Lecture1

January 7, 2020

1 Lecture 1: Welcome and Introduction

CBIO (CSCI) 4835/6835: Introduction to Computational Biology

1.1 Overview and Objectives

In this lecture, we'll define and discuss the field of "computational biology" and motivate programming in a wet lab setting. We'll also go over the logistics of CBIO 4835/6835 and how the course will proceed over the semester. By the end of this lecture, you should be able to

- Define "computational biology" and the importance of having coding skills
- Define the Python programming language
- Take the course pre-test (Google doc, linked at the end)
- Log into the course Slack channel and JupyterHub

1.2 What is "Computational Biology"?

Is it "biology, but with computers"?

What does that even mean, anyway? Is using Excel considered "computational"?

What differentiates biology from computational biology?

What about quantitative biology? Where does that fit in?

Is that a subset of computational biology, or is computational biology a subset of it? Or do they just share some overlap?

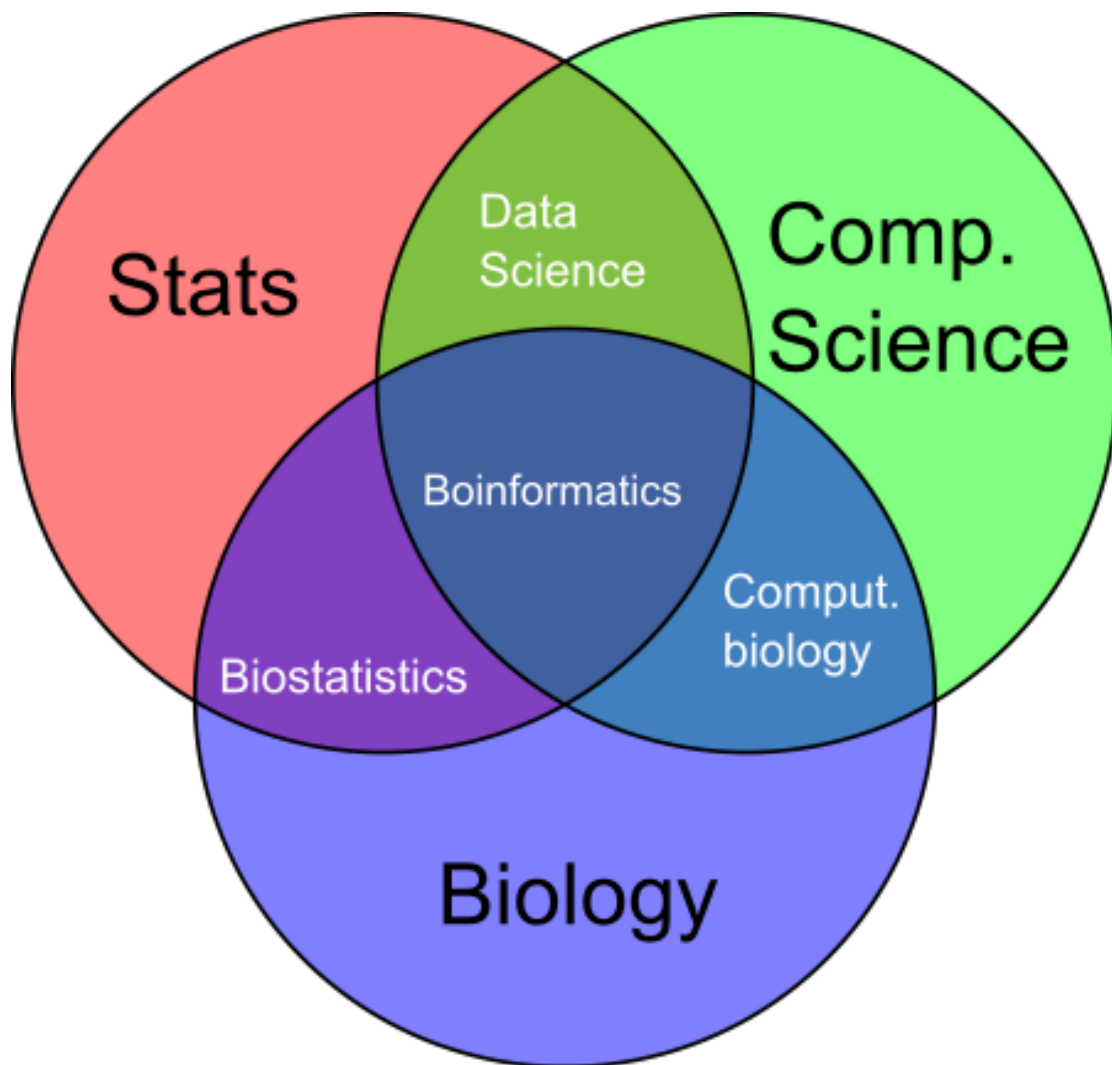And isn't all this just "bioinformatics"?
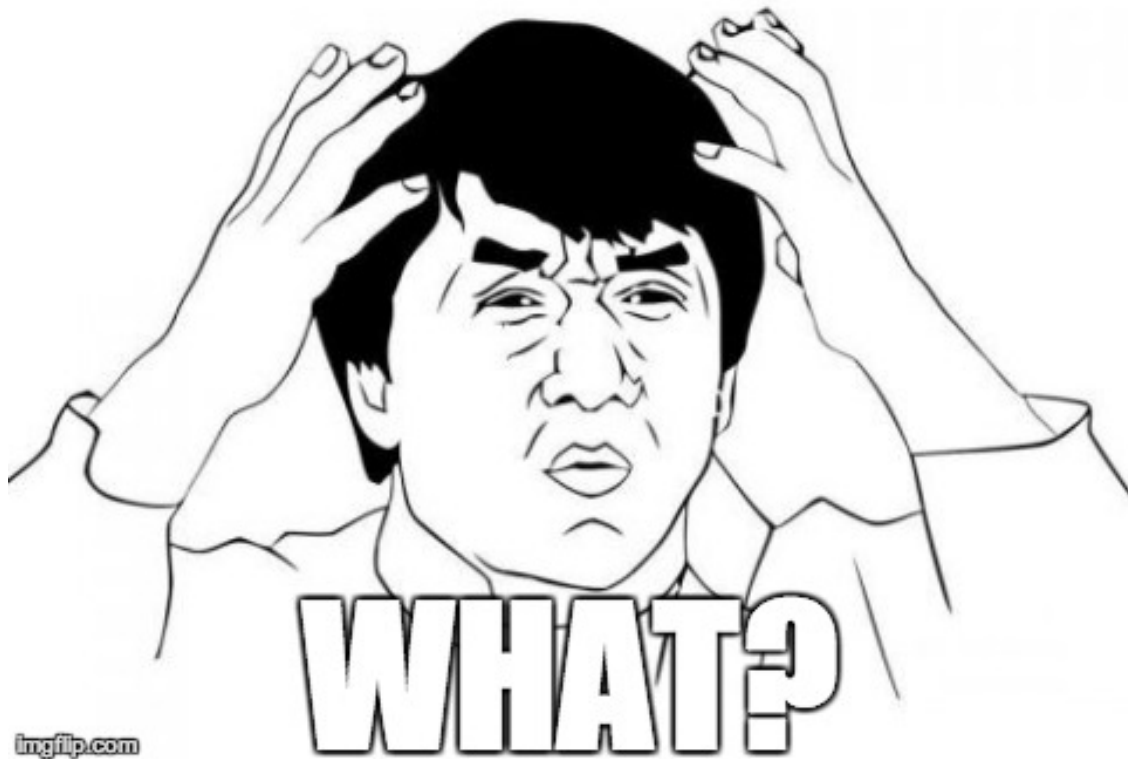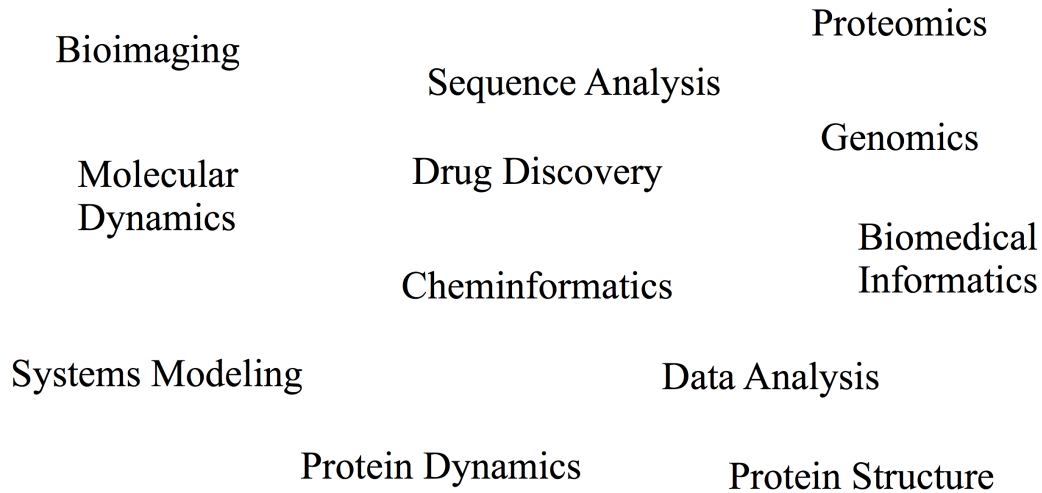
Computational biology

From Wikipedia, the free encyclopedia

Computational biology involves the development and application of data-analytical and theoretical methods, mathematical modeling and computational simulation techniques to the study of biological, behavioral, and social systems.[1] The field is broadly defined and includes foundations in computer science, applied mathematics, animation, statistics, biochemistry, chemistry, biophysics, molecular biology, genetics, genomics, ecology, evolution, anatomy, neuroscience, and visualization.[2]

Computational biology is different from biological computation, which is a subfield of computer science and computer engineering using bioengineering and biology to build computers, but is similar to bioinformatics, which is an interdisciplinary science using computers to store and process biological data.

Stats

Comp. Science

Data Science

Boinformatics

Biostatistics

Comput. biology

Biology

Bioinformatics, Computational, and Systems Biology

Proteomics

Bioimaging

Sequence Analysis

Genomics

Molecular
Dynamics

Drug Discovery

Biomedical
Informatics

Cheminformatics

Systems Modeling

Data Analysis

Protein Dynamics

Protein Structure

For the purposes of this course, I'm less concerned with properly categorizing computational biology vs bioinformatics, and more interested in

- Giving everyone the opportunity to gain experience in programming
- Teaching Python
- Surveying computational methods in a biological context
- **Improving everyone's skills to be more productive and successful researchers**

1.2.1   What this course *is*

Officially: Introduction to Computational Biology.

Officially, but more colloquially: *programming for biologists*.

1.2.2   What this course is *not*

Biology for programmers.

1.2.3   What this means

I assume *little or NO* knowledge of - programming or command-line scripting - probability and statistics - *computational biology*

I assume *substantive* knowledge of - molecular and cellular biology - chemistry and biochemistry

If you're here to learn new computational approaches and incorporate them into your current lab work, welcome!

If you're here to learn biology to apply your existing computational knowledge, you may want to look elsewhere

1.3 Why Python?

1.4 Python

The language is designed from the ground-up to be easy to use.

It's a full-featured (like C++ or Java), powerful language that can be used in a lot of different contexts.

Most importantly, it's **free**: a completely open-source platform that costs nothing.

Python is also an extremely popular language.

Of course, while that may turn off the programming language hipsters, it does mean that the language and its surrounding ecosystem has lots of momentum. This comes in handy when we want to explore specific problems!

Between 2017 and 2018, Python actually attained the status of Most Popular Programming Language!

Worldwide, Python is the most popular language, Python grew the most in the last 5 years (14.3%) and PHP lost the most (-6.5%)
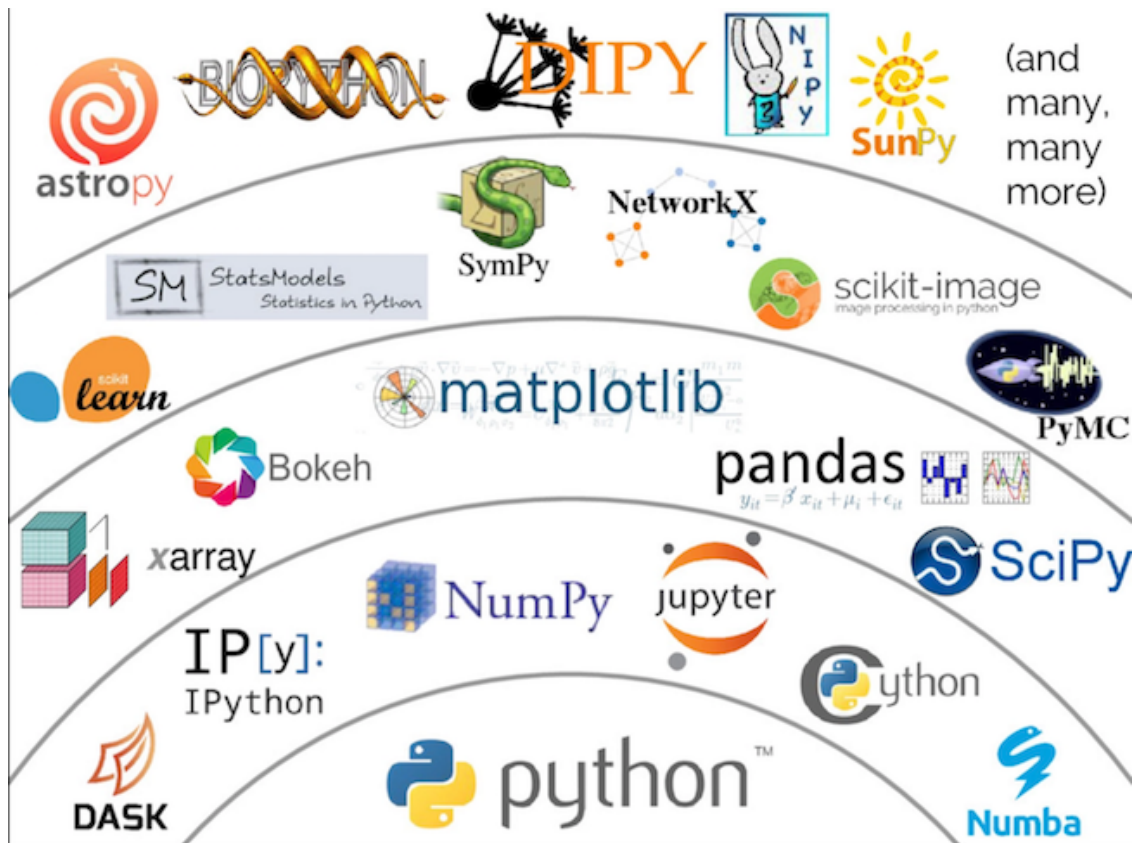
PYPL PopularitY of Programming Language
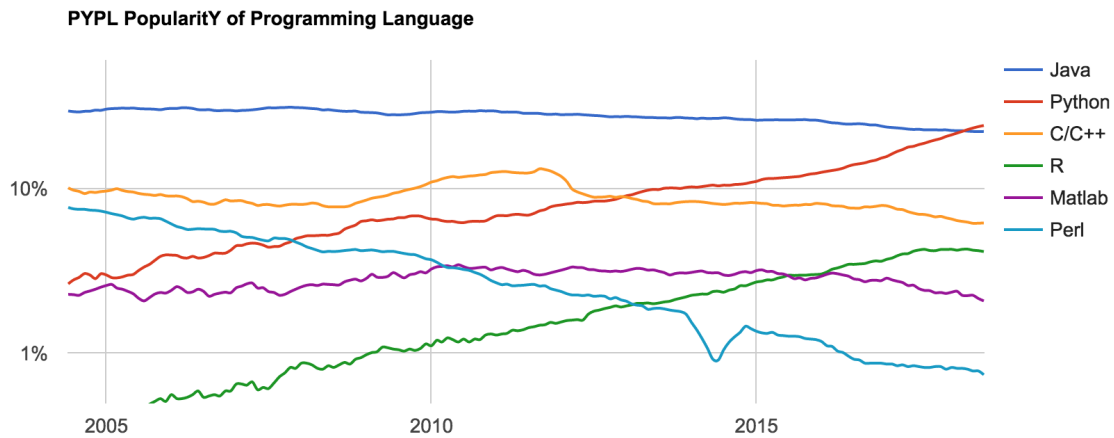
Worldwide, Jan 2020 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	29.72 %	+4.3 %
2		Java	19.03 %	-1.9 %
3		Javascript	8.2 %	+0.1 %
4		C#	7.28 %	-0.2 %
5		PHP	6.09 %	-1.1 %
6		C/C++	5.91 %	-0.3 %
7		R	3.72 %	-0.2 %
8		Objective-C	2.47 %	-0.6 %
9		Swift	2.36 %	-0.2 %
10		Matlab	1.79 %	-0.2 %

See more programming language statistics here: http://pypl.github.io/PYPL.html

1.5   ... but why Python?

The ecosystem!

1.6   Python is cool, Computational Biology is a thing…so what?

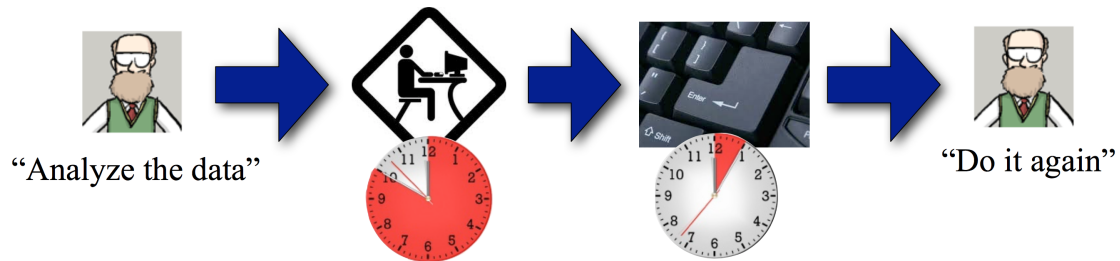Let's look at the current wet lab experimental pipeline.

"Analyze the data"

Seems good enough. Now, let's add a wrinkle.

"Analyze the data"   "Do it again"
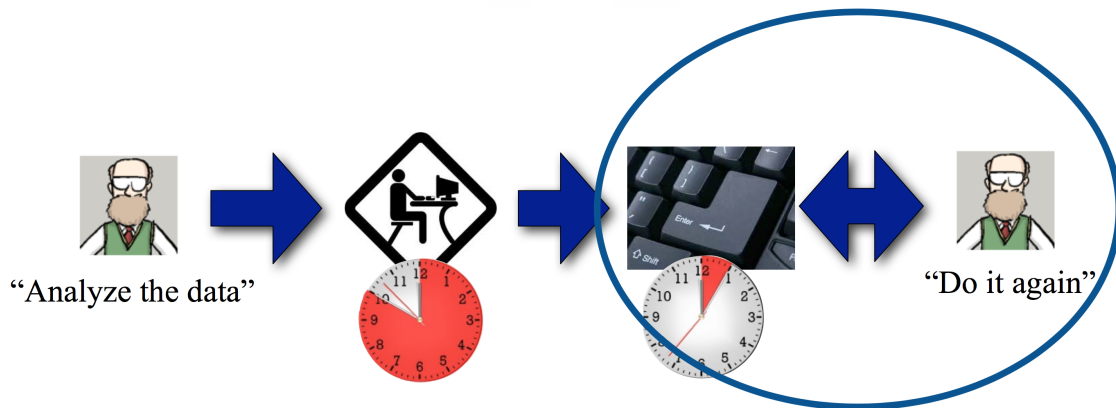
Oh, goodie. Back to the lab for another round of sleepless nights.

You can imagine how this can go on and on. Wouldn't it be nice to collect data once and, I don't know, *automate* the analysis?


"Analyze the data"   "Do it again"

It's that step at the end that's key.

"Analyze the data"   "Do it again"

In some sense, it's the goal of computational biology to *become* biology–that is, "computational biology" will be a redundant way of referring to the overall field.

If anyone is familiar with the buzzphrase of the 2010s "data science", you can think of computational biology as data science in biology.

In this course, we'll combine programming in Python with statistics to automate analyses of biological data.
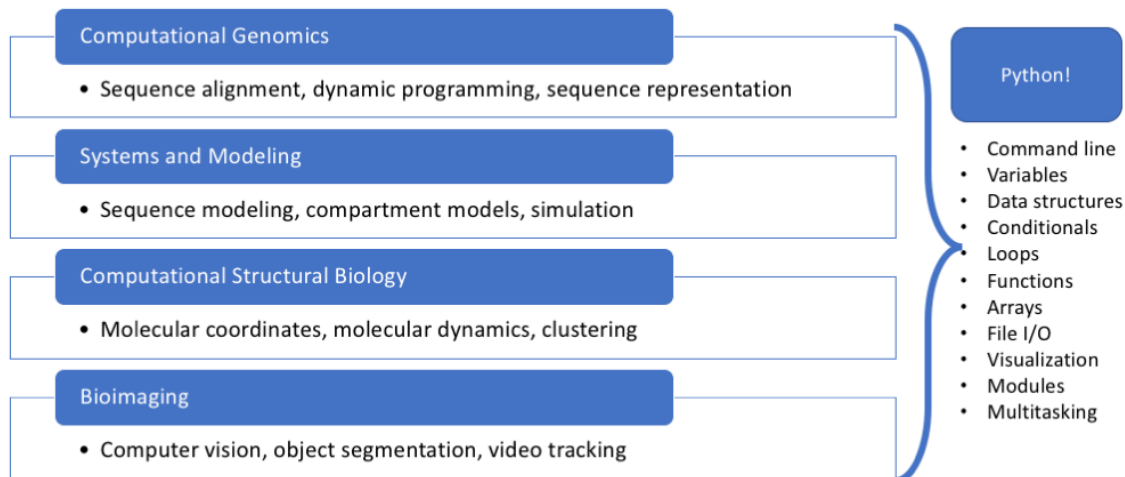
1.7 Course Logistics

All lecture materials will be posted on the course website:

https://eds-uga.github.io/cbio-x835-sp20/

You **do NOT have to** install Python on your own computer; the only requirement is that you have something with a modern web browser and an internet connection.

However, installing Python on your own machine does give you more tinkering elbow room, though (instructions to this effect to follow in a future lecture).

1.7.1 Lectures

Attendance is NOT mandatory. You're all adults.

That said, my least favorite question is from a student I've never seen in lecture coming to office hours for the first time the day before the midterm, asking me to summarize the semester for them.

Make yourself a regular in lecture, a regular on the Slack chat (asking AND answering questions), and letting me know when you need to miss lecture (we've all got things to do; you don't have to ask permission, just let me know when you won't be there), **and you'll be fine**.

1.7.2 Grading

The breakdown is summarized in the course syllabus (linked on the website, also found here: https://eds-uga.github.io/cbio-x835-sp20/syllabus.pdf).

1.7.3 Assignments

There will be 6 assignments. They'll be released on Thursdays and due by 11:59pm two weeks later.

They will all be released over JupyterHub in the form of Jupyter notebooks.

We'll go more over this format in a future lecture, but suffice to say you'll do the assignments in the entirety through a web browser. Thus, you won't need to install Python on your own machine unless you want to.

1.7.4 Projects

Final projects will consist of three main components:

- A brief proposal, outlining the project you want to run, the dataset you'll use, and the computational experiments you'll perform
- A 15-minute presentation at the end of the semester, outlining your major results
- A full conference/journal-ready paper, detailing the problem you worked on, the methods you used, and the results you obtained

More details will be released later in the semester. Be thinking about what you might want to do!

1.7.5 JupyterHub

https://jupyterhub.cs.uga.edu

Assignments will be released, completed, and submitted by JupyterHub. Everyone should have their own login (if you don't or it isn't working, let me know!).

The assignments will be in Jupyter notebook format. Jupyter notebooks will come with their own autograders, so you can run those tests on your completed sections before submitting. Occasionally there are errors in the autograders, so if you suspect your code is correct even though the autograder is throwing errors, post about it in Slack!

You'll subsequently submit completed Jupyter notebooks through JupyterHub as well. **This final step is critical; you have to click "Submit" for me to see your assignment and give it a grade!** In the past this has only rarely been a problem, but nonetheless something to keep in mind.

JupyterHub is more than just a conduit for homework assignments. You can also create your own Jupyter notebooks and experiment with Python! This is a great alternative to installing Python on your own computer. I highly encourage you to do this!

### 1.7.6 Slack

https://eds-uga-cbio4835.slack.com/

Slack is the primary way we'll keep in touch over the semester.

- I'll post announcements in the `#announcements` channel (please keep it clear).
- If you have any questions related to course concepts or homework problems, please post in `#questions`.
- If you're unable to log in, or are getting strange errors you can't diagnose, let me know in `#techprobs`.

- If you found a certain topic in class interesting and want to discuss it further, or found a cool article related to computational biology, or something else entirely, feel free to strike up a discussion in the #lounge.

You can also DM me or your classmates directly (this is WAY better than email for me!).
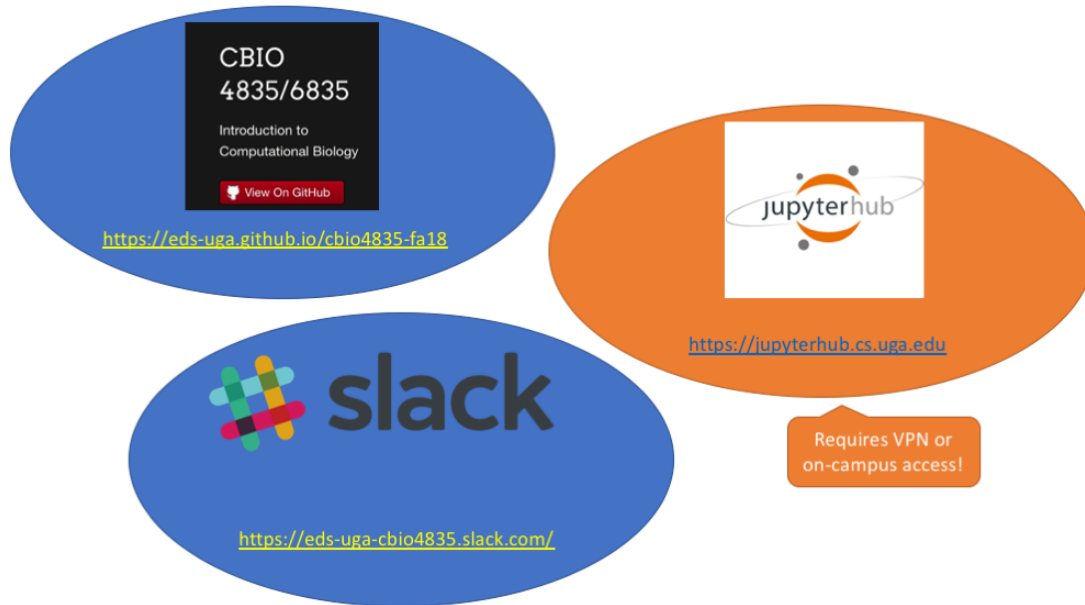
1.7.7 Office Hours

My office is located in **Boyd GSRC, room 638A**.

Office hours are **current TBD**, but they'll be in my office (surprisingly).

You are also welcome to set up a separate appointment with me! DM me over Slack or shoot me an email to set up a time to meet.

1.7.8 Academic Honesty

Do your own work. Programming and coding is a lot like writing–everyone has their own style, so it's very easy to spot copied work.

That said, **please do discuss concepts and problem solving strategies!** Either during in-person group meetings or over group chats on Slack.

One reason I like Slack is because I can jump in when I see a question, but if I'm otherwise occupied someone else might answer it first.

1.8 Pre-test

This is an ungraded survey that will help me to assess everyone's background and properly calibrate the course. It's ungraded, but it is **required that everyone finish it by Thursday's lecture.** (Let's call it: "Assignment 0")

https://docs.google.com/forms/d/1ka9yH5G3bOCfdJUTaeZXV2BdtvqqsiPaxnvKI2f4YK4/

1.9 Administrivia

- **"Assignment 0" due Thursday**
  - Finish the pre-test
  - Read the syllabus
  - Make sure you can log into JupyterHub
  - Make sure you can access Slack

1.10 Questions?

- Questions? Check the course website: https://eds-uga.github.io/cbio-x835-sp20

- Still more questions? Ask on the Slack chat: https://eds-uga-cbio4835.slack.com/

- Still more questions? Email me: squinn@cs.uga.edu (include **CBIO 4835** or **CBIO 6835** in the Subject line so it hits my filters)

- Still more questions? Erm. . .