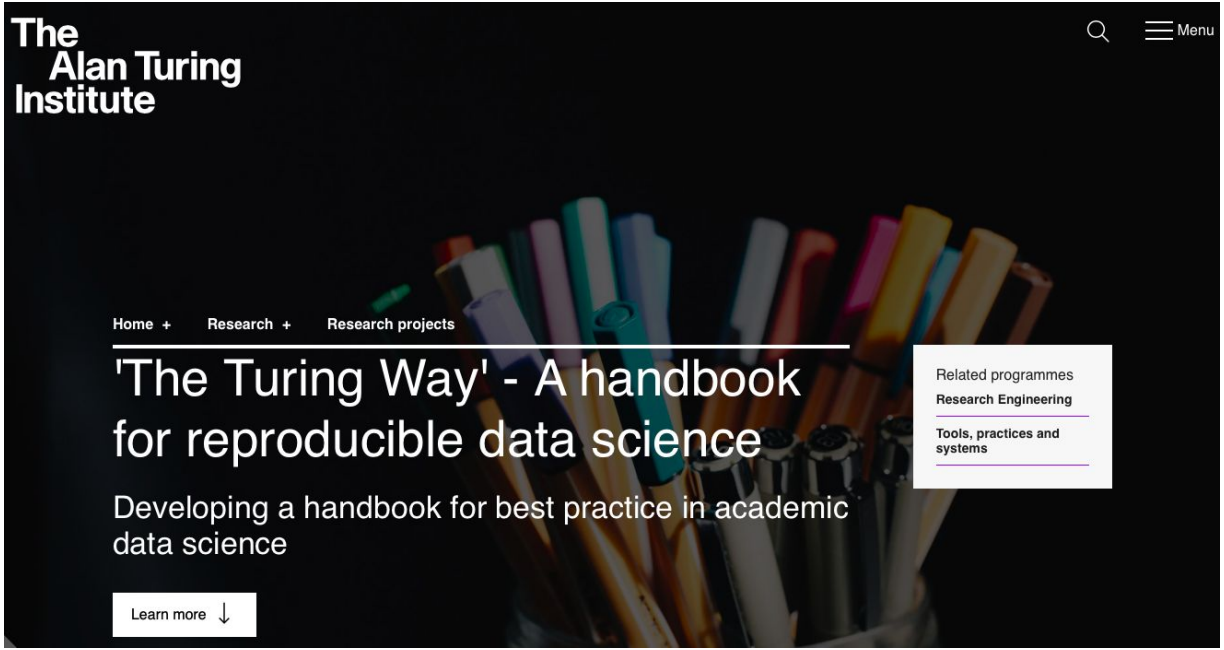
Binder for Fully Reproducible Research

Dr Andrew J. Stewart

E: drandrewjstewart@gmail.com
T: @ajstewart_lang
G: ajstewartlang

The Turing Way

This workshop draws heavily upon The Turing Way Binder workshop which I attended in 2019 - The Turing Way Handbook is a fantastic resource on reproducible data science and I very much recommend you check it out!



https://www.turing.ac.uk/research/research-projects/turing-way-handbook-reproducible-data-science

Open and Reproducible Research

Shared Data

We already know this is important for reproducibility.

Shared Code

We already know this is important for reproducibility.

Shared Computational Environment

Why is this important and how do we do it?

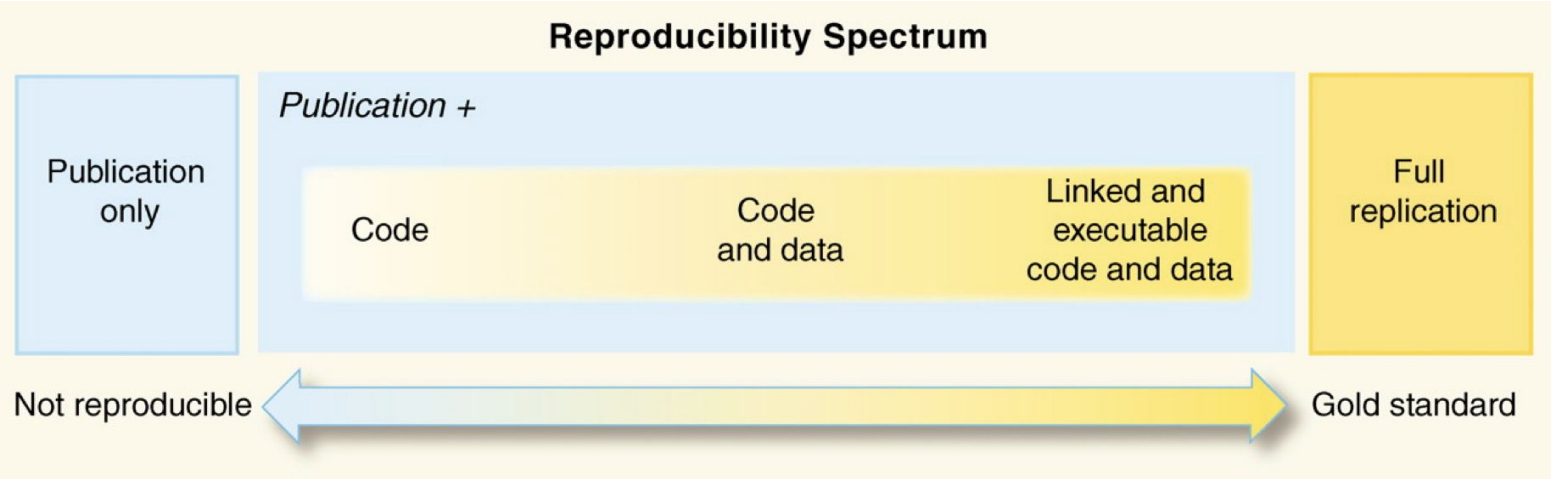
Open and Reproducible Research

PERSPECTIVE

Reproducible Research in Computational Science

Roger D. Peng
+ See all authors and affiliations

Science 02 Dec 2011:
Vol. 334, Issue 6060, pp. 1226-1227
DOI: 10.1126/science.1213847



Why do we need to reproduce the computational environment?

Quite often analysis code ‘breaks’ - often in one of two ways:

Code that worked previously now doesn’t - maybe a function in an R package was updated (e.g., `lsmeans()` became `emmeans()` so old code using the `lsmeans()` function wouldn’t now run).

Code that worked previously still works - but produces a slightly different result or now throws a warning where it didn’t previously (e.g., convergence/singular fit warnings in `lme4::` version 1.1-19 vs. version 1.1-20).

Why do we need to reproduce the computational environment?

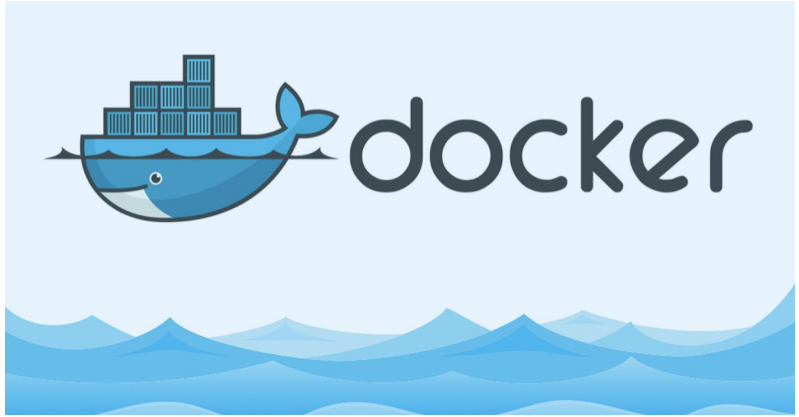
There have even been changes in the way that base R works - in R v3.6 the way in which sample() worked differed from how it had worked previously, and in R v4.0, when reading in data `stringsAsFactors = FALSE` (whereas previously `stringsAsFactors = TRUE`).

You need to capture the versions of the different R packages (plus their dependencies) and even the version of R you used in the original analysis.

Introduction to Docker

Docker packages your data, code and all its dependencies in the form called a docker container to ensure that your application works seamlessly in any environment.

When you run a docker container it's like running your analysis on a computer that has the same configuration as our own one at the point in time when you ran the original analysis.



https://medium.com/the-andela-way/docker-for-beginners-61e8e0ce6a19

So what’s Binder?

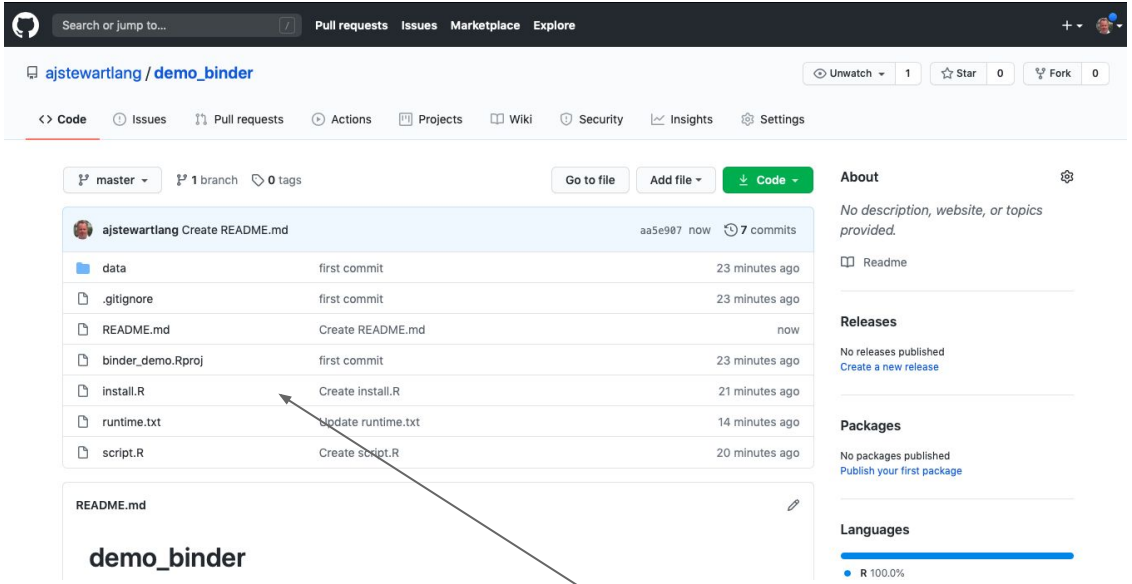
Binder is powered by BinderHub, which is an open-source tool that deploys the Binder service in the cloud.

Binder works by pulling a repository that you set up on GitHub into a Docker container using `repo2docker`.

Think of a repository as a folder containing your R code, your data, and a few other small bits and pieces - but it sits in the cloud rather than on your computer.

It pulls the versions of the R packages you specify (on the date you specify) from MRAN.

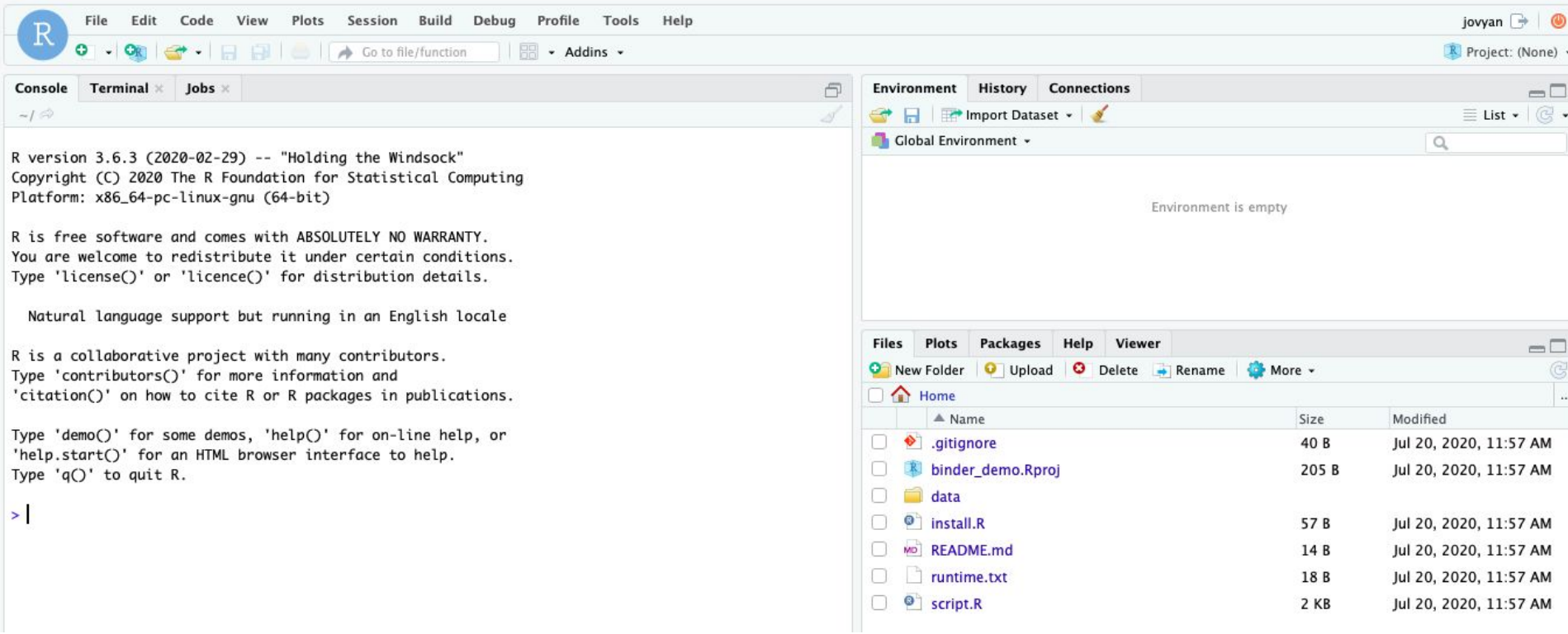
My GitHub Repo



This repo contains my data and R script (plus a few other files).

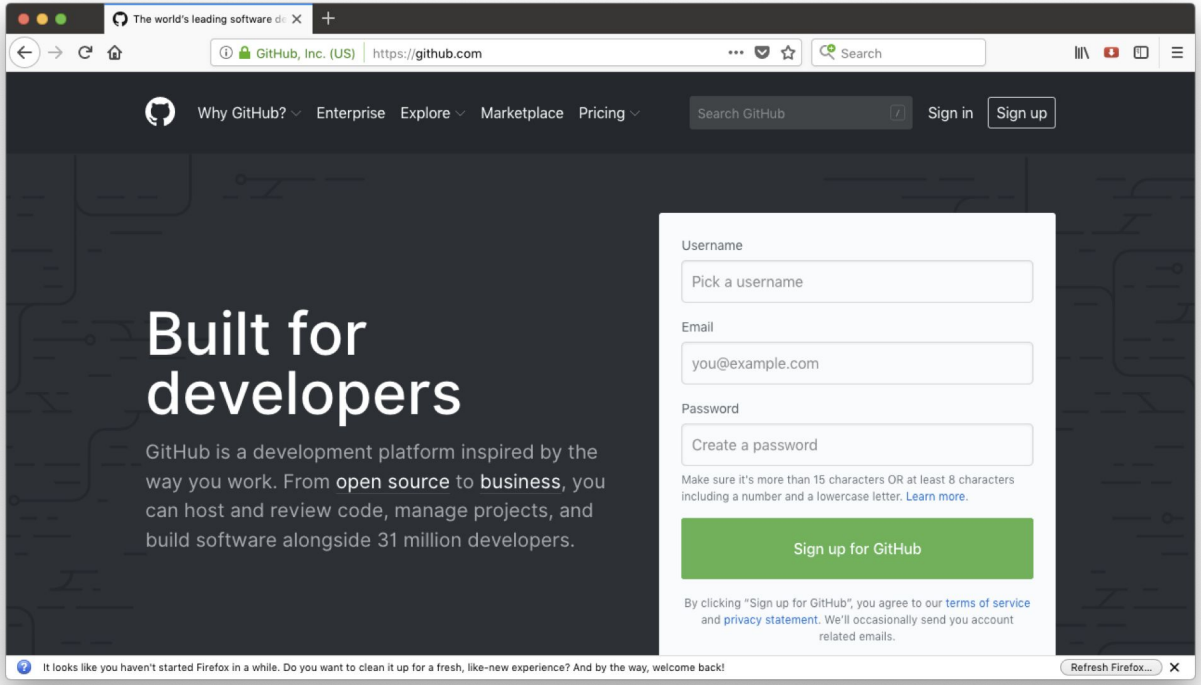
When I link my GitHub repository to Binder and launch it I then get the following in my web browser.

This is RStudio running the cloud using my code, my data and the appropriate versions of the packages that I was using when I did the analysis originally!



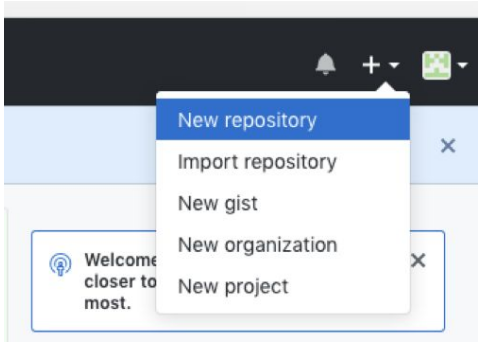
https://mybinder.org/v2/gh/ajstewartlang/demo_binder/master?urlpath=rstudio

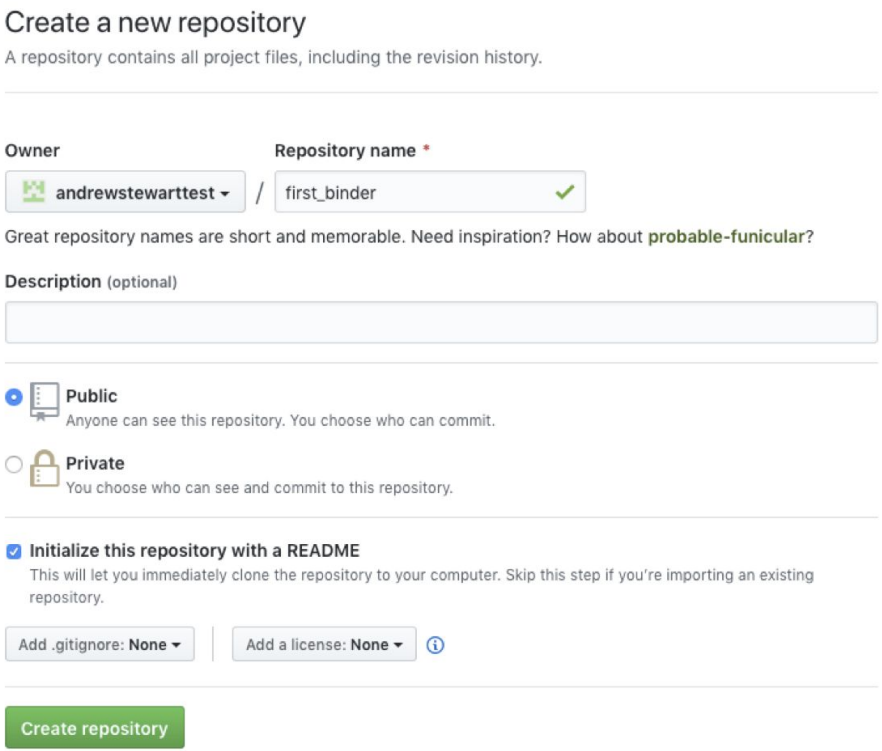
Step 1 – Set up a GitHub account



Go to github.com to set up an account.

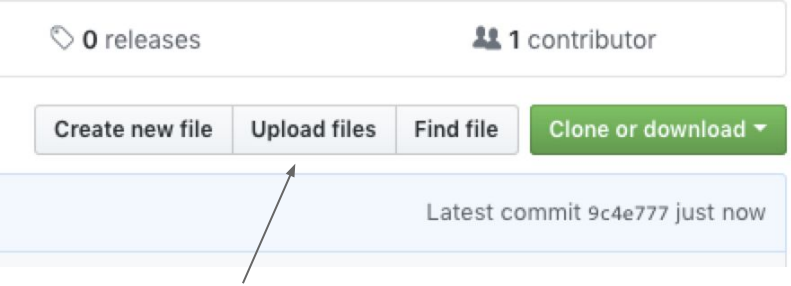
Step 2 – Create a new repository





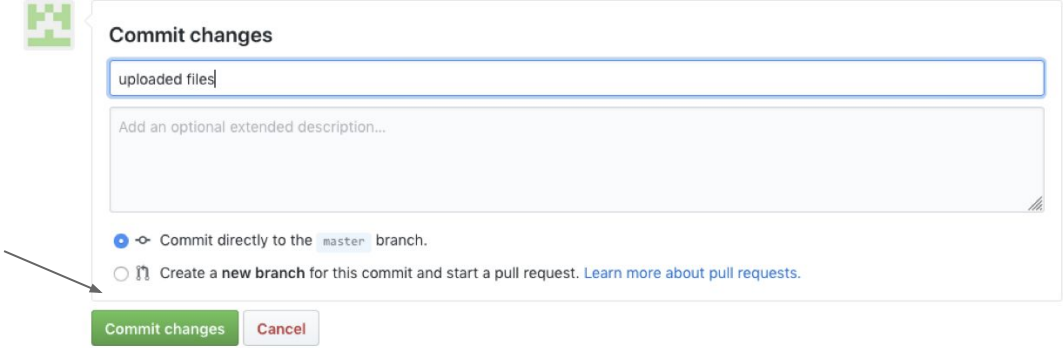
Make it 'Public' and initialise it with a README.

Step 3 – Upload your R script and data and make your first "Commit"



Click here to upload.

Click here to commit.

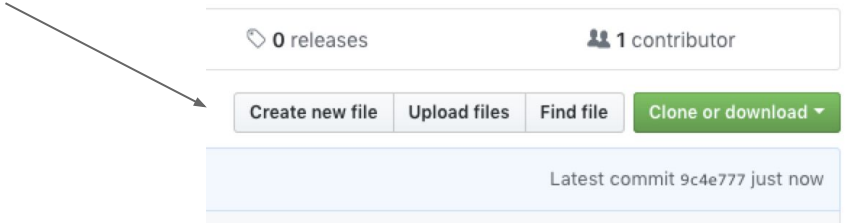


Step 4 – create the files “runtime.txt” and “install.R”

We need two other files at this point - one is called “runtime.txt” and contains the date and version of R and its associated packages that you want to simulate.

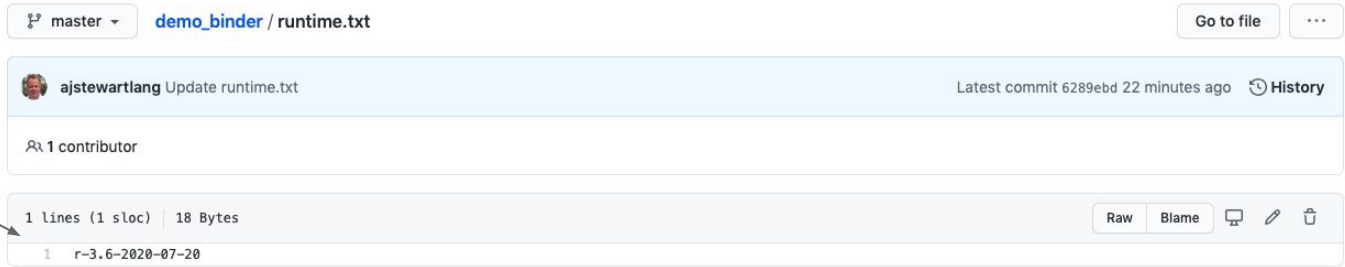
The other is called “install.R” and contains the list of R packages that need to be installed in order for your script to run.

To create a new file select “Create new file”

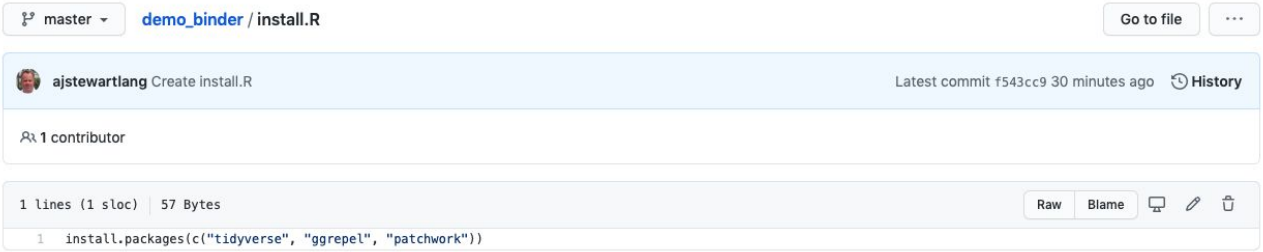


Step 4 – create the files “runtime.txt” and “install.R”

In the runtime.txt file type the R version and date of the packages on MRAN that you want in the format r-version-YYYY-MM-DD

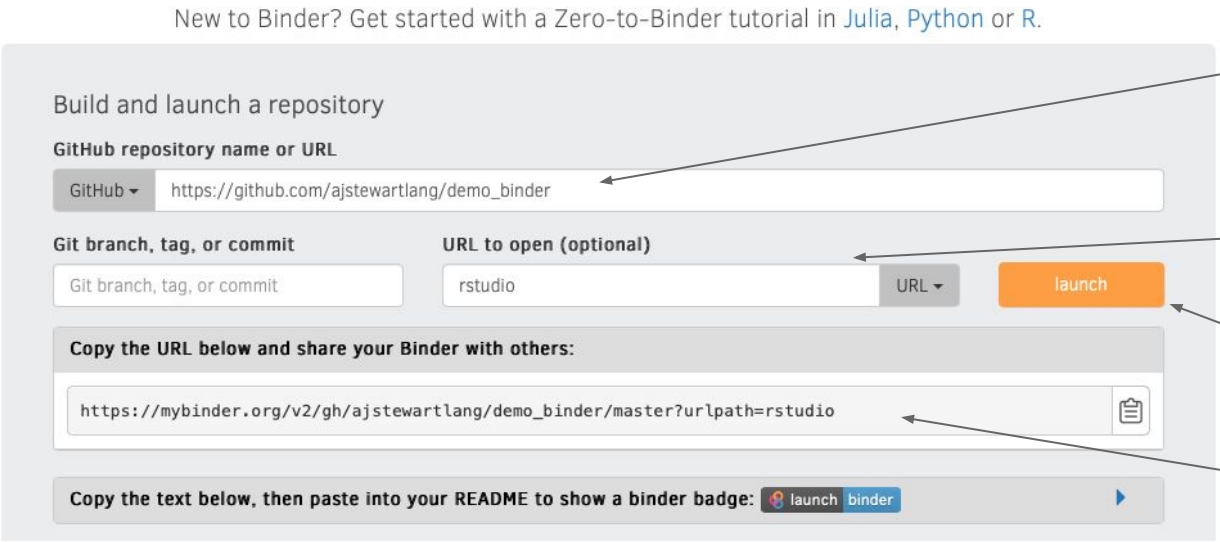


List your packages like this in the install.R file



Don’t forget to click “Commit” after you’ve created each file!

Step 5 – Now we need to link our repo to Binder (mybinder.org)



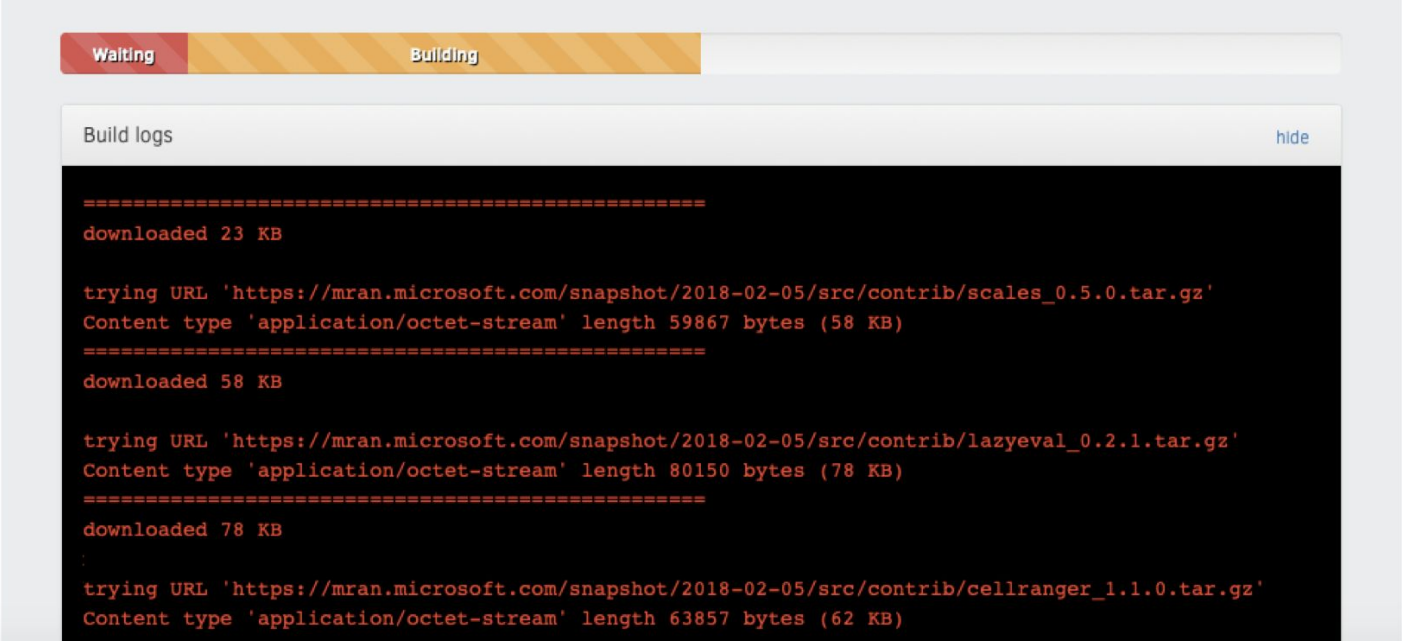
Paste the address of your GitHub repo you want to Binderise.

Type “RStudio” and select “URL”

Click on “Launch”

And this is the link to share with others.

When you first click on “Launch”



You can check the progress of the build by clicking on the “Build logs” bar.

If Binder can find an image that you've built previously, it will simply launch that. If you've made changes to your GitHub repo, it will rebuild the Docker image and create a new Binder. Either way, once Binder launches you get the following in your browser (even on mobile devices so you can even R away on your phone)…

File Edit Code View Plots Session Build Debug Profile Tools Help

jovyan

Go to file/function | Addins | Project: (None)

Console | Terminal | Jobs

```
R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (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.

>
```

Environment | History | Connections

Import Dataset | List

Global Environment

Environment is empty

Files | Plots | Packages | Help | Viewer

New Folder | Upload | Delete | Rename | More

Home

Name	Size	Modified
.gitignore	40 B	Jul 20, 2020, 11:57 AM
binder_demo.Rproj	205 B	Jul 20, 2020, 11:57 AM
data		
install.R	57 B	Jul 20, 2020, 11:57 AM
README.md	14 B	Jul 20, 2020, 11:57 AM
runtime.txt	18 B	Jul 20, 2020, 11:57 AM
script.R	2 KB	Jul 20, 2020, 11:57 AM

A few other things

Installing the entire Tidyverse in a Binder can take a long time - better to install only the packages you use (e.g., `ggplot2`, `dplyr`, `readr` etc.) - this will also ensure the individual packages are consistent with the date in your `runtime.txt` file.

Even with just a couple of packages it can take ~15 minutes or so for your Binder to be built. While your Binder builds you can close your computer as the build will continue in the cloud.

To change the version of R that Binder builds (to 3.5 say) change the R version in `runtime.txt` like this "`r-3.5-2020-07-20`"

For Ultimate Reproducibility

Make sure you have updated all your packages before you run your script locally.

Build your Binder and specify the day you ran your analysis in the runtime.txt file - and add a version of R if you don’t want it to go with the default.

Patience while your Binder builds.

Your turn

Start from Slide 11, take an R script you've already written and Binderise it!