Python Computing for Data Science

Compute & data space

`https://astro.datahub.berkeley.edu`

(log in with your CalNet ID)

Files for the Course:

`git clone https://github.com/profjsb/python-seminar.git`

(if you dont have *git,* please set it up later)

Signup (Discord):

Welcome to the
Python Computing for Data Science
Seminar

AY 250: Monday 4-7pm (Campbell Hall 131)

Instructor: Josh Bloom

GSI: Ellie Abrahams

Instructor+GSI email:
ucbpythonclass+seminar@gmail.com

Computational Science and Engineering

GORDON AND BETTY MOORE FOUNDATION

NSF

Award #0941742

Democratizing Trends in the Sciences

Data	Decreasing cost to obtain, move, store	→	open data, more freely shared
Compute	Decreasing cost, increasing specialty	→	more accessible
Technology/ Methodology	Algorithmic innovation, software tooling	→	open source

If anyone can get Data, Compute, Tech… who wins?

domain expertise → **They who asks the right questions**

\+

data science → **They who answers questions better & faster than others:**

- computational access
- methodological inference (e.g., machine learning)
- better story telling, dissemination of results
- reproducibility (acceptance)

Motivation for this Course:

short version

leverage the Python ecosystem to do cutting-edge research

long version

1) get you using Python to do cutting-edge research in the physical, biological and/or social sciences
2) help you realize that Python is a viable framework to do just about any 21st century problem well (and costs zero). “Super Glue”
3) introduce you into the Python community so you know how to navigate it yourself and so it potentially benefits from having you part of it

How we plan to do this:

- "formal" lectures on specialized topics each week (with some Guest appearances from the Python community!) (Monday)

- "breakout work sessions" interspersed within the lectures

- homework assignments based on week's lecture

- final project

Prerequisites:

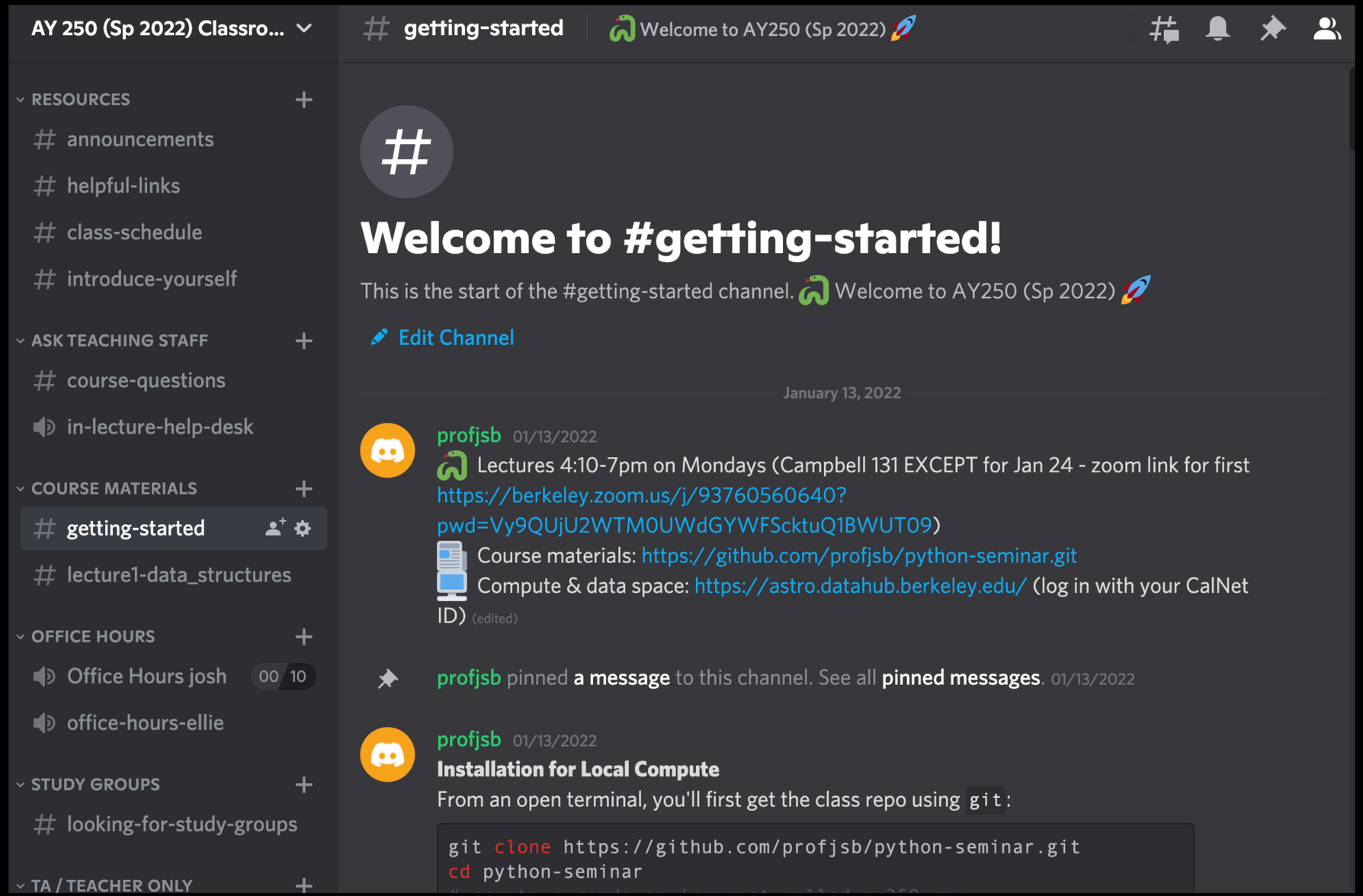
- working knowledge (or more) of the core Python language

- laptop for use in class and for homeworks

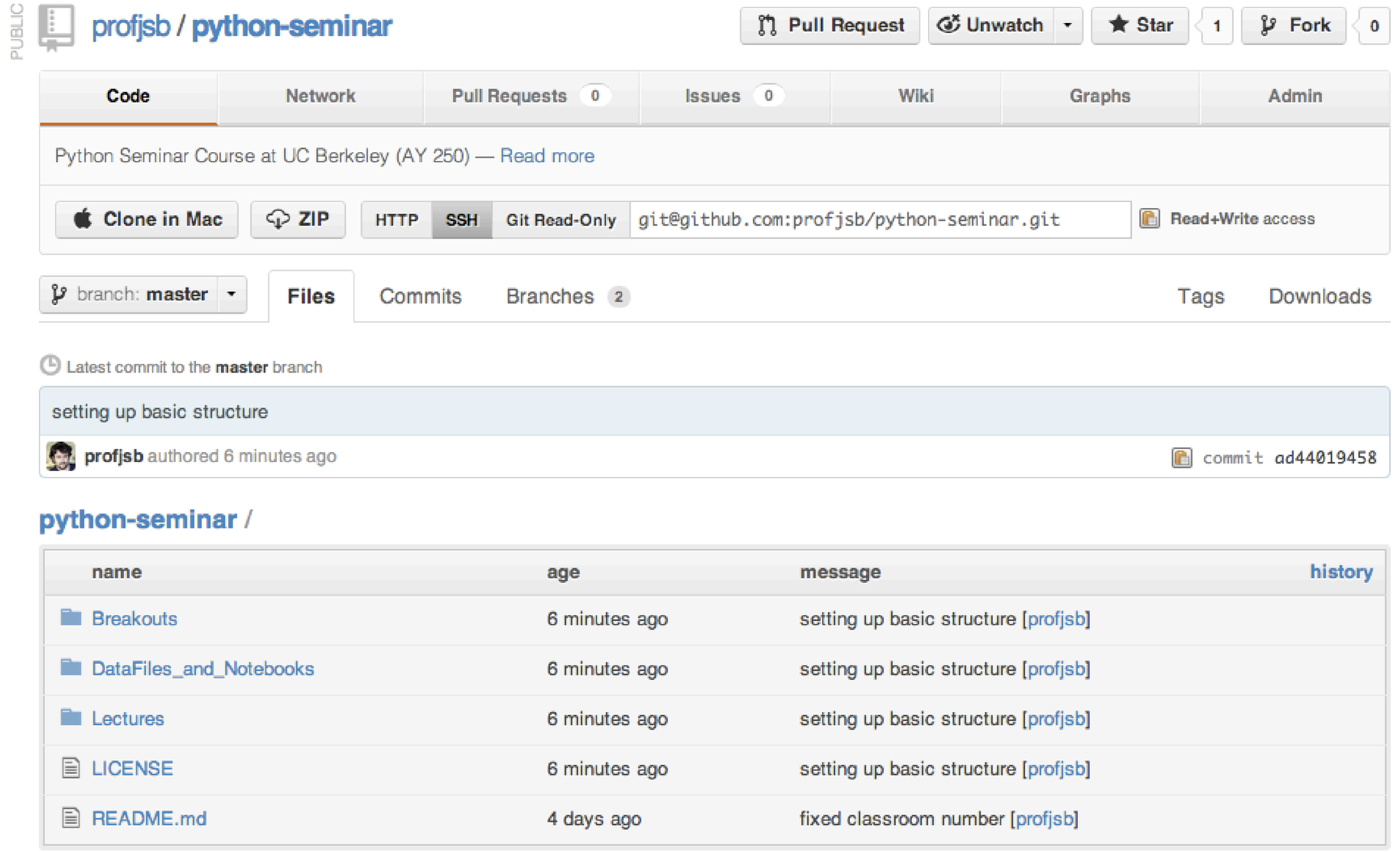
- (optional) local development: installation of Python (3.9.X), scientific 3rd party packages (Anaconda distro), & git

- tolerance for our terrible computer humor

Discord for real-time/off-line interaction homework updates, …



github is the main data portal for us...



Course Schedule

Date	Content	Reading	Leader
Jan 24 Online only	**Numpy, Scipy, & Pandas** launch binder	- scipy §§ 1.3, 1.5, 2.2 - numpy - skim chap 4/5 of McKinney	Josh
Jan 31	**Data visualization** (Matplotlib, Bokeh, Altair, Plotly)	- Skim Tufte's Visualization book - colormap talk (Scipy 2015)	Josh
Feb 7	**Application building and Testing**	None	Josh
Feb 14	**Parallelism** (asyncio, dask, ray, jax)	- ipyparallel docs	Josh
Feb 21	**Holiday** (no class)		
Feb 28	**Database interaction** (sqlite, postgres, SQLAlchemy, peewee), **Large datasets** (xarray, HDF5)	None	Josh
Mar 7	**Machine Learning I** (sklearn, NLP)	None	Josh
Mar 14	**Machine Learning II** (keras [tensorflow], pytorch)	None	Josh
Mar 21	**Spring Break**		
Mar 28	**Interacting with the world** (requests, email, IoT/pyserial)	None	Josh
Apr 1 Friday 1-3pm	**Web frameworks & RESTful APIs, Flask**	None	Josh
Apr 4	No lecture		
Apr 11	**Bayesian programming & Symbolic math**	Probabalistic Programming eBook install: pip install pymc3	Josh
Apr 18	**Image processing** (OpenCV, skimage)	None	Stefan van der Walt
Apr 25	**Speeding it up** (Numba, Cython, wrapping legacy code)	None	Josh
Onward	final project work		

Course Schedule

https://github.com/profjsb/python-seminar

Concepts/Practices in this Course

- Jupyter & JuypterLab
- using git & github
- Docker
- Data science workflows
- reproducible research
- application building
- debugging
- testing

"Data science is an interdisciplinary field about processes and systems to extract knowledge or insights from data in various forms, either structured or unstructured, which is a continuation of some of the data analysis fields such as statistics, data mining, and predictive analytics…"

-wikipedia

Workflow for a *typical* week

Friday:
email w/ special installation instructions, reading/tutorials

Monday:

4:00	cd python-seminar; git pull
4:10 - 5pm	Intro topics Lecture
5 - 5:30pm	Breakout coding
5:30-6:15pm	Detailed topics lecture (+stretch)
6:15-6:30pm	Breakout coding
6:30-7:00pm	Detailed topics lecture

Following Tuesday:
Homework project due

Course Grade

- 10% participation in lectures/breakouts

- 60% Problem Sets
  There will be 9 assignments. Do at least 5. We will keep your best 5.

- 30% Final Project, due May 10 (no final exam)

Final Project

a) Build a substantial framework for doing something in your own research, based on at least two topics from different weeks. Something you will use for a long time...

e.g., image analysis package, hardware control software, a webservice that does some crunching under the hood, provide a parallelization of some algorithm or code you use, etc.

- or -

b) Contribute code/functionality to an open-source Python project (Juypter, scipy, Cython, numpy, matplotlib, etc.)

“Parallel Image Reconstruction from Radio Interferometry Data”

“Realtime Prediction of Activity Behavior from Smartphone”

“Graph Theory Analysis of Growing Graphs”

http://mb3152.github.io/Graph-Growth/

“Bus Arrival Time Prediction in Spain”