Putting the $R$ into Reproducible Research

Directors Cut

Dr Anna Krystalli
University of Sheffield RSE

2020-12-16

UCL Tech Social
me: Dr Anna Krystalli

- **Research Software Engineer**, *University of Sheffield*
  - twitter [@annakrystalli](https://twitter.com/annakrystalli)
  - github [@annakrystalli](https://github.com/annakrystalli)
  - email a.krystalli[at]sheffield.ac.uk

- **Editor** [rOpenSci](https://ropensci.org)

- **Core Team:** [ReproHack](https://reprohack.io)

Motivation
Reproducibility has the potential to serve as a minimum standard for judging scientific claims when full independent replication of a study is not possible.

Reproducible Research in Computational Science ROGER D. PENG, SCIENCE 02 DEC 2011 : 1226-1227
Is code and data enough?

1. 
2. 
3. 
4. 
5. 
6. 

slide:  Karthik Ram: rstudio::conf 2019 talk
R for Open Reproducible Research

A whistle-stop tour of tools, practices and conventions in R for more:

- Reproducible
- Robust
- Transparent
- Reusable
- **Shareable** research materials
Project management
Rstudio Projects

Use Rstudio projects to keep materials associated with a particular analysis together

- **Self contained** and **portable**
- **Working directory set to root** of project on launch
- **Fresh session** everytime the project is launched

See Jenny Bryan's post on [project oriented workflows](#) for more details

File > New Project > New Directory

![Rstudio Projects](#)
R version 3.6.0 (2019-04-26) -- "Planting of a Tree"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

   Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and 'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or 'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

Hey Anna, welcome back, time to Rrrrrrrrock!
Use 📦 here to create robust relative paths

- Robust paths relative to project root
  - portable
  - independent of:
    - working directory
    - source code location

```r
here::here()
```

```
## [1] "/Users/Anna/Documents/workflows/talks"
```

```r
here::here("data", "summaries.csv")
```

```
## [1] "/Users/Anna/Documents/workflows/talks/data/summaries.csv"
```
Dependency management

**Minimal approach**

include an `install.R` script

```r
install.packages("dplyr")
install.packages("purrr")
```

**Most robust approach**

use 📦 `renv` *(previously `packrat`)*

- Create and manage a per project library of packages
- initialise during project set up

*will revisit later on*
Use drake to orchestrate your workflows

make plan

```r
plan <- drake::drake_plan(
  raw_data = readr::read_csv(here::here("data", "iris.csv")),
  data = raw_data %>%
    dplyr::mutate(Species =forcats::fct_inorder(Species)),
  fit = lm(Sepal.Width ~ Petal.Width + Species, data))
```

Scale the work you need.  
Skip the work you don't. 
See evidence of reproducibility.
view plan

```
plan

## # A tibble: 3 x 2
## target command
## <chr>  <expr_lst>
## 1 raw_data readr::read_csv(here::here("data", "iris.csv"))
## 2 data raw_data %>% dplyr::mutate(Species = forcats::fct_inorder(Species))
## 3 fit lm(Sepal.Width ~ Petal.Width + Species, data)
```

re-execute plan

```
drake::make(plan)
```

## All targets are already up to date.
inspect targets

drake::readd(fit)

##
## Call:
## lm(formula = Sepal.Width ~ Petal.Width + Species, data = data)
##
## Coefficients:
##   (Intercept) Petal.Width Speciesversicolor Speciesvirginica
##     3.236       0.781      -1.501       -1.844
drake::vis_drape_graph(drape::drape_config(plan))

Dependency graph
Version Control
Version Control

What is it? 🤔

The **management of changes** to documents, computer programs, large web sites, and other collections of information.

**Git** 🍃 git

Open source (free to use) **Version control software.**

**GitHub** 📖 GitHub

A **website** ([https://github.com/](https://github.com/)) that allows you to **store your Git repositories online** and makes it easy to collaborate with others.
Why use them in research?

Exhibit A

Exhibit B

Image: xkcd CC BY-NC 2.5

Image: Jorge Cham www.phdcomics.com
Git, Github & Rstudio

Before: git only through the terminal 😞

Now: Rstudio + `usethis` 📦 == ❤ Git & GitHub 😎
Configure git & GitHub

Configure git

Check your configuration

usethis::git_sitrep()

Set your configuration

Use your github username and the email you used to sign-up on GitHub

usethis::use_git_config(
  user.name = "Jane",
  user.email = "jane@example.org")
Configure GitHub authentication

Get GITHUB Personal Authorisation Token

usethis::browse_github_pat()
Store in `.Renviron` file

```r
usethis::edit_r_environ()
```
Initialise Rstudio project with Git by just checking a box!

It's now a repository

Forgot to? use `usethis::use_git()`
Git panel

Integrated graphical user interface
Git Rstudio workflow

view file status

<table>
<thead>
<tr>
<th>Environment</th>
<th>History</th>
<th>Connections</th>
<th>Git</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Staged**
  - M modified-file.R

stage files

<table>
<thead>
<tr>
<th>Environment</th>
<th>History</th>
<th>Connections</th>
<th>Git</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Staged**
  - M modified-file.R
  - A new_file.R

commit changes

<table>
<thead>
<tr>
<th>Changes</th>
<th>History</th>
<th>master</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Stage**
  - modified-file.R
- **Commit**
  - new_file.R

<table>
<thead>
<tr>
<th>Commit message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add x calculation</td>
</tr>
</tbody>
</table>

Commit
Create repo

usethis::use_github(protocol = "https")

Push further changes
Anatomy of a GitHub Repo

- **README**: Explain what your project is, and how to use it.
  - `usethis::use_readme_md()`
  - `usethis::use_readme_rmd()`

- **LICENSE**: Without a licence, the contents of the repository are technically closed.
  - Examples licence: `usethis::use_mit_license(name = "Anna Krystalli")`
  - `?licenses`: details of functions available to generate licenses
  - [https://choosealicense.com/](https://choosealicense.com/) help on choosing a licence.

- **CONTRIBUTING.md**: guidelines for contributors.
  - `usethis::use_tidy_contributing()` provides a relatively strict but instructive template

- **CODE_OF_CONDUCT.md**: set the tone for discourse between contributors.
  - `use_code_of_conduct()`
GitHub issues

use GitHub issues to plan, record and discuss tasks.

issues

- y axis radar plot
- P_adj value wilcoxon test
- Add documentation of get &, set, group & vars functions
- Publication strategy
- Anonymise variable names in examples (and tests?)
- Add significance asterisks to plots
- Collate outputs into one file

projects

- Collate outputs into one file
- Anonymise variable names in examples (and tests?)
- Publication strategy
- Determine which arguments of each statistic test should be available through map function
- Add ability to change metadata columns with variable labels
- Parametric version of resampling, standard deviation in summary
- Ability to choose between parametric/non-parametric testing
- Add significance asterisks to plots
- Determine which arguments of each statistic test should be available through map function
- Add ability to change metadata columns with variable labels
Literate programming with Rmarkdown
Literate programming

Programming paradigm first introduced by Donald E. Knuth.

Treat program as literature to be understandable to human beings

- focus on the logic and flow of human thought and understanding
- single document to integrate data analysis (executable code) with textual documentation, linking data, code, and text
Literate programming in R

Rmarkdown (.Rmd) integrates:

- a **documentantion** language (.md)
- a **programming** language (R)
- functionality to "knit" them together through 📦 knitr

features

- ✓ provides a framework for writing narratives around code and data
- ✓ Code re-run in a clean environment every time the document is "knit"
Rmarkdown outputs
File > New File > RMarkdown... > Document

---

```r
# My report
author: "Anna Krystalli"
date: "14/05/2019"
output: html_document
---

```r
r setup, include=FALSE
```n
knit: opts_chunk(set(choo - TRUE)

---

# R Markdown

R 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>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```r
r cars
```n
`, `summary(cars)

---

## Including Plots

You can also embed plots, for example:

```r
r pressure, echo=FALSE
```n
`, `plot(pressure)

---

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

---

My report

Anna Krystalli

14/05/2019

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>.

When you click the Knit button, a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```r
summary(cars)
```n

## Including Plots

You can also embed plots, for example:

![Graph](image.png)
Applications in research

Rmd documents can be useful for a number of research related long form documentation materials:

- Documentation of code & data (eg DataMaid)
- Electronic Notebooks
- Supplementary materials
- Reports
- Papers
Publish to the web for free!

- **RPubs**: Publish rendered rmarkdown documents on the web with the click of a button [http://rpubs.com/](http://rpubs.com/)

- **GitHub**: Host your site through [gh-pages](https://github.com) on GitHub. Enable in GitHub repo [⚙ Settings](https://github.com/settings/repositories)

- **Netlify**: Connect a repository, build your site, deploy and host for free. [https://www.netlify.com/](https://www.netlify.com/)
Rmarkdown extensions

Many great packages and applications build on rmarkdown.

All this makes it incredibly versatile.
Create and maintain online books

Authoring with R Markdown. Offers:

- cross-references,
- citations,
- HTML widgets and Shiny apps,
- tables of content and section numbering

The publication can be exported to HTML, PDF, and e-books (e.g. EPUB)

Examples

- rOpenSci Software Review policies
- Geocomputation in R

Thesisdown

An updated R Markdown thesis template using the bookdown package
pkgdown

For building package documentation

Produce **function references** from `.Rd` files and **demonstrate function use** through long form documentation (vignettes).
Build analyses websites and organise your project

Makes it easier for researchers to organize projects and share results. Includes **checks to ensure rendered versions correspond to up to date versions of code.**
For creating and maintaining blogs through R.

Check out [https://awesome-blogdown.com/](https://awesome-blogdown.com/), a curated list of awesome #rstats blogs in blogdown for inspiration!
presentations

A number of existing frameworks

xaringan

プレゼンテーション忍者

Presentation Ninja

Yihui Xie

2016/12/12 (updated: 2019-02-07)
Managing code
Managing analysis code

Separate function definition and application

• When a project is new and shiny, an **analysis script usually contains many lines of directly executated code**.

• As it matures, **reusable chunks get pulled into their own functions**.

• The actual analysis scripts then become relatively short, and **functions defined in separate R scripts**.
R Package Structure

Used to share functionality with the R community

- Useful **conventions**
- Useful **software development tools**
- Easy **publishing** through GitHub
Build panel

Integrated graphical user interface

Package Writing

File > New Project >
New Directory > R Package

Turn project into package,
Enable roxygen documentation with
Tools > Project Options > Build Tools

Roxygen guide at
Help > Roxygen Quick Reference
R Package conventions:

- **metadata**: in the `DESCRIPTION` file
- **functions** in `.R scripts` in the `R/` folder
- **tests** in the `tests/` folder

**Documentation:**

- functions using **Roxygen notation**
- workflows using `.Rmd documents` in the `vignettes/` folder
Package metadata

Package: gaitr
Type: Package
Title: Functions to support BMC gait analysis
Description: Helpers to analyse processed gait data.
Version: 0.0.9000
Authors@R:
  c(person(given = "Anna",
           family = "Krystalli",
           role = c("aut", "cre"),
           email = "annakrystalli@googlemail.com"),
     person(given = "Lorenza",
            family = "Angelini",
            role = "aut",
            email = "l.angelini@sheffield.ac.uk"))
License: MIT + file LICENSE
To cite package 'gaitr' in publications use:

https://github.com/annakrystalli/gaitr

A BibTeX entry for LaTeX users is

```bibtex
@Manual{,
    title = {gaitr: Functions to support BMC gait analysis},
    author = {Anna Krystalli and Lorenza Angelini},
    year = {2019},
    note = {R package version 0.1.1},
    url = {https://github.com/annakrystalli/gaitr},
}
```
Dependency management

It’s the job of the DESCRIPTION file to **list the packages that your code depends on.**

Imports:
- dplyr,
- purrr,
- here,
- broom,
- tibble,
- magrittr,
- janitor,
- ggplot2

Suggests:
- knitr,
- rmarkdown

**add dependency**

```r
usethis::use_package("forcats", type = "Imports")
```
Functions in R/

example function script

Create a new function .R file in the R/ folder

```
usethis::use_r("add")
```

R
---
add.R

0 directories, 1 files
Document functions with Roxygen

- Create help files on build (autogenerated .Rd files in man/)
- Specify which functions are exported (autogenerated NAMESPACE)

```r
#' Add together two numbers.
#' @param x A number.
#' @param y A number.
#' @return The sum of x and y.
#' @examples
#' add(1, 1)
#' add(10, 1)
add <- function(x, y) {
  x + y
}
```
Tests provide confidence in what the code is doing.
Example test

usethis::use_test("add")

Creates a `tests/` folder with the following files:

```
tests
├── testthat
│   └── test-add.R
└── testthat.R
```

`test-add.R`

```r
context("test-add")

test_that("add works", {
    expect_equal(add(2, 2), 4)
})
```
Continuous Integration w/ Travis

A cloud testing framework for automating your tests

- Monitor the effect of changes to the code
- Safe onboarding of contributions

Start with a `.travis.yml` file

```bash
usethis::use_travis()
```
Resulting `.travis.yml` file template

language: R  
sudo: false  
cache: packages  

instructions to enable TRAVIS CI

✔ Writing `.travis.yml`
✔ Adding `'^\.travis\.yml$'` to `.Rbuildignore`
● Turn on travis for your repo at https://travis-ci.org/profile/annakrystalli
● Copy and paste the following lines into `'/Users/Anna/Documents/workflows/talks/README.md`

```
```
You can now run Continuous Integration on GitHub! See Jim Hester's rstudio conf 2020 talk on [Azure Pipelines and GitHub Actions](#).

- `usethis::use_github_action_check_release()`: installs the **latest release** R version on **macOS** and runs `R CMD check` via the `rcmdcheck` package.
- `usethis::use_github_action_check_standard()`: runs `R CMD check` on the **three major OSs** (Linux, macOS and Windows) on the **release version of R and R-devel**.
- `usethis::use_github_action_check_full()`: installs the **last 5 minor R versions** and runs `R CMD check` on the **three major OSs** (Linux, macOS and Windows).

All github actions added to `.github/workflows` directory of a package.
Use other R package tests & CI configurations as inspiration

- Look in tests/ for ideas on suitable tests.
- Look at CI configuration files (eg travis.yml or files in .github/workflows) to set up testing environment.
Research compendia
The paper is the advertisement

“an article about computational result is advertising, not scholarship. The actual scholarship is the full software environment, code and data, that produced the result.”

John Claerbout paraphrased in Buckheit and Donoho (1995)

The concept of a Research Compendium

“...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 and Temple Lang, 2004
Research compendia in R

https://doi.org/10.1111/ele.13085

**Compendium**

cboettig/noise-phenomena: Supplement to: "From noise to knowledge: how randomness generates novel phenomena and reveals information"

http://doi.org/10.5281/zenodo.1219780
"The goal of rrtools is to provide instructions, templates, and functions for making a basic compendium suitable for writing reproducible research with R."

Install rrtools from GitHub

```r
# install.packages("devtools")
devtools::install_github("benmarwick/rrtools")
```
Create compendium

rrtools::create_compendium("~/Documents/workflows/rrcompendium")

✔ Setting active project to '/Users/Anna/Documents/workflows/rrcompendium'
✔ Creating 'R/'
✔ Creating 'man/'
✔ Writing 'DESCRIPTION'
✔ Writing 'NAMESPACE'
✔ Writing 'rrcompendium.Rproj'
✔ Adding '.Rproj.user' to '.gitignore'
✔ Adding '^rrcompendium\Rproj$', '^\Rproj\user$' to '.Rbuildignore'
✔ Opening new project 'rrcompendium' in RStudio
✔ The package rrcompendium has been created
✔ Opening the new compendium in a new RStudio session...

Next, you need to:  ↓ ↓ ↓
● Edit the DESCRIPTION file
● Use other 'rrtools' functions to add components to the compendium
rrtools::use_readme_rmd()

✔ Creating 'README.Rmd' from template.
✔ Adding 'README.Rmd' to `.Rbuildignore`.
● Modify 'README.Rmd'
✔ Rendering README.Rmd to README.md for GitHub.
✔ Adding code of conduct.
✔ Creating 'CONDUCT.md' from template.
✔ Adding 'CONDUCT.md' to `.Rbuildignore`.
✔ Adding instructions to contributors.
✔ Creating 'CONTRIBUTING.md' from template.
✔ Adding 'CONTRIBUTING.md' to `.Rbuildignore`.
Create analysis folder

```r
rrtools::use_analysis()
```

✔ Adding bookdown to Imports
✔ Creating 'analysis' directory and contents
✔ Creating 'analysis'
✔ Creating 'analysis/paper'
✔ Creating 'analysis/figures'
✔ Creating 'analysis/templates'
✔ Creating 'analysis/data'
✔ Creating 'analysis/data/raw_data'
✔ Creating 'analysis/data/derived_data'
✔ Creating 'references.bib' from template.
✔ Creating 'paper.Rmd' from template.

Next, you need to: ↓ ↓ ↓ ↓

● Write your article/report/thesis, start at the paper.Rmd file
● Add the citation style library file (csl) to replace the default provided here, see https:
● Add bibliographic details of cited items to the 'references.bib' file
● For adding captions & cross-referencing in an Rmd, see https://bookdown.org/yihui/bookdown
● For adding citations & reference lists in an Rmd, see http://rmarkdown.rstudio.com/author
Capturing dependencies

```r
rrtools::add_dependencies_to_description()
```

Imports:

bookdown,
ggplot2 (≥ 3.0.0),
ggthemes (≥ 3.5.0),
here (≥ 0.1),
knitr (≥ 1.20),
rticles (≥ 0.6)
Further Helpers

📦 rticles

Contains a **suite of custom R Markdown templates for popular journals**, simplifying the creation of documents that conform to research paper submission standards.
RStudio Add-in to **Insert Markdown Citations**

Upon rather than rejects the previous is noise can create novel phenomena and obscure the signal of some process. Simplicity not only makes each tractable but also allows us to see and understand noisy processes in relatively simple models, not just Bartlett (1960). Bartlett (1960). Stochastic population models in ecology and epidemiology.


[@Bartlett1960]

In parentheses

Bibliography file(s) found in YAML front matter: refs.bib

Reload file(s)
Reproducible Computational Environments
Why isn't sharing code enough?

Case Study: Sharing a Geospatial Analysis in R

On a computer without System Library `GDAL` ✗

package ‘rgdal’ successfully unpacked and MD5 sums checked

configure: gdal-config: gdal-config checking gdal-config usability ...
./conf line 1353: gdal-config: command not found

Error: gdal-config not found

... ERROR: configuration failed for package ‘rgdal’

slide: Karthik Ram: rstudio::conf 2019 talk
What are Docker containers?

standardized units of software

package up everything needed to run an application: code, runtime, system tools, system libraries and settings in a lightweight, standalone, executable package

- **Dockerfile**: Text file containing recipe for setting up computation environment.
- **Docker Image**: Executable built from the Dockerfile with all required dependencies installed. Can have many images from the same Dockerfile.
- **Docker Container**: Docker Images become containers at runtime
using the **rocker/geospatial** Docker Image ☑️

---

**KÖMPENDIUM**

1. ?
2. 4x 1x 14x 1x
3. ~
4. 
5. ✓

*slide: Karthik Ram: rstudio::conf 2019 talk*
rrtools::use_dockerfile()

✔ Creating 'Dockerfile' from template.
✔ Adding 'Dockerfile' to `.`.Rbuildignore`.
• Modify
Next:
  * Edit the dockerfile with your name & email
  * Edit the dockerfile to include system dependencies, such as linux libraries that are not in the R build
  * Check the last line of the dockerfile to specify which Rmd should be rendered in the container
# get the base image, the rocker/verse has R, RStudio and pandoc
FROM rocker/verse:3.6.0

# required
MAINTAINER Anna Krystalli <annakrystallil@googlemail.com>

COPY . /rrcompendiumDTB

# go into the repo directory
RUN . /etc/environment \
    # Install linux dependencies here \
    # e.g. need this for ggforce::geom_sina \
    && sudo apt-get update \
    && sudo apt-get install libudunits2-dev -y \
    # build this compendium package \
    && R -e "devtools::install('./rrcompendiumDTB', dep=TRUE)" \
    # render the manuscript into a docx, you'll need to edit this if you've 
    # customised the location and name of your main Rmd file \
    && R -e "rmarkdown::render('./rrcompendiumDTB/analysis/paper/paper.Rmd')"
Docker + Travis

Create **.travis.yml**

```r
rrtools::use_travis()
```

✔ Creating '.travis.yml' from template.
✔ Adding '.travis.yml' to `.Rbuildignore`.

Next:
* Add a Travis shield to your README.Rmd:
  ![Travis-CI Build Status](https://travis-ci.org/annakrystalli/rrcompendiumDTB.svg?branch=master)
* Turn on travis for your repo at https://travis-ci.org/annakrystalli/rrcompendiumDTB

* To connect Docker, go to https://travis-ci.org/`, and add your environment variables: DOCKER_EMAIL, DOCKER_USER, DOCKER_PASS to enable pushing to the Docker Hub
```yaml
env:
  global:
    - REPO=${DOCKER_USER}/rrcompendiumdtb

sudo: required

warnings_are_errors: false

language: generic

services:
  - docker

before_install:
  - docker build -t $REPO .

Create & build image using dockerfile, i.e. compile pkg and render Rmd to Word doc
```
Push our custom docker image to docker hub, env vars stored on travis-ci.org

```
after_success:
- docker login -u $DOCKER_USER -p $DOCKER_PASS
- export REPO=$DOCKER_USER/rrcompendiumdtb
- export TAG=`if [ "$TRAVIS_BRANCH" == "master" ]; then echo "latest"; else echo $TRAVIS_BRANCH; fi`
- docker build -f Dockerfile -t $REPO:$COMMIT.
- docker tag $REPO:$COMMIT $REPO:$TAG
- docker tag $REPO:$COMMIT $REPO:travis-$TRAVIS_BUILD_NUMBER
- docker push $REPO
```

Travis repository settings

**Environment Variables**

Customize your build using environment variables. For secure tips on generating private keys read our documentation

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
<th>Available to all branches</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOCKER_PASS</td>
<td>********************</td>
<td></td>
</tr>
<tr>
<td>DOCKER_USER</td>
<td>********************</td>
<td></td>
</tr>
</tbody>
</table>
Travis build passes!
Docker Image: [https://hub.docker.com/repository/docker/akrystalli/rrcompendiumdtb](https://hub.docker.com/repository/docker/akrystalli/rrcompendiumdtb)

Compendium Repository: [https://github.com/annakrystalli/rrcompendiumDTB](https://github.com/annakrystalli/rrcompendiumDTB)
Working with the Docker Image

Pull it from Dockerhub.

docker pull akrystalli/rrcompendiumdtb:latest

Run in Rstudio in your browser

More on using the RStudio image

docker container run -e PASSWORD=<password> -e USERID=$UID -p 8787:8787 --detach --name rrcompendiumdtb akrystalli/rrcompendiumdtb:latest
What is binder?  https://mybinder.org/

Give it a repo url

Launch computational environment! 🤪

Turn a Git repo into a collection of interactive notebooks

Have a repository full of Jupyter notebooks? With Binder, open those notebooks in an executable environment, making your code immediately reproducible by anyone, anywhere.

Build and launch a repository

GitHub repository name or URL

GitHub repository name or URL

Git branch, tag, or commit

Path to a notebook file (optional)

File

launch

Copy the URL below and share your Binder with others:

Fill in the fields to see a URL for sharing your Binder.

Copy the text below, then paste into your README to show a binder badge: 🚀
Binderhub Ecosystem

This image was created by Scriberia for The Turing Way community and is used under a CC-BY licence.
BAM!
# R repository options for Binder

<table>
<thead>
<tr>
<th>Basic</th>
<th>Premium</th>
<th>Pro</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>install.r</code></td>
<td><code>Dockerfile</code></td>
<td><code>Dockerfile</code></td>
</tr>
<tr>
<td><code>runtime.txt</code></td>
<td><code>install.r</code></td>
<td><code>DESCRIPTION</code></td>
</tr>
<tr>
<td><code>apt.txt</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Basic**
  - `install.r`
  - `runtime.txt`
  - `apt.txt`
  - Slow but easy to setup.
  - Recommended for beginners
  - ![Launch Binder](launch_binder.png)

- **Premium**
  - `Dockerfile`
  - `install.r`
  - Faster launch
  - ![Launch Binder](launch_binder.png)

- **Pro**
  - `Dockerfile`
  - `DESCRIPTION`
  - Best for compendia
  - ![Launch Binder](launch_binder.png)
Binderise your R projects w/ holepunch

https://github.com/karthik/holepunch

```r
remotes::install_github("karthik/holepunch")
```

Create `.binder/Dockerfile`

```
holepunch::write_dockerfile(maintainer = "Anna Krystalli")
```

→ Setting R version to 3.6.0
→ Locking packages down at 2019-11-09
✔ Dockerfile generated at ./.binder/Dockerfile
FROM rocker/binder:3.6.0
LABEL maintainer='Anna Krystalli'
USER root
COPY . ${HOME}
RUN chown -R ${NB_USER} ${HOME}
USER ${NB_USER}

RUN wget https://github.com/annakrystalli/rrcompendiumDTB/raw/master/DESCRIPTION & R
  devtools::install_deps(); devtools::install(); tinytex::install_tinytex()"

RUN rm DESCRIPTION.1; exit 0
holepunch::generate_badge()

- Copy and paste the following lines into
  `'/Users/Anna/Documents/workflows/compendia/'`:

  ![Launch Rstudio Binder](http://mybinder.org/badge_logo.svg)

  [Copied to clipboard]
Launched Binderised Compendium
Reproducibility in Practice
ReproHacks

one day reproducibility hackathons

Mission: Reproduce papers from code and data

- Record experiences and feedback to authors
- Available soon: Publish Reproducibility Report in ReScience C
ReproHack Benefits
1. Practical reproducibility they can implement in their own work
Participants

1. Practical reproducibility they can implement in their own work
1. Practical reproducibility they can implement in their own work

2. Inspiration from working with other people’s code and data.
1. Practical reproducibility they can implement in their own work

2. Inspiration from working with other people’s code and data.

3. Reproduction as community value
1. Useful feedback on the reproducibility of their work

OpenCon RepoHack feedback form

Name of participant
Anna Krystalli, Marios Georgiou

Which paper did you attempt? *
Paper #3. The archaeology, chronology and stratigraphy of Madjedbebe (Malakunanja II): a site in northern Australia with early occupation

Team issue URL *
https://github.com/OpenCon-London/OpenCon_London-Doathon/issues/4

Did you manage to reproduce it? *
- Yes
- No
- Almost

On a scale of 1 to 10, how much of the paper did you manage to reproduce? *
- None of it
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- All of it
Authors

1. Useful feedback on the reproducibility of their work

2. Appreciation for their efforts in making their work reproducible
1. Useful feedback on the reproducibility of their work

2. Appreciation for their efforts in making their work reproducible
1. Useful feedback on the reproducibility of their work

2. Appreciation for their efforts in making their work reproducible

3. An opportunity to engage others with their research.
Huge thanks to @EnviroKaty for submitting a fab 🦋🦋🦋 paper to the #CCMcr19 #ReproHack! I had loads of fun reproducing the analysis for this really cool paper

https://t.co/v1ww2D5xhg

pic.twitter.com/r8rYMAMvPm

— Jessica Ward (@JKRWard) June 27, 2019
This thing is really happening!

pic.twitter.com/d2xnHNo9z2

— ReproHack NL 🌐 (@ReproHackNL) September 20, 2019

Upcoming ReproHacks!

Newcastle, Leeds, Liverpool
Manchester, Sheffield
Jan - Mar 2020
Take home
Following conventions

This image was created by Scriberia for The Turing Way community and is used under a CC-BY licence.
Successful Reproducibility

1. Curious person
2. Symbols: CSV, disk, box, file
   - Counts: 4x, 1x, 14x, 1x
3. Person thinking
4. Person organizing items
5. Person checking items

slide: Karthik Ram: rstudio::conf 2019 talk
Enhanced Research Cycle

Enhanced Research Cycle

DATA COLLECTION

DATA PROCESSING

DATA STUDY & ANALYSIS

DATA PUBLISHING & ACCESS

RESEARCH DATA PLANNING & DESIGN

DATA RE-USE

DATA PRESERVATION

Research Ideas
Reproducibility as standard ✅

Time for a CULTURAL SHIFT...

We should value reproducibility as much as # of papers published

This image was created by Scriberia for The Turing Way community and is used under a CC-BY license.
Resources
Book

a lightly opinionated guide to reproducible data science

https://the-turing-way.netlify.com

workshops

- Boost Your Research Reproducibility with Binder materials
- Build a binderhub materials

https://github.com/alan-turing-institute/the-turing-way
Reproducibility in R

Version Control
- Happy Git and GitHub for the useR

RMarkdown
- R Markdown: The Definitive guide
- RMarkdown Driven Development (RmdDD): Blog post by Emily Riederer

R Packages
- R packages by Hadley Wickham and Jenny Bryan

Research Compendia
- Karthik Ram: rstudio::conf 2019 talk

Docker & Binder
- Getting started with binder docs
  - rOpenSci Docker tutorial

Tutorials
- Rstudio Essentials Webinar series
- rrresearch: ACCE DTP course on Research Data & Project Management
ReproHacks

reprohack-hq repository

Check out our issues

Chat to us:

chat on gitter

Sign up to our Newsletter

Enter your email address

Subscribe

powered by TinyLetter
Questions?