Week 2 lab

COGS 108, 9:00-9:50AM (B01)

https://github.com/alexavndra/cogs108-b01

Introduction

Third year undergrad Machine Learning major + Computer Science minor

Data Management Assistant at Scripps Institute of Oceanography

Avid vintage shopper, especially vintage designer and Levi's; also I have a cat!

Email: arh003@ucsd.edu

Office hours: https://calendly.com/alexandrarh/office-hours

Logistics, tips, FAQ??

1) What is section for?

Section is NOT required, but we recommend showing up BECAUSE...

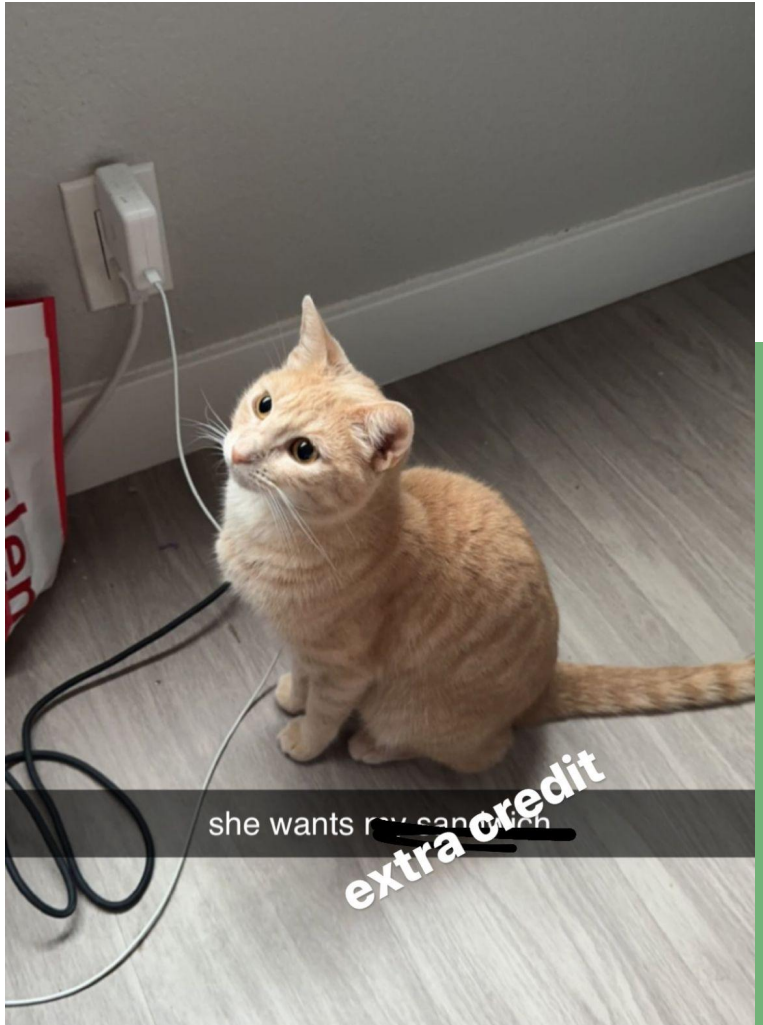
- Demos!
- IRL help, experience, and practice
- Information is on hand (Stack Overflow? Don’t know her)

2) Extra credit□□□???

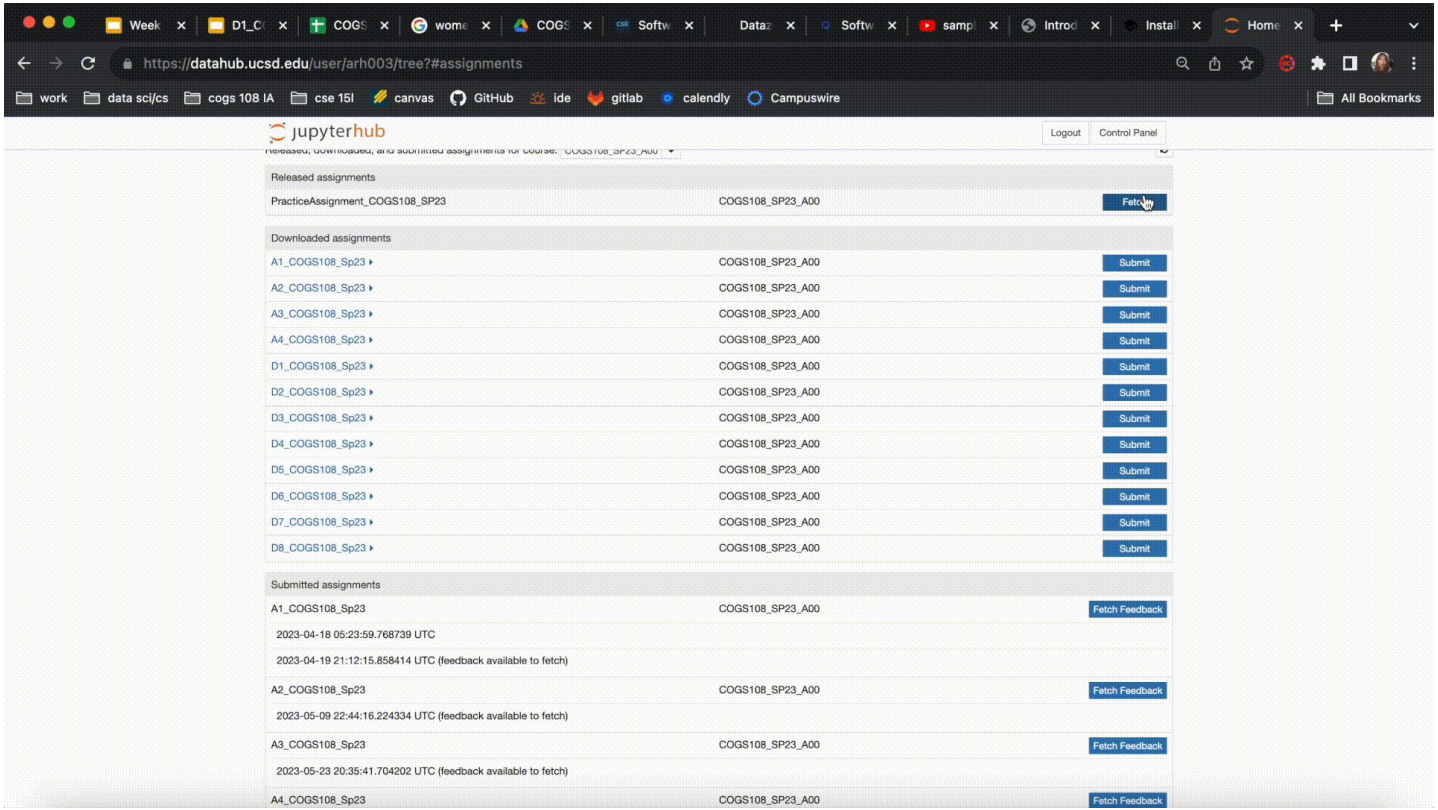
...yes

- Be a top participant on Campuswire (+0.5%)
- Do pre *and* post-course surveys (+0.25% each)
- Attend guest lectures (+0.25%)
- Submit guest lecture questions (+0.25%)
- Fill out all 7 weekly project surveys (+0.25%)

> 75% of all COGS 108 fill out SETs (+0.5%)



3) How to do/turn in labs, assignments…



4) The randomized groups are scaring me!

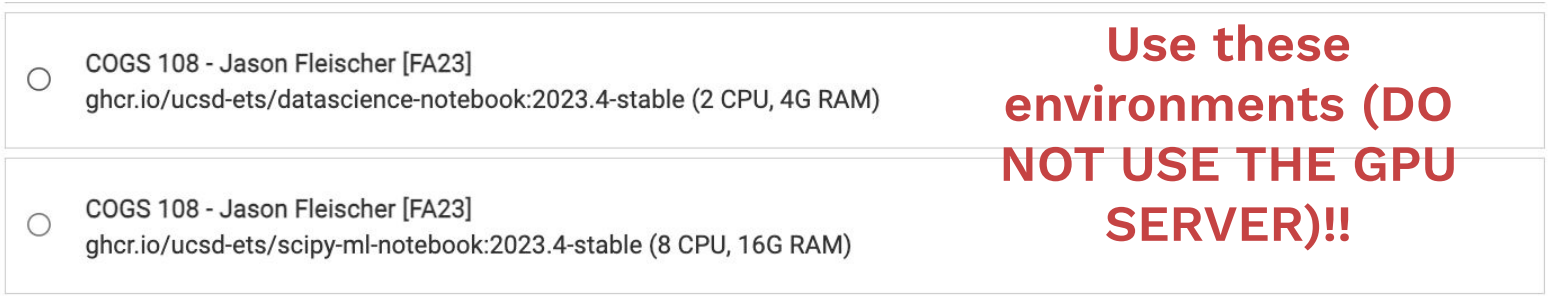
We’re running a case study, and groups will be randomized regardless of your preference (opt-in or opt-out)

Someone not participating? **Weekly surveys will help.**

5) Local machine vs. Datahub???

TL/DR: For projects, either one! Labs + assignments, **DATAHUB >>**

Why? Because that's where you fetch, validate, and submit the assignments from; plus your Datahubs are synced with the COGS108 GitHub.



6) Pieces of advice?

1) Do assignments early(!)

It's nice to get things out of the way!

2) Utilize office hours

We're here to help you succeed, promise.

3) Communicate

Please, *please*, **please** talk to your groupmates, the IAs/TAs, and even Professor Fleischer if you have questions/concerns/etc.

4) Try not to stress :)

Stressing just leads to a bad outcome (from personal experience); just breathe–YOU GOT THIS!!

Assignments and whatnot

Week 2 + 3:

- Practice assignment (Due today at 11:59pm)
- D1 (Due Friday, Oct 13 at 11:59PM)
- Quiz 2 (Due Monday, Oct 16 at 11:59PM)
- A1 (Due Wednesday, Oct 18 at 11:59PM)

Let’s get technical.

Programming, pt I

You need to have the basics...but not too much (if you don't)!

Resources + Cheat sheets:

- https://linktr.ee/cogs108resources: Pandas, Git, and Python guides/tutorials!
- https://github.com/COGS108/Resources: COGS108 Resources
- https://tinyletter.com/data-is-plural: Cool datasets
- Google "Python cheat sheet," "Pandas cheat sheet," "Git cheat sheet"

Programming, pt II

Anaconda

Data science toolkit w/ Python and libraries (also Jupyter notebooks)

How to get it:

- Download, install, and verify on anaconda.com
- Make sure it's added system path:
  - For mac: export PATH="/usr/local/anaconda3/bin:$PATH"

Jupyter Notebook

Program interface that you can type Python code (in the browser)

Datahub

Cloud server that runs Jupyter Notebooks on your computer

- Fetch and submit assignments
- Don't need to install Jupyter onto your local machine(s)

Work checks!

How to check (and submit) your work:

1. Write **assert** cells below your code for your testing
2. Click **Validate** before submitting your notebooks*
3. Click **Submit** when you're ready**

* **Validate** just replicates the autograder's mechanics, NOT hidden tests

** Don't click **Submit** AFTER the deadline (unless you're submitting late)

(see slide 6 for demo)

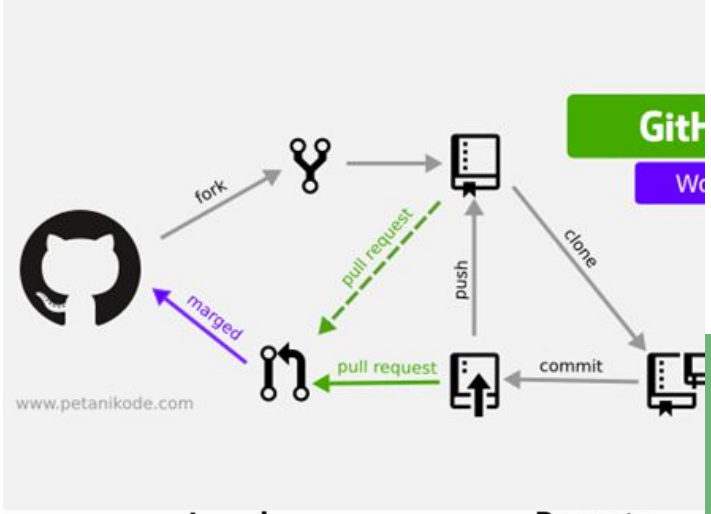
Git and GitHub

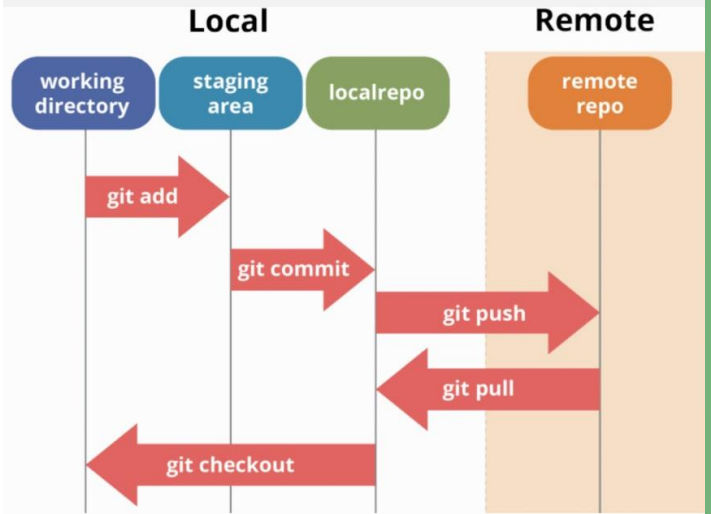
Git is a version control system; GitHub is where projects are stored–and shown–online

How to install Git:

- Go to https://git-scm.com/downloads
- Choose your OS (Windows/OS X/Linux)
- Follow the specific OS steps
- Verify installation
  - In terminal type “**git --version**”

Overall, Git is good because of version control, code reusage, backing up, and collaboration!





Demo 1: git stage, commit, and push

Git commands you'll use:

1. **git clone**: makes a clone/copy of repo at another directory
2. **git status**: displays state of working directory and staging
3. **git add**: adds changes to the working directory and staging area
4. **git commit**: commits changes made in the working directory
5. **git push**: uploads local repo content to the remote repo

You will also need to work with tokens (esp with Git and GitHub)!

Check out TA Scott Yang's guide here

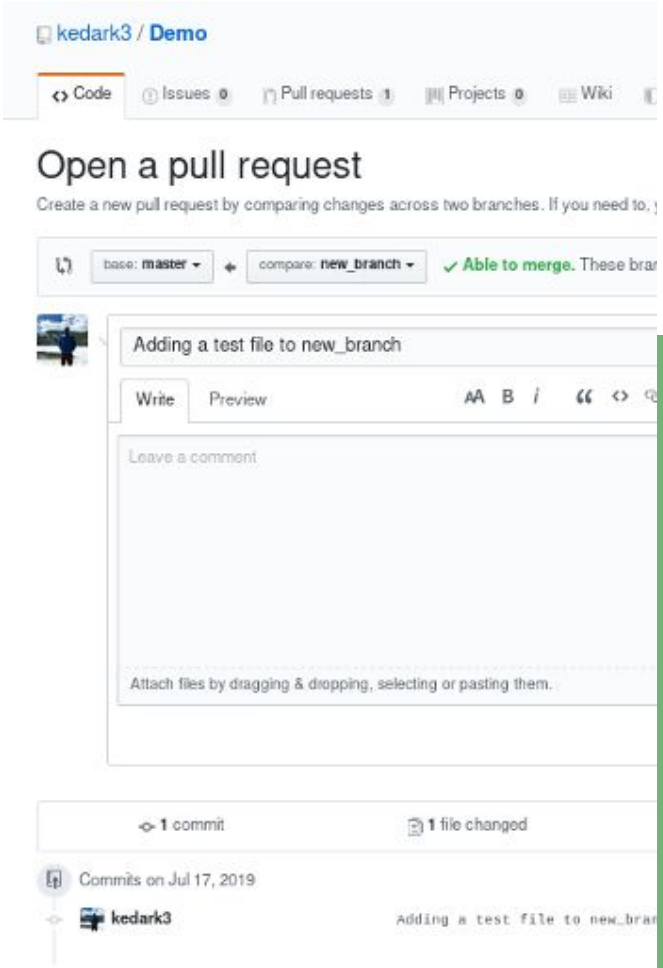
Work check, again.

Check for these things in your repo:

1. It's on your account (visible)
2. There's a README.md and .gitignore

Now...make a **pull request**:

1. Go to COGS108/MyFirstPullRequest
2. Make sure your PR shows up in that list.
3. Make sure your PR has the right title.
4. Make sure your PR has the right file (correct name).



Next week...

A1 discussion/help, more on git workflow

D1 Demo

https://datahub.ucsd.edu