Introduce myself

- Field: Materials Science and Engineering
- Research area: corrosion of Ti alloys for implants
- No formal programming training

What this presentation is about

- My first attempt to make the analysis for my recent research project compliant with computational reproducibility standards
- Describe the research
- Show the paper and some intermediary analysis/plots
- Show how I did it
- Show how I would do it now

Practical stuff

- Let's see if my analysis can be reproduced on your machine: ..- Navigate to: https://github.com/craicrai/xrd_analysis_workflow ..- Fork ..- Open terminal ..- cd Desktop ..- git clone https://github.com/*your-user-name*/xrd_analysis_workflow ..- cd xrd_analysis_workflow ..- make all

DESCRIBE THE RESEARCH

Identify and characterise corrosion products with X-ray diffraction

- More than 3000 tonnes of titanium alloys implanted in people every year
- Titanium is very corrosion resistant, but not perfect
- It is important to know its corrosion products
- New alloy: Ti40Zr10Cu34Pd14Sn2

How to analyse the corrosion products

- Shine X-rays on corrosion products which diffract them
- Diffraction pattern is like . . .

Diffraction experiment

- (show cartoon from my presentations)

SHOW THE PAPER AND SOME INTERMEDIARY ANALYSIS

Selected raw 2D diffraction image

- (show a raw diffraction image from Supplement)
- show the actual pdf outside the presentation (copyright issue)

Stack of 1D diffraction patterns

- (show Figure 1 in paper)

Calculate average values

- (show the large table)

HOW I DID THE ANALYSIS ORIGINALLY

Documentation

- no repository, no appendix with details, just this:
- (show excerpt from Experimental section in paper about the analysis)

Project organization

- very poor organization
- (show tree of glassix and inbox)

DAWN Science for calibration and azimuthal integration

- (show the azimuthal integration pipeline in DAWN)

Frustruation build-up and the enlightenment moments

- drawing thousands of lines in ppts: there must be a better way! Started learning Python
- automation, efficiency improvement, tweaks
- the first THW seminar by Matt in March 2018

HOW I WOULD DO THE ANALYSIS NOW

Resources

- Previous THW presentations
- Millman et al. (2018). Teaching Computational Reproducibility for Neuroimaging. Front. Neurosci. 12:727. doi: 10.3389/fnins.2018.00727
- https://github.com/berkeley-stat159/project-alpha
- Wilson et al. (2017). Good enough practices in scientific computing. PLoS Comput Biol 13(6): e1005510. https://doi.org/10.1371/journal.pcbi.1005510
- Matthew Brett. (2017) Curious git (0.2). https://matthew-brett.github.io/curious-git/index.html
- The Internet

Tools

- Keep it simple!
- This presentation: done in Markdown, converted to pdf with Pandoc
- Version control: git. All git actions done in Bash, used GitHub only as remote repository
- Bash, Emacs, Python

Ensuring a reproducible environment

- virtualenv
- made directory venv/ in project root
- pip freeze > requirements.txt

Somebody has done something similar, of course

- fit2d - most known
- pyFAI - Python
- DAWN Science - Java?
- GSAS-II (Python!) - does everything!
  https://subversion.xray.aps.anl.gov/trac/pyGSAS

Data processing steps

- Calibration
- Azimuthal integration

Developing the workflow

- use pyFAI module
- import function does not work on my .hdf files! -> contact dev team, report bug, contribute?
- meanwhile, use h5py module and write my own import function
- realised it would be nice to also have a function to visualize the internal .hdf file tree structure

Last slide

- Data processing workflow is reproducible
- . . . but it does not necessarily imply it is correct
- . . . but at least interested people have the chance to check it

Final impression

- This is better than I imagined because
- I can go back to it anytime and see *exactly* how the analysis was done
- and I or someone else can re-use it for other projects