**Postdoc discussion: Some discussion on good data science development practices**

**Is there a standard of ML publishing in your particular field?**

* Applied math - no rigorous standard
* Psychology - no standards, very new field to ml
* Political science / econ - no rigorous standard -- rct / focus on causal inference
* Comp sci - paperswithcode

**How do you go about solving a problem with ML?**

* Hypothesis testing - try to use ml to test these hypotheses
* Trying to understand patterns of emotion - mri scans
* Affective computing - hci
* Use ml to identify patterns in dependent or independent variables

**How do you keep track of the code?**

* github - collaborative work
* Unit testing
* Cloud computing (AWS, google cloud)

**How do you keep track of the results? How do you keep track of data cleaning/processing?**

* Csv files :c (results)
* Keep the data cleaning/processing code (annotted) (e.g. r script)
* Record on bash script
* Google spreadsheet to keep track of changes of model
* Read me file to describe dataset/code (documentation)

**What happens to the project after it’s concluded?**

* Hard drive
* Lost github
* Box account
* Cluster
* Publicly available data
* Publish with the paper

**Useful resources:**

* <https://datadryad.org/stash> : Dryad is a nonprofit membership organization that is committed to making data available for research and educational reuse
* <https://figshare.com/> :
* <https://joss.theoj.org/> : an online journal? for publishing codes
* <https://academictorrents.com/> : torrents but for academia
* <https://dataverse.harvard.edu/> for large datasets
* <https://mlflow.org/>: machine learning lifecycle
* <https://dvc.org/> : data version control
* <https://www.cs.waikato.ac.nz/ml/weka/> Weka is tried and tested open source machine learning software that can be accessed through a graphical user interface, standard terminal applications, or a Java API **(really neat, I used this for an art-project, super easy to use)**
* <https://www.h2o.ai/> auto ML, used mostly in industry as far as I know
* <https://openframeworks.cc/> creative coding c++ library -- has nice out of the box computer vision algorithms
* https://arxiv.org/pdf/1911.05929.pdf (a paper on blinding cosmological data)
* https://arxiv.org/pdf/1912.08320.pdf
* https://figshare.com/articles/dataset/Detection\_of\_anti-correlation\_of\_hot\_and\_cold\_baryons\_in\_galaxy\_clusters/8001218
* https://www.pnas.org/content/115/11/2600
* Pls add more <3

**Lessons learned:**

* Defining the problem well
* Pre-analysis plan / pre-registration plan / blinding the data to avoid p-value hacking