Supported by:

Reproducible workflows with R and GitHub

Ceres Barros

July 12th, 2023
2023 MacroBrum
Birmingham UK

https://ceresbarros.github.io/reproducible-workflowsWS

Outline

1. The importance of repeatability, reproducibility, reusability and transparency – R^{3}T

2. General guidelines

3. A working example in R and GitHub

Outline

1. The importance of repeatability, reproducibility, reusability and transparency – R³T

2. General guidelines

3. A working example in R and GitHub

Repeatability, reproducibility, reusability and transparency R³T

What?

Repeatability ≠ Reproducibility ≠ Reusability

Repeatability, reproducibility, reusability and transparency R³T

What?

Repeatability ≠ Reproducibility ≠ Reusability

agreement of results obtained by the <u>same individual</u> using <u>same methods</u>
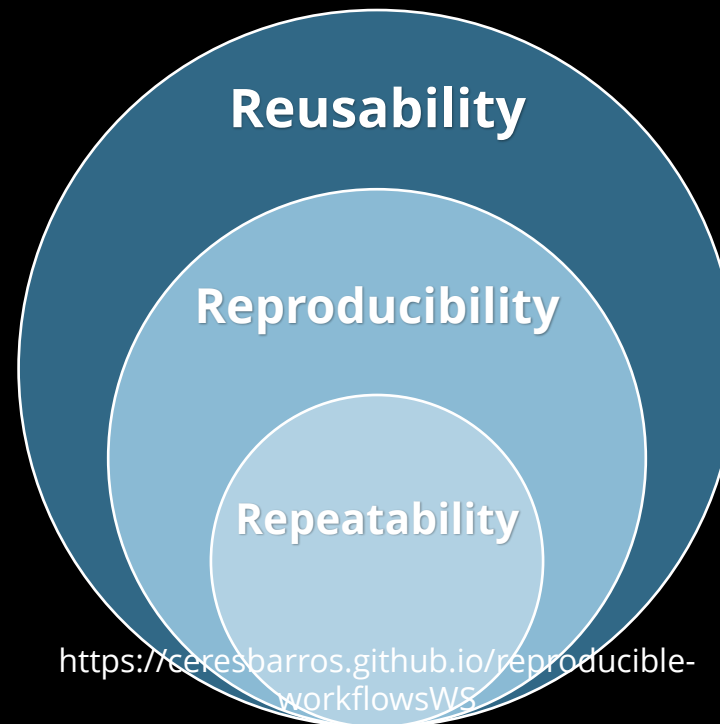
Repeatability, reproducibility, reusability and transparency
R³T
What?

Repeatability ≠ Reproducibility ≠ Reusability

agreement of results obtained by
the same individual using same
methods

agreement of results obtained by
two individuals/groups using
same methods

Repeatability, reproducibility, reusability and transparency
R³T
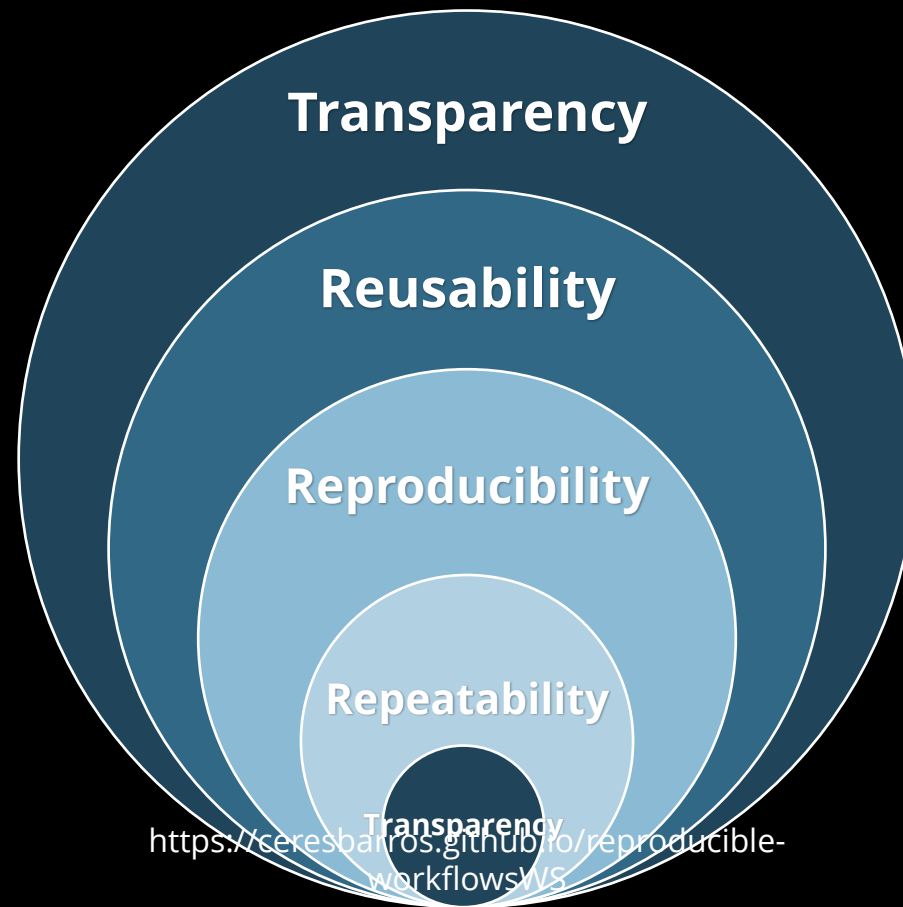### What?
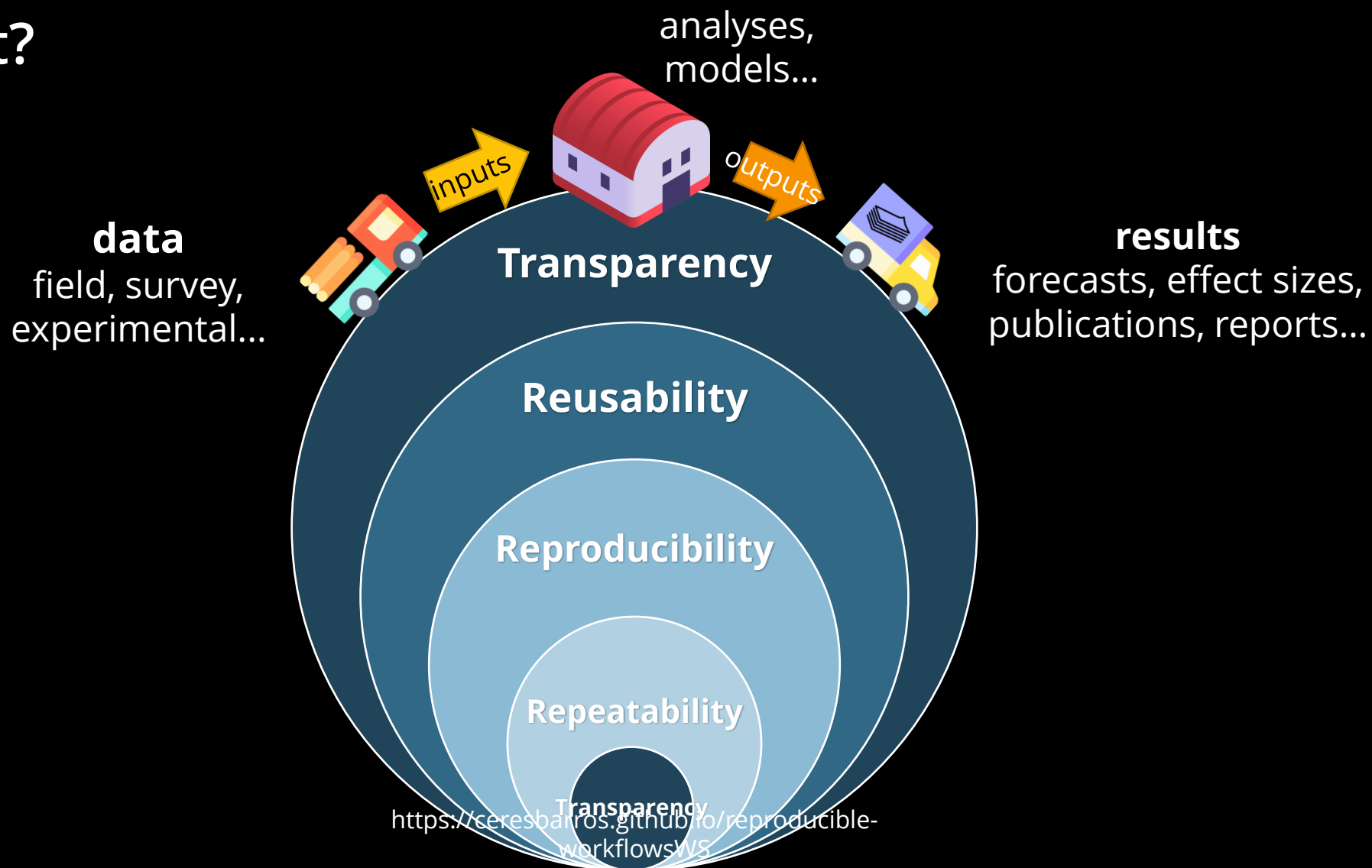
Repeatability ≠ Reproducibility ≠ Reusability

agreement of results obtained by the <u>same individual</u> using <u>same methods</u>

ability to <u>re-use the same methods</u> in a <u>different context</u> (e.g. new study area)

agreement of results obtained by <u>two individuals/groups</u> using <u>same methods</u>

Repeatability, reproducibility, reusability and transparency R³T
What?

Reusability

Reproducibility

Repeatability

https://ceresbarros.github.io/reproducible-workflowsWS

Repeatability, reproducibility, reusability and transparency R³T
What?

Transparency

Reusability

Reproducibility

Repeatability

Transparency

https://ceresbarros.github.io/reproducible-workflowsWS

Repeatability, reproducibility, reusability and transparency R³T
What?

analyses, models...

inputs

outputs

data
field, survey, experimental...

Transparency

Reusability

Reproducibility

Repeatability

Transparency

results
forecasts, effect sizes, publications, reports...

https://ceresbarros.github.io/reproducible-workflowsWS

Repeatability, reproducibility, reusability and transparency R³T

Why?

Trust

Repeatability, reproducibility, reusability and transparency R³T
Why?

Trust

Benchmarking & meta-analyses

Repeatability, reproducibility, reusability and transparency
R³T
Why?

Trust

Benchmarking &
meta-analyses

Building-on & improving
analyses/models/workflows

Repeatability, reproducibility, reusability and transparency
R³T
How?

Depends on **context**

- Project type and size
- Purpose
- Audience

https://ceresbarros.github.io/reproducible-workflowsWS

Repeatability, reproducibility, reusability and transparency R³T
How?

Depends on **context**

- Project type and size

- Purpose

- Audience

Data (both input and output) types
Input and output management
Suitable workflow

https://ceresbarros.github.io/reproducible-workflowsWS

Repeatability, reproducibility, reusability and transparency
R³T
How?

Methods in Ecology and Evolution

RESEARCH ARTICLE

Realising the Promise of Large Data and Complex Models

Empowering ecological modellers with a PERFICT workflow: Seamlessly linking data, parameterisation, prediction, validation and visualisation

Ceres Barros[1] | Yong Luo[1,2,3] | Alex M. Chubaty[4] | Ian M. S. Eddy[2] | Tatiane Micheletti[1] | Céline Boisvenue[1,2] | David W. Andison[5] | Steven G. Cumming[6] | Eliot J. B. McIntire[1,2]

https://ceresbarros.github.io/reprodu...
workflowsWS

Repeatability, reproducibility, reusability and transparency R³T
How?

Ecological (iterative) forecasting (continuous and integrated) workflow based on monitoring data
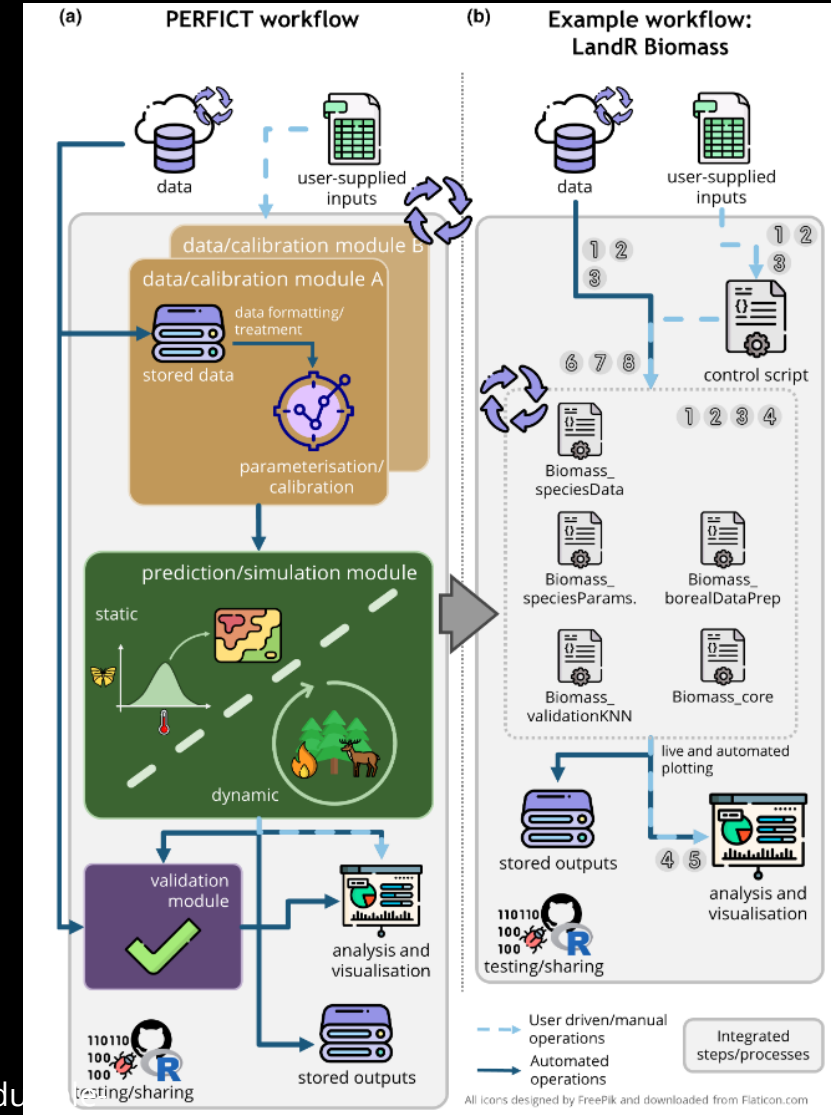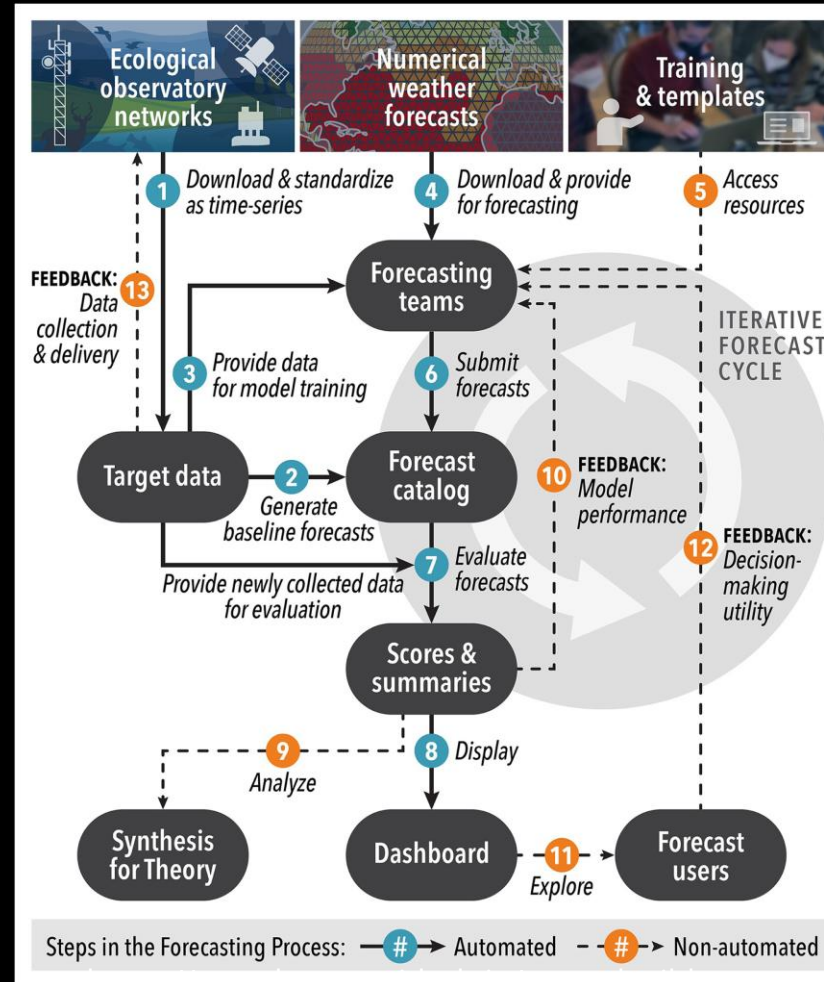
NEON Forecasting Challenge workflow Thomas et al. (2023)

https://ceresbarros.github.io/reproducible workflowsWS

Repeatability, reproducibility, reusability and transparency R³T
How?

engagement/education
point of view

R-shiny apps can be useful
for education, engaging
stakeholders/public and
delivering an interactive
product to end-users

https://vnijs.shinyapps.io/radiant/?SSUID=03eddd27f4

https://ceresbarros.github.io/reproducible-
workflowsWS
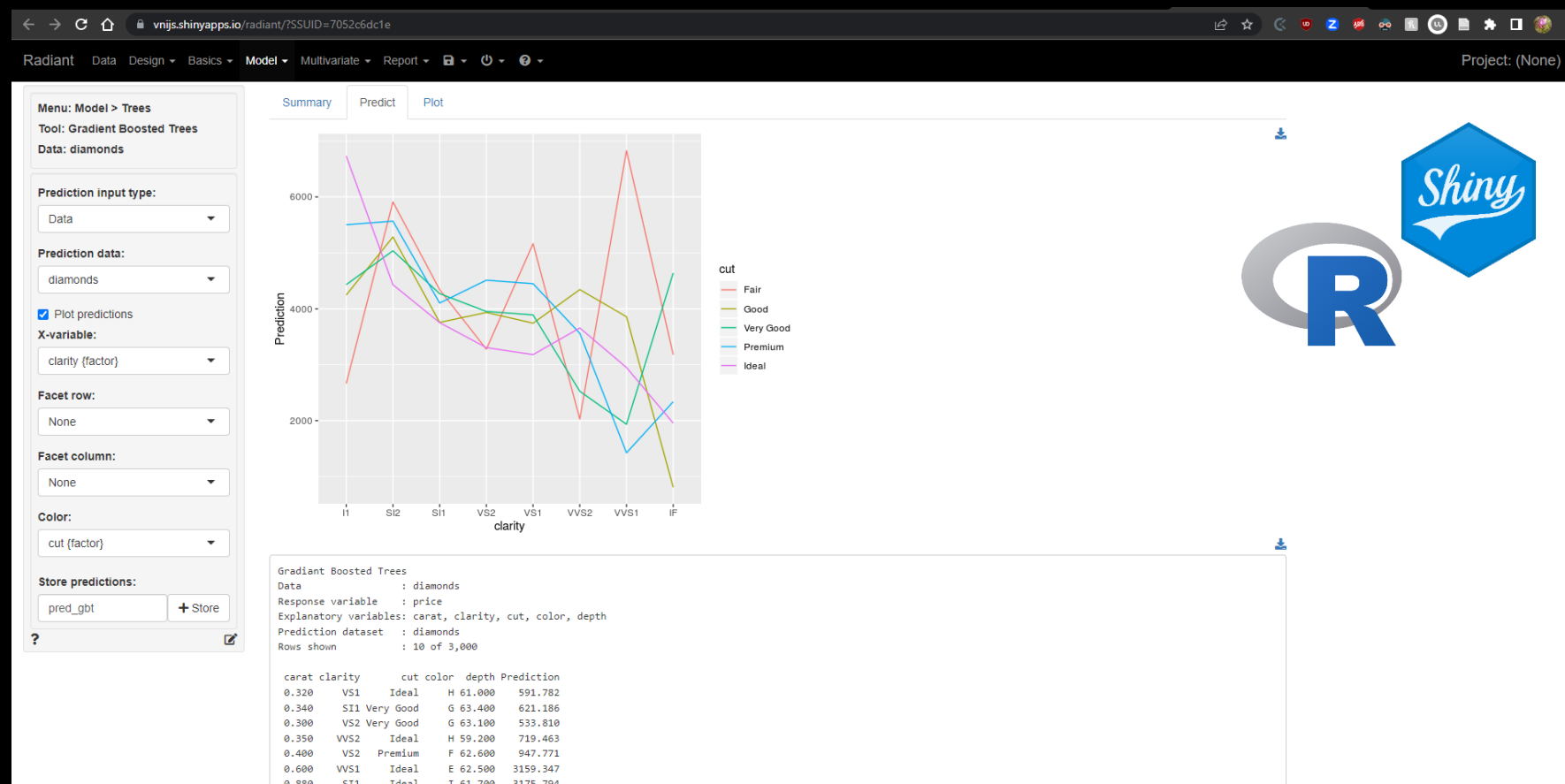
Repeatability, reproducibility, reusability and transparency R³T
How?

Most ecological research likely benefits from using a R³T approach, but the tools used to accomplish it can be varied
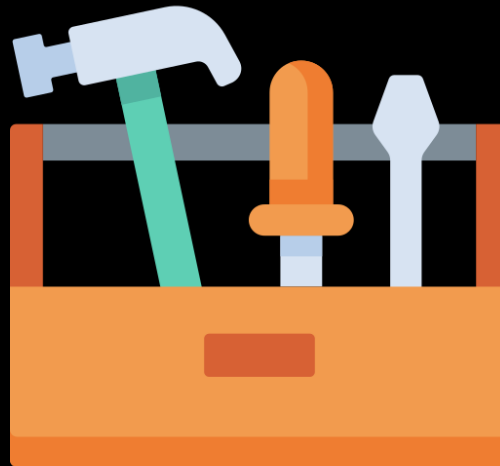
Repeatability, reproducibility, reusability and transparency R³T
How?

All steps, from processing *raw data* to producing *final figures* are integrated and automated*

Data is FAIR
(Wilkinson *et al.* 2016)

Final outputs can be repeated and are integrated in the reporting**

*as much as possible
https://certasbarros.github.io/reproducible-
**directly, or indirectly via links
workflowsWS

Repeatability, reproducibility, reusability and transparency R³T
How?

Self-contained

All steps, from processing *raw data* to producing *final figures* are integrated and automated*

Data is FAIR
(Wilkinson *et al.* 2016)

Final outputs can be repeated and are integrated in the reporting**

*as much as possible
**directly, or indirectly via links

https://aretxabarros.github.io/reproducible-workflowsWS

Outline

1. The importance of repeatability, reproducibility, reusability and transparency – R3T

2. General guidelines

3. A working example in R and GitHub

https://ceresbarros.github.io/reproducible-workflowsWS
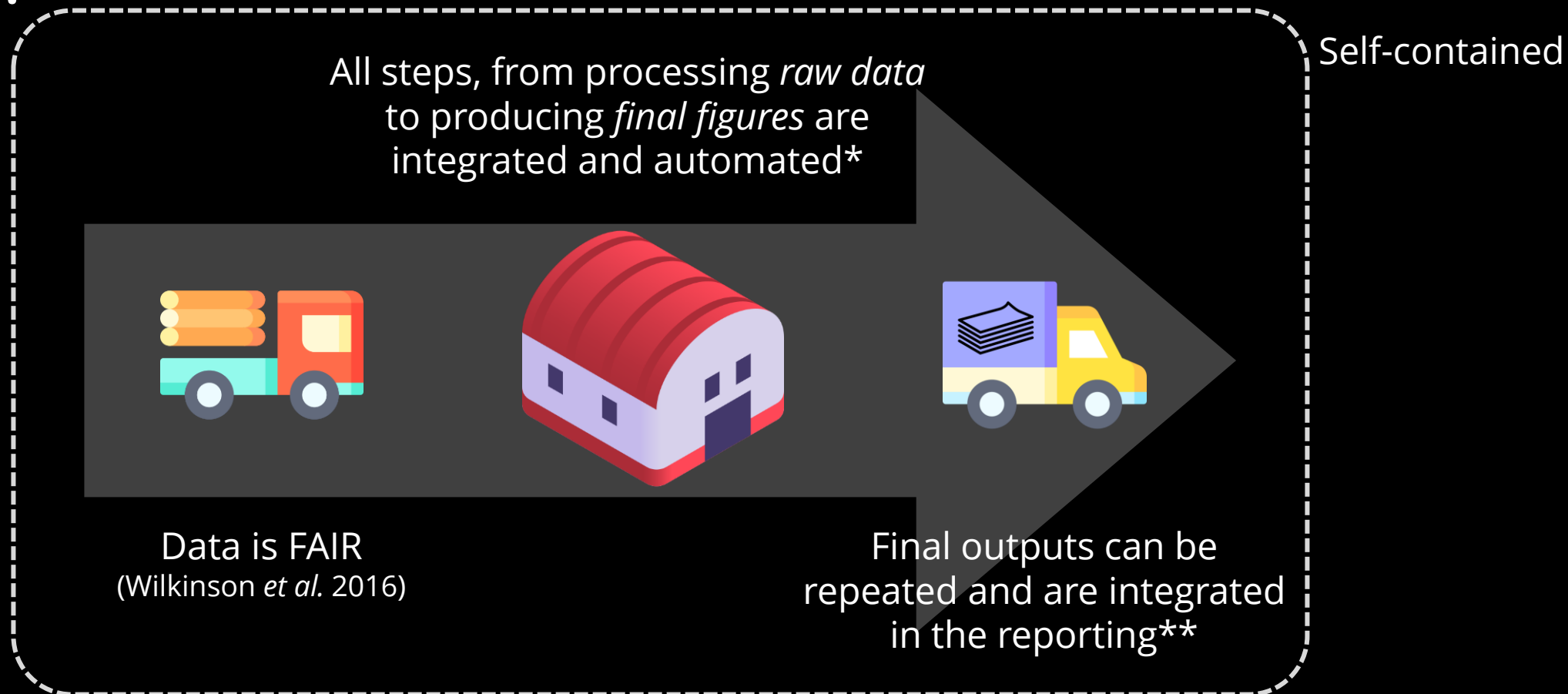
General guidelines
1. Scripting/executing the workflow

1.1. Script, script, script

- **Goal**: no "secret handshakes" + record all steps of an analysis
- <u>ALL steps</u> – this includes <u>package/library installation/loading</u> and <u>sourcing data</u>

DOComment your code

General guidelines
1. Scripting/executing the workflow

1.1. Script, script, script

1.2. Minimise software/languages used

- **Goal**: increase workflow robustness - fewer "moving parts", fewer "secret handshakes", fewer manual operations
- Interpreted languages (real-time user interaction)
    - R, Julia, Python…
- Compiled languages (pre-compiled programs)
    - C, C++, C#, Fortran,… Do you really need this?

General guidelines
1. Scripting/executing the workflow

1.1. Script, script, script

1.2. Minimise software/languages used

1.3. Modularise and "functionise" (!)

- **Goal:** code organisation/readability; easier propagation of code updates/changes
- Avoid loooooooooong scripts
- Break scripts into logical pieces
- Encapsulate code into functions, *especially* when used multiple times/in multiple places
- Consider "packaging" your functions.

General guidelines
1. Scripting/executing the workflow
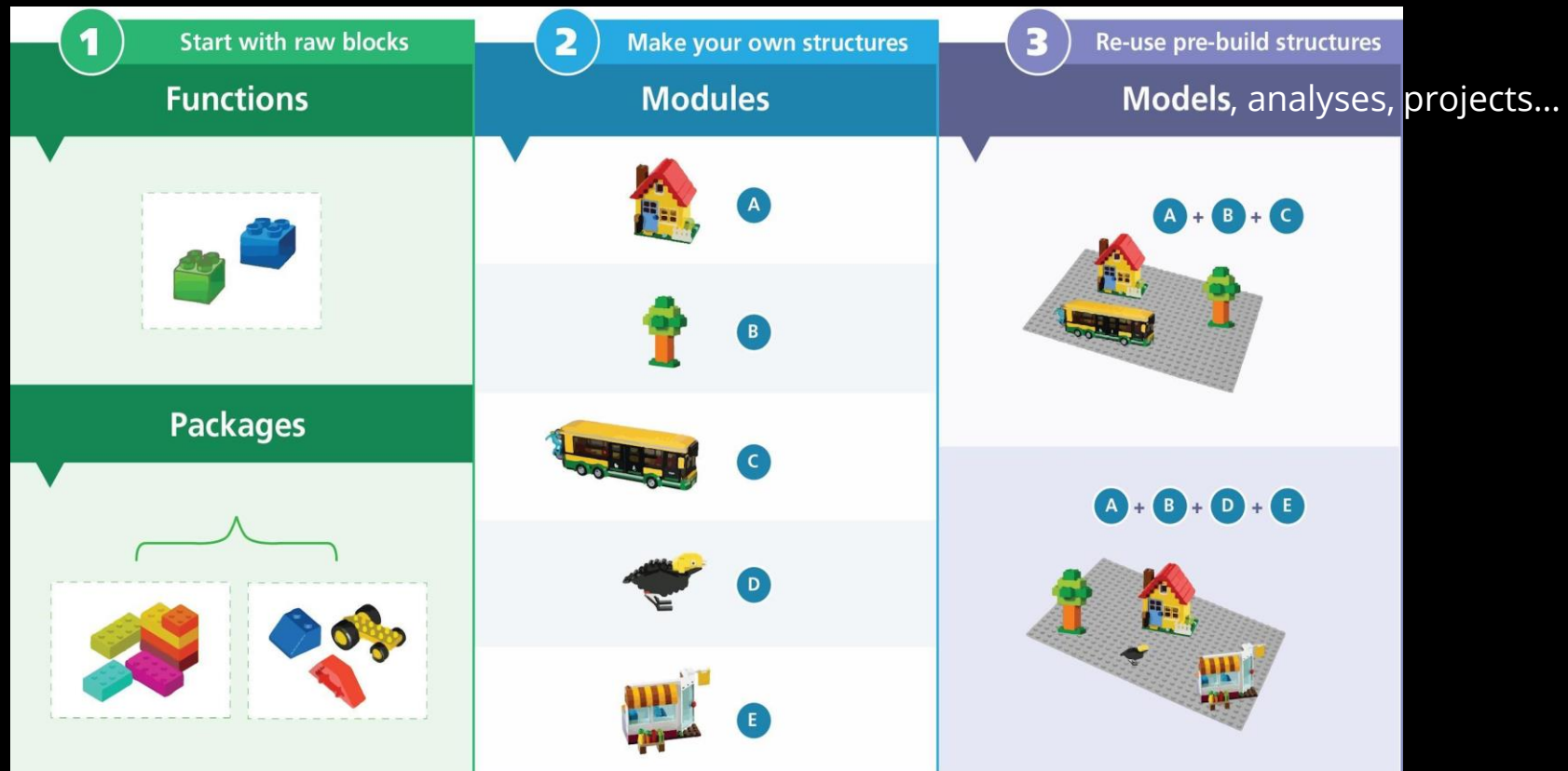
Functions and modules as key tools for R³T, but also for building integrated and continuous workflows
McIntire *et al.* (2022)

https://ceresbarros.github.io/reproducible-workflowsWS

General guidelines
1. Scripting/executing the workflow

1.1. Script, script, script

1.2. Minimise software/languages used

1.3. Modularise and "functionise" (!)

1.4. Centralise workflow in a single script

- **Goal:** no "secret handshakes" - all scripts are utilised in correct way/sequence
- Call/execute scripts/steps from central ("control") script
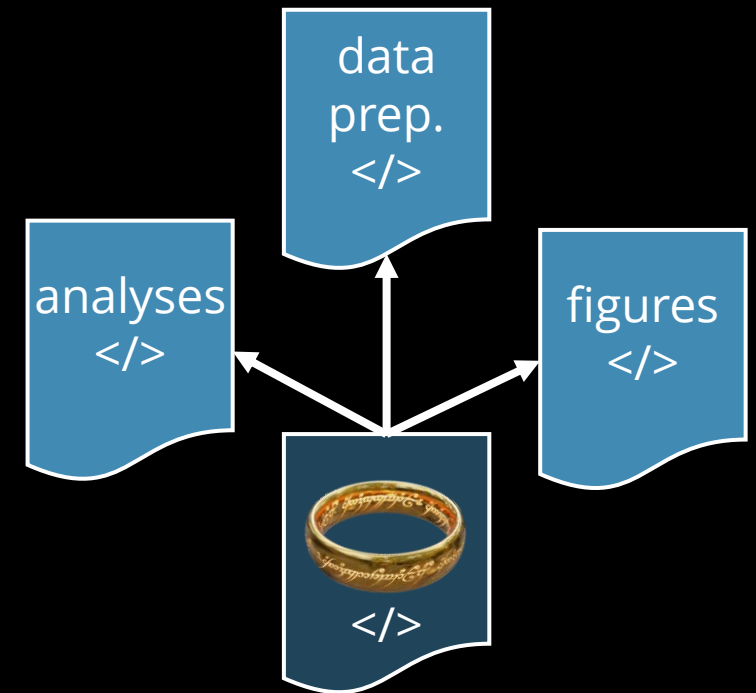
General guidelines
2. Project structure

2.1. Project-oriented workflows

- **Goal:** the entirely workflow can be re-run easily, and without changing code or files
- Choose a structure that is self-explanatory
- <u>Relative paths</u> *vs.* absolute paths
- Project-libraries

Source: https://ecoforecast.org/reproducible-forecasting-workflows/

Adapted from EFI
https://ceresbarros.github.io/reproducible-
https://ecoforecast.org/reproducible-forecasting-workflows/
workflows/5

General guidelines
2. Project structure

2.1. Project-oriented workflows

2.2. Self-contained workflows

- **Goal:** ensure reproducibility
- E.g. RStudio-projects
- Containerisation – encapsulates the whole system (even OS) – e.g. Docker

Self-contained

Adapted from EFI
https://ceresbarros.github.io/reproducible-
https://ecoforecast.org/reproducible-forecasting-workflows/
workflows/
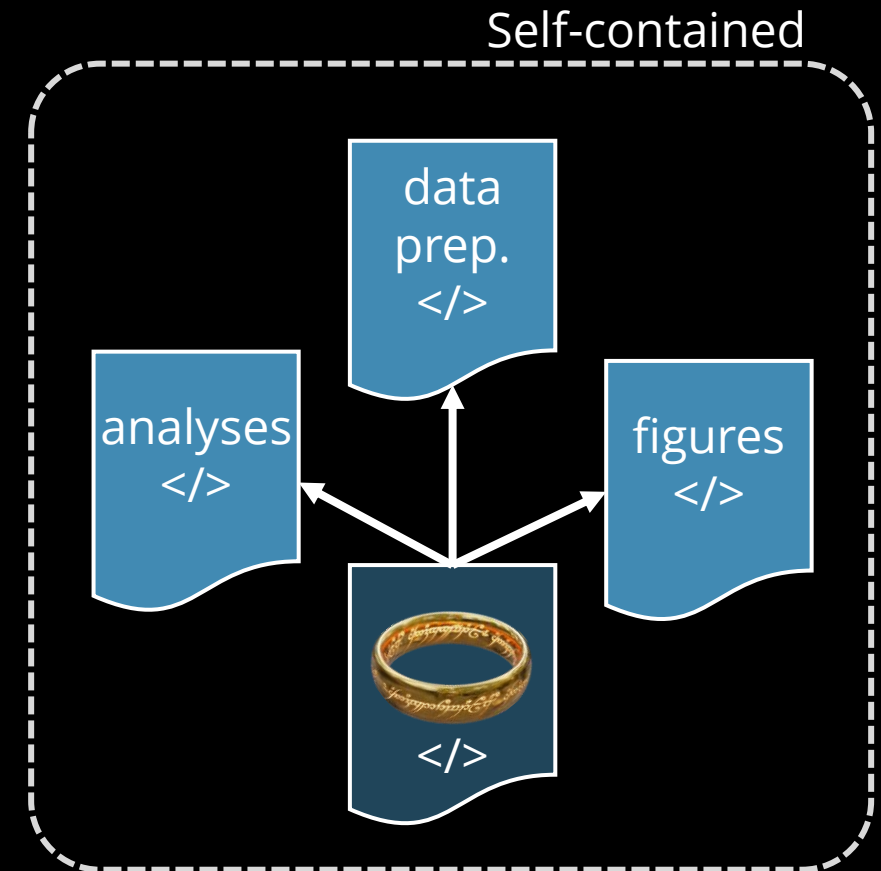
General guidelines
3. Project management

3.1. Version control

- **Goal:** <u>change tracking</u> in code/files + continuous and <u>collaborative</u> development
- Keeps a formal record of all changes
- Allows recovering old versions
- Allows keeping/working on multiple versions of the same code/project
- E.g. Git, CVS, SVN, …
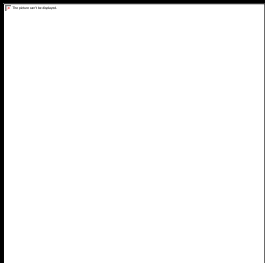
We'll come back to this!
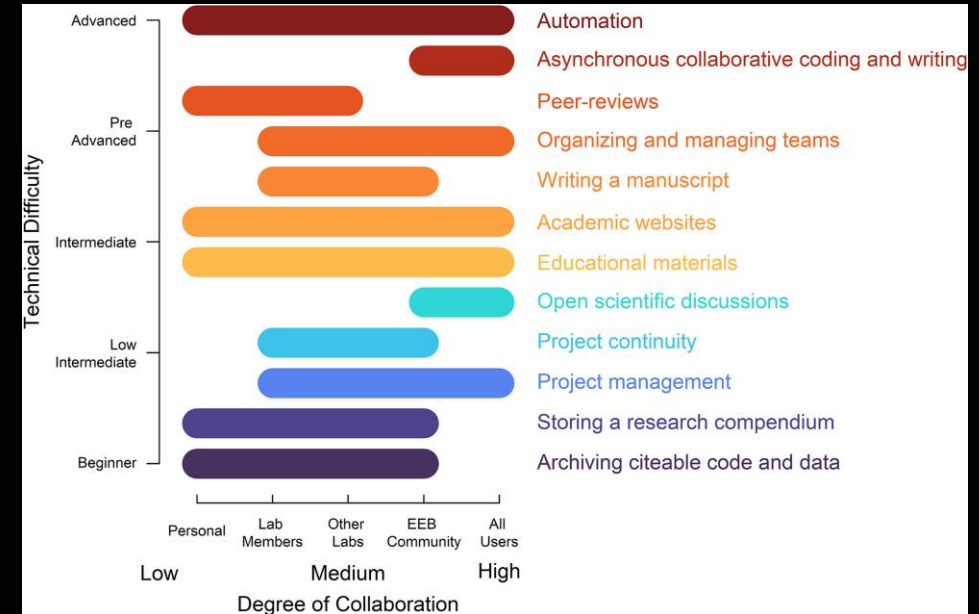
GitHub

GitHub is a multifaceted tool that can be appropriate to manage, track and collaborate on projects for various purposes and at various levels of complexity.
(Braga *et al*. 2023)

General guidelines
3. Project management

3.1. Version control

3.2. Integrated testing

- **Goal:** enhance code robustness and longevity
- Not always necessary, but always a good idea ;)
- Integration tests *vs* unit tests *vs* assertions
- Manual *vs* automated execution
- E.g.
  - `testthat` in R (unit tests)
  - simple code/object checks (assertions)
  - GitHub Actions and Travis CI for automated testing – all types.

General guidelines
4. Literate programming

4.1. Integrate code and reporting/publication

- **Goal**: establishing explicit links between report/publication, data and analyses
- Integrates code and text in a single file
- Enhances transparency/reproducibility of reported outputs.
- E.g.
  - RMarkdown, Quarto – static or interactive; multiple languages in a single file
  - Jupyter – interactive; single language at a time (Julia, R or Python)

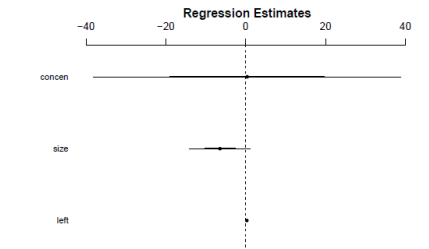A Pandoc Markdown Article Starter and Template *

Steven V. Miller   *Clemson University*

Figure 1: A Coefficient Plot

Outline

1. The importance of repeatability, reproducibility, reusability and transparency – R3T

2. General guidelines

3. A working example in R + RStudio + GitHub

Shall we try this?

What we will cover:

Project structure and management
- Version control – using GitHub and GitKraken
- Self-contained workflows – using R and Rstudio

Scripting/executing the workflow
- Script, script, script
- Modularise and "functionise" (!)
- Centralise workflow in a single script

Adapted from EFI
https://ceresbarros.github.io/reproducible-
https://ecoforecast.org/reproducible-forecasting-workflows/
workflows/

Shall we try this?

What we will cover:

Project structure and management
- Version control – using GitHub and GitKraken
- Self-contained workflows – using R and Rstudio

Scripting/executing the workflow
- Script, script, script
- Modularise and "functionise" (!)
- Centralise workflow in a single script

The order is variable;
it depends on the stage of the project and your own preference

Tools used in each step can also vary

*you can do this in...

1. Create a repository for your project

Assuming you already have an account on GitHub.com...

Create a repo

1. Create a repository for your project

Assuming you already have an account on GitHub.com...

Or fork someone else's

2. Create a self-contained project

In RStudio, go to
File > New Project... > Version Control > Git
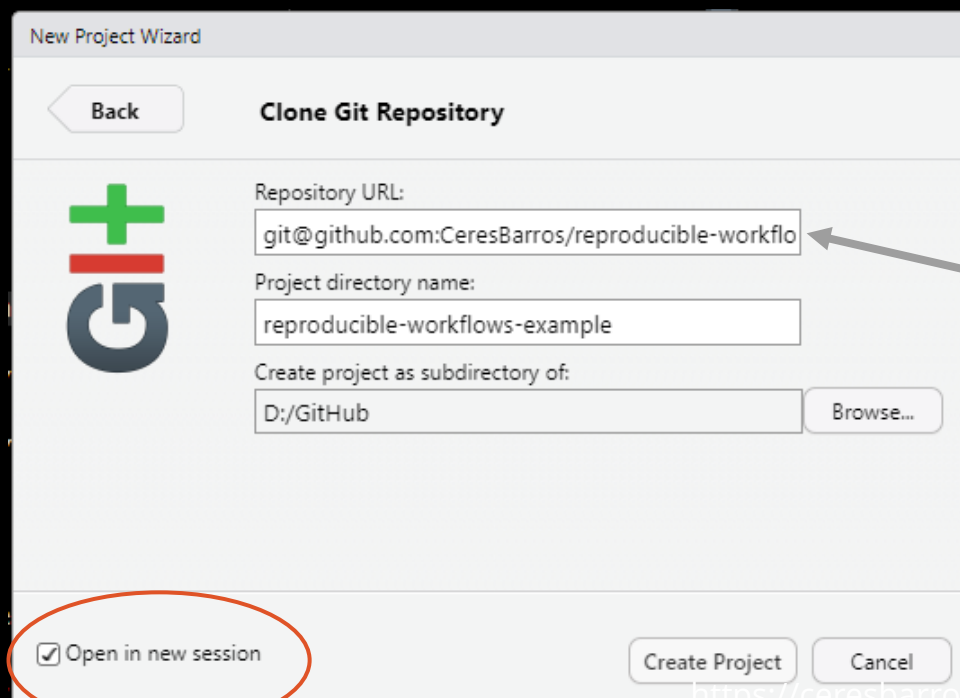
Get repo URL from
GitHub.com/your_username/your_repo

https://ceresbarros.github.io/reproducible-workflowsWS
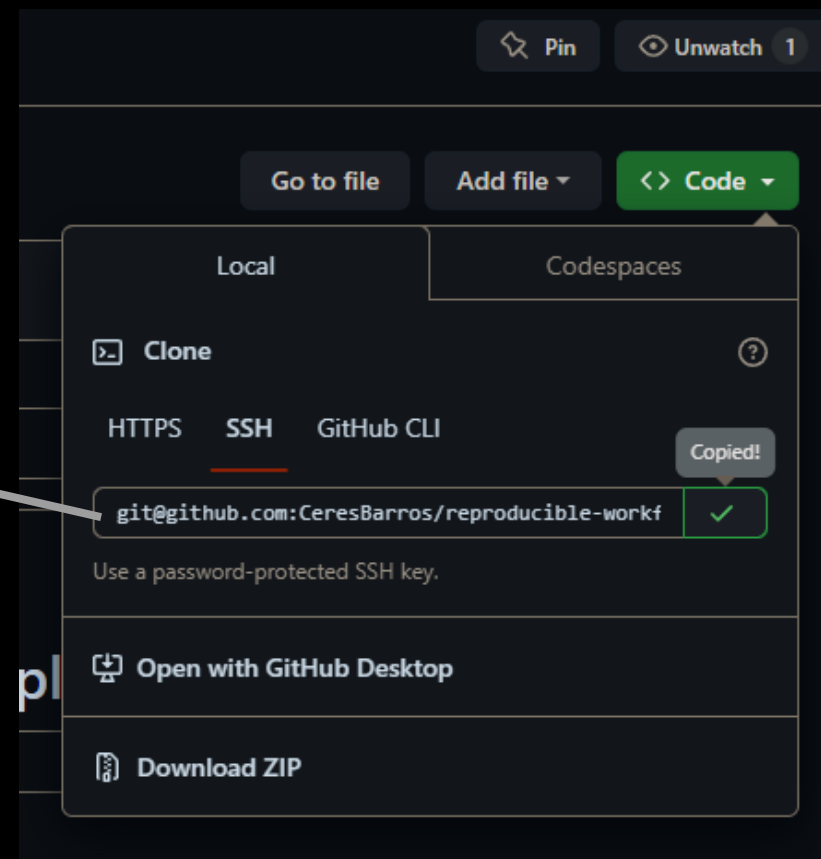
2. Create a self-contained project

If you already have a project folder (e.g. created by GitKraken, or from an existing project):

In RStudio, go to
File > Existing Directory

New Project Wizard

Back   **Create Project from Existing Directory**

Project working directory:

D:/GitHub/reproducible-workflows-example   Browse...

☑ Open in new session   Create Project   Cancel

https://ceresbarros.github.io/reproducible-workflowsWS
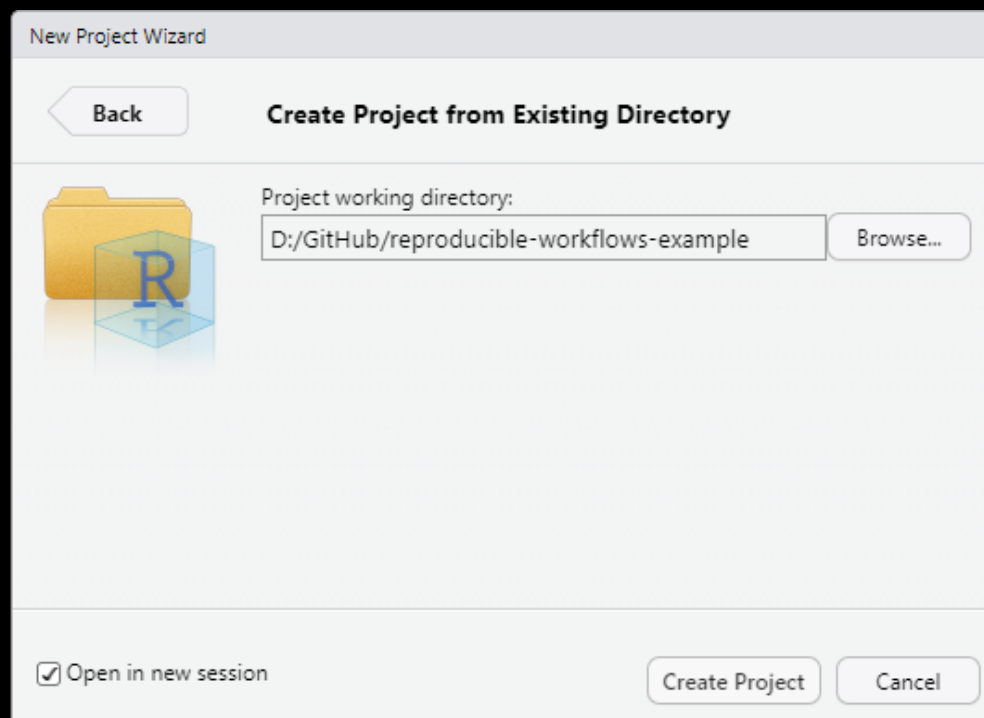
2. Create a self-contained project

You can now manage your Git repo
from RStudio
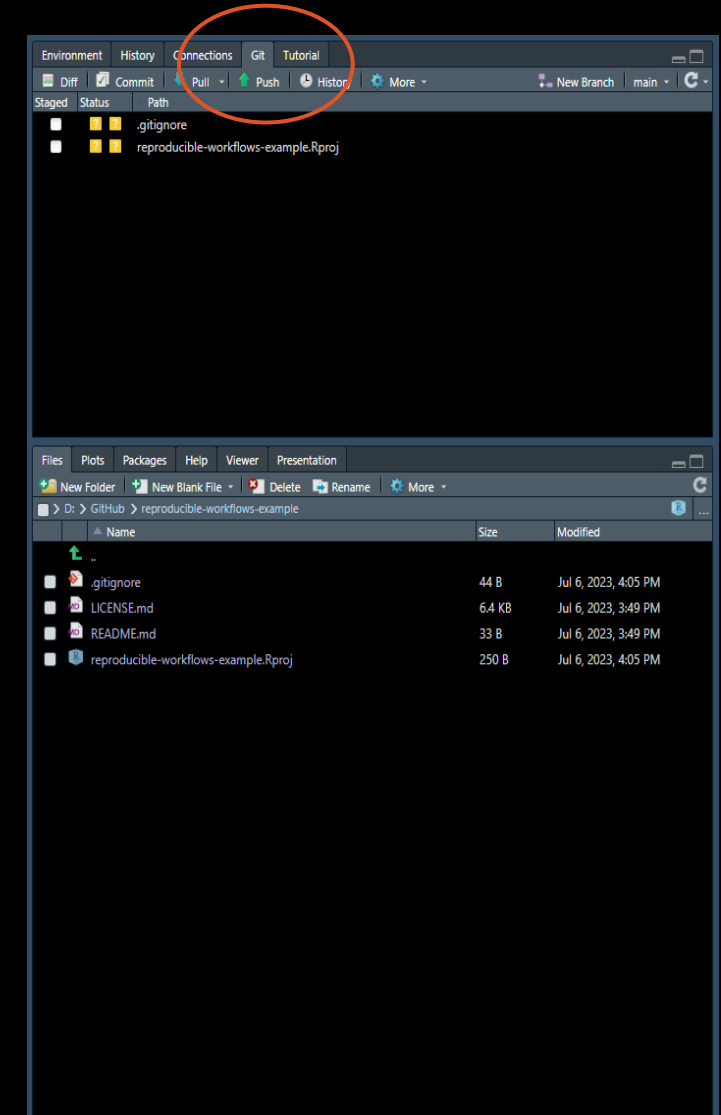
2. Create a project

You can now manage your Git repo from RStudio,
GitKraken
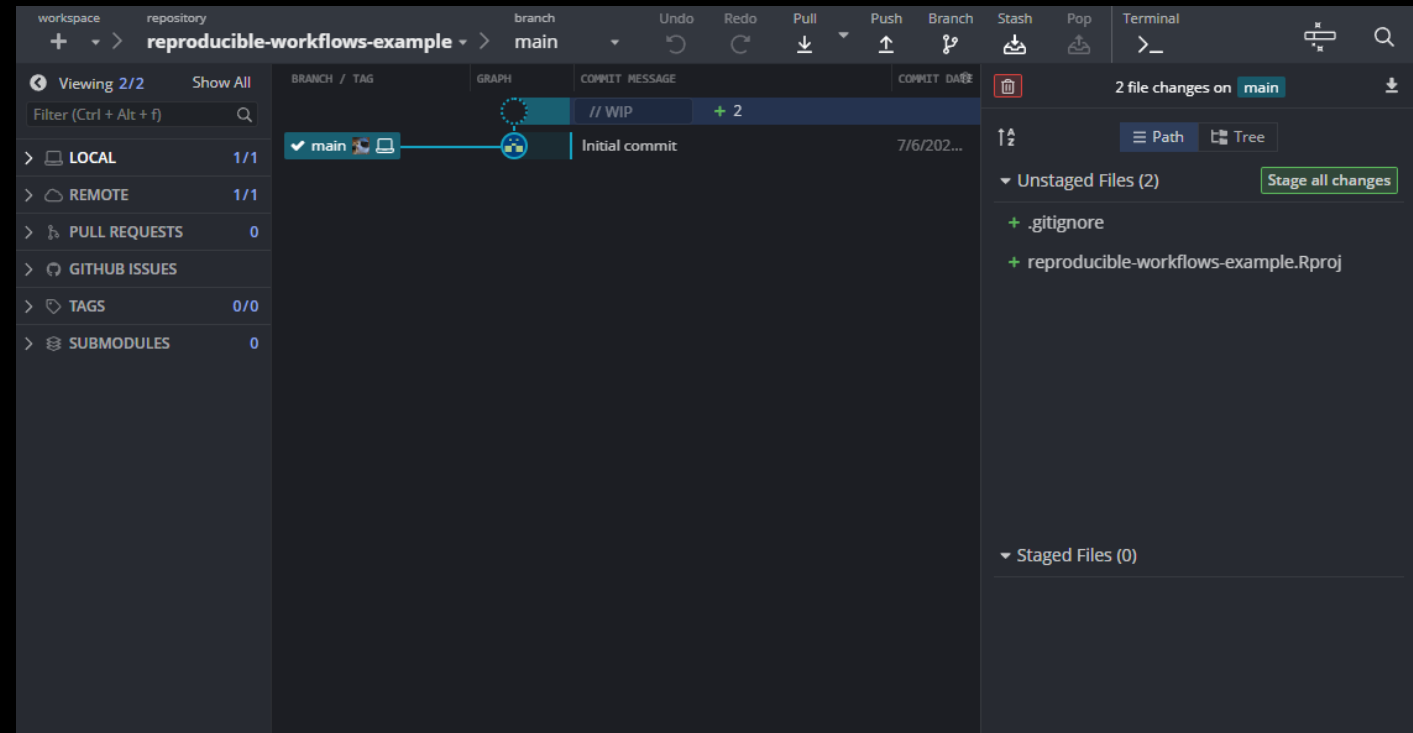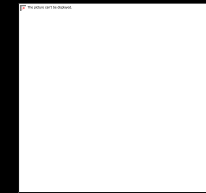
2. Create a project

You can now manage your Git repo from RStudio,
GitKraken,
or even the command-line
(e.g., git bash for Windows)

W-VIC-A127584+cbarros@W-VIC-A127584 MINGW64 /d/GitHub/reproducible-workflows-example (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore
        reproducible-workflows-example.Rproj

nothing added to commit but untracked files present (use "git add" to track)

W-VIC-A127584+cbarros@W-VIC-A127584 MINGW64 /d/GitHub/reproducible-workflows-example (main)
$
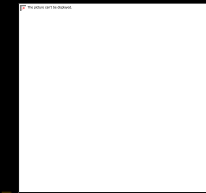
3. Version control
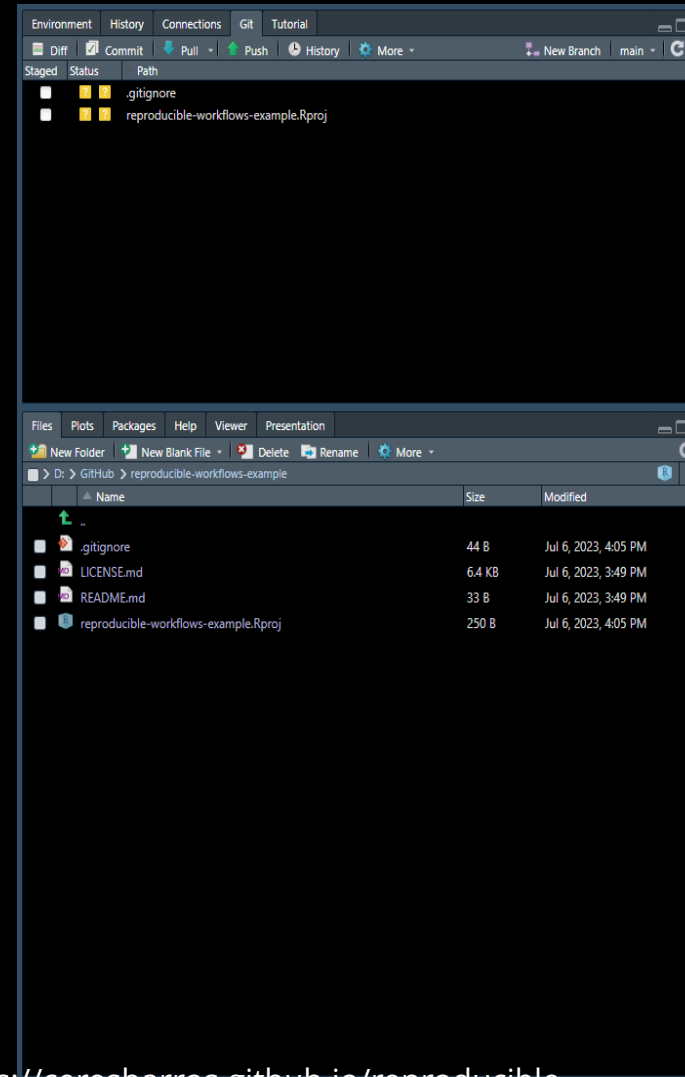
✓ keep master/main branch clean; develop in other branches

✓ small, incremental, commits

✓ *.gitignore* **sensitive and large files** – think about data storage

✓ pull first, push after

https://ceresbarros.github.io/reproducible-workflowsWS

Source: https://www.earthdatascience.org/workshops/intro-version-control-git/pull-request/
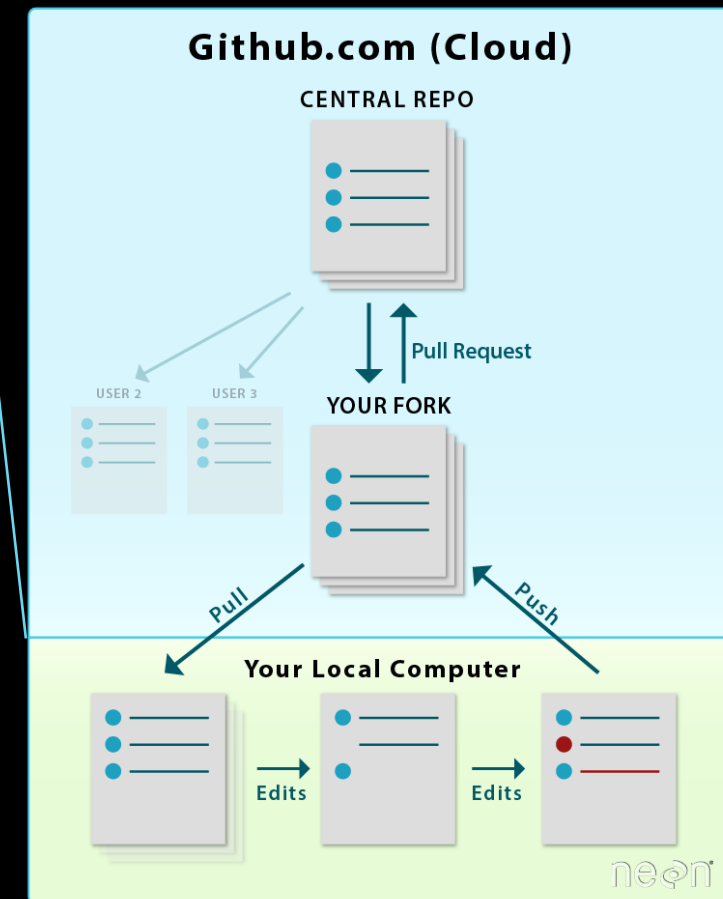
3. Example of a reproducible workflow in R, RStudio and GitHub

<https://github.com/CeresBarros/reproducible-workflows-example>

https://ceresbarros.github.io/reproducible-workflowsWS

Useful resources

Peer-reviewed:

- Barros, C., Luo, Y., Chubaty, A.M., Eddy, I.M.S., Micheletti, T., Boisvenue, C., *et al.* (2023). Empowering ecological modellers with a PERFICT workflow: Seamlessly linking data, parameterisation, prediction, validation and visualisation. *Methods Ecol Evol*, 14, 173–188.

- Braga, P.H.P., Hébert, K., Hudgins, E.J., Scott, E.R., Edwards, B.P.M., Sánchez Reyes, L.L., *et al.* (2023). Not just for programmers: How GitHub can accelerate collaborative and reproducible research in ecology and evolution. *Methods in Ecology and Evolution*, 14, 1364–1380.

- Brousil, M.R., Filazzola, A., Meyer, M.F., Sharma, S. & Hampton, S.E. (2023). Improving ecological data science with workflow management software. *Methods in Ecology and Evolution*, 14, 1381–1388.

- Ellison, A.M. (2010). Repeatability and transparency in ecological research. *Ecology*, 91, 2536–2539.

- McIntire, E.J.B., Chubaty, A., Cumming, S., Andison, D., Barros, C., Boisvenue, C., *et al.* (2022). PERFICT: a Re-imagined Foundation for Predictive Ecology. *Ecology Letters*.

- Thomas, R.Q., Boettiger, C., Carey, C.C., Dietze, M.C., Johnson, L.R., Kenney, M.A., *et al.* (2023). The NEON Ecological Forecasting Challenge. *Frontiers in Ecology and the Environment*, 21, 112–113.

- Wilkinson, M.D., Dumontier, M., Aalbersberg, Ij.J., Appleton, G., Axton, M., Baak, A., *et al.* (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Sci Data*, 3, 160018.

Reproducible workflows:

- Ecological Forecasting Initiative. (2020). *Reproducible Forecasting Workflows*. *Ecological Forecasting Initiative*. Available at: https://ecoforecast.org/reproducible-forecasting-workflows/. Last accessed 6 July 2023.

- **The Practice of Reproducible Research** (http://www.practicereproducibleresearch.org/)

- **R Markdown: The Definite Guide** (https://bookdown.org/yihui/rmarkdown/)

- **R Markdown cheat sheets** (https://www.rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf)

- **GitHub Quickstart** (https://docs.github.com/en/get-started/quickstart/hello-world)

Software:

RStudio

R

GitKraken

Git

References

GIFs/Images:

- https://www.reddit.com/r/gifs/comments/4a3exq/cat_typing_a_document_on_laptop/

- https://en.wikipedia.org/wiki/One_Ring#/media/File:One_Ring_Blender_Render.png

- All icons designed by Freepik and downloaded from Flaticon.com

Literature

- Barros, C., Luo, Y., Chubaty, A.M., Eddy, I.M.S., Micheletti, T., Boisvenue, C., *et al.* (2023). Empowering ecological modellers with a PERFICT workflow: Seamlessly linking data, parameterisation, prediction, validation and visualisation. *Methods Ecol Evol*, 14, 173–188.

- Braga, P.H.P., Hébert, K., Hudgins, E.J., Scott, E.R., Edwards, B.P.M., Sánchez Reyes, L.L., *et al.* (2023). Not just for programmers: How GitHub can accelerate collaborative and reproducible research in ecology and evolution. *Methods in Ecology and Evolution*, 14, 1364–1380.

- Ecological Forecasting Initiative. (2020). *Reproducible Forecasting Workflows*. *Ecological Forecasting Initiative*. Available at: https://ecoforecast.org/reproducible-forecasting-workflows/. Last accessed 6 July 2023.

- McIntire, E.J.B., Chubaty, A., Cumming, S., Andison, D., Barros, C., Boisvenue, C., *et al.* (2022). PERFICT: a Re-imagined Foundation for Predictive Ecology. *Ecology Letters*.

- Thomas, R.Q., Boettiger, C., Carey, C.C., Dietze, M.C., Johnson, L.R., Kenney, M.A., *et al.* (2023). The NEON Ecological Forecasting Challenge. *Frontiers in Ecology and the Environment*, 21, 112–113.

- Wilkinson, M.D., Dumontier, M., Aalbersberg, Ij.J., Appleton, G., Axton, M., Baak, A., *et al.* (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Sci Data*, 3, 160018.