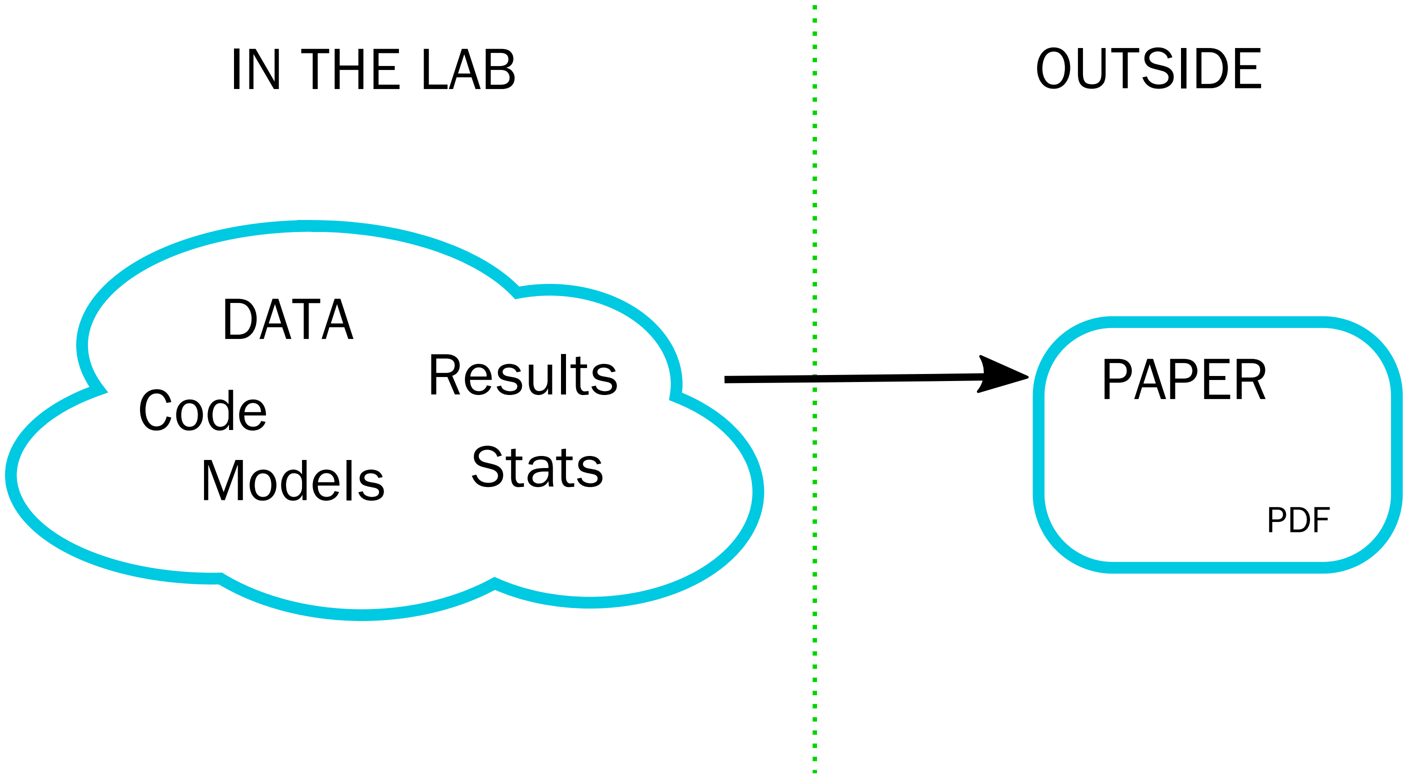
Ben Marwick: How computers broke science – and what we can do to fix it

---

"Claerbout's claim" in Donoho (2010), An invitation to reproducible computational research

Claerbout & Karrenbach (1992), Electronic documents give reproducible research a new meaning

Premise



We should be sharing material on the left, not the right; *"Paper as advert for Scholarship"* Buckheit & Donoho (1995)

Traditional and modern research(ers)

FROM

T

- broad knowledge: cross-discipline, collaboration
- deep knowledge: domain speciality (expertise and skills)

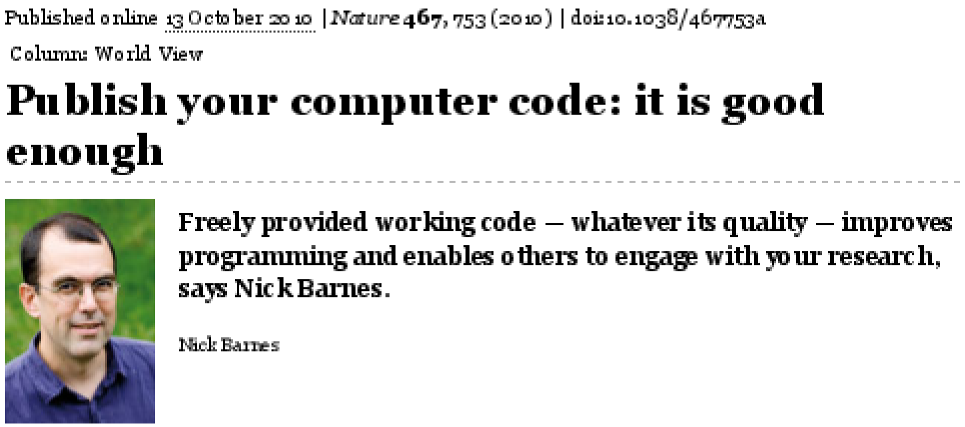
TO

Π

- broad knowledge: cross-discipline, collaboration
- deep knowledge
  - domain speciality (expertise and skills)
  - stats/computing/reproducibility

Sources: T-shaped skills; Hacking Academia: Data Science and the University; When All Science Becomes Data Science; Community-level data science Public domain images: Wikipedia, CDC, Wikipedia, kallerma

Approaches to code sharing



Published online 13 October 2010 | *Nature* **467**, 753 (2010) | doi:10.1038/467753a

Column: World View

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.

Nick Barnes

- Informal 'code buddy' system
- Community-led *research compendia*
- Online workspaces/infrastructure (e.g., Code Ocean Nature trial, see also Konkol et al., 2020, for a review)
- Certify reproducibility with confidential data (CASCAD) (Pérignon et al., 2019)

CODE
CHECK ✓

https://codecheck.org.uk/

Independent execution of computations underlying research articles.

The four CODECHECK principles

1. Codecheckers record but don’t investigate or fix.

2. Communication between humans is key.

3. Credit is given to codecheckers.

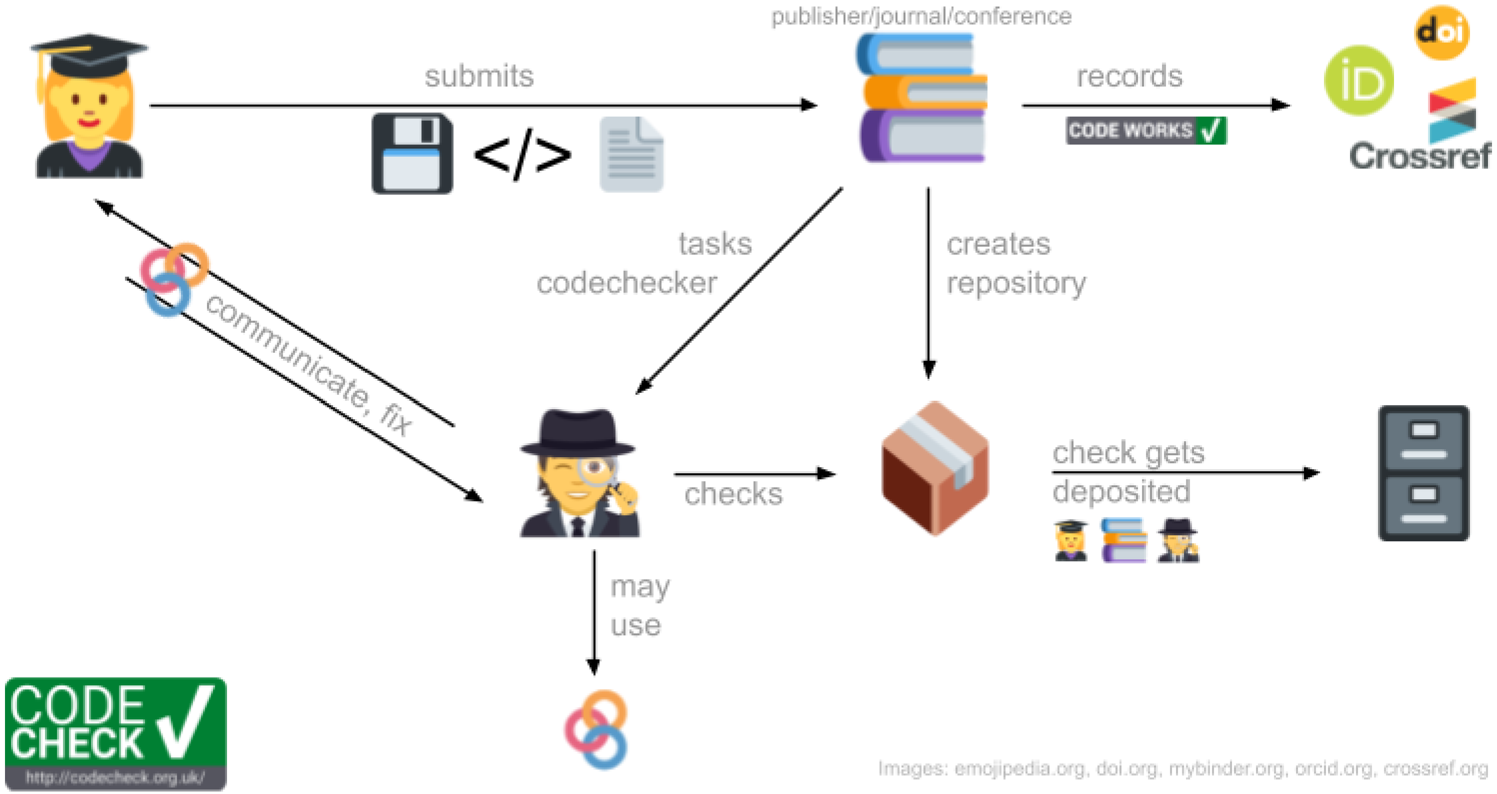
4. Workflows must be auditable.

CODE WORKS ✓

The CODECHECK philosophy

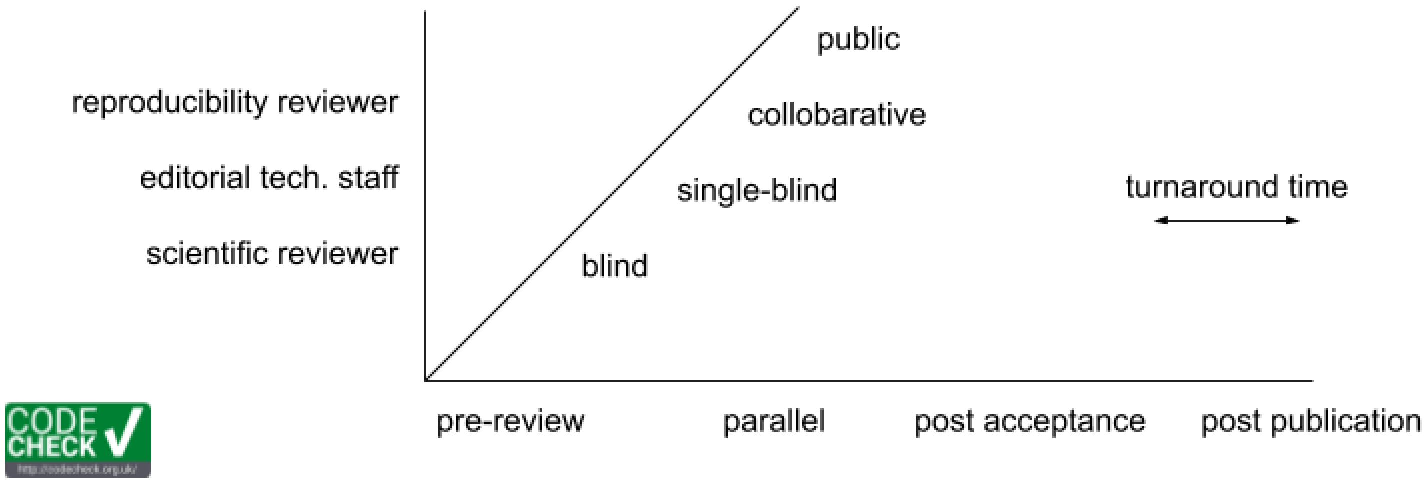
- Systems like Code Ocean set the bar high by "making code reproducible *forever* for *everyone*".
- CODECHECK simply asks "was the code reproducible *once* for *someone* else?"
- We check the code runs and generates the expected number of output files.
- The contents of those output files are not necessarily checked, but are available for others to see.
- The validity of the code is *not* checked.

Implementing the CODECHECK process



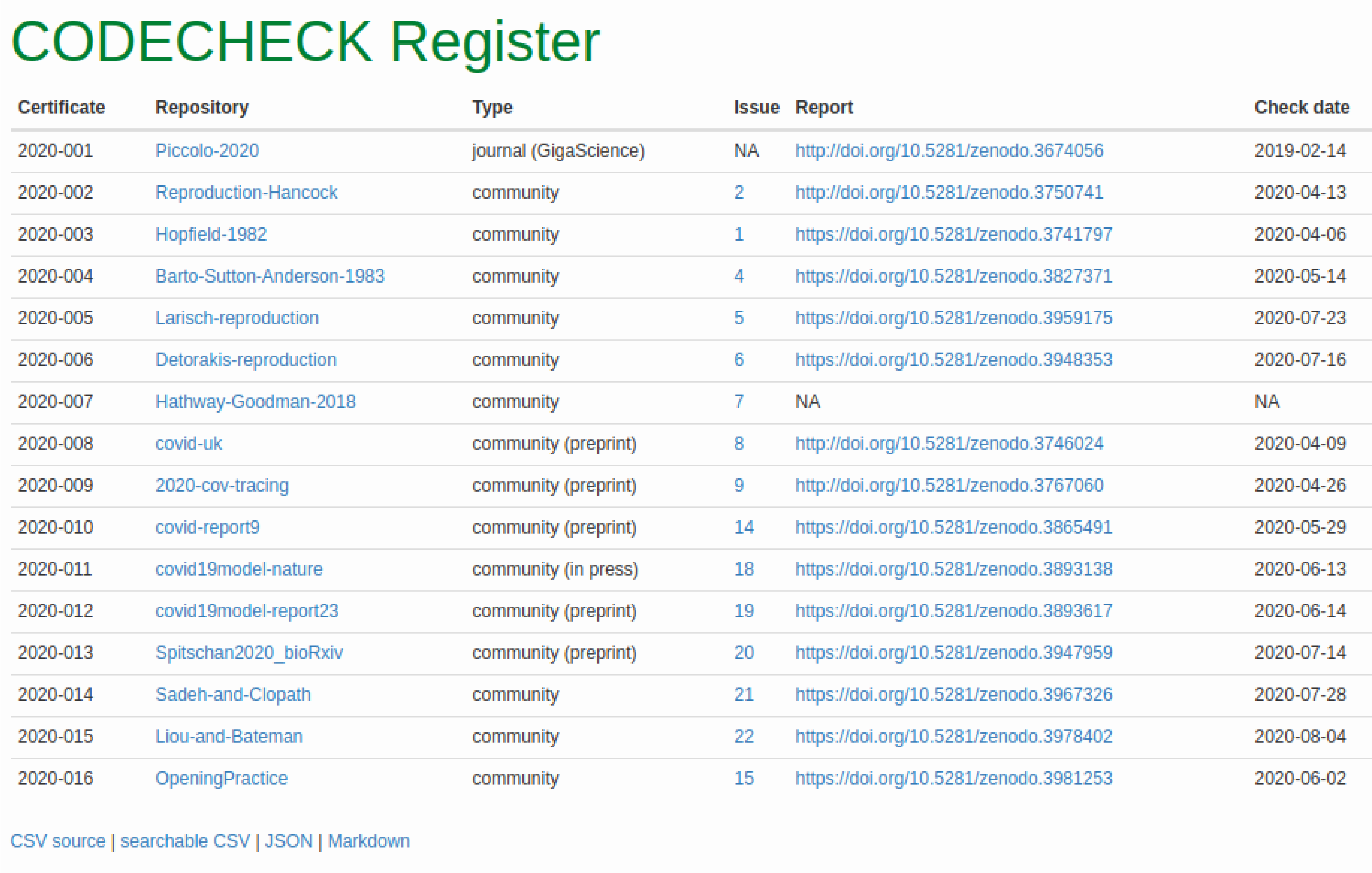
Sketch of steps and responsibilities to implement a CODECHECK, see https://codecheck.org.uk/process/.

Process variations



- Post review with extra role (ECR opportunity)
- Pre-review with extra role on staff
- Parallel reproducibility reviewer (flexible skill matching)
- Regular reviewer (though not widely established during last 20 years ¯_(ツ)_/¯)
- Independent community checks for preprints or postprints

https://codecheck.org.uk/process/



CODECHECK Register

Certificate	Repository	Type	Issue	Report	Check date
2020-001	Piccolo-2020	journal (GigaScience)	NA	http://doi.org/10.5281/zenodo.3674056	2019-02-14
2020-002	Reproduction-Hancock	community	2	http://doi.org/10.5281/zenodo.3750741	2020-04-13
2020-003	Hopfield-1982	community	1	https://doi.org/10.5281/zenodo.3741797	2020-04-06
2020-004	Barto-Sutton-Anderson-1983	community	4	https://doi.org/10.5281/zenodo.3827371	2020-05-14
2020-005	Larisch-reproduction	community	5	https://doi.org/10.5281/zenodo.3959175	2020-07-23
2020-006	Detorakis-reproduction	community	6	https://doi.org/10.5281/zenodo.3948353	2020-07-16
2020-007	Hathway-Goodman-2018	community	7	NA	NA
2020-008	covid-uk	community (preprint)	8	http://doi.org/10.5281/zenodo.3746024	2020-04-09
2020-009	2020-cov-tracing	community (preprint)	9	http://doi.org/10.5281/zenodo.3767060	2020-04-26
2020-010	covid-report9	community (preprint)	14	https://doi.org/10.5281/zenodo.3865491	2020-05-29
2020-011	covid19model-nature	community (in press)	18	https://doi.org/10.5281/zenodo.3893138	2020-06-13
2020-012	covid19model-report23	community (preprint)	19	https://doi.org/10.5281/zenodo.3893617	2020-06-14
2020-013	Spitschan2020_bioRxiv	community (preprint)	20	https://doi.org/10.5281/zenodo.3947959	2020-07-14
2020-014	Sadeh-and-Clopath	community	21	https://doi.org/10.5281/zenodo.3967326	2020-07-28
2020-015	Liou-and-Bateman	community	22	https://doi.org/10.5281/zenodo.3978402	2020-08-04
2020-016	OpeningPractice	community	15	https://doi.org/10.5281/zenodo.3981253	2020-06-02

CSV source | searchable CSV | JSON | Markdown

https://codecheck.org.uk/register/

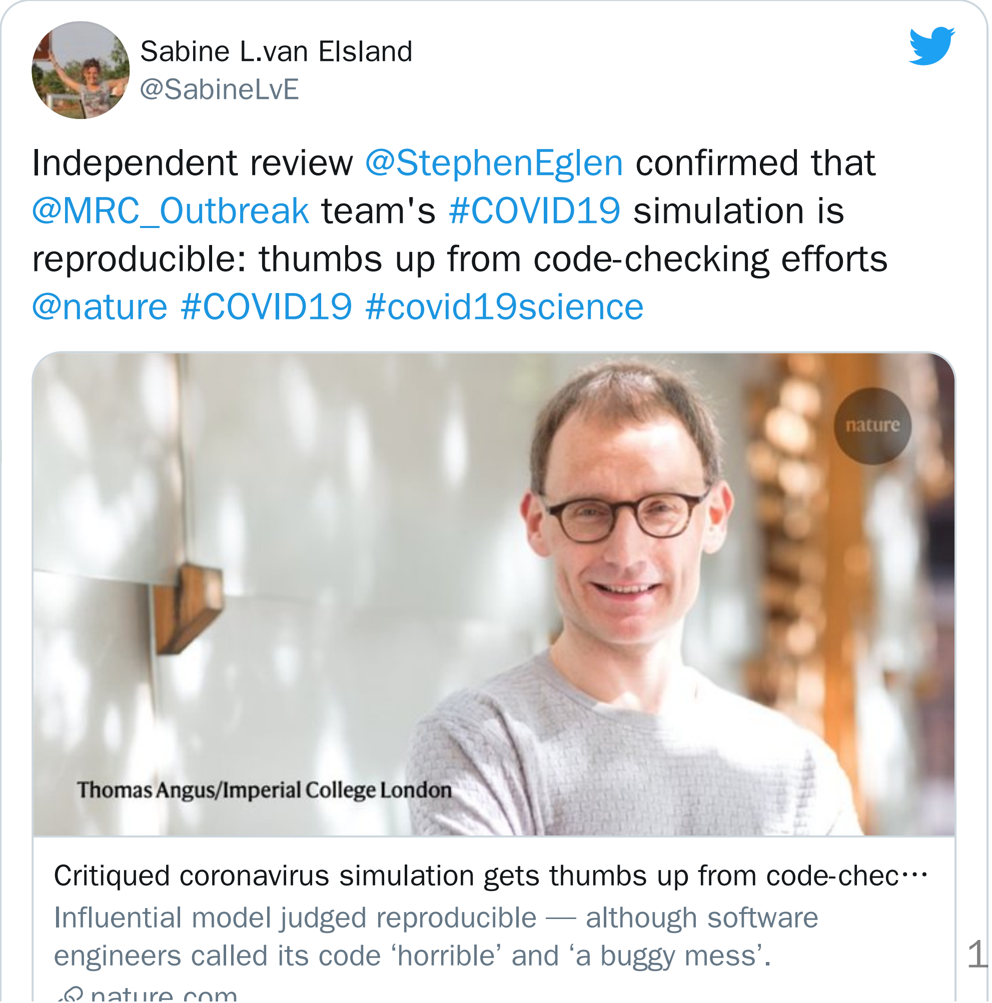
Example 1 (Stephen)

DOI 10.5281/zenodo.3865491

https://zenodo.org/record/3865491/files/codecheck.pdf

"It ain't pretty, but it works" (Hilda Bastian)

(Tweet on the right:
https://twitter.com/SabineLvE/status/127078972705934950



Sabine L.van Elsland
@SabineLvE

Independent review @StephenEglen confirmed that @MRC_Outbreak team's #COVID19 simulation is reproducible: thumbs up from code-checking efforts @nature #COVID19 #covid19science

Critiqued coronavirus simulation gets thumbs up from code-chec…
Influential model judged reproducible — although software engineers called its code ‘horrible’ and ‘a buggy mess’.
nature.com

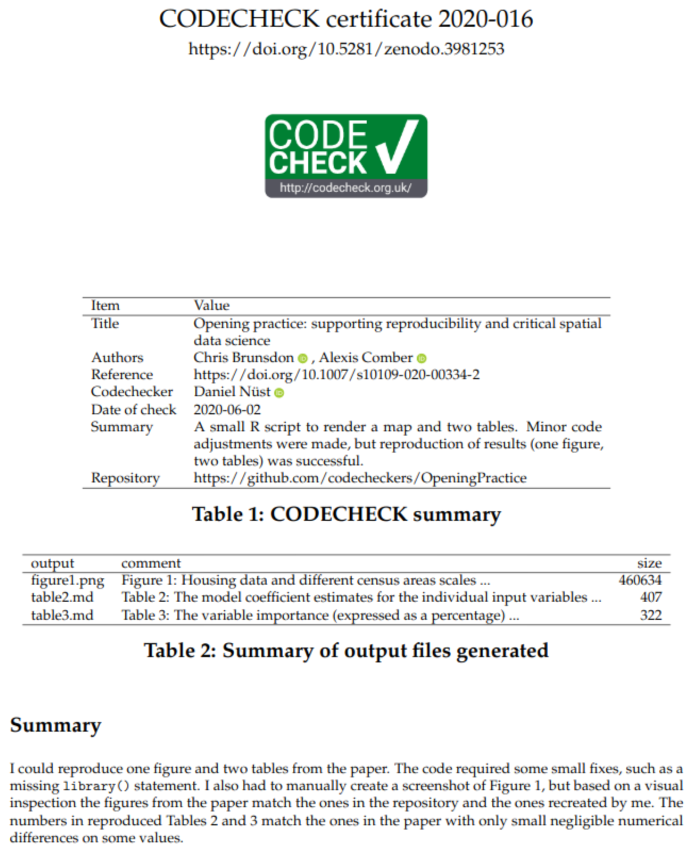
Example 2 (Daniel)

DOI 10.5281/zenodo.3981253

Report:
https://zenodo.org/record/3981253/files/codecheck.pdf

Repository:
https://github.com/codecheckers/OpeningPractice

Paper (acknowledgement):
https://link.springer.com/article/10.1007/s10109-020-00334-2#Ack1



CODECHECK certificate 2020-016
https://doi.org/10.5281/zenodo.3981253

CODE
CHECK
http://codecheck.org.uk/

Item	Value
Title	Opening practice: supporting reproducibility and critical spatial data science
Authors	Chris Brunsdon , Alexis Comber
Reference	https://doi.org/10.1007/s10109-020-00334-2
Codechecker	Daniel Nüst
Date of check	2020-06-02
Summary	A small R script to render a map and two tables. Minor code adjustments were made, but reproduction of results (one figure, two tables) was successful.
Repository	https://github.com/codecheckers/OpeningPractice

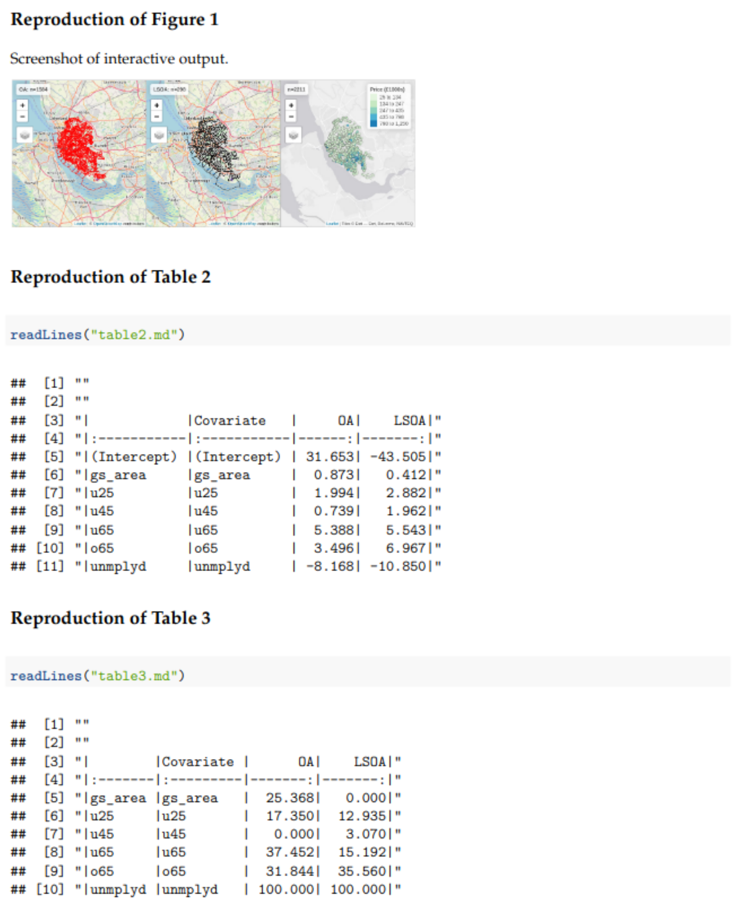
Table 1: CODECHECK summary

output	comment	size
figure1.png	Figure 1: Housing data and different census areas scales ...	460634
table2.md	Table 2: The model coefficient estimates for the individual input variables ...	407
table3.md	Table 3: The variable importance (expressed as a percentage) ...	322

Table 2: Summary of output files generated

Summary

I could reproduce one figure and two tables from the paper. The code required some small fixes, such as a missing `library()` statement. I also had to manually create a screenshot of Figure 1, but based on a visual inspection the figures from the paper match the ones in the repository and the ones recreated by me. The numbers in reproduced Tables 2 and 3 match the ones in the paper with only small negligible numerical differences on some values.



Reproduction of Figure 1

Screenshot of interactive output.

Reproduction of Table 2

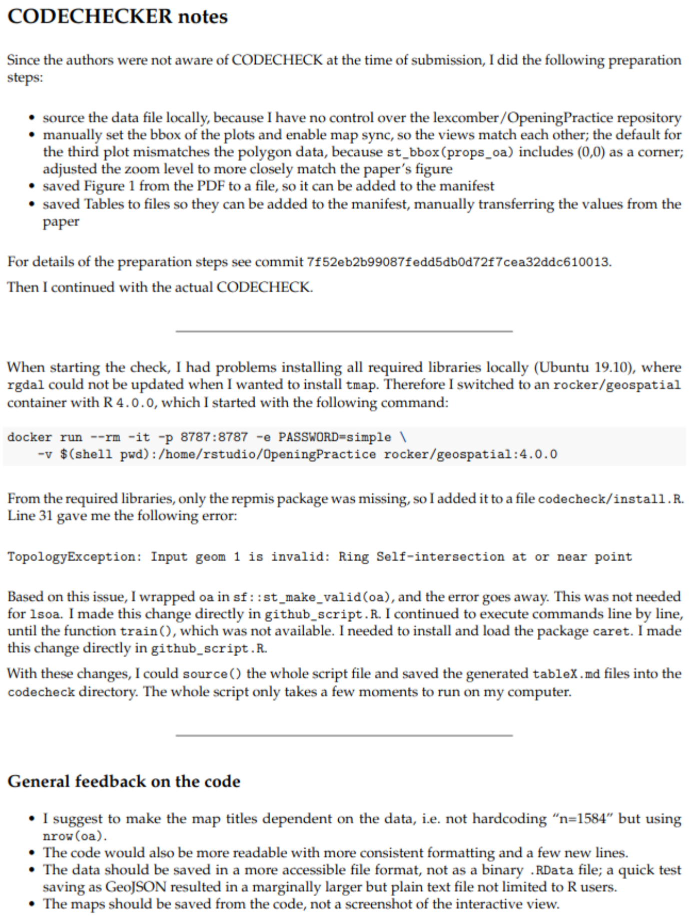
```
readLines("table2.md")
```

```
## [1] ""
## [2] ""
## [3] "|             |Covariate   |     OA|    LSOA|"
## [4] "|:-----------|:-----------|------:|-------:|"
## [5] "|(Intercept) |(Intercept) | 31.653| -43.505|"
## [6] "|gs_area     |gs_area     |  0.873|   0.412|"
## [7] "|u25         |u25         |  1.994|   2.882|"
## [8] "|u45         |u45         |  0.739|   1.962|"
## [9] "|u65         |u65         |  5.388|   5.543|"
## [10] "|o65         |o65         |  3.496|   6.967|"
## [11] "|unmplyd     |unmplyd     | -8.168| -10.850|"
```

Reproduction of Table 3

```
readLines("table3.md")
```

```
## [1] ""
## [2] ""
## [3] "|        |Covariate |      OA|    LSOA|"
## [4] "|:-------|:---------|-------:|-------:|"
## [5] "|gs_area |gs_area   |  25.368|   0.000|"
## [6] "|u25     |u25       |  17.350|  12.935|"
## [7] "|u45     |u45       |   0.000|   3.070|"
## [8] "|u65     |u65       |  37.452|  15.192|"
## [9] "|o65     |o65       |  31.844|  35.560|"
## [10] "|unmplyd |unmplyd   | 100.000| 100.000|"
```



CODECHECKER notes

Since the authors were not aware of CODECHECK at the time of submission, I did the following preparation steps:

- source the data file locally, because I have no control over the lexcomber/OpeningPractice repository
- manually set the bbox of the plots and enable map sync, so the views match each other; the default for the third plot mismatches the polygon data, because `st_bbox(props_oa)` includes (0,0) as a corner; adjusted the zoom level to more closely match the paper's figure
- saved Figure 1 from the PDF to a file, so it can be added to the manifest
- saved Tables to files so they can be added to the manifest, manually transferring the values from the paper

For details of the preparation steps see commit `7f52eb2b99087fedd5db0d72f7cea32ddc610013`.

Then I continued with the actual CODECHECK.

When starting the check, I had problems installing all required libraries locally (Ubuntu 19.10), where `rgdal` could not be updated when I wanted to install `tmap`. Therefore I switched to an `rocker/geospatial` container with R `4.0.0`, which I started with the following command:

```
docker run --rm -it -p 8787:8787 -e PASSWORD=simple \
    -v $(shell pwd):/home/rstudio/OpeningPractice rocker/geospatial:4.0.0
```

From the required libraries, only the repmis package was missing, so I added it to a file `codecheck/install.R`. Line 31 gave me the following error:

```
TopologyException: Input geom 1 is invalid: Ring Self-intersection at or near point
```

Based on this issue, I wrapped `oa` in `sf::st_make_valid(oa)`, and the error goes away. This was not needed for `lsoa`. I made this change directly in `github_script.R`. I continued to execute commands line by line, until the function `train()`, which was not available. I needed to install and load the package `caret`. I made this change directly in `github_script.R`.

With these changes, I could `source()` the whole script file and saved the generated `tableX.md` files into the `codecheck` directory. The whole script only takes a few moments to run on my computer.

General feedback on the code

- I suggest to make the map titles dependent on the data, i.e. not hardcoding "n=1584" but using `nrow(oa)`.
- The code would also be more readable with more consistent formatting and a few new lines.
- The data should be saved in a more accessible file format, not as a binary `.RData` file; a quick test saving as GeoJSON resulted in a marginally larger but plain text file not limited to R users.
- The maps should be saved from the code, not a screenshot of the interactive view.

https://zenodo.org/record/3981253/files/codecheck.pdf

What would help to conduct CODECHECKs?

1. Good README files

2. Advanced control of computing environments

3. Good practices around research compendia

4. Suggest and reward little steps

5. Tools for codecheckers

https://codecheck.org.uk/guide/community-process

Every little step helps

Concrete steps for authors

- have a `README` (*"all else is details"*)
- "document for future you"
- use good file names and paths
- use text-based, open file formats
- publish data subsets/mock data
- apply templates & follow community good practices (e.g., `rrtools`)
- write and publish notebooks
- use only scripts, no point-and-click
- embrace openness & be-ne fi-ts
- work/review in the spirit of preproducibility
- only work in containers

Sustainable culture change

- every step towards openness and reproducibility matters (spectrum!)
- acknowledge challenges of computational reproducibility in education and publications
- reward early adopters
- *change policies*

Next steps

1. Embedding into journal workflows

2. Training a community of codecheckers

3. Funding for a codecheck editor

CODECHECK is an open community:

Get involved as a...

- codechecker
- author
- reviewer
- editor
- publisher
- conference organiser

https://codecheck.org.uk/get-involved/

More information: codecheck.org.uk

Team

- 🐦 StephenEglen
- 🐦 nordholmen
- ✉ Stephen Eglen, University of Cambridge
- ✉ Daniel Nüst, University of Münster

Acknowledgements

The project is supported by a Mozilla Open Science Mini-Grant (see official announcement) from February 2019 to May 2020.

moz://a

Supported by the UK Software Sustainability Institute.

Material

Everything is on GitHub https://github.com/codecheckers/.

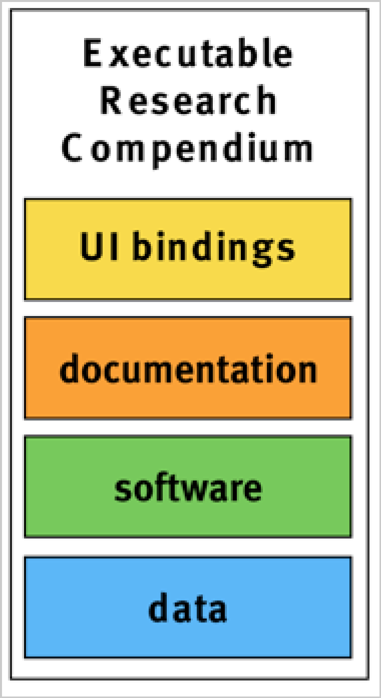
Watch Stephen's talks on CODECHECK at The 14th Munin Conference on Scholarly Publishing 2019 and at the UK Open Research Working Group meeting in September 2020.

Declarations

SJE is affiliate editor of bioRxiv and senior editor of Scientific Data.

Advanced control of computing environments

- Binder-ready research compendium binder
- Research compendium + container + bindings = Executable Research Compendium (o2r.info/results/)
- Ten Simple Rules for Writing Dockerfiles for Reproducible Data Science



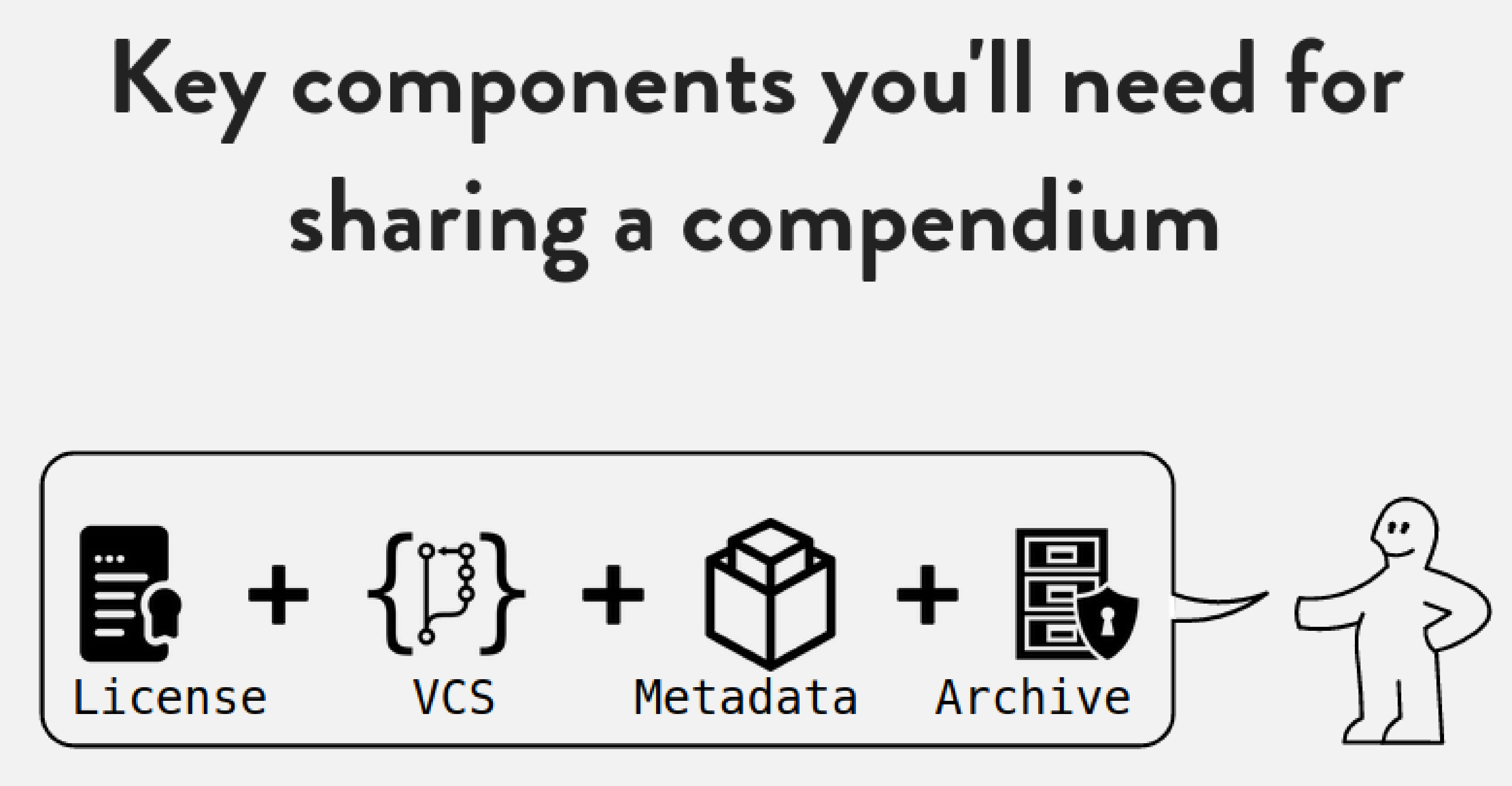
Research Compendium

research-compendium.science

Research compendia

> ...We introduce the concept of a *compendium as* both *a container for* the different elements that make up the document and its computations (i.e. *text, code, data, ...*), and as a means for *distributing*, *managing* and *updating* the collection.

Gentleman, Robert, and Duncan Temple Lang. 2007. "Statistical Analyses and Reproducible Research". Journal of Computational and Graphical Statistics 16 (1): 1–23. https://doi.org/10.1198/106186007X178663



Key components you'll need for sharing a compendium

Source: Ram (2019), How To Make Your Data Analysis Notebooks More Reproducible

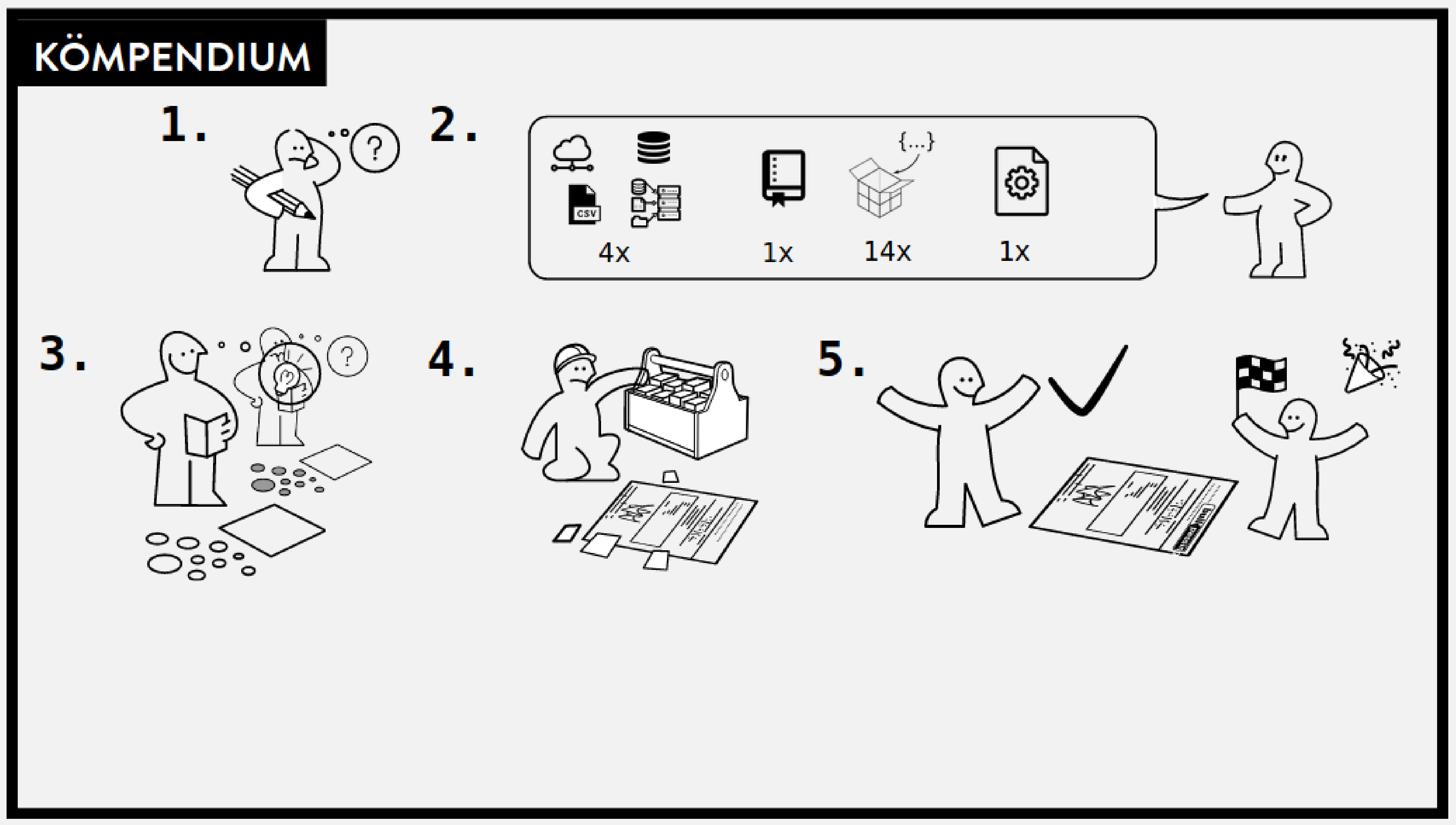
Research compendia

Ram (2019), How To Make Your Data Analysis Notebooks More Reproducible

- small, medium, large compendia

- Stick with the conventions of your peers
- Keep data, methods and outputs separate
- Specify your computational environment as clearly as you can

- Leverage the R package structure and support tools/services as much as possible
- Use modern tools to make your compendia more accessible (repo2docker, containerit, holepunch, drake)
- Don't forget long-term archives and simpler formats (Zenodo)



Source: Ram (2019), How To Make Your Data Analysis Notebooks More Reproducible

Transparent

Credit

Discover

Reuse

Colaborate

More on research compendia at research-compendium.science

You want to introduce changes in your community?

Reproducible Publications at AGILE Conferences

AGILE Reproducible Paper Guidelines

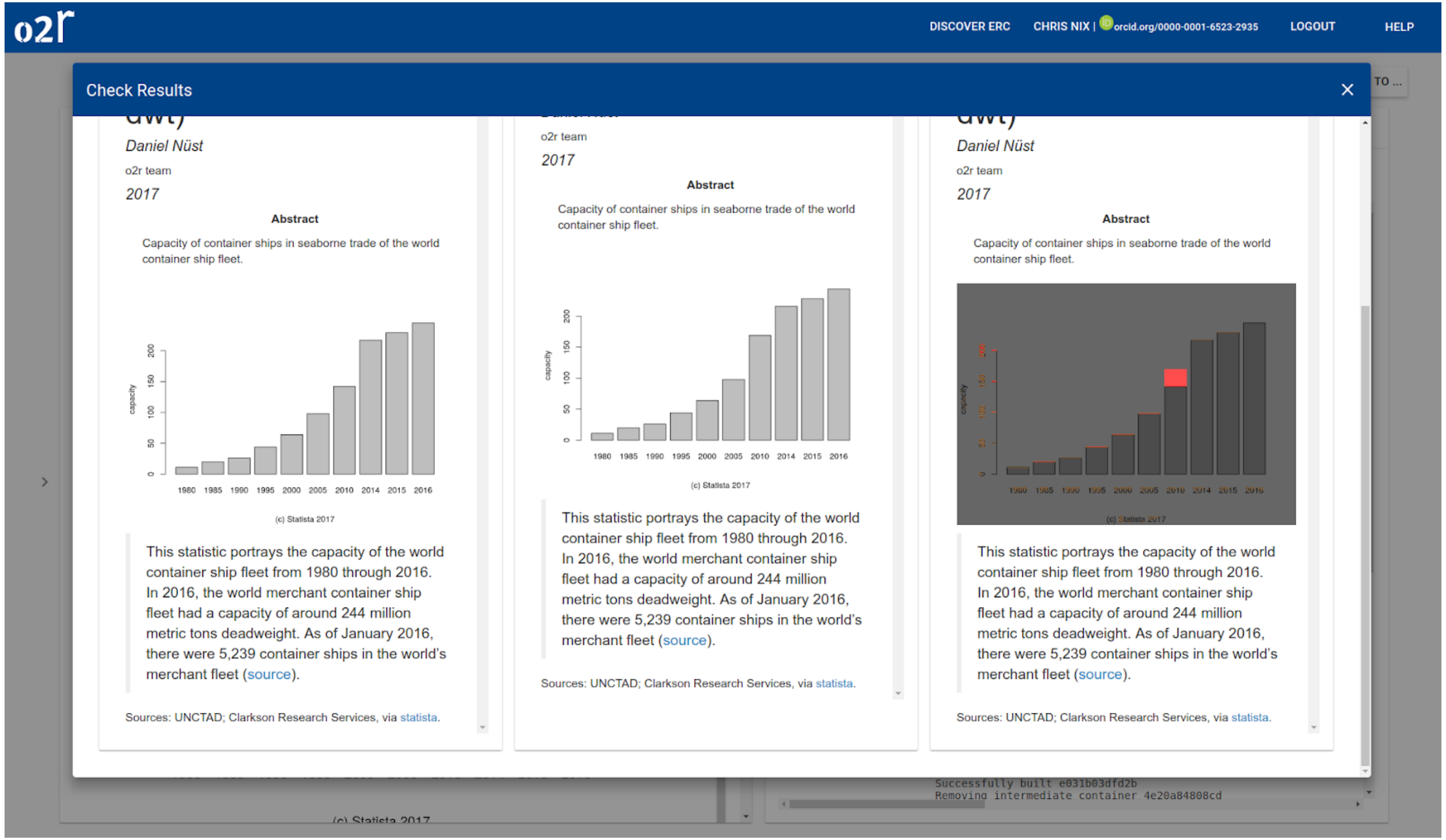
https://reproducible-agile.github.io/

ERCs in peer review

https://o2r.info/pilots/

- Collaboration pilots
- OJS pilot

o2r goals and benefits



Check Results

Daniel Nüst

o2r team

2017

Abstract

Capacity of container ships in seaborne trade of the world container ship fleet.

(c) Statista 2017

This statistic portrays the capacity of the world container ship fleet from 1980 through 2016. In 2016, the world merchant container ship fleet had a capacity of around 244 million metric tons deadweight. As of January 2016, there were 5,239 container ships in the world's merchant fleet (source).

Sources: UNCTAD; Clarkson Research Services, via statista.

o2r team

2017

Abstract

Capacity of container ships in seaborne trade of the world container ship fleet.

(c) Statista 2017

This statistic portrays the capacity of the world container ship fleet from 1980 through 2016. In 2016, the world merchant container ship fleet had a capacity of around 244 million metric tons deadweight. As of January 2016, there were 5,239 container ships in the world's merchant fleet (source).

Sources: UNCTAD; Clarkson Research Services, via statista.

Daniel Nüst

o2r team

2017

Abstract

Capacity of container ships in seaborne trade of the world container ship fleet.

(c) Statista 2017

This statistic portrays the capacity of the world container ship fleet from 1980 through 2016. In 2016, the world merchant container ship fleet had a capacity of around 244 million metric tons deadweight. As of January 2016, there were 5,239 container ships in the world's merchant fleet (source).

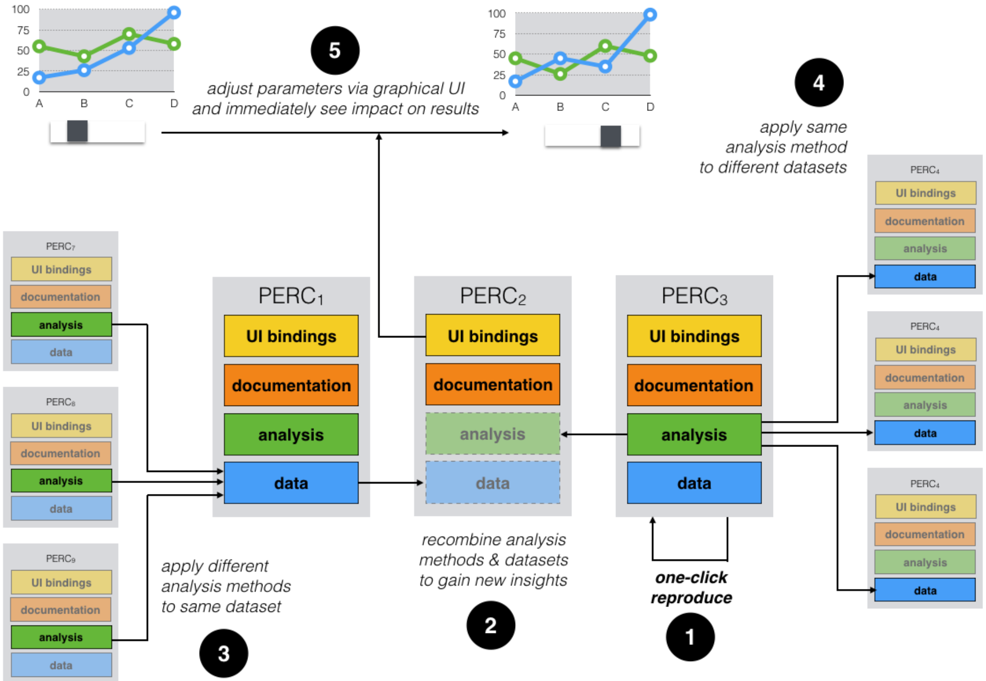
Sources: UNCTAD; Clarkson Research Services, via statista.

(c) Statista 2017

```
Successfully built e031b03dfd2b
Removing intermediate container 4e20a84808cd
```

https://o2r.info/results | https://o2r.uni-muenster.de/

o2r goals and benefits



adjust parameters via graphical UI and immediately see impact on results

apply same analysis method to different datasets

apply different analysis methods to same dataset

recombine analysis methods & datasets to gain new insights

one-click reproduce

Nüst, Daniel, Markus Konkol, Edzer Pebesma, Christian Kray, Marc Schutzeichel, Holger Przibytzin, and Jörg Lorenz. 2017. Opening the Publication Process with Executable Research Compendia. D-Lib Magazine 23 (1/2). https://doi.org/10.1045/january2017-nuest.

Kray, Christian, Pebesma, Edzer, Konkol, Markus, Nüst, Daniel (2019). Reproducible Research in Geoinformatics: Concepts, Challenges and Benefits. https://doi.org/10.4230/LIPIcs.COSIT.2019.8

Computational Research / Data Science 2020

Challenges

- dependency hell
- FAIR
- licensing
- sensitive data
- big data

Opportunities/Solutions

- version control
- containerisation
- openness (data, software, preprints)
- community
- self-education
- collaboration
- research integrity
- The Carpentries
- subsets/enclaves/domains