UCL FESTIVAL OF CODE

Celebrating Research Software & Coding Communities

Code sharing as an Early Career Researcher

The good, the bad & the ugly!

Dr Louise Mc Grath-Lone, Rachel Pearson, Dr Ania Zylbersztejn,

Aims of this session

- Highlight the benefits (and potential pitfalls) of code sharing
- Outline the key steps in establishing, maintaining and contributing to a code repository, drawing on our experiences
- Gather your views on the barriers to code sharing that are faced by Early Career Researchers at UCL

What is code sharing?

Code sharing is a loosely defined process that includes:

Sharing code publicly

- E.g., through a journal or online repository like GitHub

Sharing code informally

- E.g., with a colleague to help them or for review purposes

What are the benefits of code sharing?

aka “the Good”

Note: This is not an exhaustive list, but we chose some of the main benefits that are likely to impact many early career researchers at UCL who code.

1. Reduces duplication of effort

- As researchers, we aim to make novel contributions to our field of knowledge.

- But, there are lots of parts of our work that aren't novel. For example:
  - "Wrangling" (e.g., extracting the data you need, linking it together)
  - Cleaning (e.g., de-duplicating data, identifying outliers/errors)
  - Preparation (e.g., harmonizing variables over time, deriving variables)
  - Visualisation (e.g., formatting charts)

- One benefit of code sharing is that it reduces the duplication of effort among researchers for these non-novel activities, freeing up more time for the fun stuff!

2. Improves the transparency of your research

- As researchers, we are good at making our results openly available (e.g., though paper).
- However, the way you arrived at the results is just as important as what you found!

Child Abuse & Neglect 63 (2017) 73–83

Contents lists available at ScienceDirect

Child Abuse & Neglect

Factors associated with re-entry to out-of-home care among children in England

Louise Mc Grath-Lone [a,b,*], Lorraine Dearden [a,c], Katie Harron [d], Bilal Nasim [a,c], Ruth Gilbert [a,b]

[a] Administrative Data Research Centre for England, University College London, London, UK
[b] UCL Great Ormond Street Institute of Child Health, University College London, London, UK
[c] UCL Institute of Education, University College London, London, UK
[d] Department of Health Service Research and Policy, London School of Hygiene and Tropical Medicine, London, UK

ARTICLE INFO

Article history:
Received 11 August 2016
Received in revised form
11 November 2016
Accepted 16 November 2016
Available online 28 November 2016

Keywords:
Looked after children
Out-of-home care
Re-entry
Permanence

ABSTRACT

Exiting and re-entering out-of-home care (OHC) is considered a disruption to permanence which may have long-lasting, negative consequences for children due to a lack of stability and continuity. Each year approximately one-third of children in OHC in England exit, but information is lacking on rates of re-entries and associated factors. Using national administrative data, we calculated rates of re-entry among children exiting OHC from 2007 to 2012, identified key child and care factors associated with re-entry using Cox proportional hazards modelling, and developed a simple probability calculator to estimate which groups of children are most likely to re-enter OHC within three months. Between 2007 and 2012 re-entries to OHC in England decreased (from 23.3% to 14.4% within one year of exit, $p < 0.001$), possibly due to concurrent changes in the way children exited OHC. Overall, more than one-third of children exiting OHC in 2008 re-entered within five years (35.3%, $N = 4076$), but rates of re-entry varied by child and care characteristics including age, ethnicity, mode of exit, and placement stability. Based on these associated factors, we developed a calculator that can estimate the likelihood of rapid re-entry to OHC for a group of children and could be used by social care practitioners or service planners. Our findings provide insight into which groups of children are most likely to re-enter OHC, who may benefit from additional support or ongoing monitoring.

1. Introduction

A central goal of England's social care system is to ensure that children have permanence (Department for Education, 2015a). This permanence (i.e. emotional, physical, and legal security, stability, and continuity (Department for Education, 2013)) helps children develop and maintain a sense of identity and belonging during childhood and beyond (Thomas, 2013). Most children in the care of the State (who are known as looked-after children) are placed in out-of-home care (OHC), such as with a foster carer or in a children's group home. OHC can provide permanence to children – through stable, long-term foster care, for example. However, current policy favours achieving permanence in a permanent family setting outside of the OHC system, with a particular focus on adoption (Department for Education, 2016a; Department of Health, 2000).

* Corresponding author at: ADRC-E, 222 Euston Road, London, NW1 2DA, UK.
E-mail address: louise.mc-grath-lone.13@ucl.ac.uk (L. Mc Grath-Lone).

http://dx.doi.org/10.1016/j.chiabu.2016.11.012
0145-2134/

Scriberia

2. Improves the transparency of your research

Sharing data may not always be in your control, but sharing your code is one way to help people to understand what exactly it is that you did in your research.

3. Captures your “preparatory” work

Child Abuse & Neglect 63 (2017) 73–83

Contents lists available at ScienceDirect

Child Abuse & Neglect

Factors associated with re-entry to out-of-home care among children in England

Louise Mc Grath-Lone[a,b,*], Lorraine Dearden[a,c], Katie Harron[d], Bilal Nasim[a,c], Ruth Gilbert[a,b]

[a] Administrative Data Research Centre for England, University College London, London, UK
[b] UCL Great Ormond Street Institute of Child Health, University College London, London, UK
[c] UCL Institute of Education, University College London, London, UK
[d] Department of Health Service Research and Policy, London School of Hygiene and Tropical Medicine, London, UK

ARTICLE INFO

Article history:
Received 11 August 2016
Received in revised form
11 November 2016
Accepted 16 November 2016
Available online 28 November 2016

Keywords:
Looked after children
Out-of-home care
Re-entry
Permanence

ABSTRACT

Exiting and re-entering out-of-home care (OHC) is considered a disruption to permanence which may have long-lasting, negative consequences for children due to a lack of stability and continuity. Each year approximately one-third of children in OHC in England exit, but information is lacking on rates of re-entries and associated factors. Using national administrative data, we calculated rates of re-entry among children exiting OHC from 2007 to 2012, identified key child and care factors associated with re-entry using Cox proportional hazards modelling, and developed a simple probability calculator to estimate which groups of children are most likely to re-enter OHC within three months. Between 2007 and 2012 re-entries to OHC in England decreased (from 23.3% to 14.4% within one year of exit, $p < 0.001$), possibly due to concurrent changes in the way children exited OHC. Overall, more than one-third of children exiting OHC in 2008 re-entered within five years (35.3%, $N = 4076$), but rates of re-entry varied by child and care characteristics including age, ethnicity, mode of exit, and placement stability. Based on these associated factors, we developed a calculator that can estimate the likelihood of rapid re-entry to OHC for a group of children and could be used by social care practitioners or service planners. Our findings provide insight into which groups of children are most likely to re-enter OHC, who may benefit from additional support or ongoing monitoring.

1. Introduction

A central goal of England's social care system is to ensure that children have permanence (Department for Education, 2015a). This permanence (*i.e.* emotional, physical, and legal security, stability, and continuity (Department for Education, 2013)) helps children develop and maintain a sense of identity and belonging during childhood and beyond (Thomas, 2013). Most children in the care of the State (who are known as looked-after children) are placed in out-of-home care (OHC), such as with a foster carer or in a children's group home. OHC can provide permanence to children – through stable, long-term foster care, for example. However, current policy favours achieving permanence in a permanent family setting outside of the OHC system, with a particular focus on adoption (Department for Education, 2016a; Department of Health, 2000).

* Corresponding author at: ADRC-E, 222 Euston Road, London, NW1 2DA, UK.
E-mail address: louise.mc-grath-lone.13@ucl.ac.uk (L. Mc Grath-Lone).

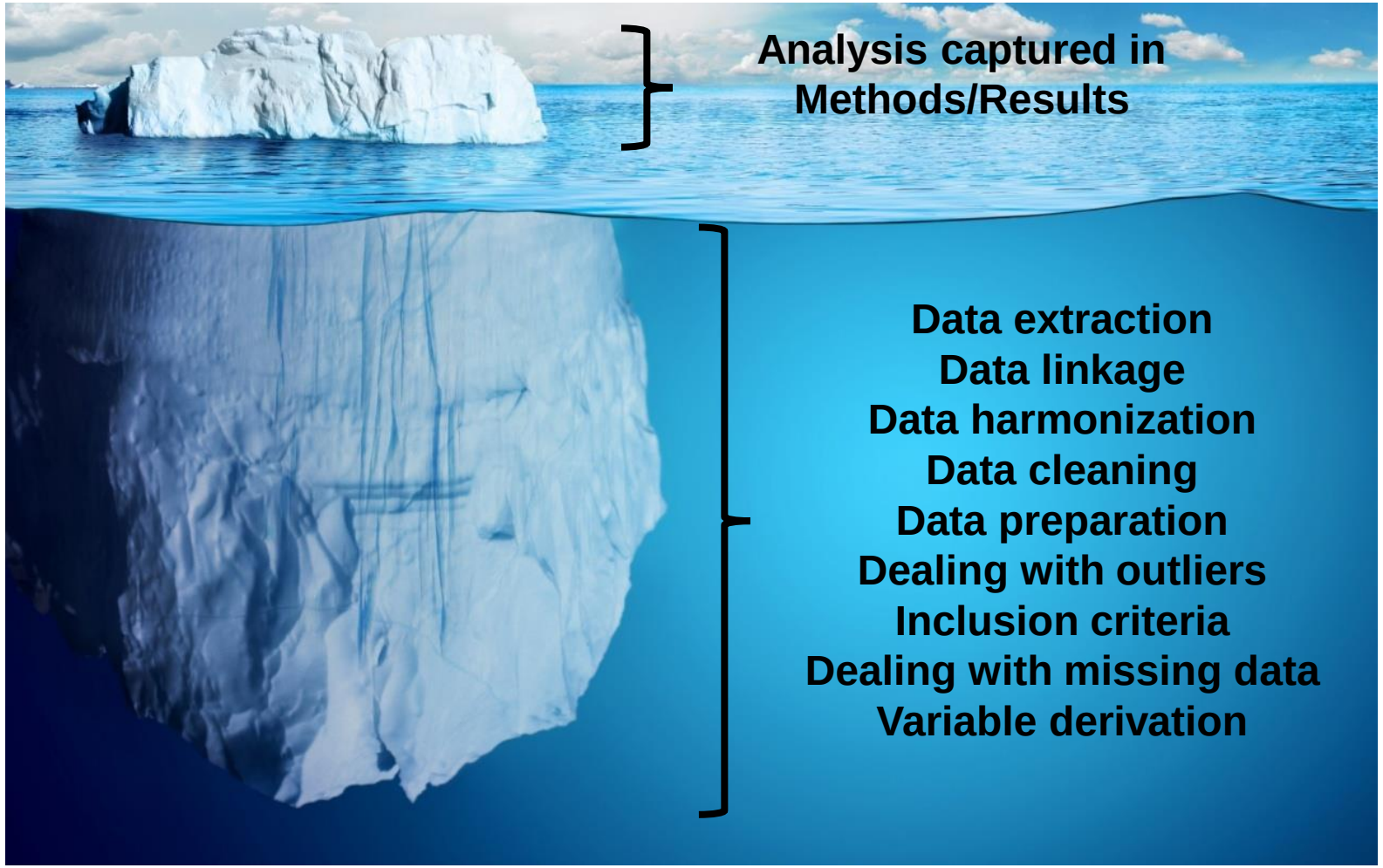
http://dx.doi.org/10.1016/j.chiabu.2016.11.012
0145-2134/

- In my field, it can take months to prepare a dataset and the “actual” research where the dataset is queried using a model takes minutes.
- For example, this extract from a paper describes the work I did to prepare data for analysis:

“For this study, we derived a data extract for children who were placed in care for non-respite reasons. Exits were identified and categorized, as per Supplementary Table S1.”

- These 28 words summarise 1,000+ lines of code and 6 months of my work!

3. Captures your “preparatory” work



- The information we provide in the methods/results sections of a paper is often just the tip of the iceberg in terms of the work we have done.
- The data management and preparation activities we undertake is often hidden.
- Code sharing is one way to demonstrate and get a tangible output for your “preparatory” work.

4. Increases the reproducibility of your work

- Reproducing research without the full instructions is a real challenge!
- As researchers, we tend to rely on describing our methods clearly as a means of making our research reproducible.
- However, code sharing is another way to increase the reproducibility of your work.
- For example, code sharing will allow others to validate your work (e.g., through Reprothons) or apply your methods to new data sources.
- From an ECR perspective, this can translate into more citations or new collaborations.

4. Increases the reproducibility of your work

- We are in the midst of a reproducibility crisis in research.
- Code sharing is a way to tackle the problem.

npj | digital medicine

Explore content ⌄ Journal information ⌄ Publish with us ⌄

nature › npj digital medicine › perspectives › article

Perspective | Open Access | Published: 29 January 2019

The reproducibility crisis in the age of digital medicine

Aaron Stupple, David Singerman & Leo Anthony Celi ✉

npj Digital Medicine **2**, Article number: 2 (2019) | Cite this article

5715 Accesses | **21** Citations | **39** Altmetric | Metrics

Medical studies have replication rates as low as **44%!**

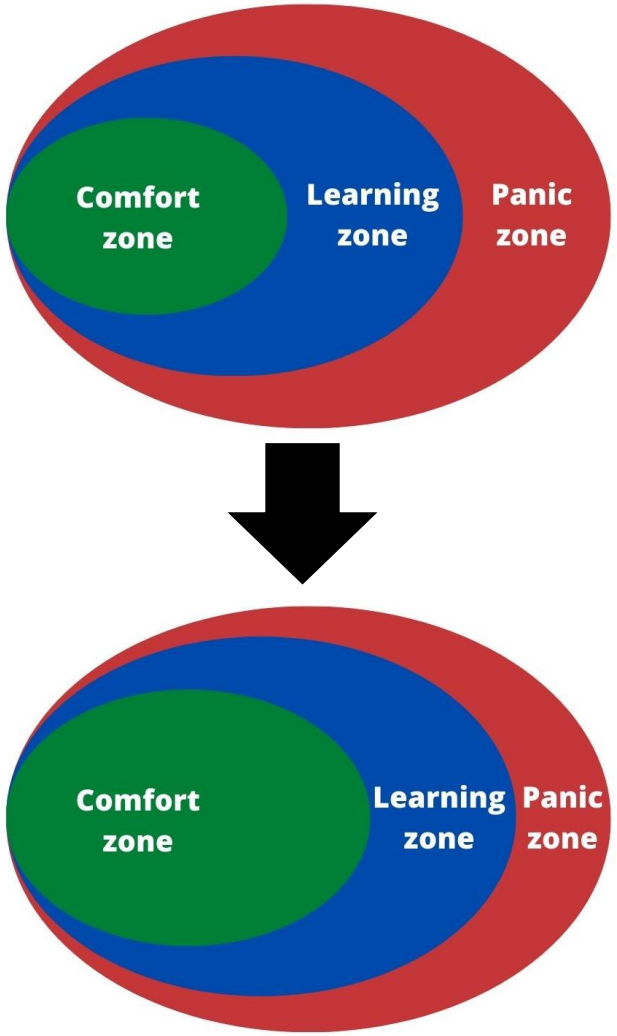
5. Enables the continuity of your work

- The reality for many ECRs is a period of fixed-term and often short research contracts.
- When your project comes to the end of the road, code sharing is a way to enable the continuity of your work after you've moved on.
- This increases the chances of the work reaching the paper stage and your efforts and inputs being recognized in a peer-reviewed publication.

6. Builds your reputation & networks

- The stereotype of the solitary academic working in isolation in their lab or office is not how modern research works.
- Even if you are not part of a large team or working directly with colleagues, you will be part of a broader network of researchers.
- Code sharing is a way for you to build your reputation and develop your networks.

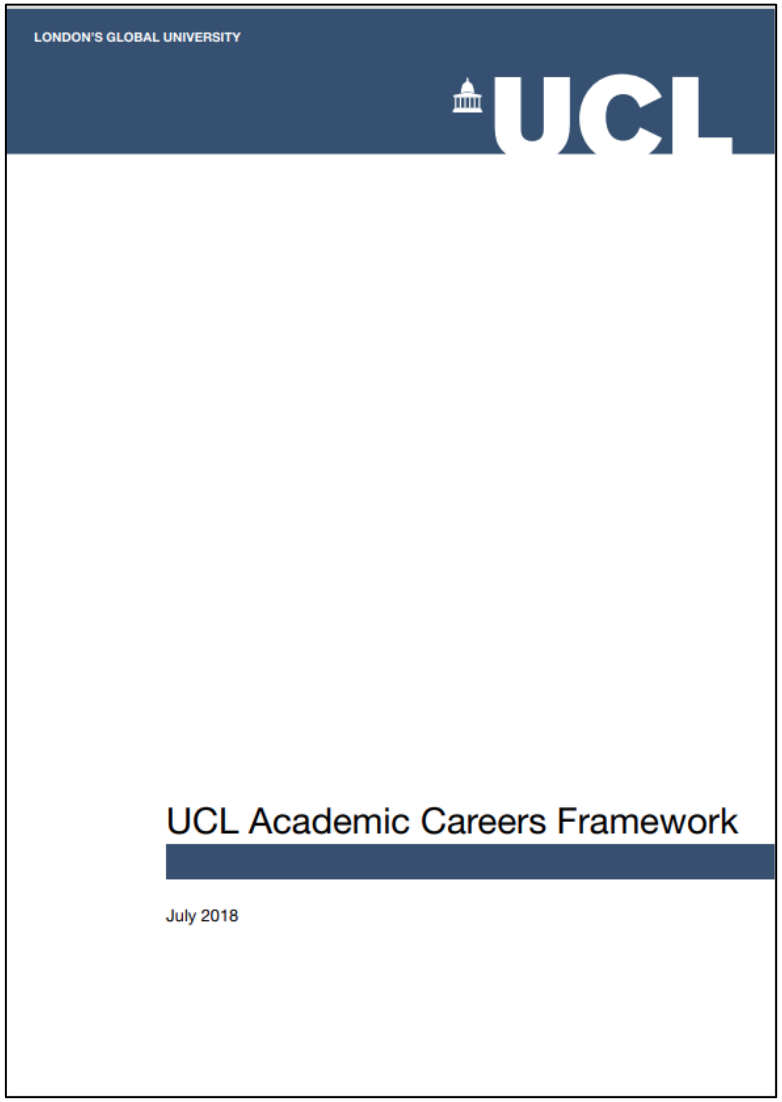
7. Provides opportunities for learning and teaching

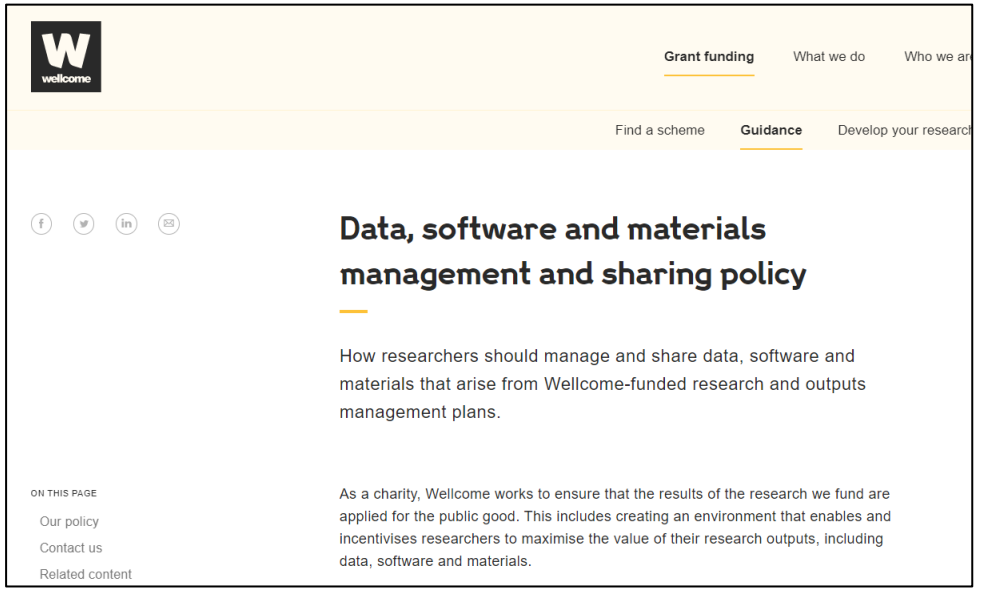


- In Vygotsky's zones of proximal development (a model of knowledge acquisition) for any skill, there will be:
  - things you can do well (comfort zone)
  - things you can do with some help and support (learning zone)
  - things you can't do yet even with help (sometimes called the panic zone!).

- By sharing code, and seeing code that other people have shared, you have opportunities to both teach and learn thereby expanding your comfort zone and improving your coding skills.

8. Shows a commitment to Open Science Principles





- Demonstrating a commitment to Open Science is increasingly important for progressing in your academic career (e.g., it is part of the UCL Academic Careers Framework).
- Code sharing is a tangible way to demonstrate your commitment to Open Science.
- Code sharing is also increasingly required by funders (e.g., Welcome Trust) and journals (e.g., Biometrical Journal).

What are the barriers to code sharing?

aka “the Bad” and “the Ugly”

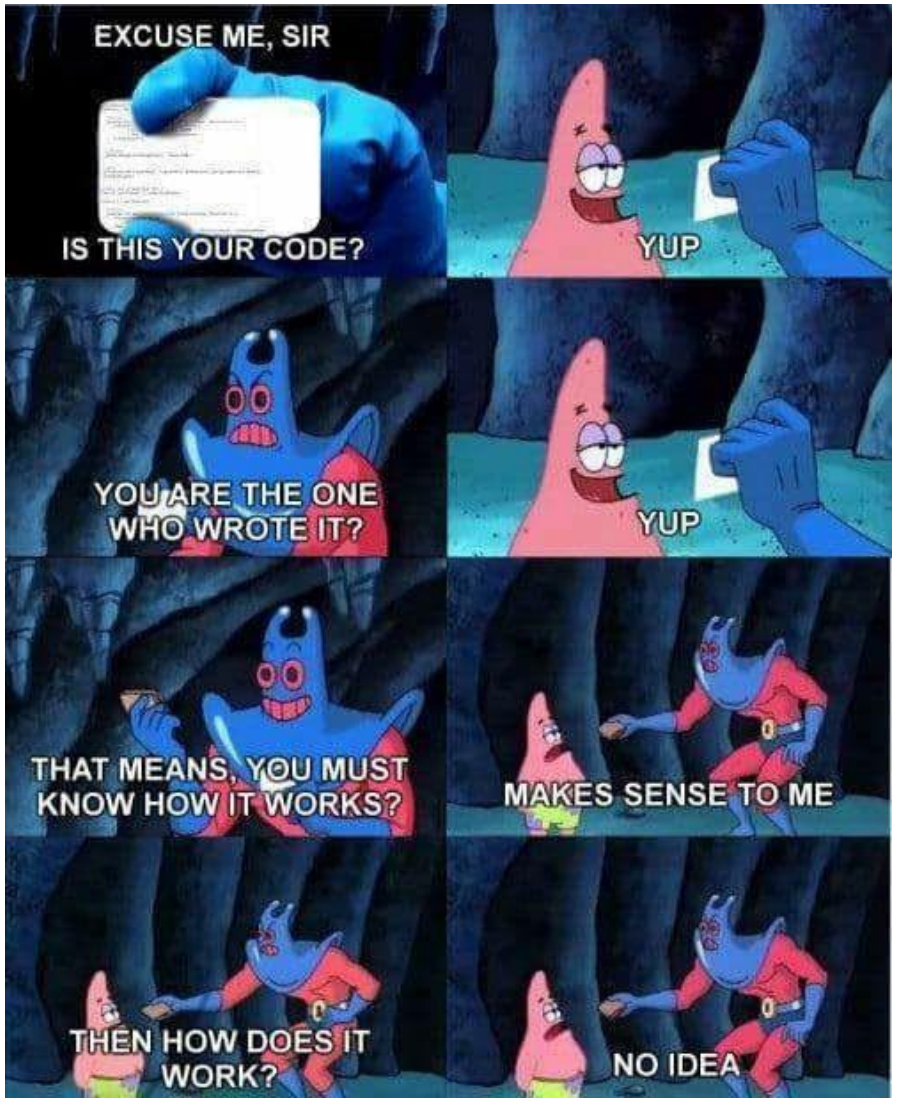
Note: This is not an exhaustive list but we chose some of the main barriers that are likely to impact many early career researchers at UCL who code

(Thanks also to Emma Vestesson who shared her insights on barriers/facilitators to code sharing which informed this section)

Thinking code not ‘good’ enough to be shared

- Maybe you’ve not written something particularly elegantly, or you’ve taken an unconventional approach that might confuse others
- Maybe you’ve given up on commenting your code after making several updates
- Perhaps what started off as a small and simple program has turned into a very large program that ideally would be split into a number of smaller scripts.

Time and effort required



- Sharing code often requires significant investment of time and effort – one more task to add to the list
- By the time your paper gets published you've likely forgotten what all the code does & might feel you need to tidy it up
- If working in a Trusted Research Environment, you may need to thoroughly check code to remove person identifiable information before code can be output
- You might be required to register with a repository like GitHub which could require learning how to use it

No one to share code with

- Maybe you are the only person in your group using a specific type of programming language
- Maybe you are the only person on your project or in your group using a programming language at all

Fear, worry or a lack of confidence

A number of people in the polls mentioned they worried about their code being reused without permission. Hosting your code somewhere like GitHub allows you choose a suitable license for your code which can help prevent undesired use while still supporting open science! You can also have a private GitHub repository just for sharing code among your research group/project team (see our later slides).

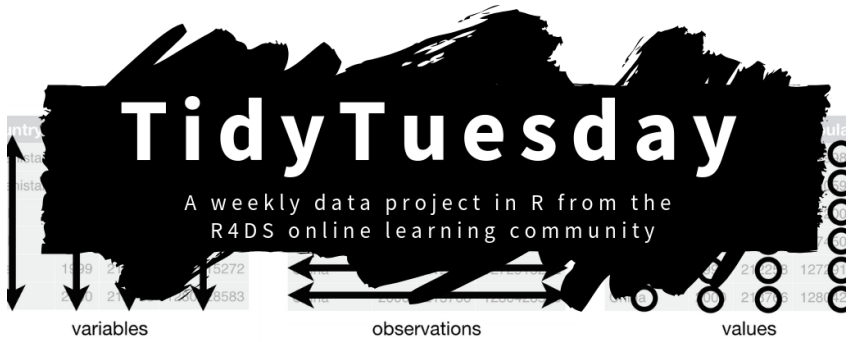
Then there's the 'the ugly' of code sharing (though this is very rare!!)

- we might worry that sharing code publicly could invite unconstructive or nasty criticism
- That we might get called out on our inelegant code or worse than that, someone might tell us our code is wrong and not doing what we think it should be.
- A big worry can be that someone will find a serious error in our code after we've already published the results

How can you overcome these barriers?

Get involved in inclusive, supportive online coding communities

Please use the chat to share further suggestions!



- Getting involved with online coding communities can help build your confidence with sharing code
- For example, R4DS's Tidy Tuesday project offers the opportunity to try out new data viz and analytical techniques each week & to learn from one another
- Via hashtags for these communities on twitter, you can easily connect with other people & you are almost guaranteed to get supportive and constructive feedback (or at the very least some confidence boosting likes or retweets!)

Look for opportunities to share your code

- research group/department ECR groups
- journal clubs or special interest groups
- networking events like seminar series or conferences
- UCL coding communities – they specifically encourage people to get together to share code and provide advice/support
- Set up a GitHub to easily share a URL to your code & make it easy for others to contribute if collaborating

Create your own internal code review network

If all goes well and you find lots of people wanting to receive and give code review, then you could even formalize it by setting up your own code review group (like a journal club but for sharing code).

You could also share what you've learnt with staff/students by delivering workshops on code review (with some funding from UCL). For example:

- UCL ChangeMakers funding supports staff-student partnership work – the award is for £450 (small project) or £700 (large project) towards reward and recognition of students. The money can pay student stipends e.g. for delivering an event to staff and students

- UCL Researcher-led Initiative Awards offered by UCL HR Organisational Development for 'short-term, well-defined initiatives that develop and deliver transferable skills training experiences and/or resources to the applicants' peers across all departments'. These must be delivered by and for early career researchers (i.e. PGR student and grades 7/8)

Shift your perception of “shareable” code

Sharing any code is often better and more helpful to others than sharing nothing at all.

Invest in yourself

- Join UCL code clubs to learn from others
  - http://github-pages.ucl.ac.uk/CodeClubs/
  - https://www.ucl.ac.uk/research/domains/eresearch/developing-technical-skills-good-practice-careers/develop-better-research-software
  - Learn how to use code repositories like GitHub - https://www.ucl.ac.uk/isd/services/research-it/research-software-development-tools/support-for-ucl-researchers-to-use-github
  - Follow/develop a coding style guide (examples below) to help you produce shareable code (that doesn't require hours of tidying up!) and to help standardise your coding across projects (making it easier to work out what is going when you return to your code months/years later).
    - https://jef.works/R-style-guide/ (R)
    - https://michaelshill.net/2015/07/31/in-stata-coding-style-is-the-essential/ (Stata)
  - endeavour to write your code in a way it could be shared (clear, concise and well commented) – this will also help future you!

Where do I start?

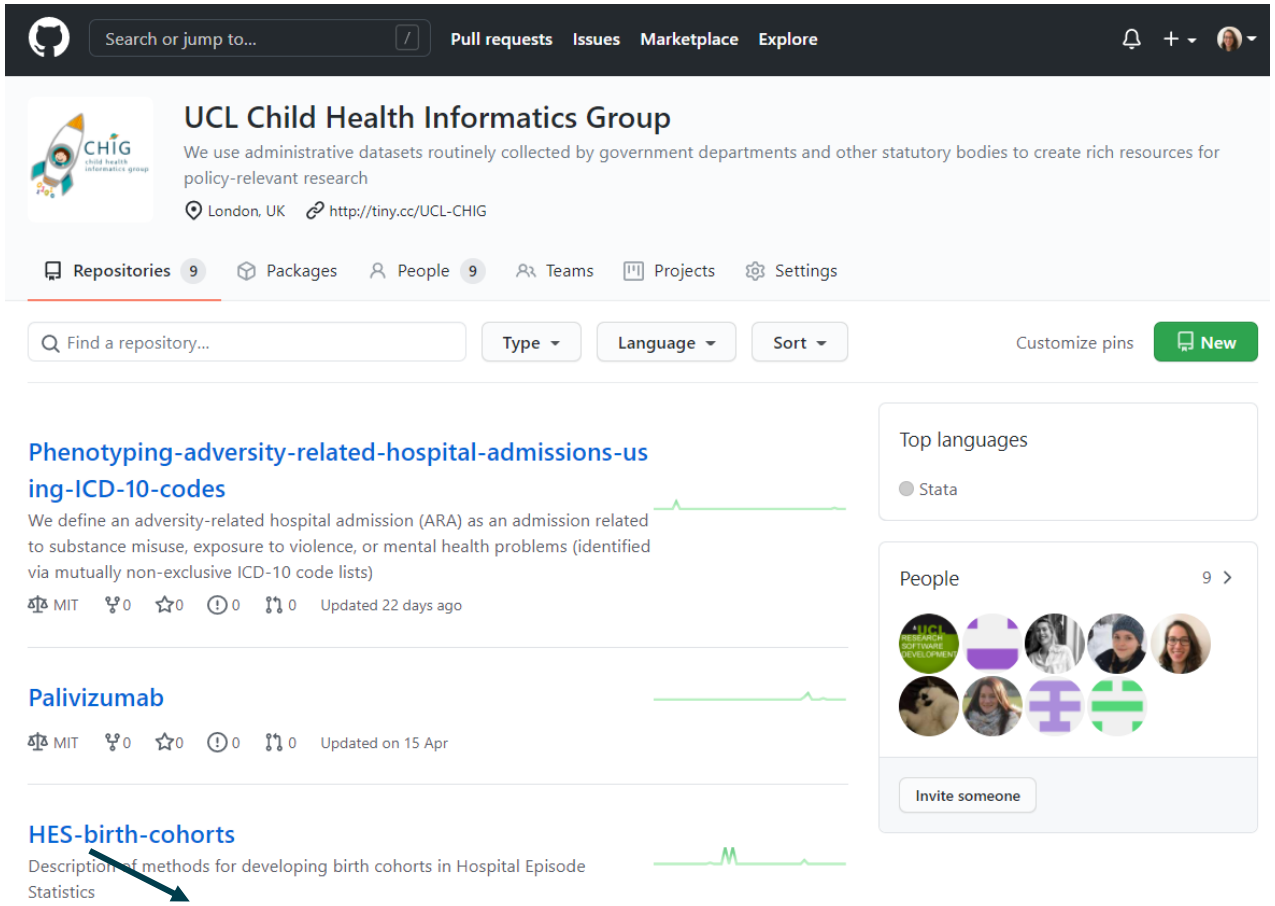
Tips for code sharing using GitHub

Why use GitHub? Our experience

Why we chose GitHub

We wanted to have space where we can share:

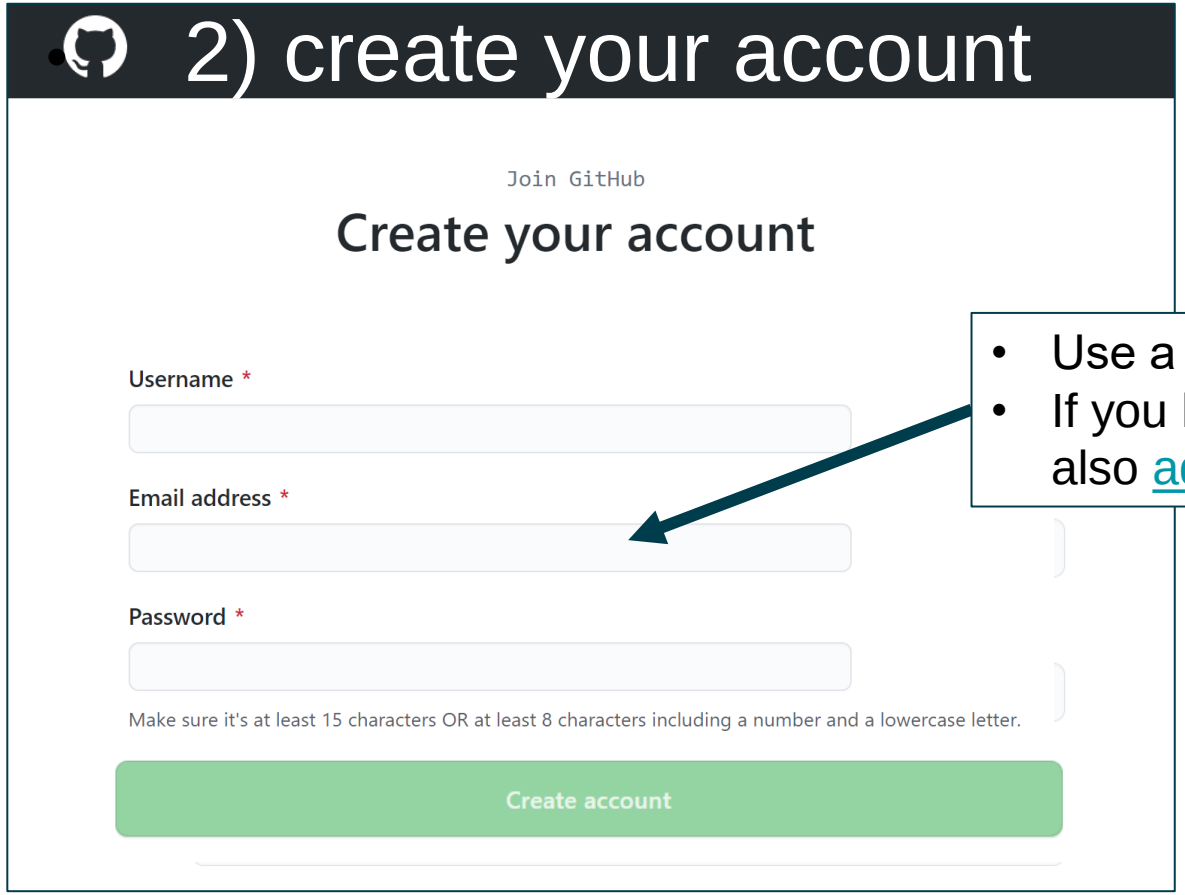
1. Finalised code (to include in publications) and keep a record of how often it's used
2. Code for ongoing work to share privately within our research team



removed infants who were not resident in England. Details of data cleaning are described in S2 Appendix, and Stata code for cohort derivation can be found at https://github.com/UCL-CHIG/HES-birth-cohorts.

1. Set up your personal GitHub account

1) Go to https://github.com

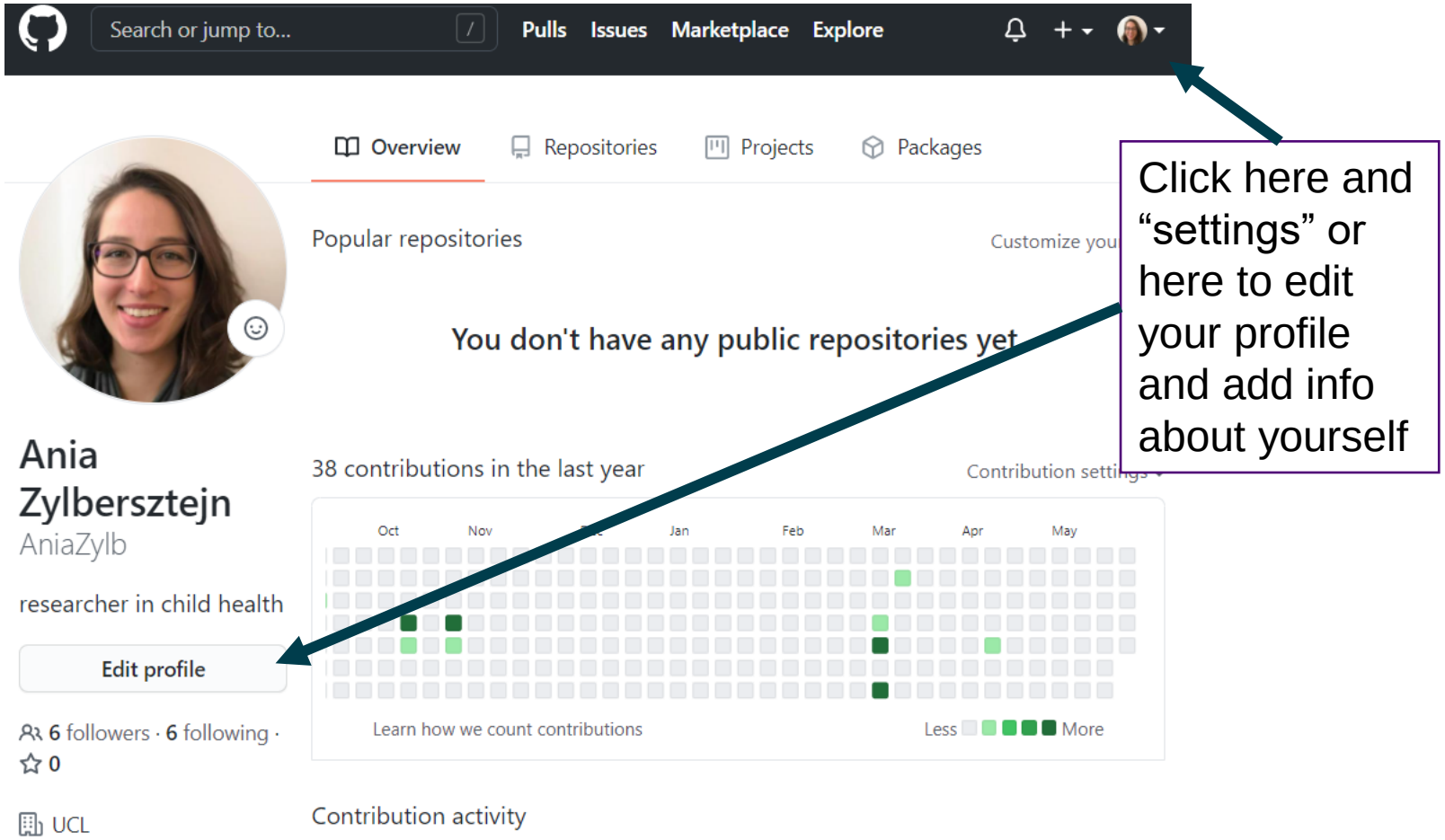


Useful resources:
Have a look at tutorials on creating personal GitHub account from:
1) UCL
2) HDR UK (hosted on GitHub)!

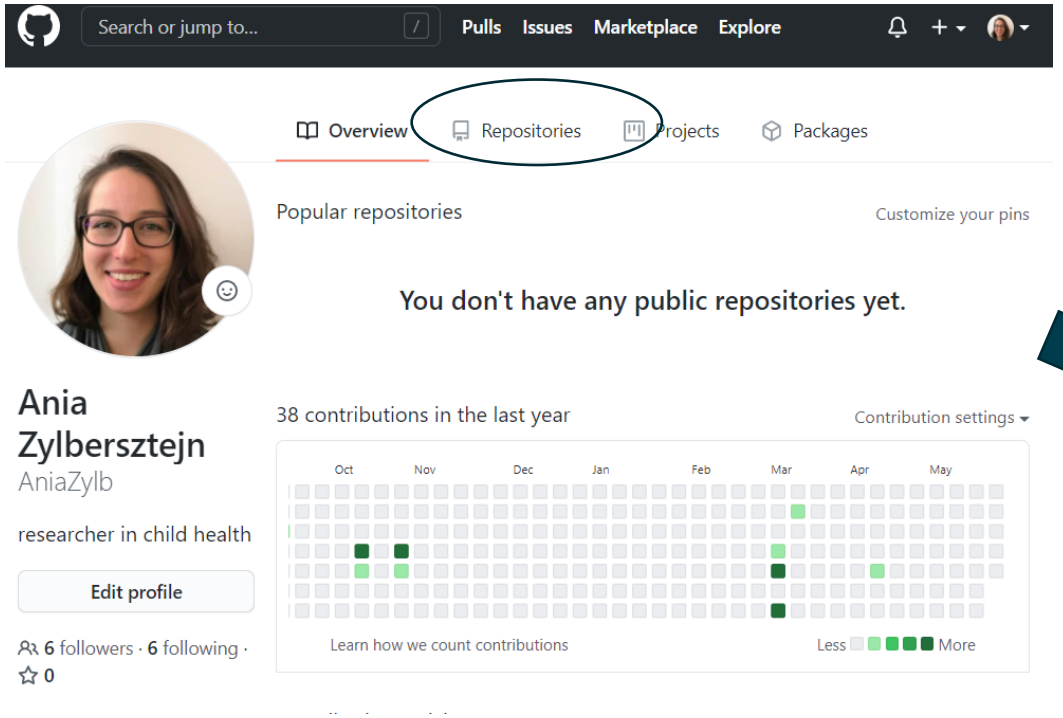
- Use a “ucl.ac.uk” email address
- If you have already an account you can also add your ucl email to your account

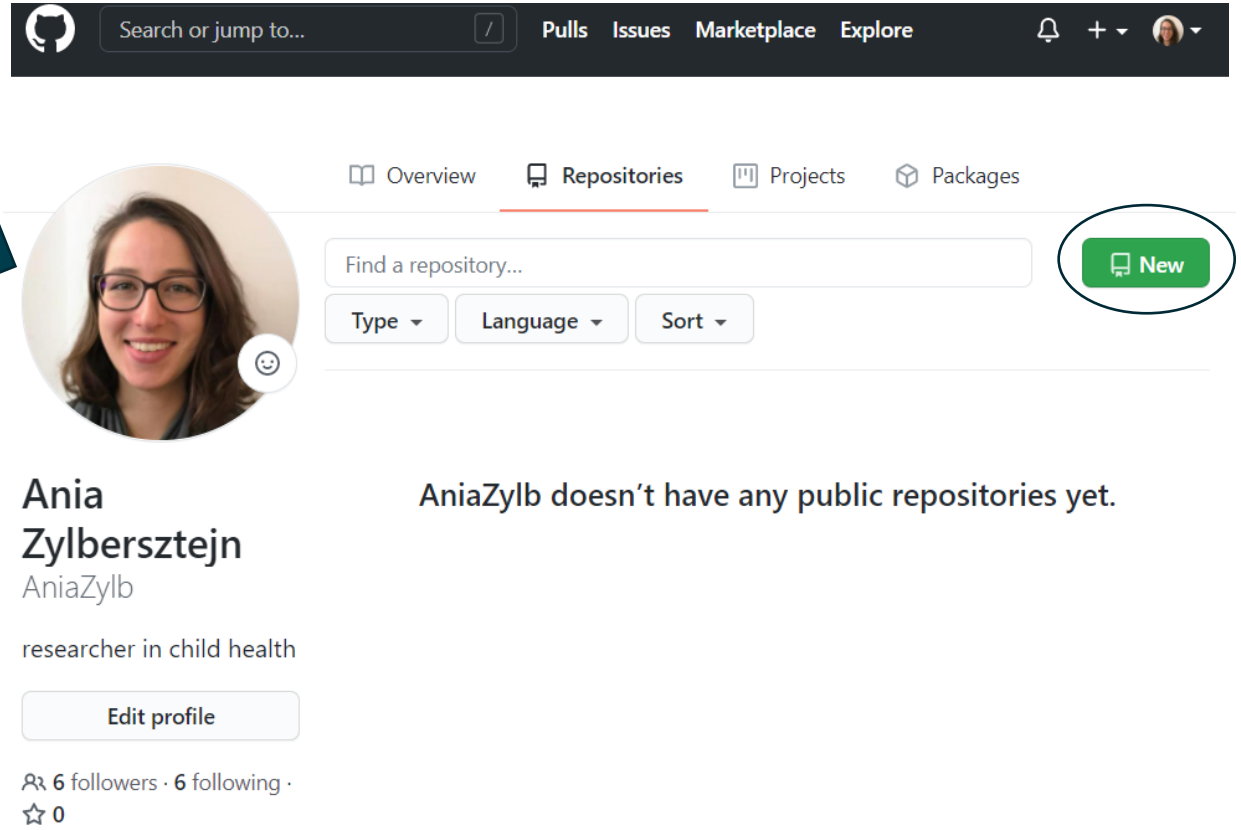
3) Pick a plan:
free plan covers unlimited private and public repositories

You’re ready to go!

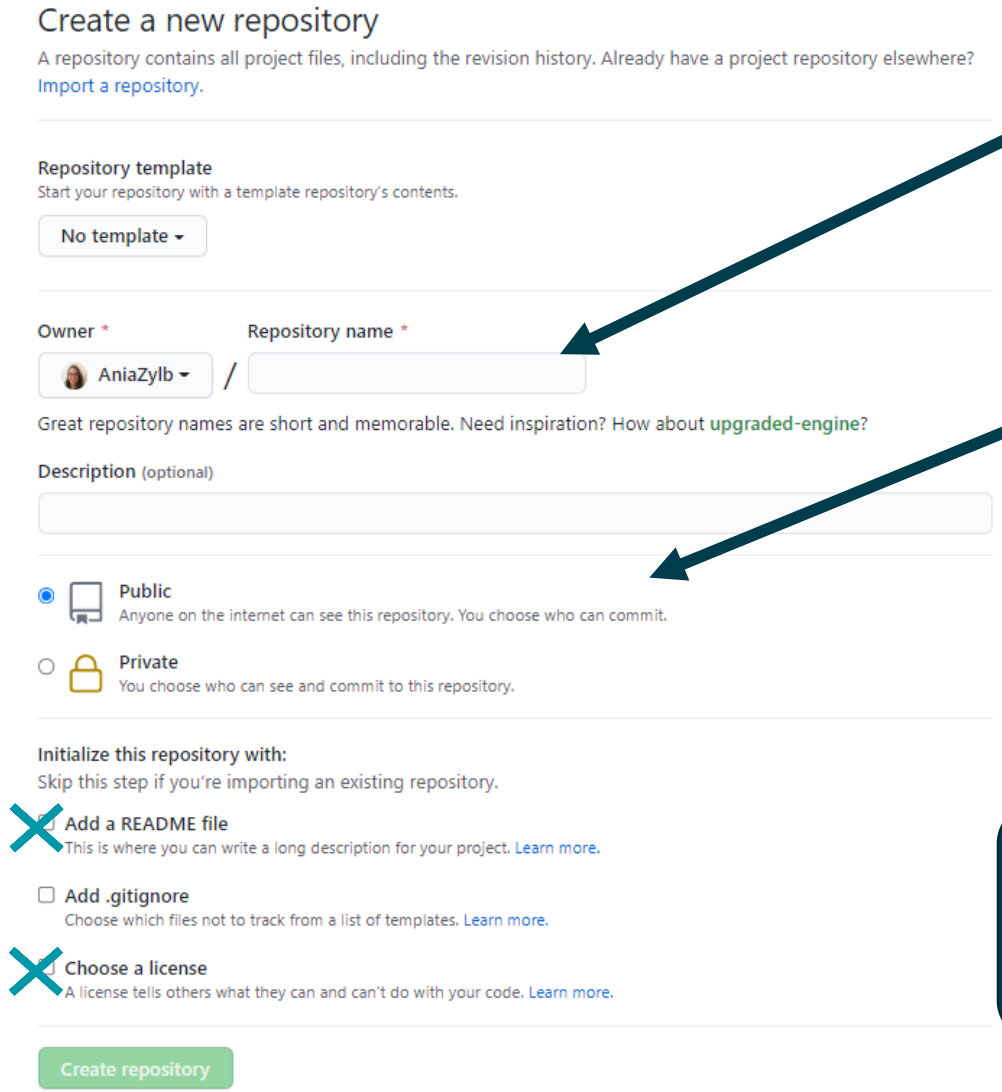


2. Create a repository





2. Create a repository



Name (no spaces)

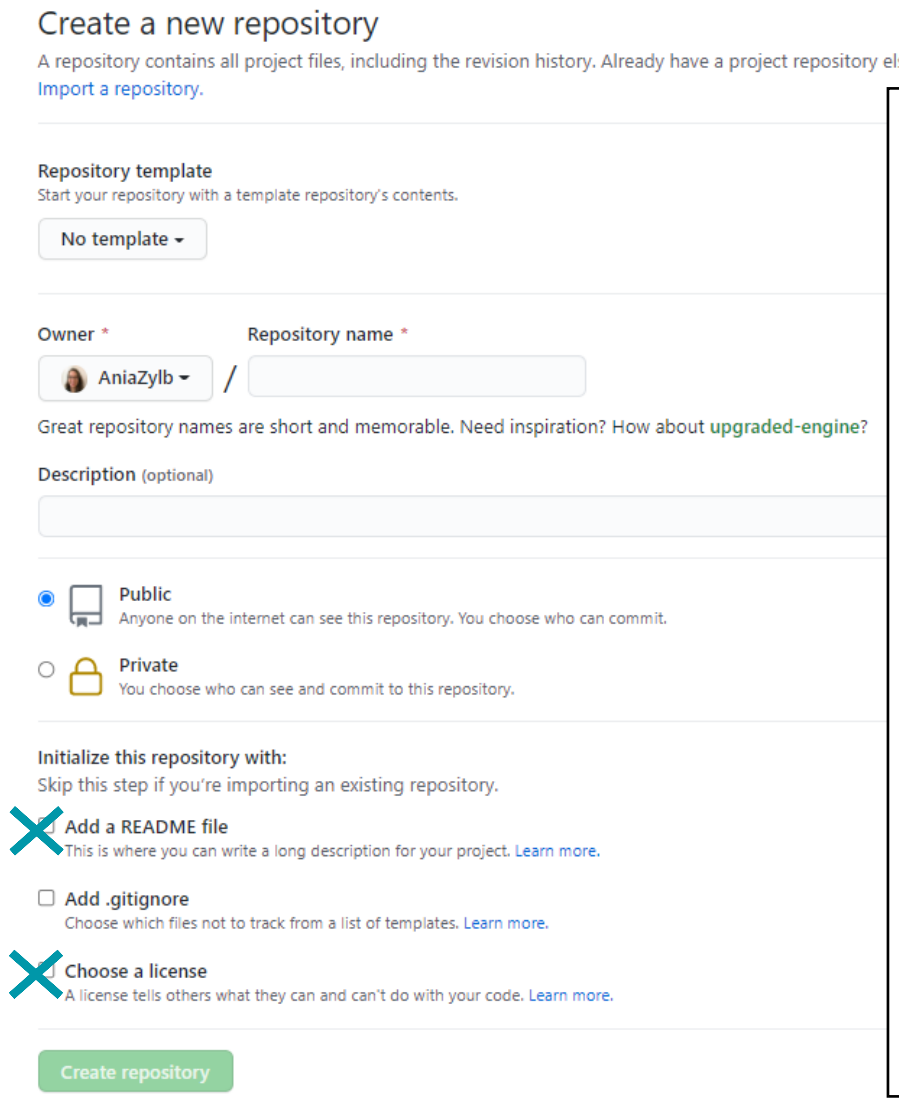
Public repositories can be browsed and downloaded by anyone;
Private repositories are only visible by you and ppl you decide to share with;

You can switch between them:

- Still work in progress – private?
- Ready to include in papers – public?

Useful resources:
Have a look at this tutorial from HRD UK on creating your first repository

Which licence to choose?



Licence: it's also important to tell people how they can use your code

We use MIT licence – most flexible. But, this will depend on your project.

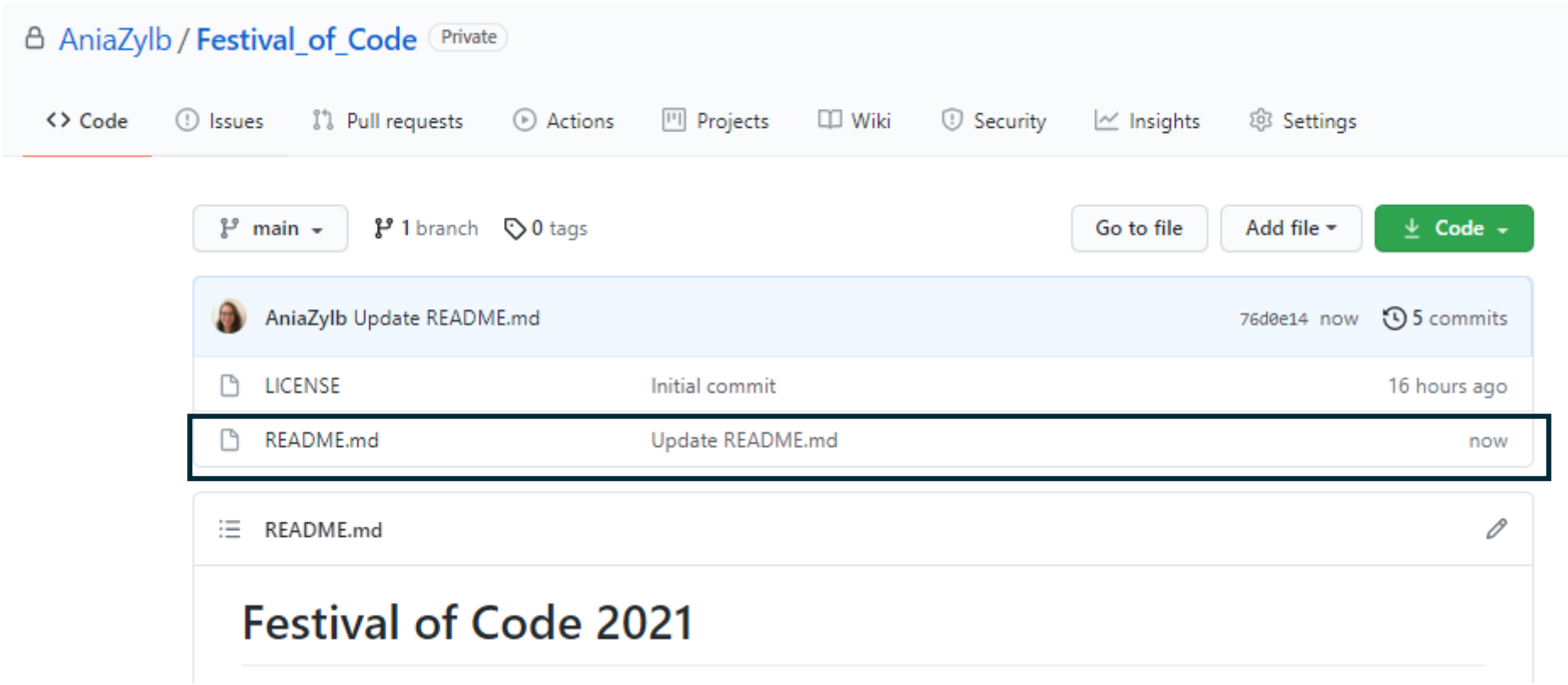
Here are some resources to help you pick the right licence:

- https://choosealicense.com/ - support from GitHub
- https://ufal.github.io/public-license-selector/ - includes support for projects where part of the code is based on existing software
- UCL support on Software Licensing
- You don't have to stick to licences listed by GitHub - you can copy-paste whatever licence you want.

Note: CC licences are NOT recommended for software and they list a number of other alternatives

2. Create a repository

All done! Now let's add a README file



3. README – describe your project & code

README.md

Festival of Code 2021

Why is it important?

README is the first thing people will see when they open your repository. Describe brieflt what your code does, why it is worth looking at, what your project is about.

What info to include?

- Project title (reflected by repository name), a short overview of project aims
- Repository description / overview of what code does / info about order files need to be used in?
- Information needed to re-use your code: info on data sources, software used, any special requirements (e.g. R packages)
- What have you used it for: add links to relevant outputs
- Who contributed?: authors, acknowledgments (with links to Github twitter)

Useful resources

Have a look at what other teams / people do for inspiration

- CHIG
- The Health Foundation
- Public HEalth England
- useful template from The Health Foundation

Useful resources:
README file template from the Health Foundation

3. README – describe your project & code

README.md

Festival of Code 2021

Why is it important?

README is the first thing people will see when they open yo
is worth looking at, what your project is about.

What info to include?

- Project title (reflected by repository name), a short over
- Repository description / overview of what code does / i
- Information needed to re-use your code: info on data s
  packages)
- What have you used it for: add links to relevant outputs
- Who contributed?: authors, acknowledgments (with link

Useful resources

Have a look at what other teams / people do for inspiration

- CHIG
- The Health Foundation
- Public HEalth England
- useful template from The Health Foundation

Useful resources:
Great markdown tutorial

Nice formatting – some people use it even for writing their PhD thesis!

Festival_of_Code / README.md in main

Edit file | Preview | Spa

```
1  # Festival of Code 2021
2  ## Why is it important?
3  README is the first thing people will see when they open your repository.
4  Describe brieflt what your code does, why it is worth looking at, what your project is about.
5
6  ## What info to include?
7  - Project title (reflected by repository name), a short overview of project aims
8  - Repository description / overview of what code does / info about order files need to be used in?
9  - Information needed to re-use your code: info on data sources, software used, any special requirements (e.g. R packages)
10 - What have you used it for: add links to relevant outputs
11 - Who contributed?: authors, acknowledgments (with links to [Github](https://github.com/) [twitter](https://twitter.com/) )
12
13 ## Useful resources
14 Have a look at what other teams / people do for inspiration
15 - [CHIG](https://github.com/UCL-CHIG)
16 - [The Health Foundation](https://github.com/HFAnalyticsLab)
17 - [Public HEalth England](https://github.com/PublicHealthEngland/)
18 - useful template from [The Health Foundation](https://user-images.githubusercontent.com/57709358/119957897-67aa1280-bf9a-11eb-9c7c-5c756b21e757.png)
19
20
```

4. Prepare your code

```
*************************************************************
* Title: XXXx
* Authors: XXXX
* Date created: XXX
* Date modified: XXX
*************************************************************

/*************************************************************
DO-FILE DESCRIPTION:
Briefly describe what the do-file does, where it fits with the other ones

For example:
"this do-file contains some preliminary cleaning rules for variables
recorded in HES. The code ensures e.g. consistent coding of missing values,
removes implausible values etc.
We used this do-file on an extract of HES admissions in children aged under 1 year old
prior to identifying births and linking episodes into admissions."
*************************************************************/


********************* housekeeping  ********************************

* 1. use global macro filepath to define where you save the data created in the process
* don't release absolute file paths if you work with sensitive data
global filepath "write filepath here X:\...."

* 2. load the data
use "XXXXX.dta", clear

* 3. Start a log
log using "XXXXXX.log", replace

***************************** clean variables ********************************
* since coding of some variables changed from numerical to characters over time
* we save these variables as string for consistency over years
tostring admimeth, replace
tostring nhsnoind, replace
tostring sushrg, replace
```

No one way of doing this - find template that works for you and use it from start

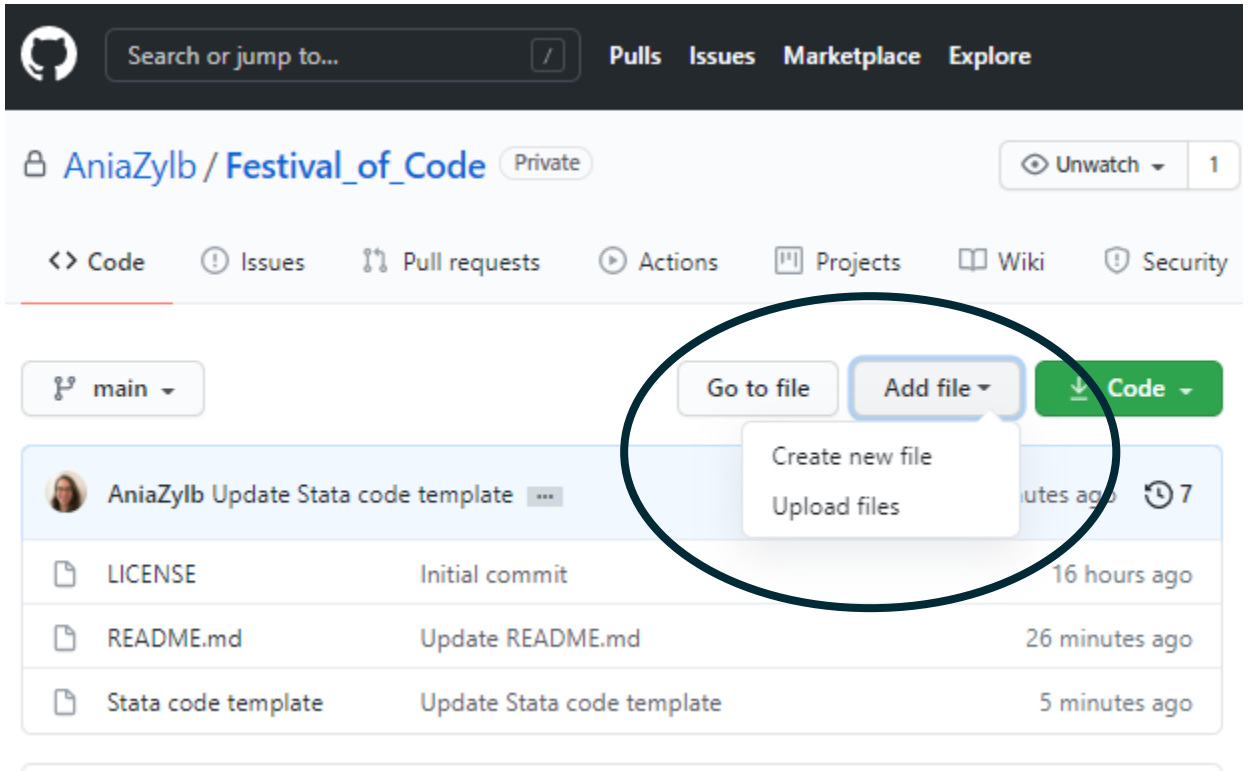
Examples of style guides for R and Stata and general coding style guide

e.g. Break your code into sections to make easier to navigate

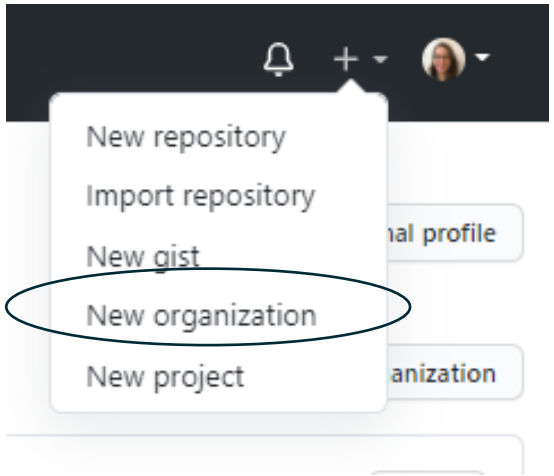
If you work with sensitive data, have a look at principles for sharing code safely from the Health Foundation:

1. Keep server architecture confidential & don't release absolute file paths (or any usernames, passwords etc)
2. Don't refer to raw data (e.g. IDs) or statistical results in the comments

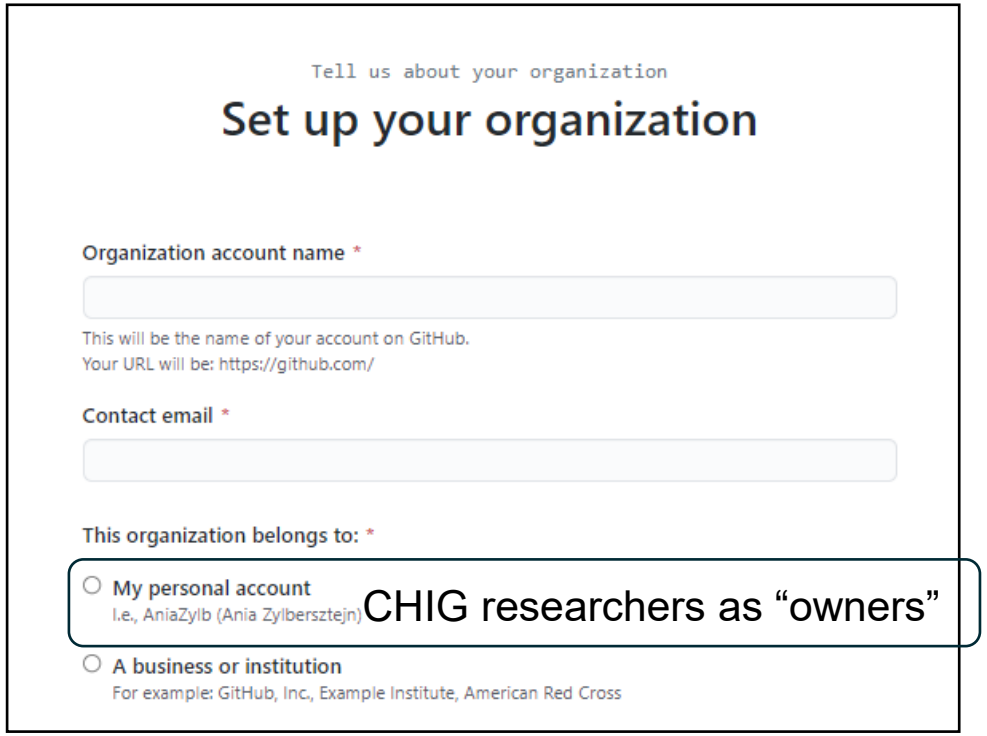
5. Upload the files



Setting up GitHub for a group / organisation



Pick a plan: free plan covers unlimited private and public repositories



CHIG researchers as “owners”

- "billing email“: any “.ac.uk” email address
- Include the user **rc-softdev-admin** as admin (for support from UCL)
- Add members to your organisation – make sure you ask them to make their membership public!
- Get in touch with GitHub if got problems

Useful resources:
Help on creating research organisation account from GitHub and UCL

Summary of useful resources for GitHub

- Support pages on GitHub from UCL
- GitHub tutorial from HDR UK
- Markdown tutorial (for writing your README files)
- Setting up GitHub for code sharing – blog on experiences from the Health Foundation
- Tutorial on using GitHub for version control in IDHS
- Use choosealicense.com or this licence selector to figure out which licence to use
- Organisational GitHub pages: CHIG, The Health Foundation Analytics Lab, NHS Digital, Public Health England
- UCL Coding Communities

Thank you!
Please get in touch if you have any questions.

Contact details:
- Dr Louise Mc Grath-Lone, Institute of Health Informatics l.mcgrath-lone@ucl.ac.uk
- Rachel Pearson, Institute of Child Health rachel.pearson@ucl.ac.uk
- Dr Ania Zylbersztejn, Institute of Child Health ania.zylbersztejn@ucl.ac.uk

UCL FESTIVAL OF CODE

Celebrating Research Software & Coding Communities