NHS England

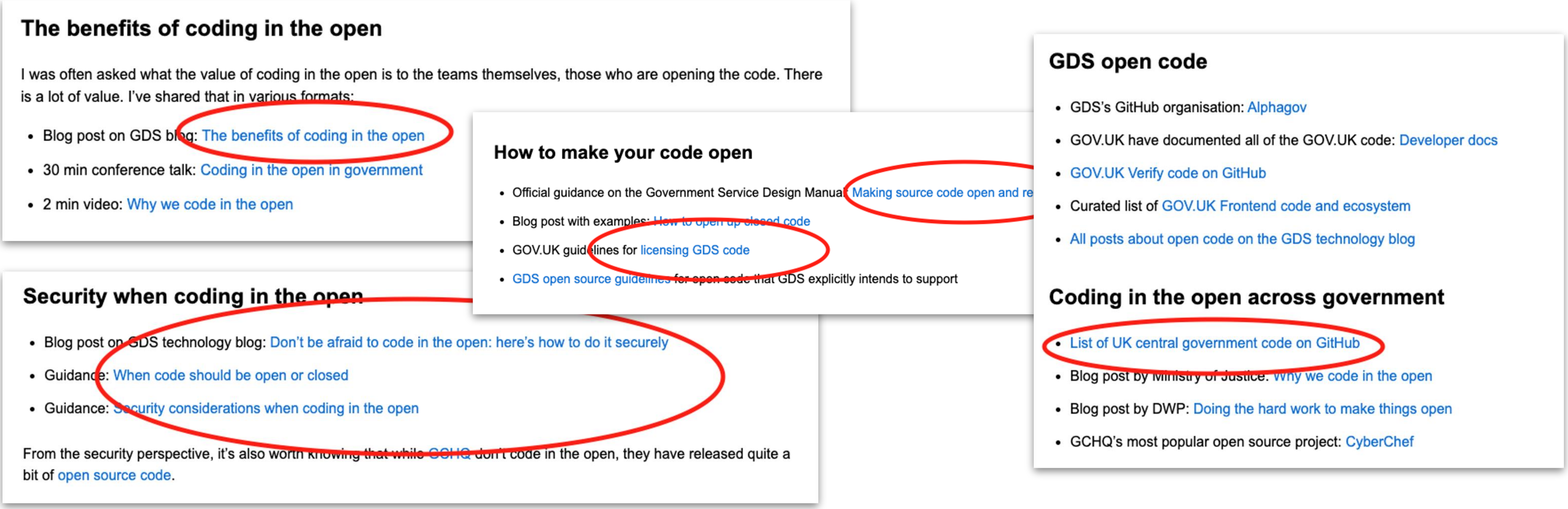
Sharing in the Open

NHSR conference 2022

Jonny Pearson,
Lead Data Scientist,
Digital Analytics and Research Team (DART)
NHS England
jonathanpearson@nhs.net

16th November 2022

Current Materials



The benefits of coding in the open

I was often asked what the value of coding in the open is to the teams themselves, those who are opening the code. There is a lot of value. I've shared that in various formats:

- Blog post on GDS blog: The benefits of coding in the open
- 30 min conference talk: Coding in the open in government
- 2 min video: Why we code in the open

How to make your code open

- Official guidance on the Government Service Design Manual: Making source code open and re
- Blog post with examples: How to open up closed code
- GOV.UK guidelines for licensing GDS code
- GDS open source guidelines for open code that GDS explicitly intends to support

Security when coding in the open

- Blog post on GDS technology blog: Don't be afraid to code in the open: here's how to do it securely
- Guidance: When code should be open or closed
- Guidance: Security considerations when coding in the open

From the security perspective, it's also worth knowing that while GCHQ don't code in the open, they have released quite a bit of open source code.

GDS open code

- GDS's GitHub organisation: Alphagov
- GOV.UK have documented all of the GOV.UK code: Developer docs
- GOV.UK Verify code on GitHub
- Curated list of GOV.UK Frontend code and ecosystem
- All posts about open code on the GDS technology blog

Coding in the open across government

- List of UK central government code on GitHub
- Blog post by Ministry of Justice: Why we code in the open
- Blog post by DWP: Doing the hard work to make things open
- GCHQ's most popular open source project: CyberChef

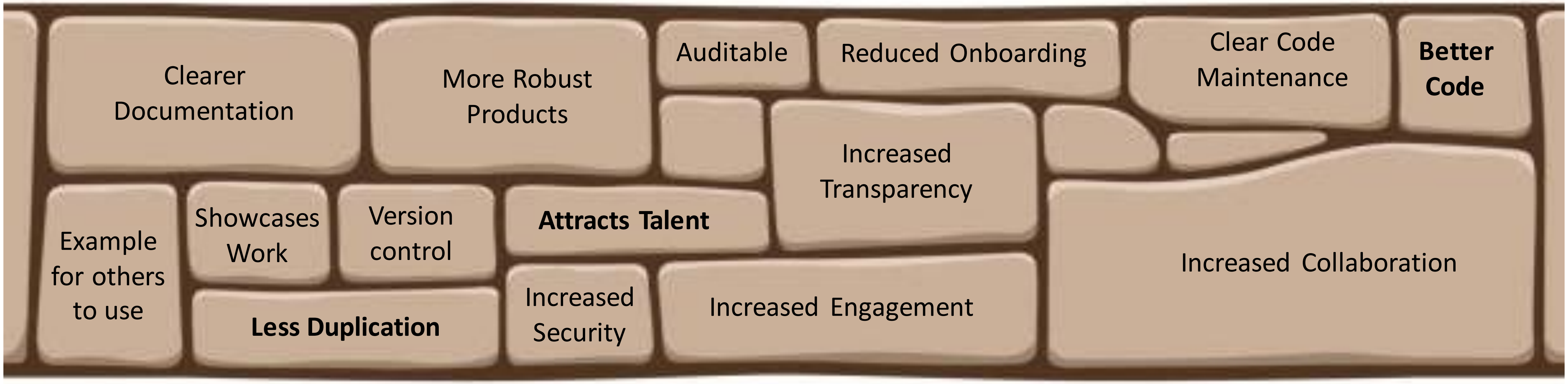
If you take one thing from this talk then take this

In 2018, Anna Shipman as Open Source Lead at GDS published these relevant materials:
https://www.annashipman.co.uk/jfdi/open-code-resources.html

Why Should I (Mandate & Benefits)?

For a formal mandate then see the **Goldacre Review** & **12. Make new source code open - Service Manual - GOV.UK** copied into **NHS service standard - 12. Make new source code open**.
Also see **Be open and use open source - GOV.UK** & **Open Data Charter - GOV.UK**

> *"Public services are built with public money. So unless there's a good reason not to, the code they're based should be made available for other people to reuse and build on.*
>
> *Open source code can be reused by developers working in government, avoiding duplication of work and reducing costs for government as a whole. And publishing source code under an open license means that you're less likely to get locked in to working with a single supplier."*



Clearer Documentation

More Robust Products

Auditable

Reduced Onboarding

Clear Code Maintenance

Better Code

Increased Transparency

Example for others to use

Showcases Work

Version control

Attracts Talent

Increased Collaboration

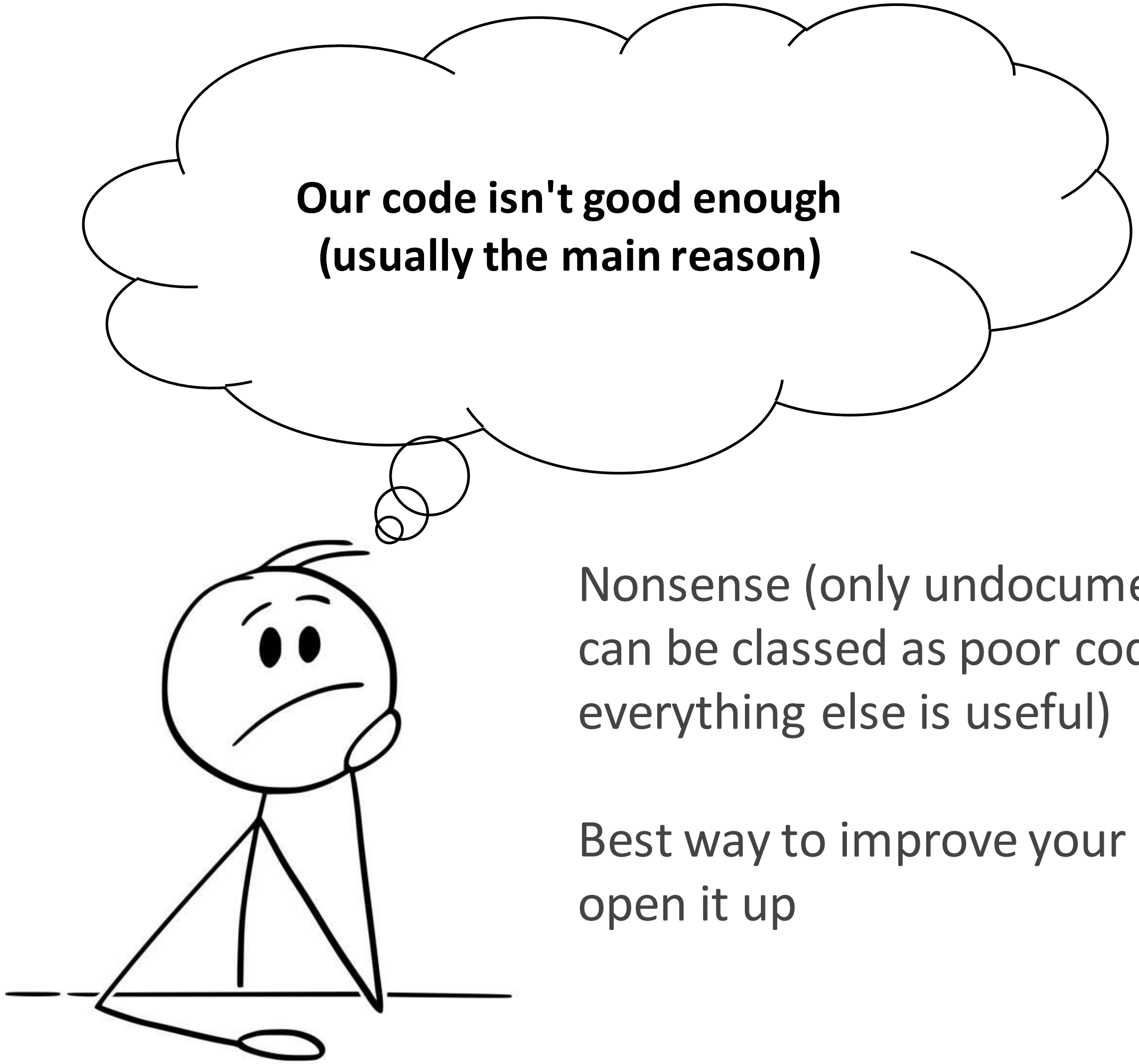
Less Duplication

Increased Security

Increased Engagement

What stops us?

Based off Terence Eden talk on overcoming barriers to open code



Our code isn't good enough (usually the main reason)

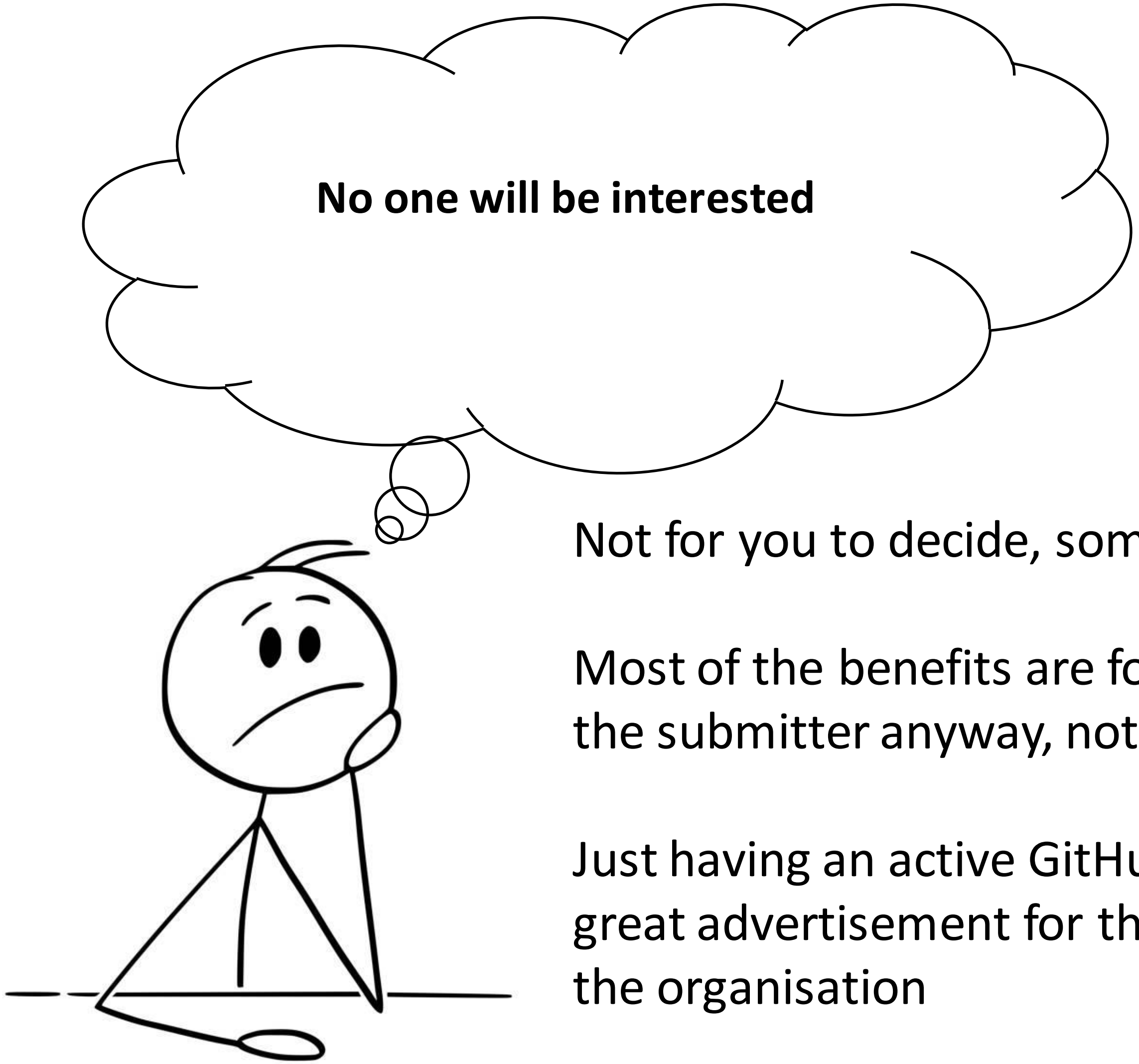
Nonsense (only undocumented chaos can be classed as poor code, everything else is useful)

Best way to improve your coding is to open it up

What stops us?

Based off Terence Eden talk on overcoming barriers to open code

- ~~Our code isn't good enough (usually the main reason)~~



Not for you to decide, someone might

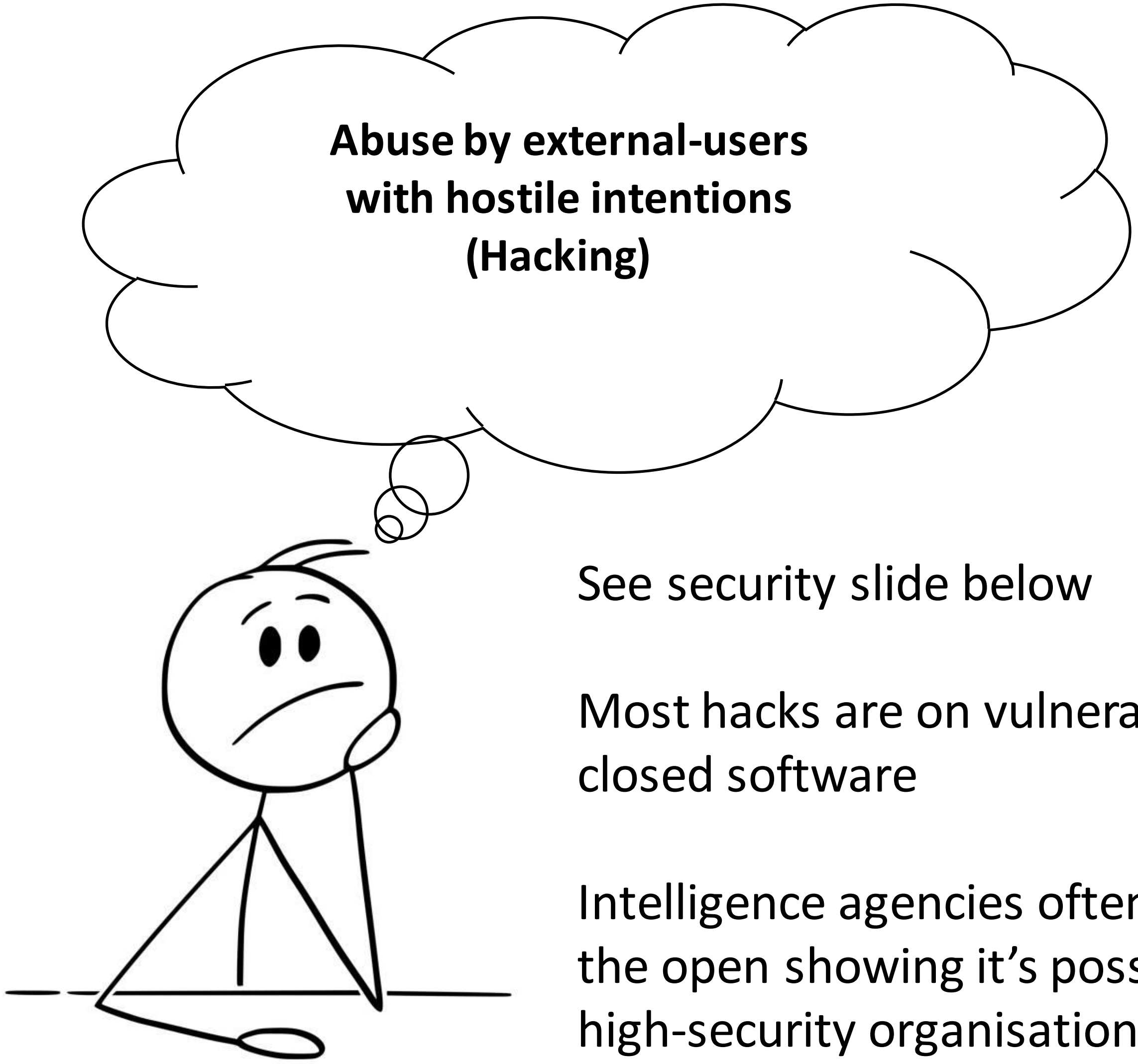
Most of the benefits are for the submitter anyway, not the reader

Just having an active GitHub repo is a great advertisement for the analysis in the organisation

What stops us?

Based off Terence Eden talk on overcoming barriers to open code

- ~~Our code isn't good enough (usually the main reason)~~
- ~~No one will be interested~~



Abuse by external-users with hostile intentions (Hacking)

See security slide below

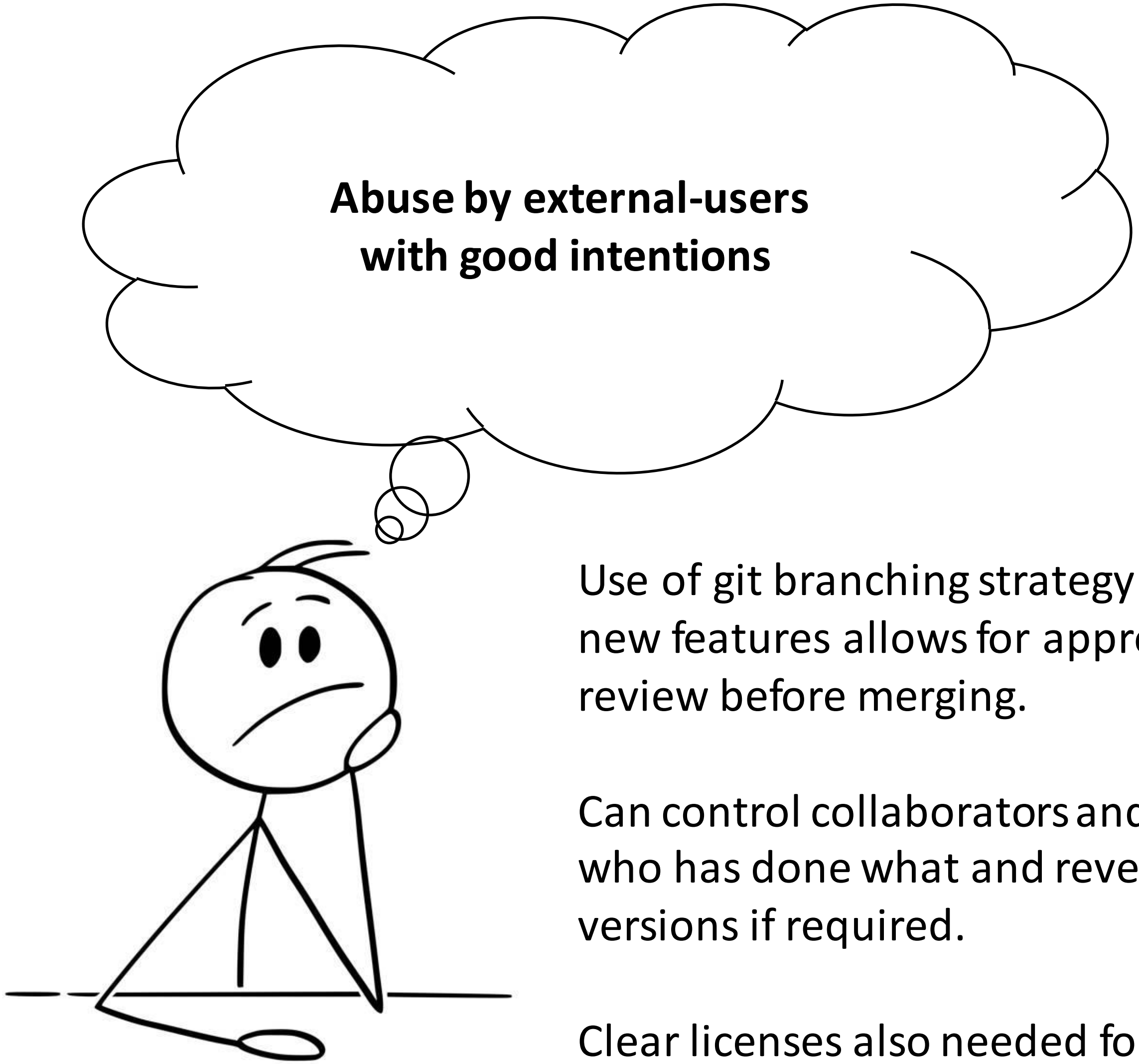
Most hacks are on vulnerabilities of closed software

Intelligence agencies often work in the open showing it's possible for high-security organisations

What stops us?

Based off Terence Eden talk on overcoming barriers to open code

- ~~Our code isn't good enough (usually the main reason)~~
- ~~No one will be interested~~
- ~~Abuse by external-users with hostile intentions (Hacking)~~



Abuse by external-users with good intentions

Use of git branching strategy to isolate new features allows for appropriate review before merging.

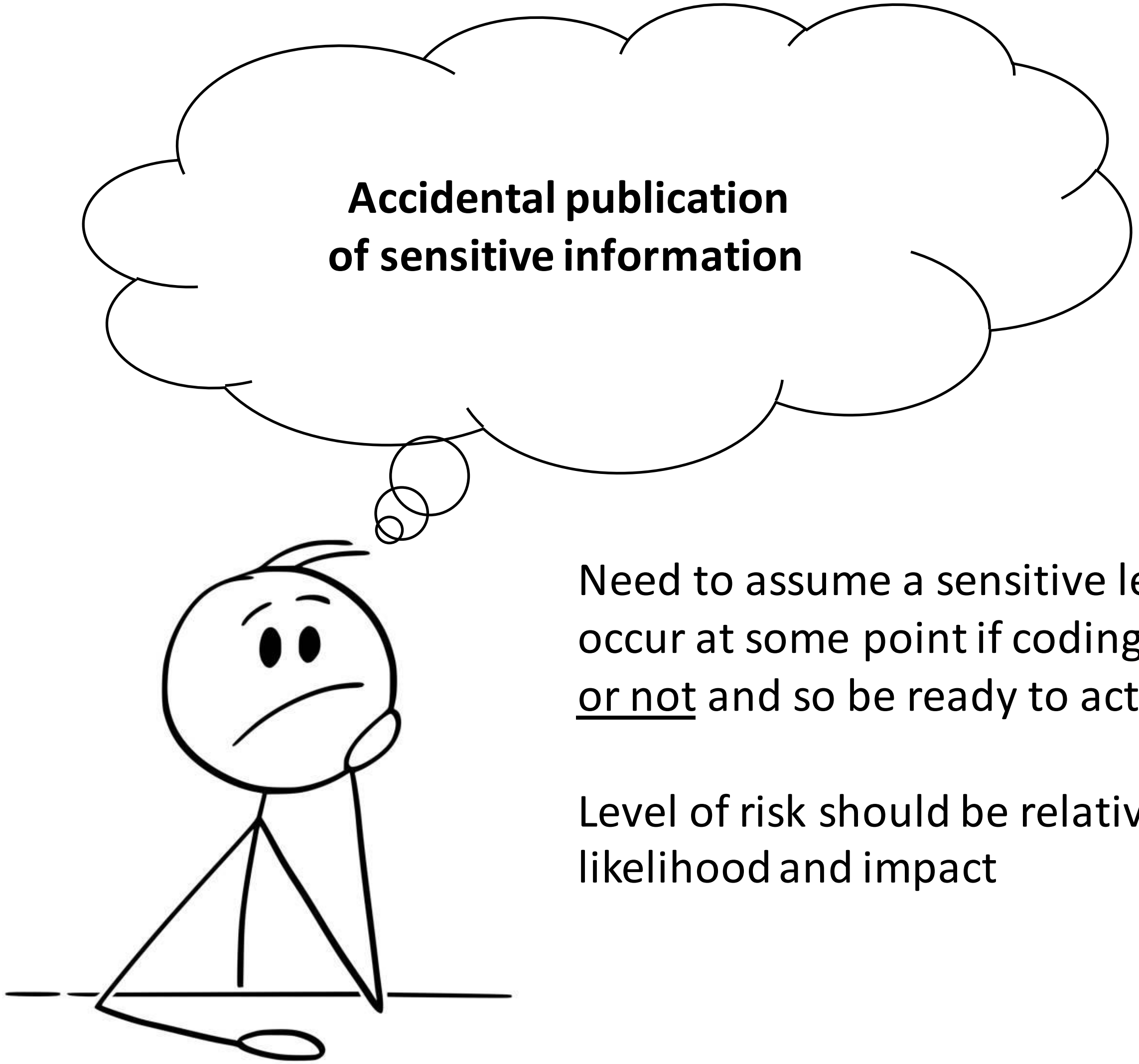
Can control collaborators and clearly see who has done what and revert to earlier versions if required.

Clear licenses also needed for clarity about use and responsibility

What stops us?

Based off Terence Eden talk on overcoming barriers to open code

- ~~Our code isn't good enough (usually the main reason)~~
- ~~No one will be interested~~
- ~~Abuse by external-users with hostile intentions (Hacking)~~
- ~~Abuse by external-users with good intentions~~



Accidental publication of sensitive information

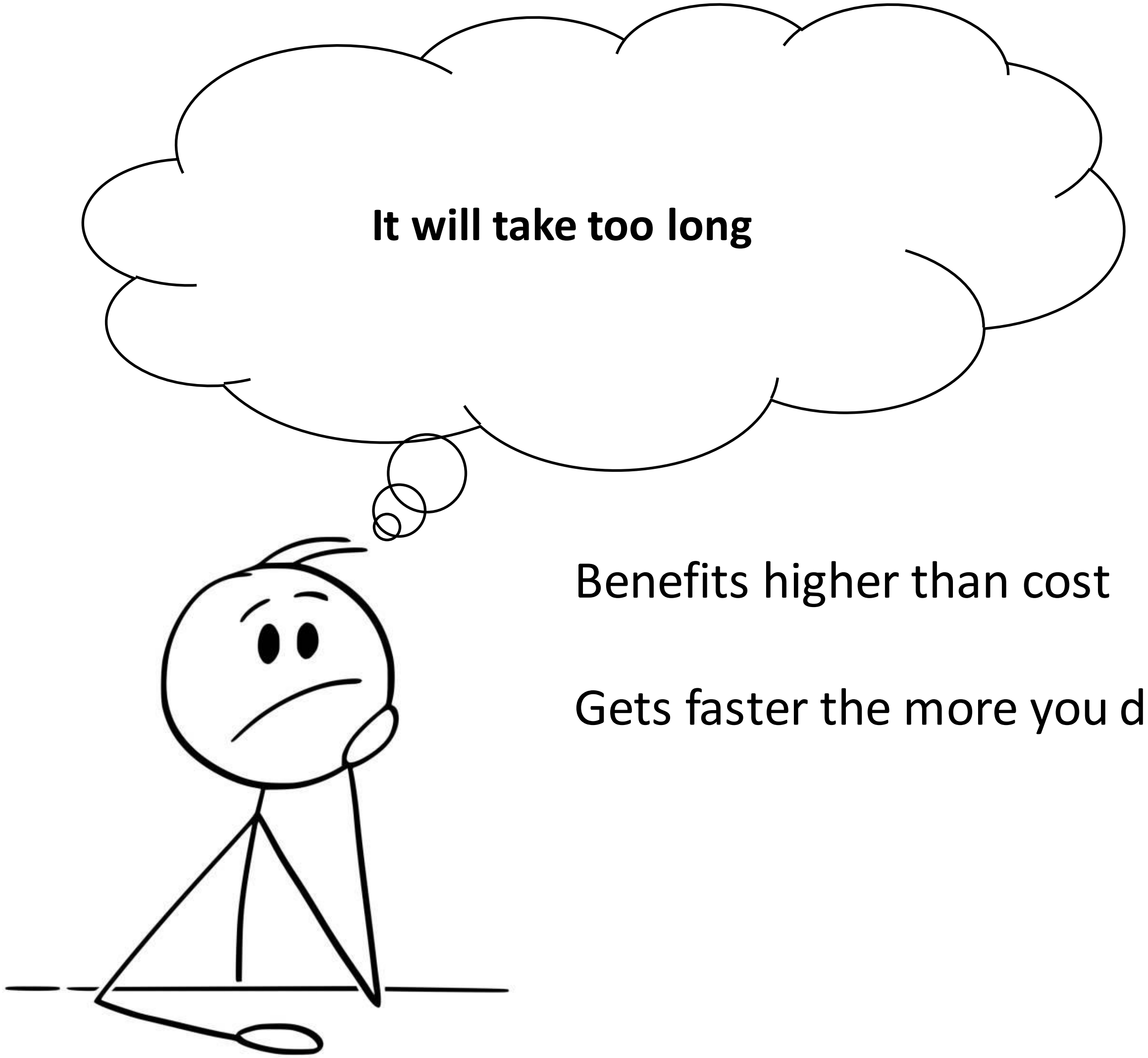
Need to assume a sensitive leak will occur at some point if coding in the open <u>or not</u> and so be ready to act!

Level of risk should be relative to likelihood and impact

What stops us?

Based off Terence Eden talk on overcoming barriers to open code

- ~~Our code isn't good enough (usually the main reason)~~
- ~~No one will be interested~~
- ~~Abuse by external-users with hostile intentions (Hacking)~~
- ~~Abuse by external-users with good intentions~~
- ~~Accidental publication of sensitive information~~



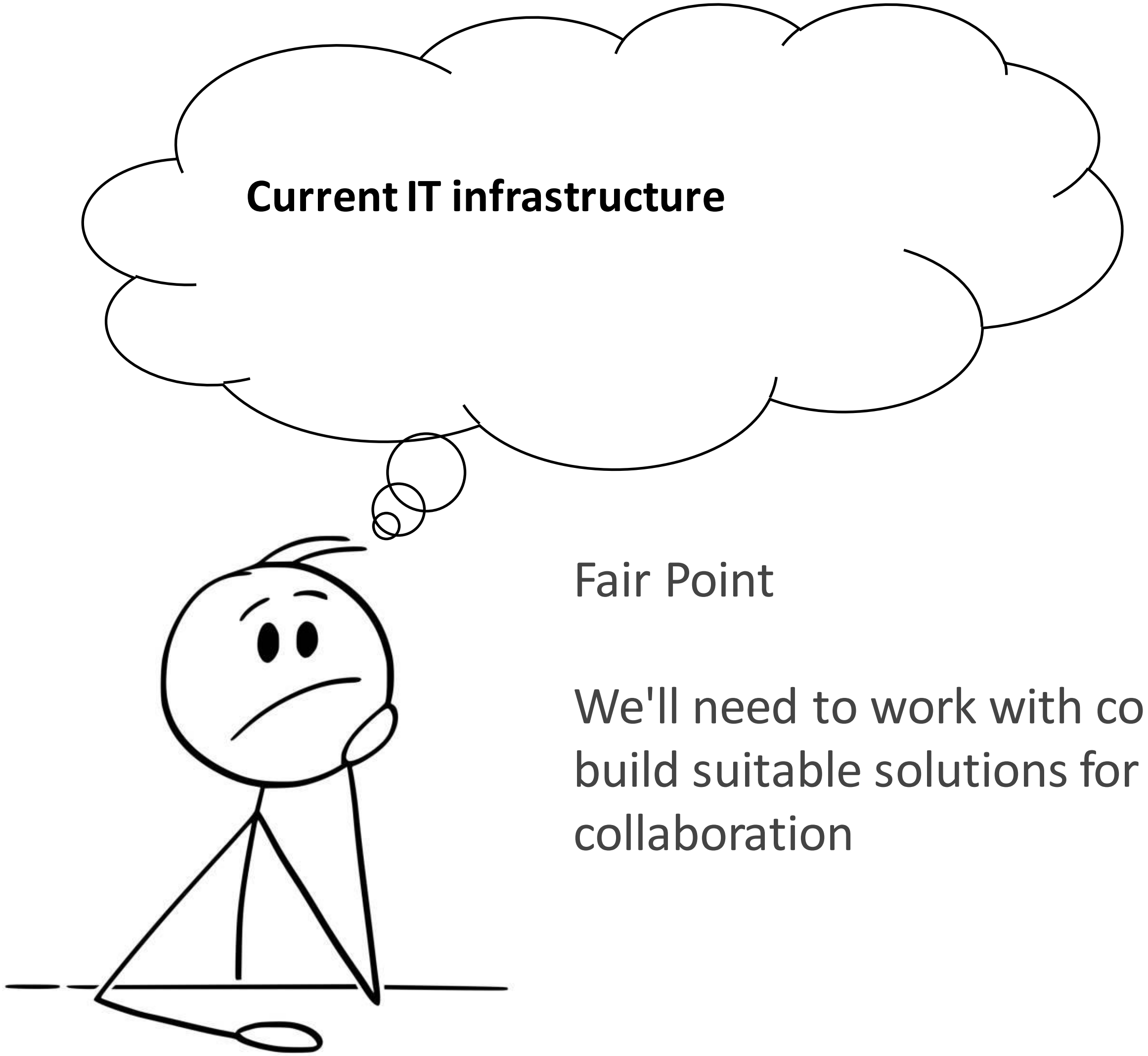
Benefits higher than cost

Gets faster the more you do it

What stops us?

Based off Terence Eden talk on overcoming barriers to open code

- ~~Our code isn’t good enough (usually the main reason)~~
- ~~No one will be interested~~
- ~~Abuse by external-users with hostile intentions (Hacking)~~
- ~~Abuse by external-users with good intentions~~
- ~~Accidental publication of sensitive information~~
- ~~It will take too long~~



Current IT infrastructure

Fair Point

We'll need to work with colleagues to build suitable solutions for open collaboration

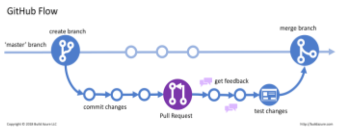
What do I need to know / have?

NHS England

Required

Understanding of Sensitive Data

Code with Version Control



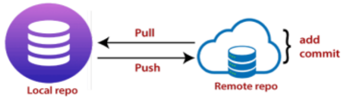
Peer Review

Understanding of Licenses

Understanding of Security Considerations

Knowledge of ig escalation routes

Open Source Software and push/pull permissions



Useful

Awareness of What Others are Doing

Non-sensitive useful starter project

Open data for testing and demo

Community of Support

What do I need to know / have?

Required

Understanding of Sensitive Data

More than just sensitive, secret, top secret information

Ideally no data stored alongside the code but if so, then written permission needs to be obtained by the data owner.

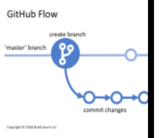
Consider checking the data content, code content, notebook outputs, commit messages and git history for:

- Credentials,
- Connection strings,
- SQL server addresses,
- Secret keys
- Unreleased Policy
- Business sensitive algorithms

Best to write config code separately.

If unsure about the git history then play it safe and create blank repo and copy across before making public

Code with Version Control



Peer Review

Understanding of Licenses

Understanding of Security Considerations

Useful

Awareness of What Others are Doing

Non-sens starte

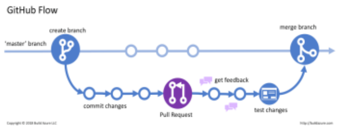
pport

What do I need to know / have?

Required

Understanding of Sensitive Data

Code with Version Control



Peer Review

Understanding of Licenses

Understanding of Se
Considerations

Even if you think you'll be the only person to ever use the code or that it will remain static, still use a versioning system

Establishing a branching strategy - recommend github flow

Consider templates (e.g. government cookie cutter)

Consider Semantic Versioning

Good commit notes - See Maintaining version control in coding - Service Manual

Useful

Awareness of What Others are Doing

ity of Support

What do I need to know / have?

Understanding of Sensitive Data

Code with Version Control

Peer Review

Understanding of Licenses

Require

Beyond quality assurance and accountability peer reviews increase knowledge sharing and produce better code.

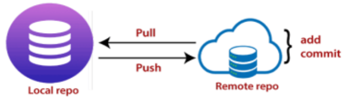
Colleagues need to be ready to do informal and formal reviews of the code and approach used.

Recommend using the quality assurance of code for analysis and research checklist

A minimum level of testing includes "can a colleague run the code?". If the data is sensitive then simple fake data can be used to allow a smooth initial run

edge of ig escalation routes

Open Source Software and push/pull permissions



Useful

Awareness of What Others are Doing

Non-sensitive useful starter project

Open data for testing and demo

Community of Support

What do I need to know / have?

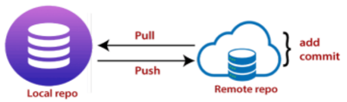
Required

Ur
S

Review

Understanding of Licenses

Open Source Software and push/pull permissions



[MIT Licence](#)

Allow unrestricted use but protects developer from liability and acknowledges contributions - Default licence for all new code

[APLv2](#)

If your code has regulatory requirements then an accompanying legal notice can be used here

[GPLv3](#)

If you want to prevent proprietary or closed re-use of code

[Open Government 3.0 Licence](#)

Recommended as default licence for all documentation

You should also include a copyright notice

Useful

Awaren

r testing and
mo

Community of Support

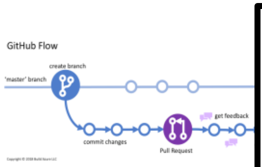
What do I need to know / have?

NHS England

Required

Understanding of Sensitive Data

Code with Version Control



Peer Review

Understanding of Licenses

Understanding of Security Considerations

Kr

Ensure good development practice:

- Open the code early
- a set way of managing changes - GDS pull request guidance

Ensure you have checked any third party tools used for data transfers:

- Assess against the NCSC's cloud security principles

Ensure the libraries used are reputable:

- Do the developers behind the library have a track record?
- Is the development team adequately supported or just an interested individual?
- Is the library kept up-to-date or static?

Useful

Awareness of What Others are Doing

Non-sensitive use starter project

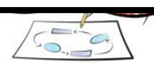
What do I need to know / have?

Assume that any accidental leakage of data or sensitive content is a breach.

Thus, a route for reporting and dealing with this possibility needs to be clear from the offset. This should already be in place for any organisation dealing with sensitive data.

Required

Understanding of Sensitive Data

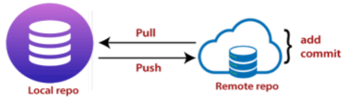


Understanding of Licenses

Understanding of Security Considerations

Knowledge of ig escalation routes

Open Source Software and push/pull permissions



Useful

Awareness of What Others are Doing

Non-sensitive useful starter project

Open data for testing and demo

Community of Support

What do I need to know / have?

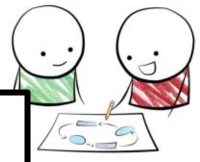
Required

Understanding of Sensitive Data

Code with Version Control

GitHub Flow
'master' branch
create branch
merge branch

Peer Review



Understanding of Licenses

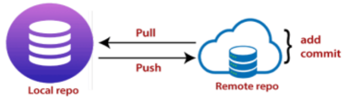
This gap in knowledge, combined with a culture of risk-aversion greatly hampers the process of adoption of new software in the NHS, leading many developers to look for workarounds.

Used a stepwise approach to engage properly.

Share both success and failures

n

Open Source Software and push/pull permissions



Useful

Awareness of What Others are Doing

Non-sensitive useful starter project

Open data for testing and demo

Community of Support

What do I need to know / have?

Required

Understanding of Sensitive Data

Code

Understanding of Secu Considerations

Balance between:

- Useful for the business
- Achievable
- Interesting

Suggestions:

1. Comparison of multiple standard models on open prescribing data
2. Novel visualization of GBD data
3. Geospatial mapping to highlight inequalities across multiple hierarchical boundaries
4. Sentiment analysis of survey data

Useful

Awareness of What Others are Doing

Non-sensitive useful starter project

Open data for testing and demo

Community of Support

What do I need to know / have?

NHS England

Structured Activity and HER records

- Opendata.nhsbsa
- PHE Fingertips
- NHS England Data Catalogue
- OpenPrescribing
- **Global Burden of Disease (GBD)**

Requir

Text

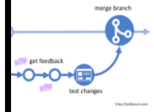
- **MIMIC III**
- **n2c2 PII tasks**
- **Diameter Health - GPT-2 generated notes**
- **Nottinghamshire Healthcare Foundation Trust (NHFT) - Friends & Family Test (FFT) Feedback Dataset**

Images

- **MIMIC CXR**
- **OASIS**
- **The Cancer Genome Atlas Lung Adenocarcinoma data**
- **NIHCC - DeepLesion**
- **CDAS**

Usefu

ion Control

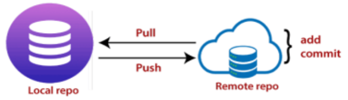


Peer Review

Understanding of Licenses

Knowledge of ig escalation routes

Open Source Software and push/pull permissions



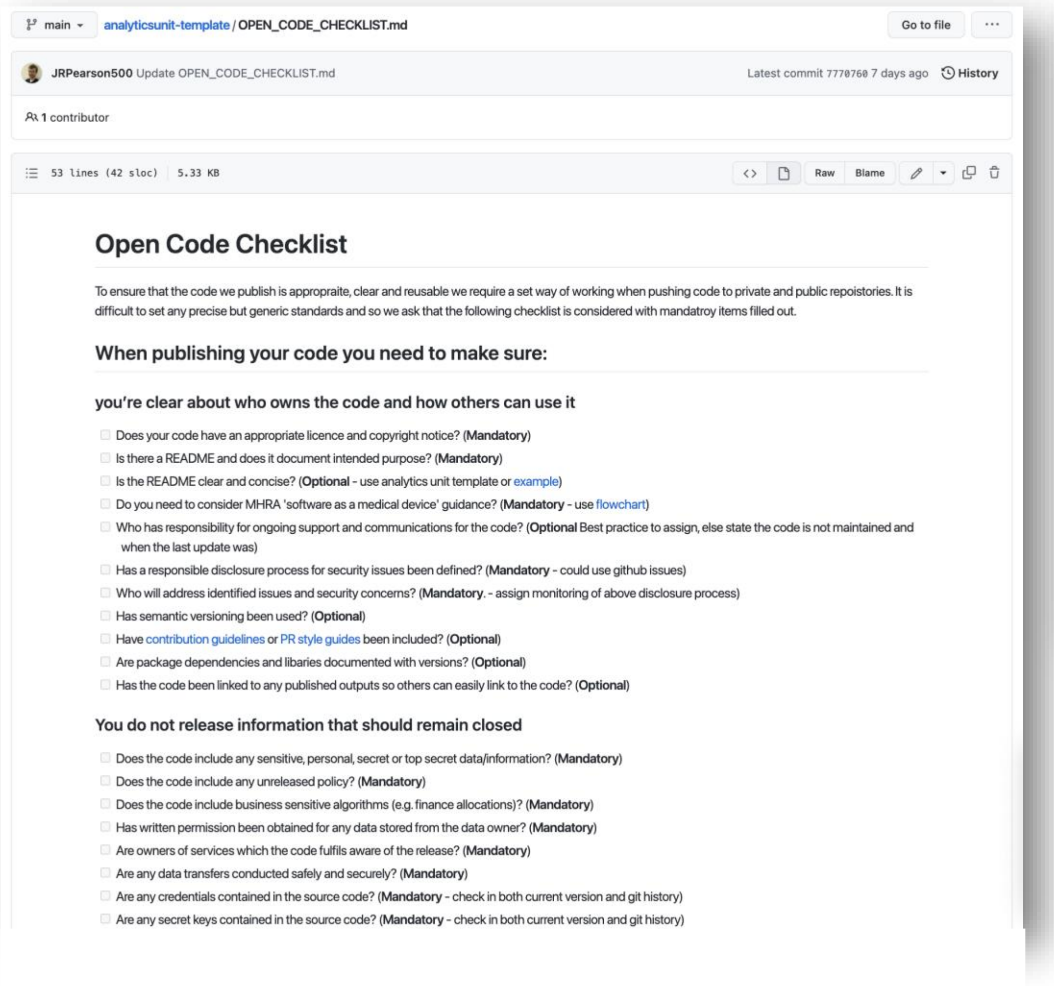
e useful
oject

Open data for testing and demo

Community of Support

Open Code Checklist

We use https://github.com/nhsengland/analyticsunit-template/blob/main/OPEN_CODE_CHECKLIST.md



analyticsunit-template / OPEN_CODE_CHECKLIST.md

JRPearson500 Update OPEN_CODE_CHECKLIST.md

Latest commit 7770760 7 days ago

1 contributor

53 lines (42 sloc) | 5.33 KB

Open Code Checklist

To ensure that the code we publish is appropraite, clear and reusable we require a set way of working when pushing code to private and public repoistories. It is difficult to set any precise but generic standards and so we ask that the following checklist is considered with mandatroy items filled out.

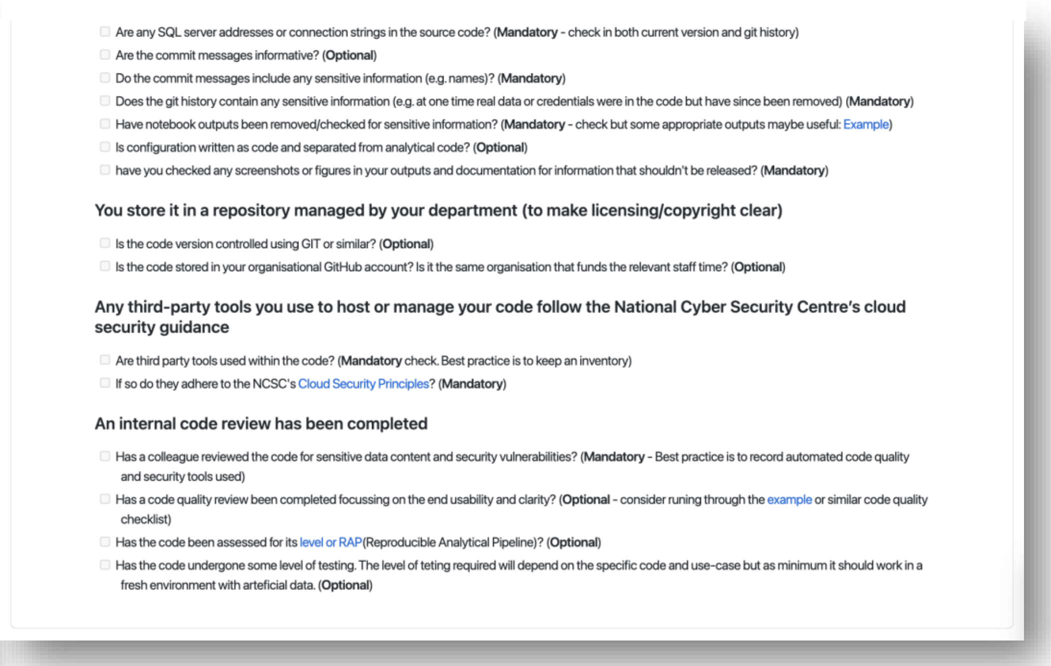
When publishing your code you need to make sure:

you're clear about who owns the code and how others can use it

- □ Does your code have an appropriate licence and copyright notice? (**Mandatory**)
- □ Is there a README and does it document intended purpose? (**Mandatory**)
- □ Is the README clear and concise? (**Optional** - use analytics unit template or example)
- □ Do you need to consider MHRA 'software as a medical device' guidance? (**Mandatory** - use flowchart)
- □ Who has responsibility for ongoing support and communications for the code? (**Optional** Best practice to assign, else state the code is not maintained and when the last update was)
- □ Has a responsible disclosure process for security issues been defined? (**Mandatory** - could use github issues)
- □ Who will address identified issues and security concerns? (**Mandatory**. - assign monitoring of above disclosure process)
- □ Has semantic versioning been used? (**Optional**)
- □ Have contribution guidelines or PR style guides been included? (**Optional**)
- □ Are package dependencies and libaries documented with versions? (**Optional**)
- □ Has the code been linked to any published outputs so others can easily link to the code? (**Optional**)

You do not release information that should remain closed

- □ Does the code include any sensitive, personal, secret or top secret data/information? (**Mandatory**)
- □ Does the code include any unreleased policy? (**Mandatory**)
- □ Does the code include business sensitive algorithms (e.g. finance allocations)? (**Mandatory**)
- □ Has written permission been obtained for any data stored from the data owner? (**Mandatory**)
- □ Are owners of services which the code fulfils aware of the release? (**Mandatory**)
- □ Are any data transfers conducted safely and securely? (**Mandatory**)
- □ Are any credentials contained in the source code? (**Mandatory** - check in both current version and git history)
- □ Are any secret keys contained in the source code? (**Mandatory** - check in both current version and git history)



- □ Are any SQL server addresses or connection strings in the source code? (**Mandatory** - check in both current version and git history)
- □ Are the commit messages informative? (**Optional**)
- □ Do the commit messages include any sensitive information (e.g. names)? (**Mandatory**)
- □ Does the git history contain any sensitive information (e.g. at one time real data or credentials were in the code but have since been removed) (**Mandatory**)
- □ Have notebook outputs been removed/checked for sensitive information? (**Mandatory** - check but some appropriate outputs maybe useful: Example)
- □ Is configuration written as code and separated from analytical code? (**Optional**)
- □ have you checked any screenshots or figures in your outputs and documentation for information that shouldn't be released? (**Mandatory**)

You store it in a repository managed by your department (to make licensing/copyright clear)

- □ Is the code version controlled using GIT or similar? (**Optional**)
- □ Is the code stored in your organisational GitHub account? Is it the same organisation that funds the relevant staff time? (**Optional**)

Any third-party tools you use to host or manage your code follow the National Cyber Security Centre's cloud security guidance

- □ Are third party tools used within the code? (**Mandatory** check. Best practice is to keep an inventory)
- □ If so do they adhere to the NCSC's Cloud Security Principles? (**Mandatory**)

An internal code review has been completed

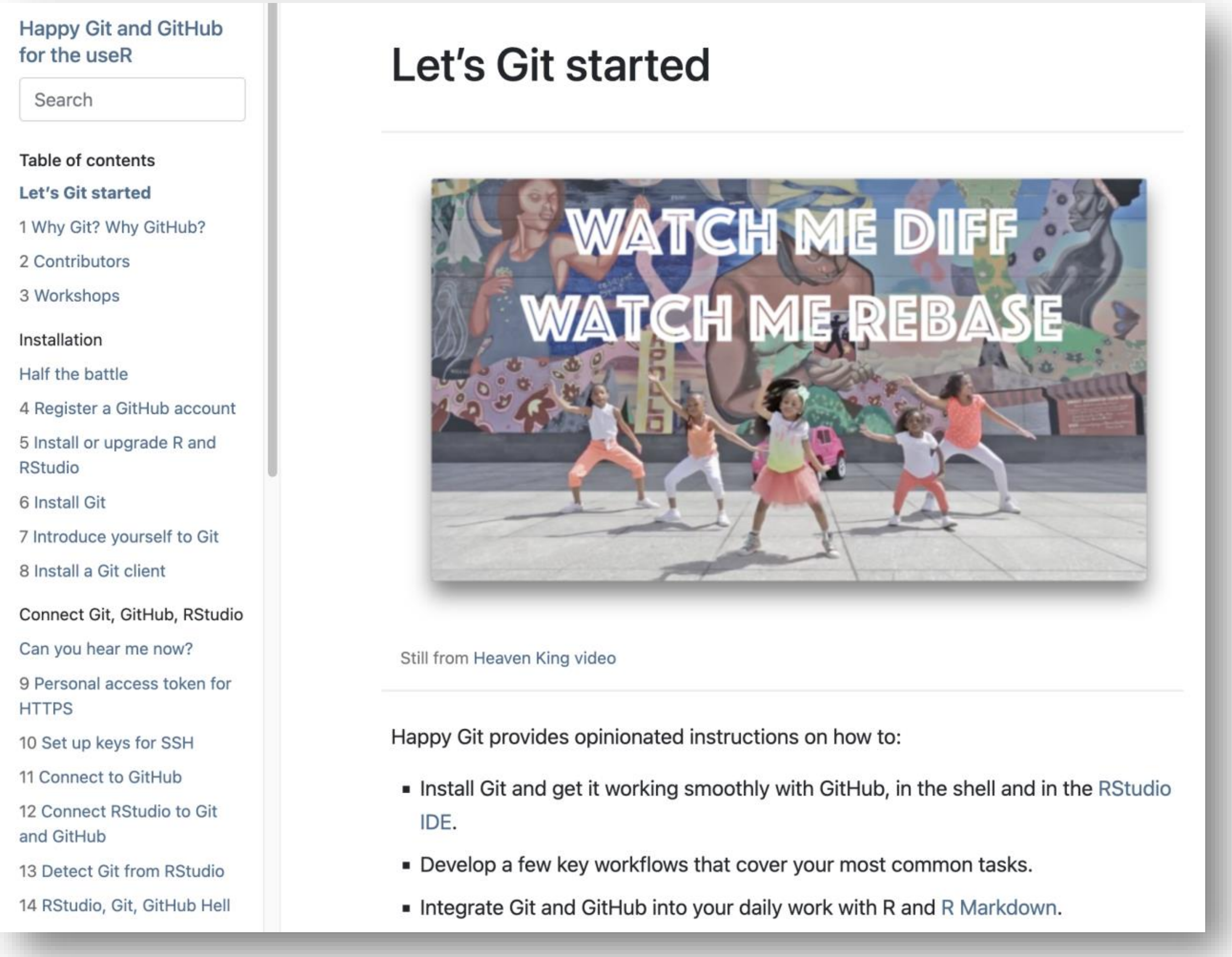
- □ Has a colleague reviewed the code for sensitive data content and security vulnerabilities? (**Mandatory** - Best practice is to record automated code quality and security tools used)
- □ Has a code quality review been completed focussing on the end usability and clarity? (**Optional** - consider runing through the example or similar code quality checklist)
- □ Has the code been assessed for its level or RAP(Reproducible Analytical Pipeline)? (**Optional**)
- □ Has the code undergone some level of testing. The level of teting required will depend on the specific code and use-case but as minimum it should work in a fresh environment with arteficial data. (**Optional**)

Using Git and Github with RStudio

Great materials on connecting git and github to RStudio can be found in happy git with r created by Jenny bryan (also see her other work on use this and tidyverse addins)

Helen Richardson (NHSD) is running a workshop on an Introduction to git and github on the 23rd November.

On NHS-R github there are git training materials also created by Helen.



Happy Git and GitHub for the useR

Search

Table of contents

Let's Git started

1 Why Git? Why GitHub?

2 Contributors

3 Workshops

Installation

Half the battle

4 Register a GitHub account

5 Install or upgrade R and RStudio

6 Install Git

7 Introduce yourself to Git

8 Install a Git client

Connect Git, GitHub, RStudio

Can you hear me now?

9 Personal access token for HTTPS

10 Set up keys for SSH

11 Connect to GitHub

12 Connect RStudio to Git and GitHub

13 Detect Git from RStudio

14 RStudio, Git, GitHub Hell

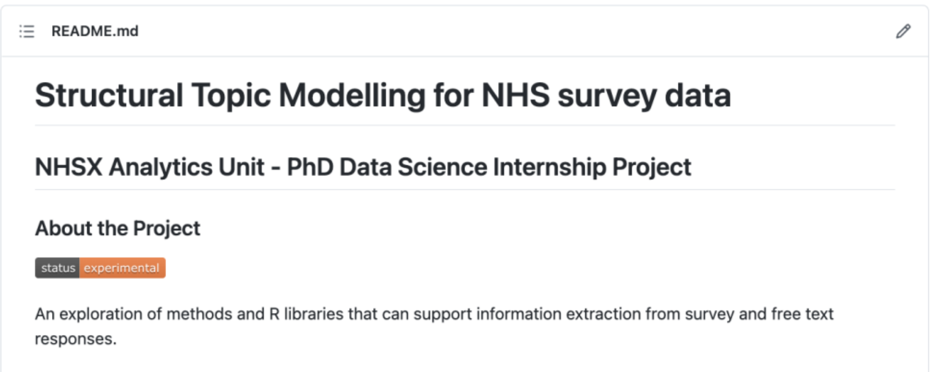
Let's Git started

Still from Heaven King video

Happy Git provides opinionated instructions on how to:

- Install Git and get it working smoothly with GitHub, in the shell and in the RStudio IDE.
- Develop a few key workflows that cover your most common tasks.
- Integrate Git and GitHub into your daily work with R and R Markdown.

Example - https://github.com/nhsx/stm-survey-text



README.md

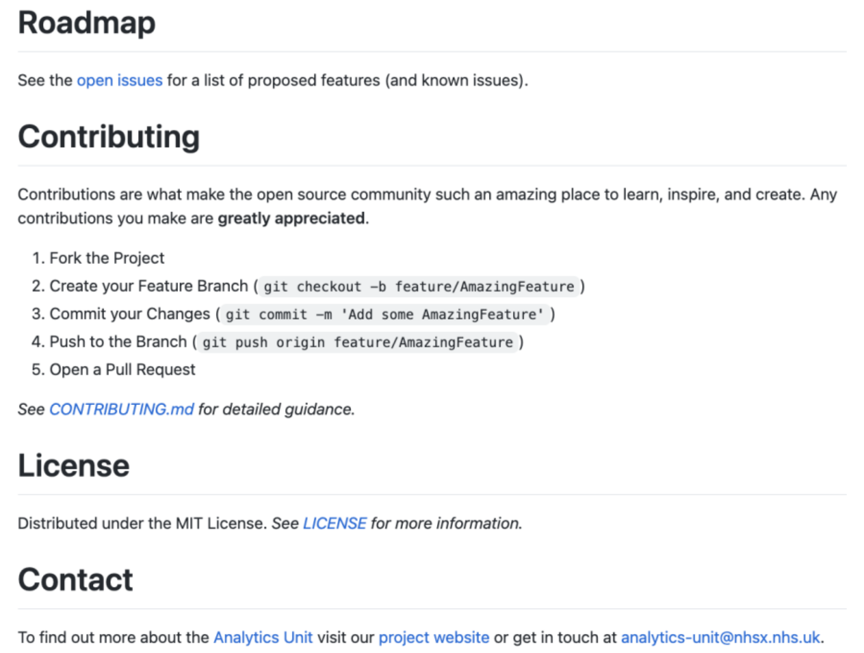
Structural Topic Modelling for NHS survey data

NHSX Analytics Unit - PhD Data Science Internship Project

About the Project

status experimental

An exploration of methods and R libraries that can support information extraction from survey and free text responses.



Roadmap

See the open issues for a list of proposed features (and known issues).

Contributing

Contributions are what make the open source community such an amazing place to learn, inspire, and create. Any contributions you make are **greatly appreciated**.

1. Fork the Project
2. Create your Feature Branch (`git checkout -b feature/AmazingFeature`)
3. Commit your Changes (`git commit -m 'Add some AmazingFeature'`)
4. Push to the Branch (`git push origin feature/AmazingFeature`)
5. Open a Pull Request

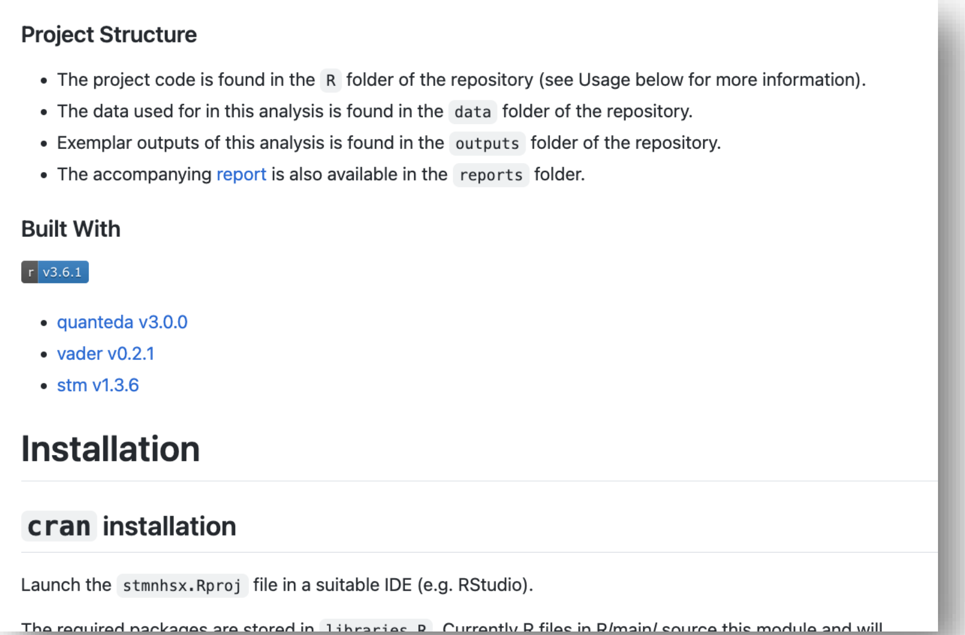
See CONTRIBUTING.md for detailed guidance.

License

Distributed under the MIT License. *See LICENSE for more information.*

Contact

To find out more about the Analytics Unit visit our project website or get in touch at analytics-unit@nhsx.nhs.uk.



Project Structure

- The project code is found in the `R` folder of the repository (see Usage below for more information).
- The data used for in this analysis is found in the `data` folder of the repository.
- Exemplar outputs of this analysis is found in the `outputs` folder of the repository.
- The accompanying report is also available in the `reports` folder.

Built With

r v3.6.1

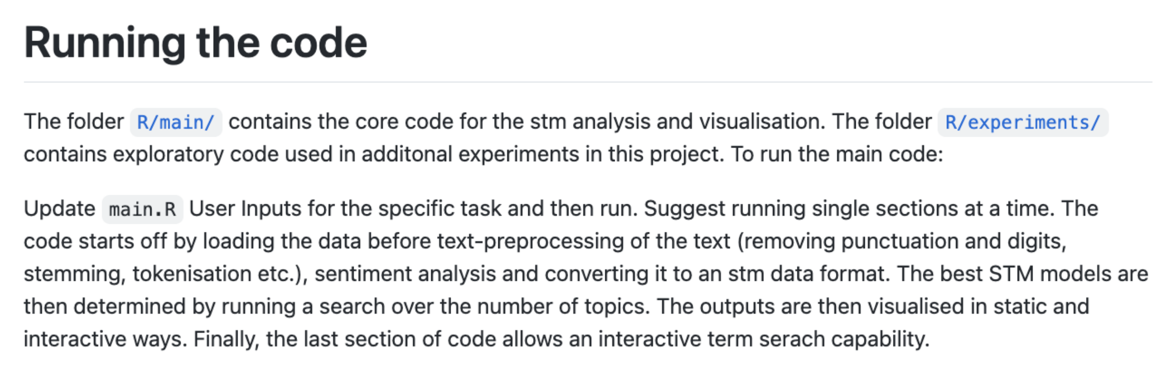
- quanteda v3.0.0
- vader v0.2.1
- stm v1.3.6

Installation

`cran` installation

Launch the `stmnhsx.Rproj` file in a suitable IDE (e.g. RStudio).

The required packages are stored in `libraries.R` . Currently R files in R/main/ source this module and will



Running the code

The folder `R/main/` contains the core code for the stm analysis and visualisation. The folder `R/experiments/` contains exploratory code used in additonal experiments in this project. To run the main code:

Update `main.R` User Inputs for the specific task and then run. Suggest running single sections at a time. The code starts off by loading the data before text-preprocessing of the text (removing punctuation and digits, stemming, tokenisation etc.), sentiment analysis and converting it to an stm data format. The best STM models are then determined by running a search over the number of topics. The outputs are then visualised in static and interactive ways. Finally, the last section of code allows an interactive term serach capability.

Questions?

Contact

See our open work here

Contact the team using analytics-unit@nhsx.nhs.uk