

# Slides

[http://ajstewartlang.github.io/Binder\\_slides.pdf](http://ajstewartlang.github.io/Binder_slides.pdf)

# Binder for fully reproducible research in R (data, code, and computational environment).

Andrew Stewart  
University of Manchester  
@ajstewart\_lang



The Turing Way

<https://github.com/alan-turing-institute/the-turing-way>

# 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?

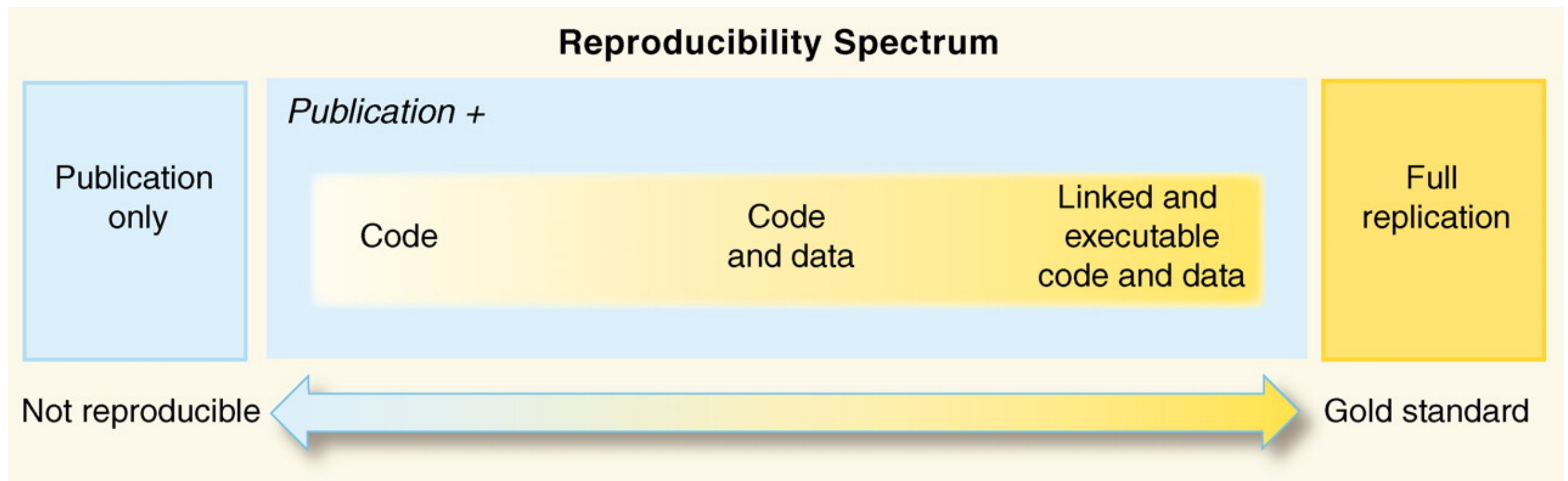
PERSPECTIVE

# 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 `lsmeans` 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).

# Capturing your local computational environment

- You need to capture the versions of the different R packages (plus their dependencies).
- May sound trivial but trying running some old R code and be amazed at how many things now don't work as they once did!

# Docker for beginners

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 virtual computer that has the same configuration as our own one at the point in time when you ran the 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.
- 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.



github.com/ajstewartlang/Turing\_way2

Google Scholar Scopus jobs.ac.uk Apple BBC News Chester Weather The Telegraph The Grauniad The Independent Google Maps Chester Weather Station Favourites

Search or jump to... Pull requests Issues Marketplace Explore

ajstewartlang / Turing\_way2 Watch 0 Star 0 Fork 1

Code Issues 0 Pull requests 1 Projects 0 Wiki Insights Settings

markdown\_for\_Turing\_Way Edit

Manage topics

6 commits 1 branch 0 releases 1 contributor

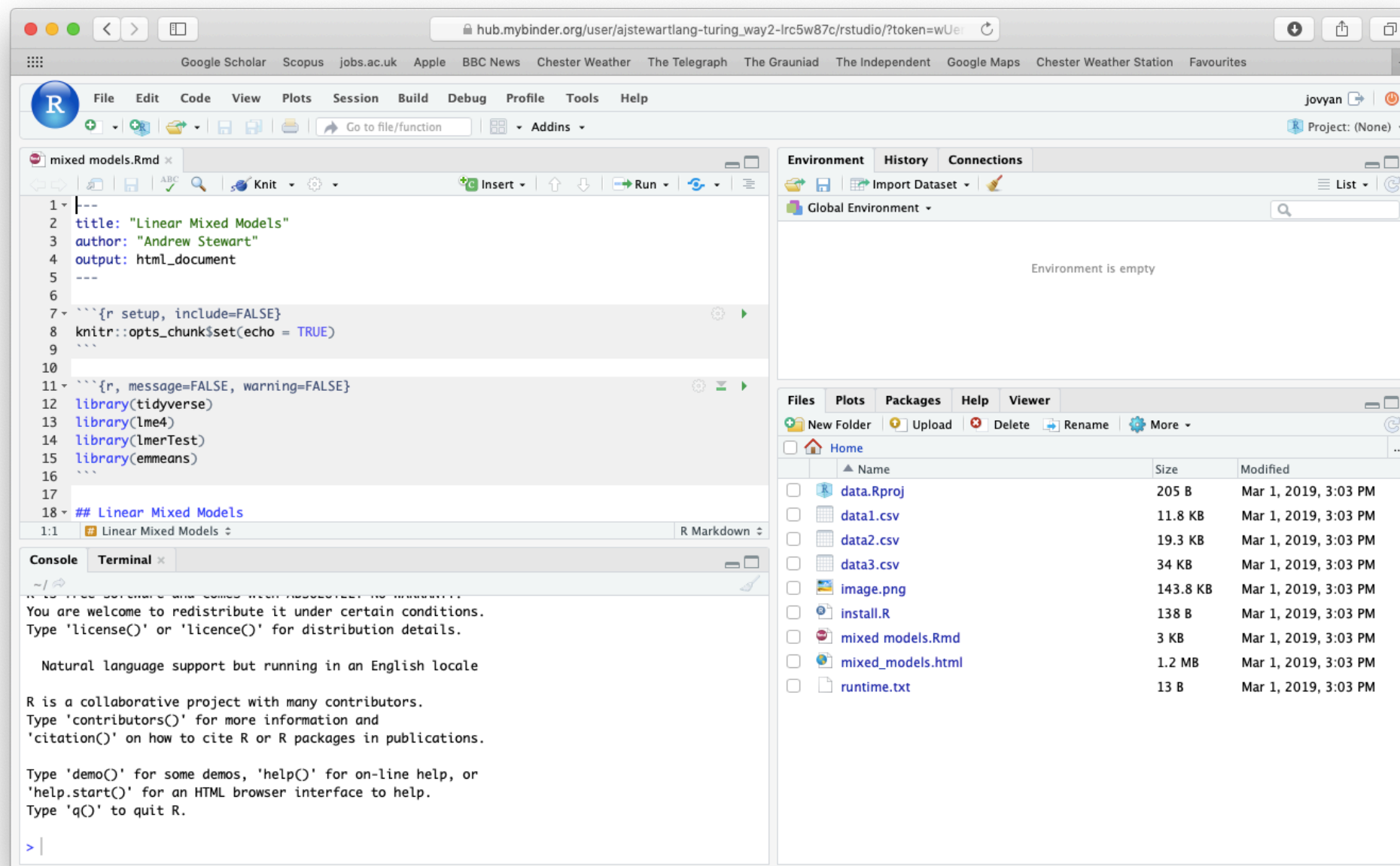
Branch: master New pull request Create new file Upload files Find file Clone or download

ajstewartlang	Create install.R	Latest commit 36d3181 2 hours ago
.Rproj.user	commit	2 hours ago
data.Rproj	first commit	3 hours ago
data1.csv	first commit	3 hours ago
data2.csv	first commit	3 hours ago
data3.csv	first commit	3 hours ago
image.png	first commit	3 hours ago
install.R	Create install.R	2 hours ago
mixed models.Rmd	commit	2 hours ago
mixed_models.html	first commit	3 hours ago
runtime.txt	Update runtime.txt	2 hours ago

GitHub Desktop

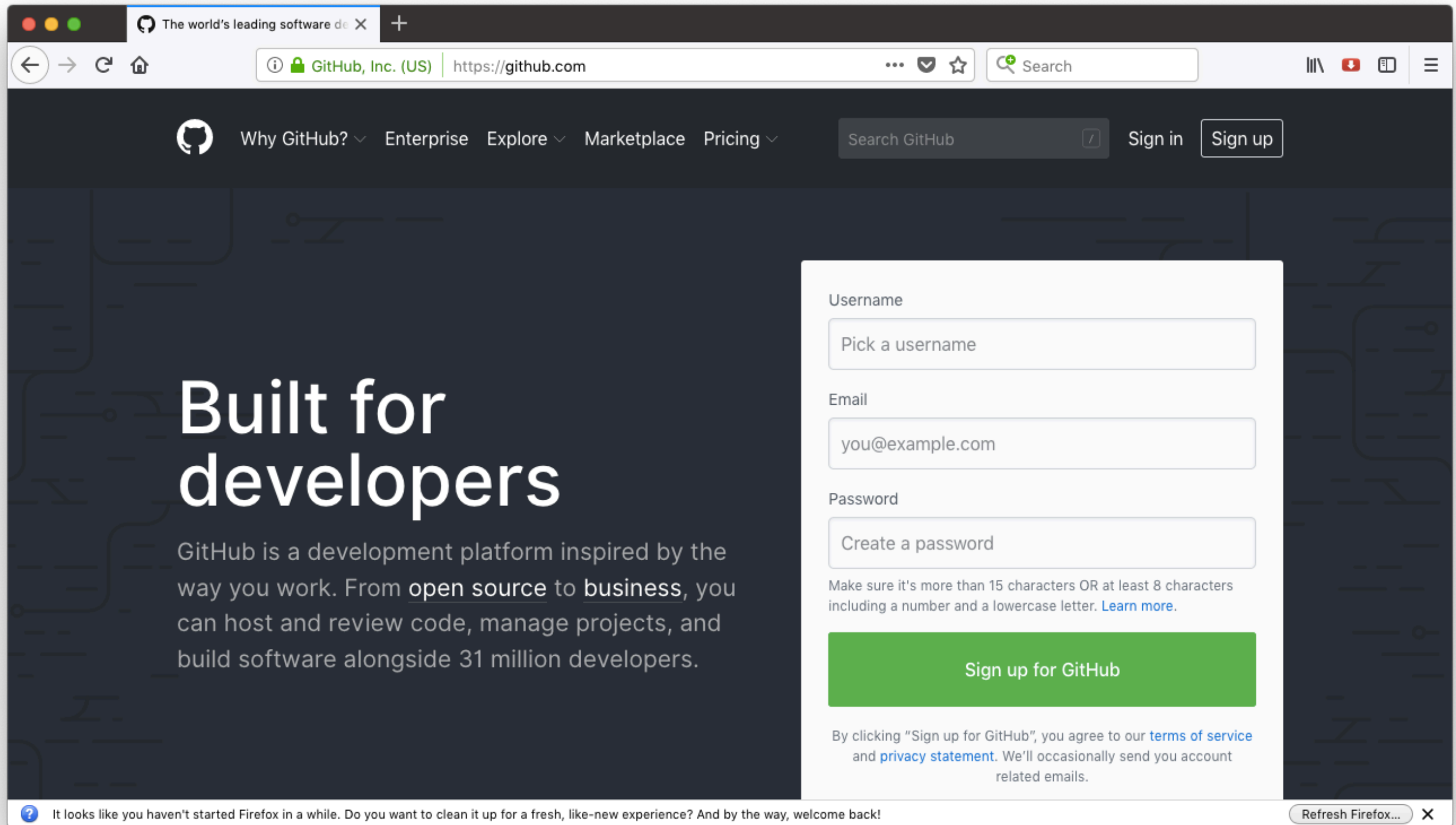
My R code and data files.

- 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/Turing\\_way2/master?urlpath=rstudio](https://mybinder.org/v2/gh/ajstewartlang/Turing_way2/master?urlpath=rstudio)

# Step 1 - Set up a GitHub account



The screenshot shows the GitHub homepage in a web browser. The browser's address bar displays "https://github.com". The page features a dark blue header with the GitHub logo, navigation links ("Why GitHub?", "Enterprise", "Explore", "Marketplace", "Pricing"), a search bar, and "Sign in" and "Sign up" buttons. The main content area has a large heading "Built for developers" and a subheading "GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 31 million developers." On the right side, there is a white sign-up form with fields for "Username", "Email", and "Password". The "Username" field contains the placeholder text "Pick a username". The "Email" field contains "you@example.com". The "Password" field contains the placeholder text "Create a password". Below the password field, there is a note: "Make sure it's more than 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)". A large green button labeled "Sign up for GitHub" is positioned below the form. At the bottom of the form, there is a disclaimer: "By clicking 'Sign up for GitHub', you agree to our [terms of service](#) and [privacy statement](#). We'll occasionally send you account related emails." At the very bottom of the browser window, there is a Firefox notification bar that says: "It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!" with a "Refresh Firefox..." button and a close button.

Username

Pick a username

Email

you@example.com

Password

Create a password

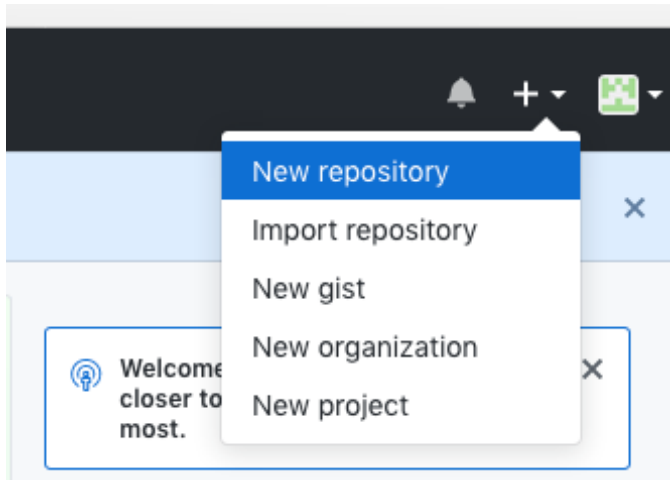
Make sure it's more than 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy statement](#). We'll occasionally send you account related emails.

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back! Refresh Firefox... X

# Step 2 - Create a new repository



Give it a name,  
set it to public  
and tick  
“Initialise this  
repository with a  
README”.

## Create a new repository

A repository contains all project files, including the revision history.

Owner

 andrewstewarttest ▾


Repository name \*

first\_binder ✓

Great repository names are short and memorable. Need inspiration? How about **probable-funicular**?

Description (optional)


☒  **Public**  
Anyone can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

☒ **Initialize this repository with a README**

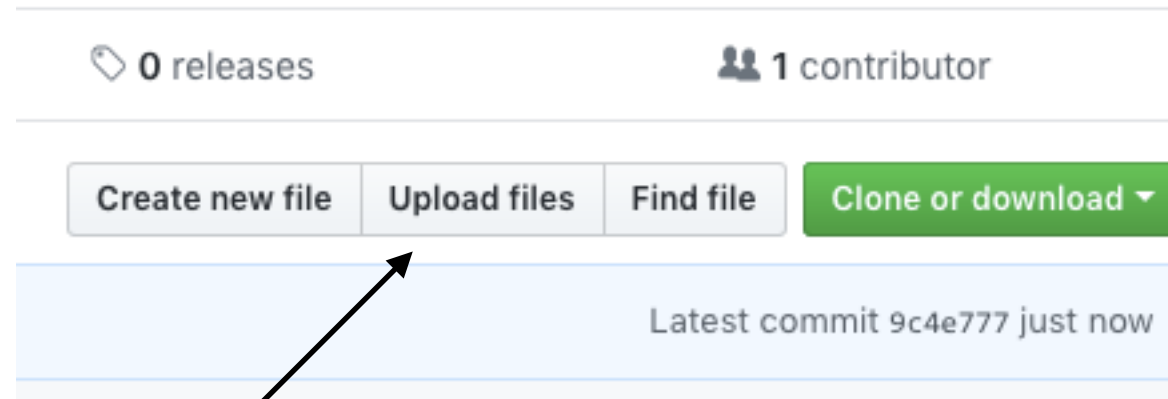
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

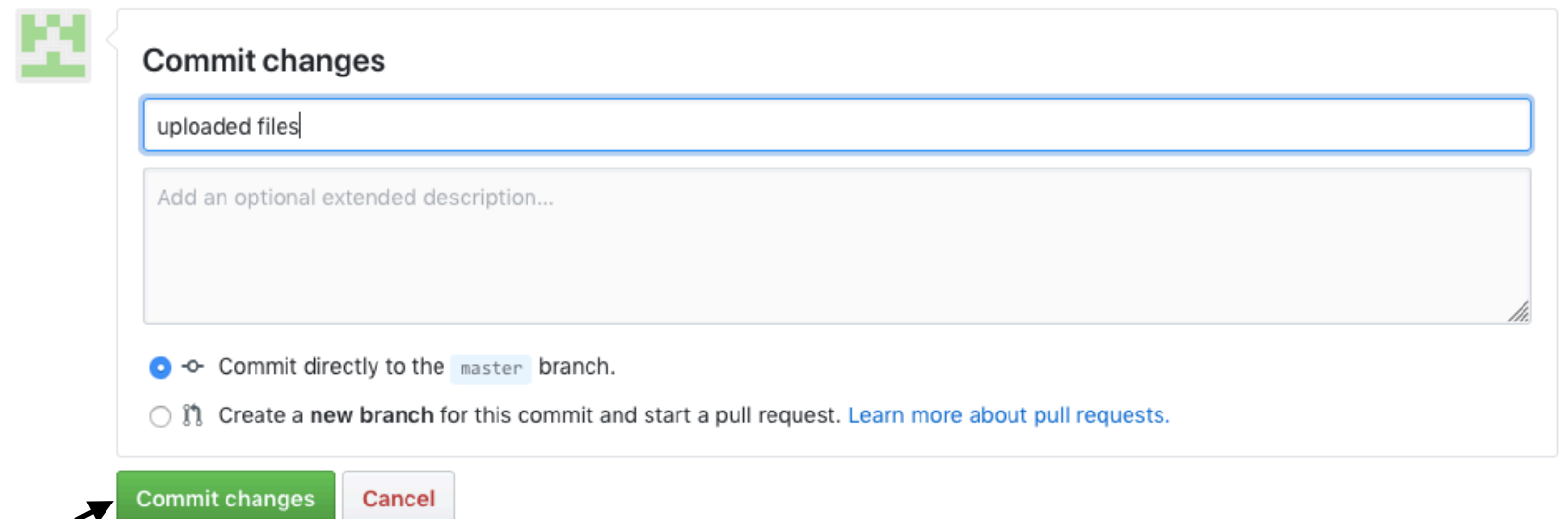
Add a license: **None** ▾ 

Create repository

# Step 3 - Upload your R script and data and make your first “Commit”



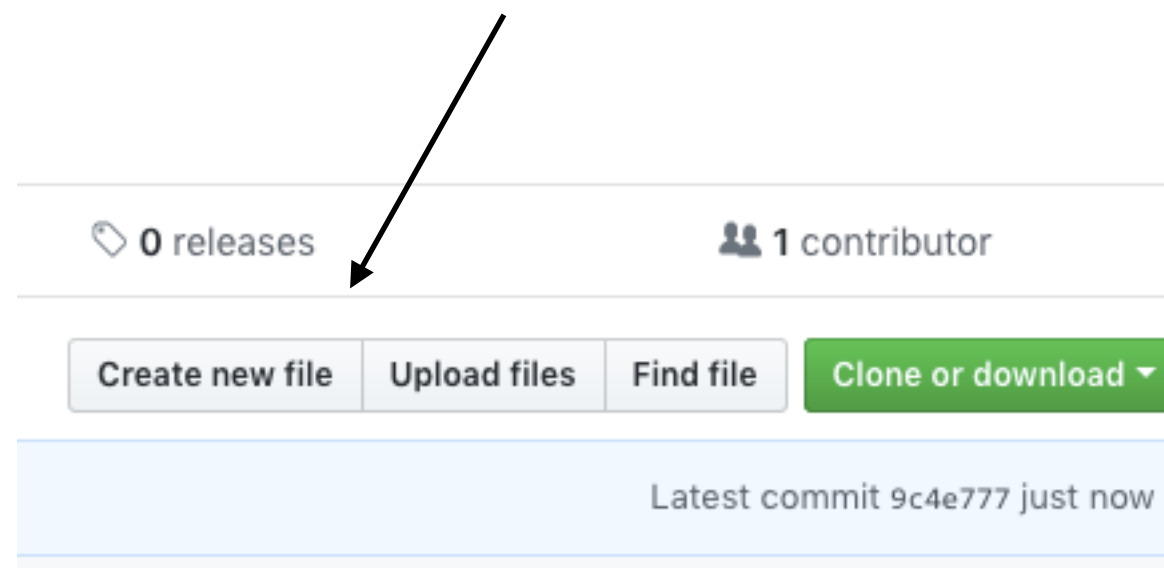
Click here to upload



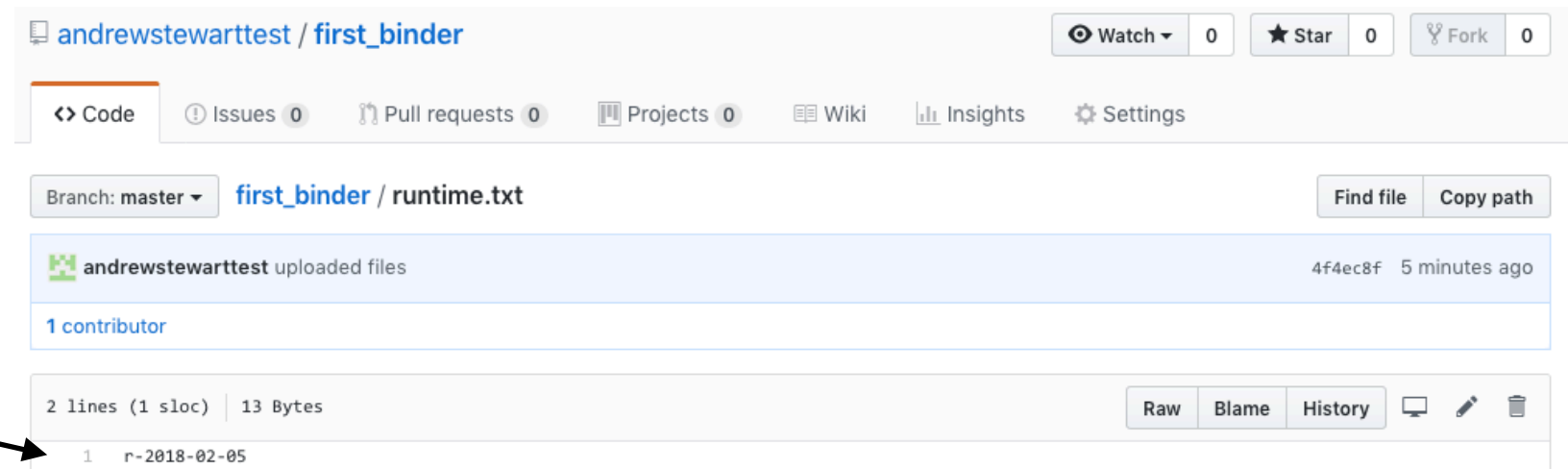
Click here to Commit

# Step 3 - Upload your R script and data and make your first “Commit”

- We need two other files at this point - one is called “runtime.txt” and contains the date 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”



In the  
runtime.txt file  
type the date  
you want in the  
format r-YYYY-  
MM-DD



andrewstewarttest / first\_binder

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Branch: master first\_binder / runtime.txt Find file Copy path

andrewstewarttest uploaded files 4f4ec8f 5 minutes ago

1 contributor

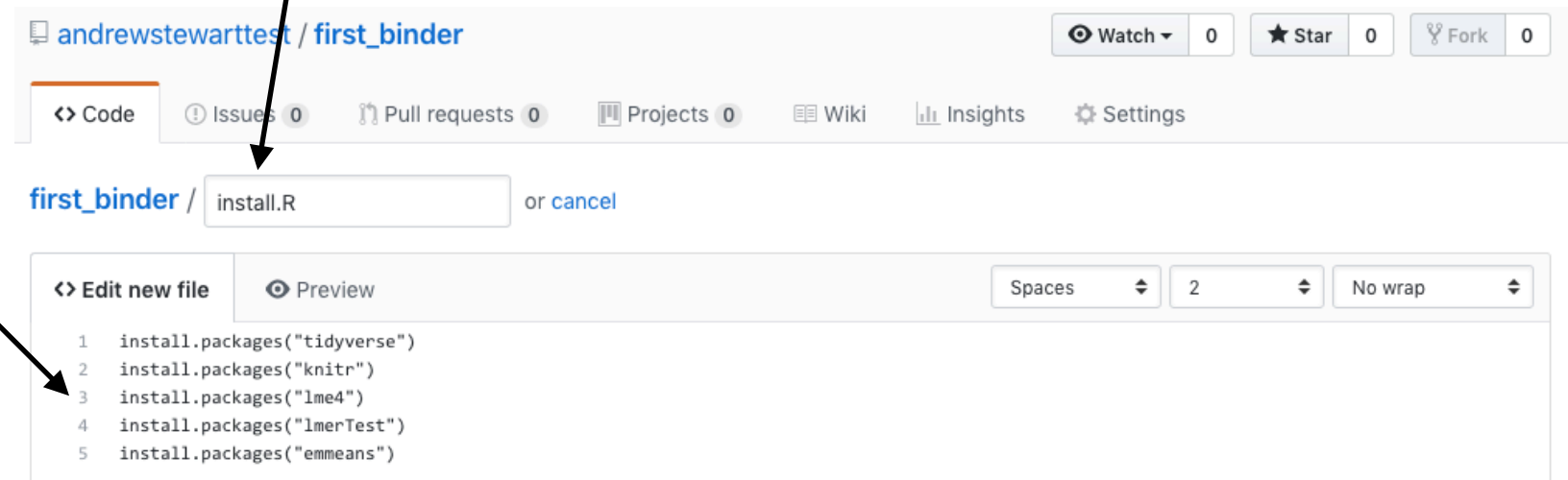
2 lines (1 sloc) | 13 Bytes

Raw Blame History

```
1 r-2018-02-05
```

Name your file

List your  
packages like  
this in the  
install.R file



andrewstewarttest / first\_binder

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

first\_binder / install.R or cancel

Edit new file Preview Spaces 2 No wrap

```
1 install.packages("tidyverse")
2 install.packages("knitr")
3 install.packages("lme4")
4 install.packages("lmerTest")
5 install.packages("emmeans")
```

Don't forget to click "Commit" after you've created each file!

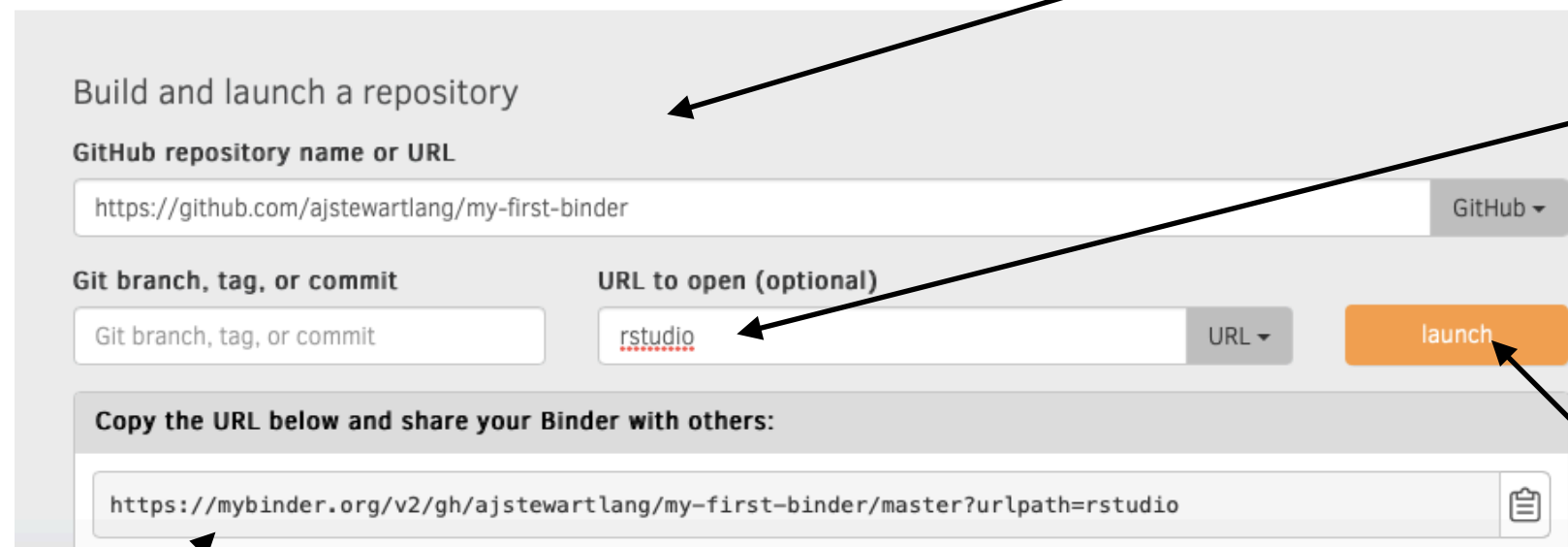


# Step 5 - Now we need to link our repo to Binder (mybinder.org)



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.

A screenshot of the Binder web interface. It shows a form titled "Build and launch a repository". The first field is "GitHub repository name or URL" with the value "https://github.com/ajstewartlang/my-first-binder" and a "GitHub" dropdown. Below this are two fields: "Git branch, tag, or commit" (empty) and "URL to open (optional)" with the value "rstudio" and a "URL" dropdown. To the right of these is an orange "launch" button. Below the form is a section titled "Copy the URL below and share your Binder with others:" containing a text box with the URL "https://mybinder.org/v2/gh/ajstewartlang/my-first-binder/master?urlpath=rstudio" and a clipboard icon. Arrows from the numbered instructions point to the repository URL field, the "rstudio" input, the "launch" button, and the final URL.

1. Paste the link to your repo here.

2. Type rstudio here and select "URL"

3. Then click on "launch"

4. This is the URL to share with others.



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

m

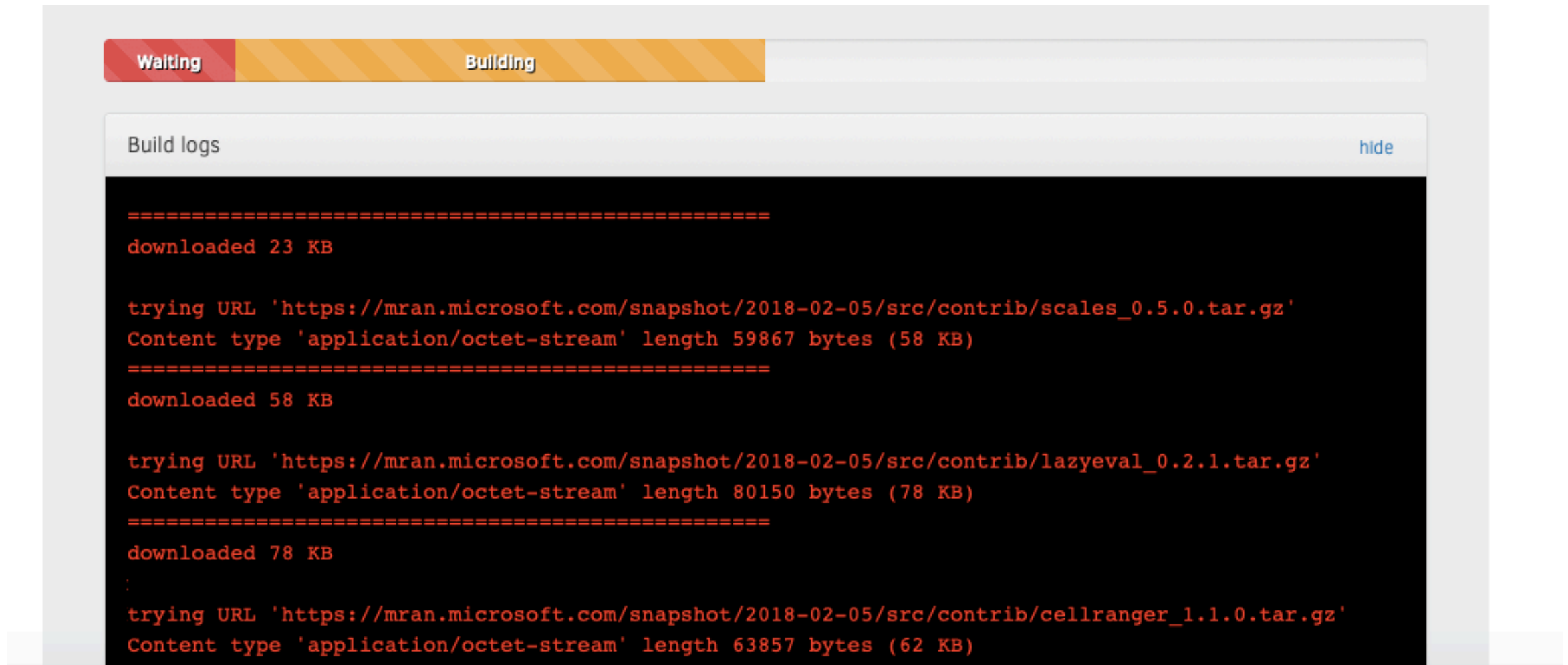
```
[![Binder](https://mybinder.org/badge_logo.svg)](https://mybinder.org/v2/gh/ajstewartlang/Binder_demo/master?urlpath=rstudio)
```

.rst

```
.. image:: https://mybinder.org/badge_logo.svg
   :target: https://mybinder.org/v2/gh/ajstewartlang/Binder_demo/master?urlpath=rstudio
```

- Paste this code into your GitHub repo README.md - you'll then be able to click on the 'launch binder' button in your repository to launch the actual binder once it has been built - makes it easy for others to go from you GitHub repo to your code running in Binder.

# Once you click ‘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)...

# And then...

The screenshot displays the RStudio web interface running on mybinder.org. The browser address bar shows the URL: `hub.mybinder.org/user/andrewstewarttest-first_binder-z4kwp6gl/rstudio/?token=`. The RStudio interface includes a menu bar (File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help) and a toolbar with icons for file operations and running code. The main editor window shows a file named `mixed models.Rmd` with the following R Markdown code:

```
1 ---
2 title: "Linear Mixed Models"
3 author: "Andrew Stewart"
4 output: html_document
5 ---
6
7 ```{r setup, include=FALSE}
8 knitr::opts_chunk$set(echo = TRUE)
9 ```
10
11 ```{r, message=FALSE, warning=FALSE}
12 library(tidyverse)
13 library(lme4)
14 library(lmerTest)
15 library(emmeans)
16 ```
17
18 ## Linear Mixed Models
```

The console window at the bottom shows the R startup messages:

```
~|
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.

> |
```

The file explorer on the right shows the contents of the binder environment:

Name	Size	Modified
data.Rproj	205 B	Mar 1, 2019, 5:39 PM
data1.csv	11.8 KB	Mar 1, 2019, 5:39 PM
data2.csv	19.3 KB	Mar 1, 2019, 5:39 PM
data3.csv	34 KB	Mar 1, 2019, 5:39 PM
image.png	143.8 KB	Mar 1, 2019, 5:39 PM
install.R	138 B	Mar 1, 2019, 5:39 PM
mixed models.Rmd	3 KB	Mar 1, 2019, 5:39 PM
mixed_models.html	1.2 MB	Mar 1, 2019, 5:39 PM
README.md	14 B	Mar 1, 2019, 5:39 PM
runtime.txt	13 B	Mar 1, 2019, 5:39 PM

# Speeding things up...

- You can create a Dockerfile in your GitHub repo which allows a pre-built version of R to be run in Binder (thus saving time) - note that not all packages are in these pre-builds and runtime.txt is ignored via this route.
- Create a new file called “Dockerfile” and paste in the following code:

```
FROM rocker/binder:3.5.0
```

```
# Copy your repository contents to the image  
COPY --chown=rstudio:rstudio . ${HOME}
```

```
## Run an install.R script, if it exists.
```

```
RUN if [ -f install.R ]; then R --quiet -f install.R; fi
```

Branch: master ▾ **Binder\_demo** / Dockerfile Find file Copy path

 **ajstewartlang** Update Dockerfile 1e73a3a 17 minutes ago

1 contributor

8 lines (5 sloc) | 214 Bytes Raw Blame History   

```
1 FROM rocker/binder:3.5.0
2
3 # Copy your repository contents to the image
4 COPY --chown=rstudio:rstudio . ${HOME}
5
6 ## Run an install.R script, if it exists.
7 RUN if [ -f install.R ]; then R --quiet -f install.R; fi
```

- Click on ‘commit’ and then build your Binder as normal...

# 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 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.
- Some R packages need system-level packages to also be installed - you can do that via an additional apt.txt file which lists those packages - this is used by apt-install to install those packages from the Ubuntu apt repository.

# A few other things...

- At the moment, you can't change the version of R that runs on Binder (currently set to 3.4.4.) so need to go down the Rocker route but be aware that you may not get the right version of the packages that you want...
- Something to think about if you're using `set.seed()` and `sample()` in R 3.6 - if so, consider using (e.g.) `RNGversion("3.5.3")` for pre-3.6 reproducibility.
- You can close your laptop if Binder is taking too long - the image and your Binder will continue to be built in the Cloud. And it's always a good excuse for another coffee...



# For Ultimate Reproducibility

- Make sure you have updated all your packages before you run your script.
- Build your Binder and specify the day you ran your analysis in the runtime.txt file
- Don't use pre-built Dockerfiles (as these likely won't have the same package versions that you are using).
- Patience while your Binder builds...

# Slides

[http://ajstewartlang.github.io/Binder\\_slides.pdf](http://ajstewartlang.github.io/Binder_slides.pdf)

**My step-by-step guide (including an example R script if you don't have one)**

<https://hackmd.io/gO3cehAVSpuCB2EDEvepzg?view>