

Reproducibility

Principles and Practice

Daniela Palleschi

Humboldt-Universität zu Berlin

2024-04-30

Learning Objectives

Today we will learn about...

- reproducibility rates in linguistics
- FAIR principles
- concepts for building a reproducible workflow

Reproducibility

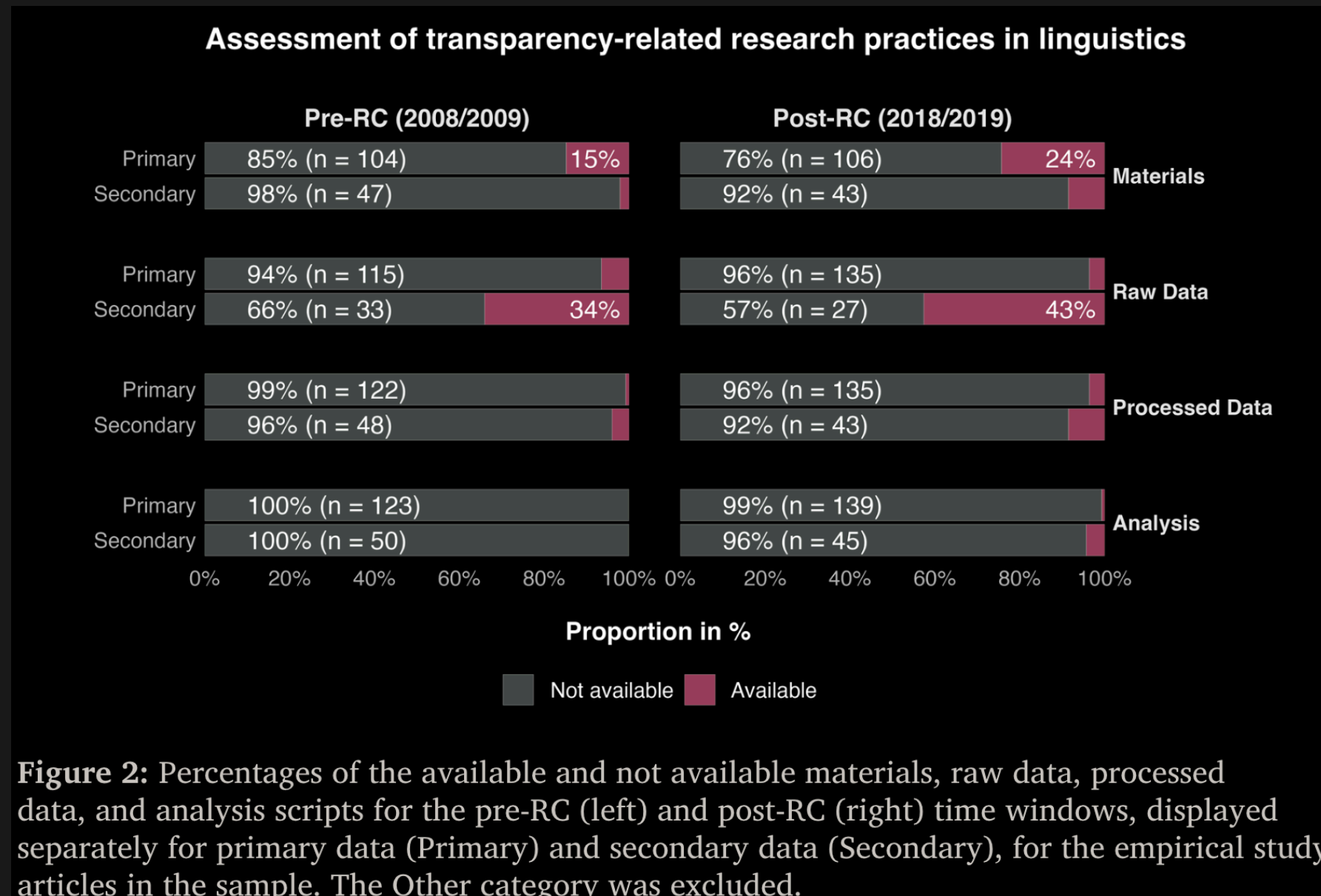
- generating the same results with the same data and analysis scripts
 - seems obvious, but requires organisation and forethought
- bare minimum: share the code and the data ([Laurinavichyute et al., 2022](#))
- rates of reproducibility vary across fields ([Bochynska et al., 2023](#))
 - open access: 25-65%
 - data and analyses sharing: 11-33%
 - pre-registrations: 0-3%
- what constitutes “reproducibility”?

What should (ideally) be shared?

- materials
 - protocols
 - stimuli
 - experiment set-up
- documentation
 - README
 - metadata
- materials are helpful for replication
 - but also for inspection of e.g., design
- data
 - raw
 - e.g., text files, audio, video, or images
 - processed
- analysis code
 - pre-processing
 - analyses
- data and code are necessary for reproducibility
 - along with proper documentation of software used

Reproducibility rates in linguistic research

Figure 1: Source: Bochynska et al. (2023), p. 11 (all rights reserved)



- meta-analysis of 519 randomly sampled articles from various linguistic journals
 - pre- and post-reproducibility crisis (2008/9, 2018/19) (Bochynska et al., 2023)
 - differentiated between primary (collected for study) and secondary (pre-existing) data
- reported a post-RC increase in shared materials, data, and analyses
 - but still low rates of each
- higher rates of secondary data sharing, presumably due to publicly available corpora
- data shared more often than analyses, pre- and post-RC

Journal of Memory and Language

- meta-analysis of articles from JML ([Laurinavichyute et al., 2022](#))
 - before and after an Open Science Policy was introduced in 2019

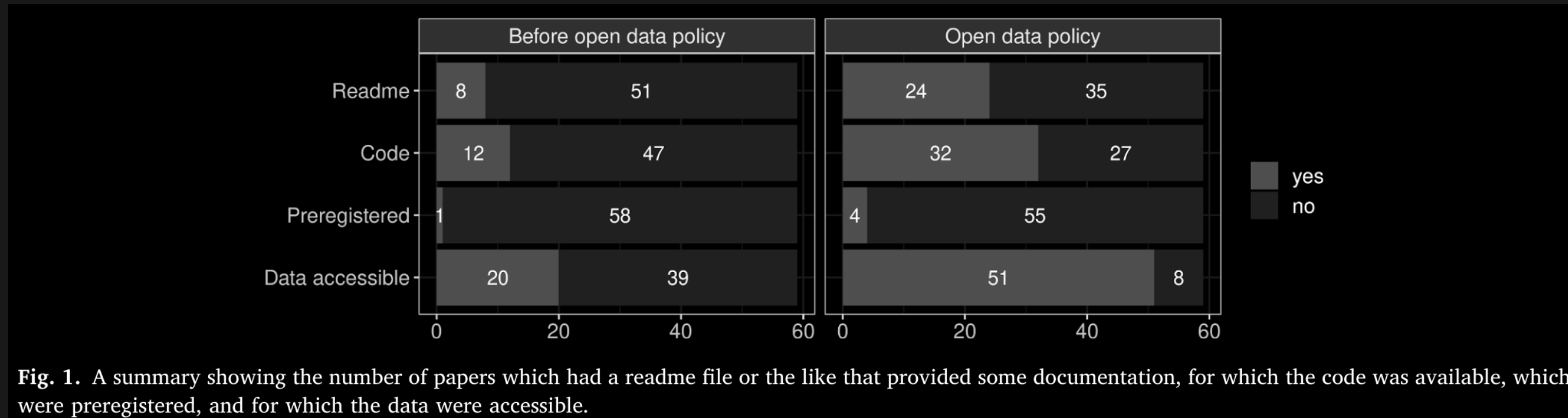


Figure 2: Source: Laurinavichyute et al. ([2022](#)), p. 5 (all rights reserved)

- code and data availability improved
- but reproducibility rate ranged from 34-56%, depending on criteria
- higher rates compared to field-wide meta-analysis ([Bochynska et al., 2023](#))

FAIR principles

- guidelines for sharing digital resources
- refers broadly to data, but we'll consider it in terms of analyses

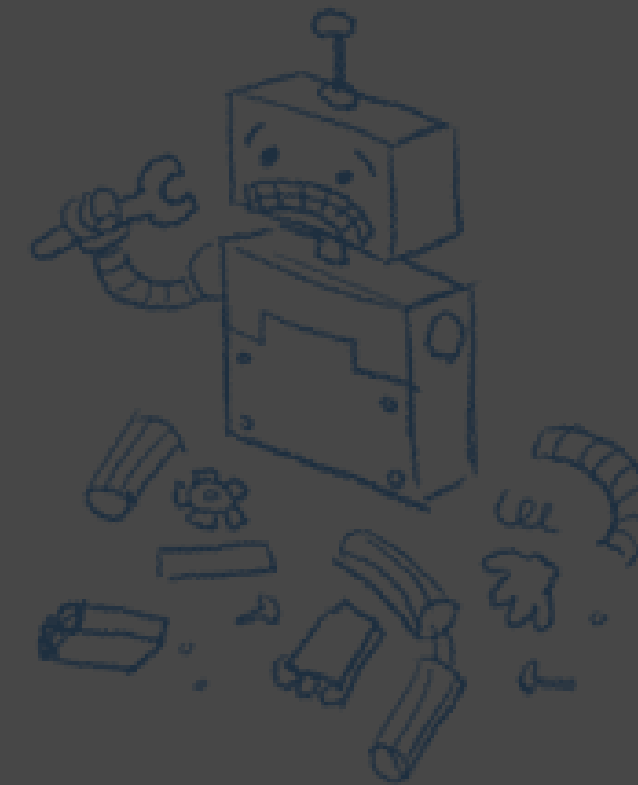
Figure 3: Source: [National Library of Medicine](#) (all rights reserved)



- findable and accessible refer to where materials are stored
 - in *findable* repositories
 - that are *accessible*, i.e., do not require an account
- interoperable and reusable emphasise the format of data (and code)
 - the importance of future use
 - and use beyond your precise computational environment
- a great way to test the FAIR principles
 - code review!
 - i.e., have a colleague try to access your data/run your code
 - either via an online repository
 - or send them your project folder

Findable

- refers to data and supplementary materials
- materials should have a “persistent identifier”
 - e.g., Digital Object Identifier (DOI) for scholarly articles
- a digital, long-term storage of data
 - *not* on a personal or professional website
 - GitHub files don’t typically have sufficient metadata
 - ideally: OSF, Zenodo or some other repository
- in recent papers, an OSF link is typically provided
- also: *discoverable*
 - e.g., in data-specific search engines (Google’s Dataset search)





Get access to the full version of this article. View access options below.

Institutional Login

Accessible

- data (and code) should be
 - machine- and human-readable
 - available on a trusted repository, e.g., the OSF
 - Open Access
 - not behind a paywall
 - nor require a login

Log in to the Online Library

If you have previously obtained access with your personal account, please log in.

Log in

Purchase Instant Access

☐ **48-Hour online access** | **\$12.00**

Details



☐ **Online-only access** | **\$20.00**

Details



☒ **PDF download and online access** | **\$49.00**

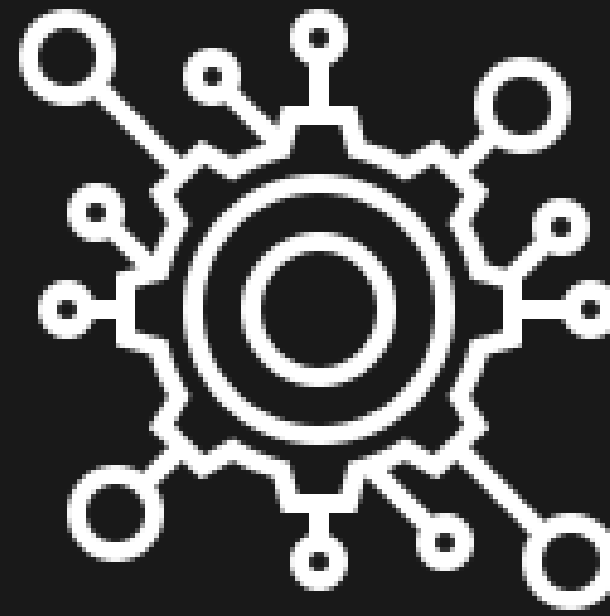
Details



Check out

Interoperable

- data (and code) should
 - not dependent on an operating system
 - nor entirely on software/package versions
- easiest work around:
 - document your software versions
 - this doesn't automatically facilitate interoperability
 - but may help pinpoint where problems are coming from



Reusable

- data (and code) should
 - be reusable for future research
- data format should be generic
 - i.e., not tied to a specific program
 - for tabular data, I recommend **.CSV** format
- we can swap with ‘reproducible’ in the context of analyses



Task: finding data

Go to datasetsearch.research.google.com/

- do a search for data related to a topic of interest to you
- what type of information does the search provide?
- what type of links?
- do you find analysis code, or just data?
- do the same search at osf.io
- and at zenodo.org/
 - are there the same amount of hits?

Data and code availability

- “data available upon (reasonable) request”
 - generally not true
- data was not available in 68% of the most cited psychology studies (2006-2016) ([Hardwicke & Ioannidis, 2018](#))
 - a further 18% were available with restrictions
 - only 11% available without restriction
- data alone is not sufficient
 - ‘Data Analysis’ sections are rarely exhaustive/unambiguous
 - very difficult to re-create analyses without code
 - e.g., is data trimming explicitly defined?
 - this will even affect descriptive statistics

Figure 4: Source: Hardwicke & Ioannidis (2018), p. 6 (all rights reserved)

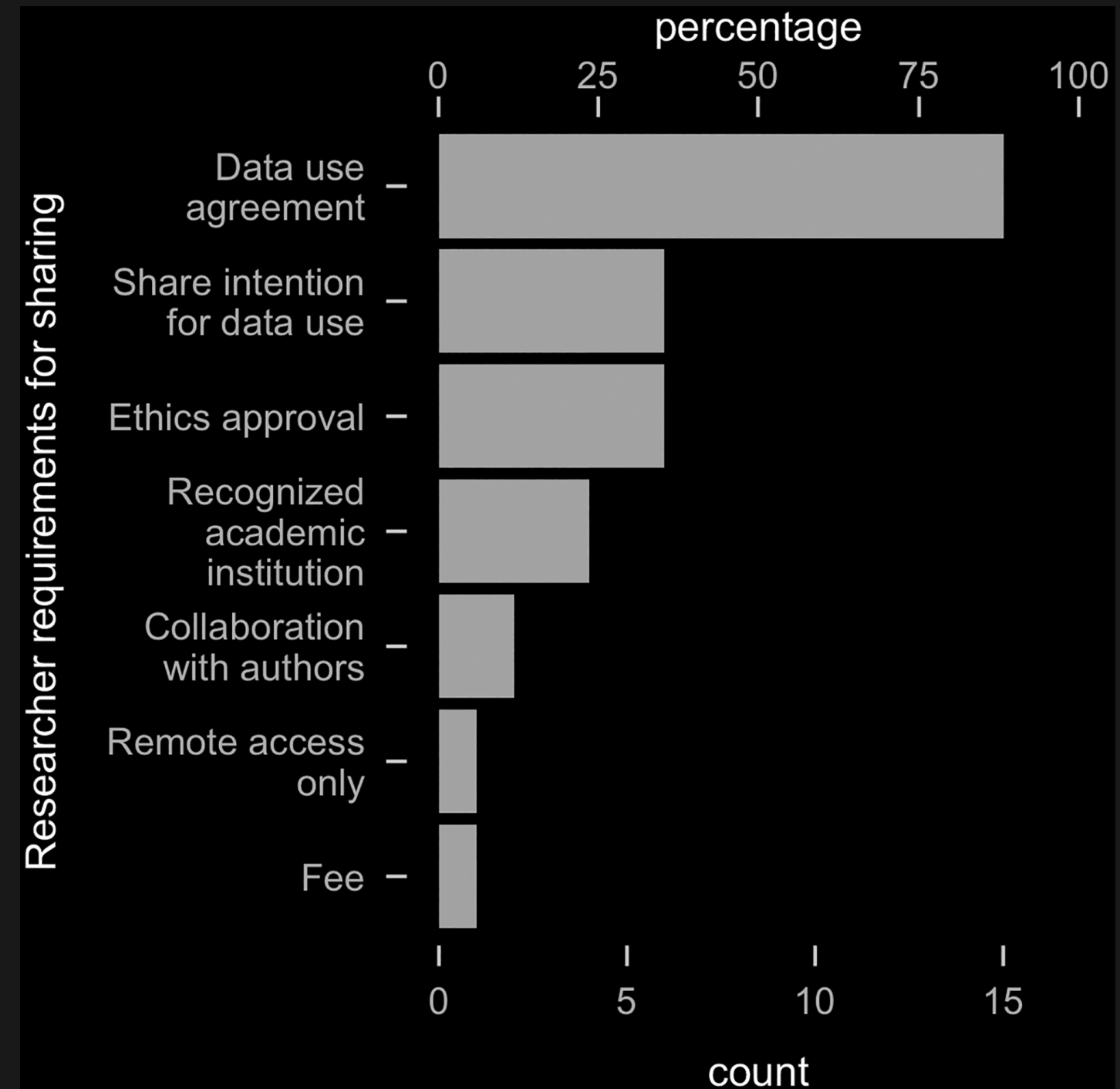


Fig 3. Requirements imposed by researchers before data can be shared. X-axes represent counts and percentages (of n = 17 with restricted data sharing systems in place). Note that these data are not mutually exclusive: individual sharing schemes often entailed multiple restrictions.

Data and code \neq Reproducibility

- even including code does not guarantee reproducibility
- access to data and code do not mean analyses are reproducible
- what can go wrong? Examples from Laurinavichyute et al. ([2022](#))

1. Data problems

- inaccessible data
- incomplete data (e.g., 2/3 experiments)

2. Code problems

- incomplete code
- error messages
- code rot: outdated syntax or environment
- proprietary software

3. Documentation problems

- data difficult to interpret
- no README file/data dictionary
- unclear folder/file/variable naming convention
- manuscript contradicts code

4. Unclear terms of use

- no licence specification

Share the code, not just the data

- Why?
 - key details are often missing from ‘Methods’ sections
- suggestions for researchers from Laurinavichyute et al. ([2022](#))

1. Share data in usable form

- with pre-processing code

2. Use publicly accessible repositories

- e.g., OSF

3. Use non-proprietary data formats

- e.g., not **.xls** files (Excel)

4. Provide documentation

- e.g., README, data dictionaries

5. Share code *and* data

- they estimate a 38% increase in reproducibility

6. Teach data management and computing skills

- that’s what this course is for!

Building a reproducible workflow

- there are different levels of reproducibility
 - the *bare minimum* is sharing the code and data
 - *and* including session information:
 - which operating system was used
 - which software/package versions were used
- going bigger:
 - project-oriented workflow
 - project-specific filepaths
 - contained in a single project folder
- we will be using RProjects to achieve this

Project management

- folder structure
- project-relative file paths
- appropriate documentation
 - e.g., README
- it's great to map out your project structure early on
 - but it will grow as you go along
 - reproducible principles facilitate adapting as it grows

Literate programming

Instead of imagining that our main task is to instruct a *computer* what to do, let us concentrate rather on explaining to *human beings* what we want a computer to do.

— Knuth (1984), p. 97

- originally used to refer to writing programs
- but also applies to analysis code
 - especially if we're aiming for reproducibility
- main concepts:
 - code is linear (this pre-dates Knuth, 1984)
 - informative but concise commenting
- main benefits:
 - facilitates maintenance
 - helpful for future-you, collaborators, etc.

Documentation

- metadata
 - project README
 - codebook/data dictionary
- README should contain
 - a project description
 - relevant links
 - description of folder structure
- can be updated as the project develops
- README.md files in GitHub/Lab are automatically used as a project description
 - **.md** is a plaintext document
 - uses markdown syntax

Version control

- git: local tracking
- useful for the analysis and writing phases
 - but can be tricky for collaboration
- GitHub/GitLab: remote tracking
 - store your changes to your local git repository
 - then push them to your remote repository
- safe guards against local hardware/software issues
 - lost or damaged computer or local files
- and allows for collaboration or sharing

Persistent (public) storage

- GitHub/Lab are sub-optimal
 - developer-focused
 - typically lack thorough documentation/metadata
 - not very user-friendly for non-users
- OSF, Zenodo
 - Open Science-focused
 - can be linked to a GitHub/Lab repository
 - facilitate thorough documentation
 - user-friendly

Writing




- dynamic reports with Markdown syntax
 - e.g., Rmarkdown, Quarto
 - integration of data, code, and prose
 - facilitates cross-referencing within document
 - integration of citation management tools
 - supports LaTeX syntax for example sentences and tables
- **papa****j****a** package for APA-formatted Rmarkdown documents
- challenge: collaboration
 - not all collaborators know these tools
 - track changes not currently possible

Setting up a project

- next week: hands-on
- required installations/recent versions of:
 - R
 - version **4.4.0**, “Puppy Cup”
 - check current version with **R.version**
 - download/update: <https://cran.r-project.org/bin/macosx/>
 - RStudio
 - version **2023.12.1.402**, “Ocean Storm”
 - Help > Check for updates
 - new install: <https://posit.co/download/rstudio-desktop/>

Learning objectives

Today we learned...

- reproducibility rates in linguistics 
- FAIR principles 
- concepts for building a reproducible workflow 

References

- Bochynska, A., Keeble, L., Halfacre, C., Casillas, J. V., Champagne, I.-A., Chen, K., Röthlisberger, M., Buchanan, E. M., & Roettger, T. B. (2023). Reproducible research practices and transparency across linguistics. *Glossa Psycholinguistics*, 2(1).
<https://doi.org/10.5070/G6011239>
- Corker, K. S. (2022). An Open Science Workflow for More Credible, Rigorous Research. In M. J. Prinstein (Ed.), *The Portable Mentor* (3rd ed., pp. 197–216). Cambridge University Press. <https://doi.org/10.1017/9781108903264.012>
- Hardwicke, T. E., & Ioannidis, J. P. A. (2018). Populating the Data Ark: An attempt to retrieve, preserve, and liberate data from the most highly-cited psychology and psychiatry articles. *PLOS ONE*, 13(8), e0201856. <https://doi.org/10.1371/journal.pone.0201856>
- Knuth, D. (1984). Literate programming. *The Computer Journal*, 27(2), 97–111.
- Laurinavichyute, A., Yadav, H., & Vasishth, S. (2022). Share the code, not just the data: A case study of the reproducibility of articles published in the Journal of Memory and Language under the open data policy. *Journal of Memory and Language*, 125, 12.

