Welcome to BILD62

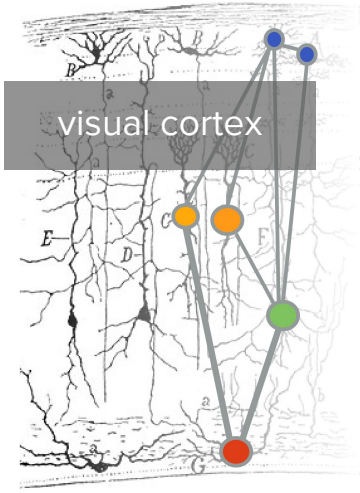
Dr. Juavinett
jah-vah-nett
(or, Dr. J)

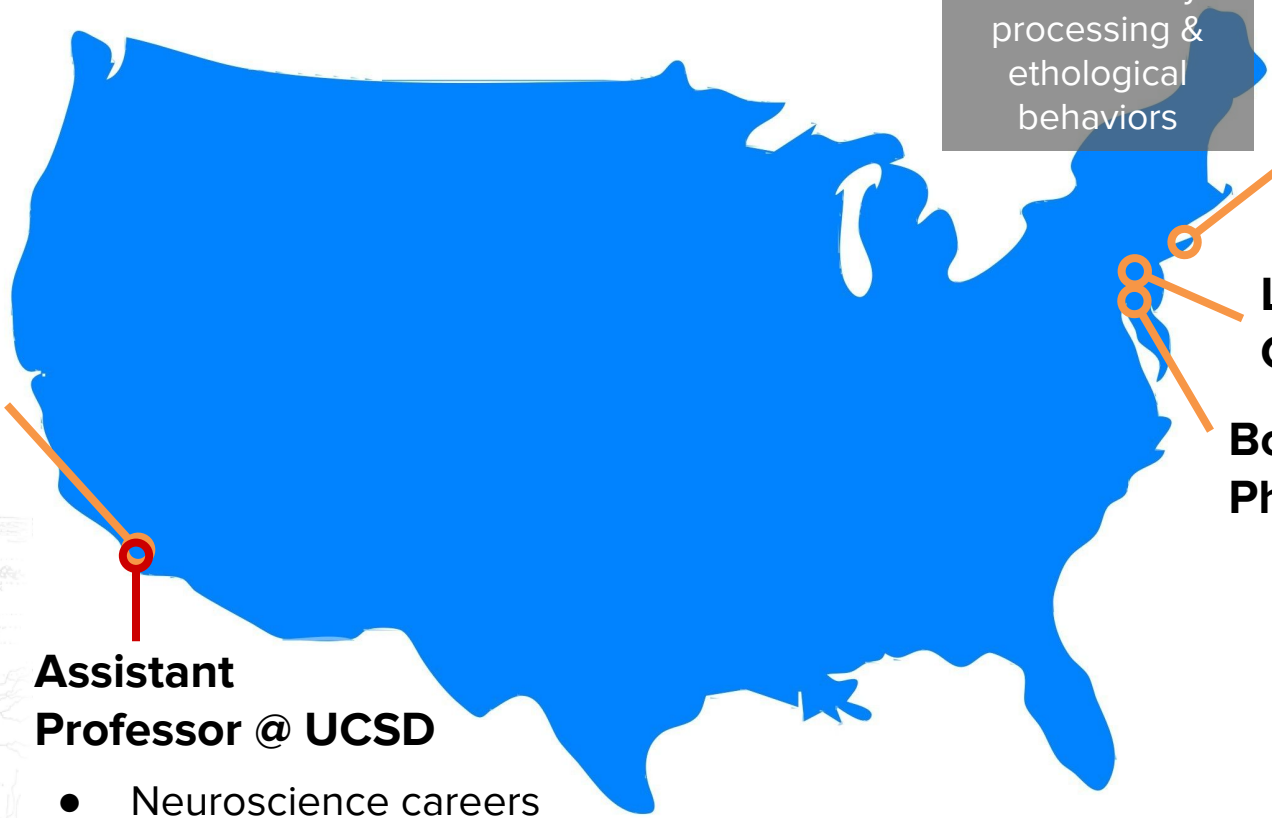
Image: garetsworkshop/Shutterstock

Objectives for this morning

- Introduce the teaching staff, students, and class
- Motivate learning how to code as a biology student
- Discuss course logistics, expectations, & tools

PhD in Neuroscience @ UCSD





multisensory processing & ethological behaviors

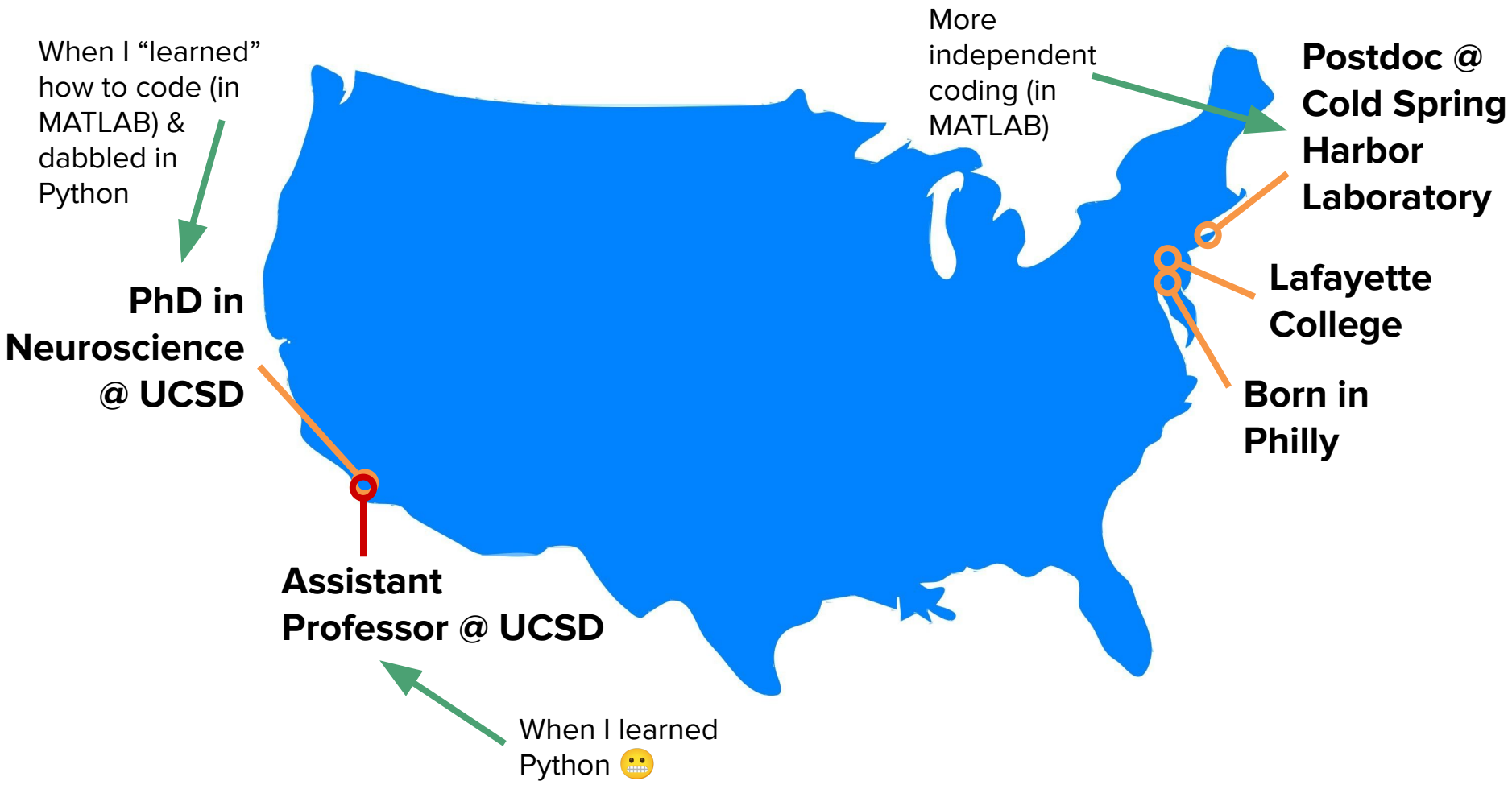
Postdoc @ Cold Spring Harbor Laboratory

Lafayette College

Born in Philly

Assistant Professor @ UCSD

- Neuroscience careers & education
- Open-source data


When I “learned” how to code (in MATLAB) & dabbled in Python
PhD in Neuroscience @ UCSD
Assistant Professor @ UCSD
When I learned Python 😬
More independent coding (in MATLAB)
Postdoc @ Cold Spring Harbor Laboratory
Lafayette College
Born in Philly

Introduction to our Instructional Assistant!

Jiawei Shen
MS Student in the Gleeson Lab
jis215@ucsd.edu

Discussion Sections:
- Monday, 3-3:50 pm in U301 122
- Wednesday, 9-9:50 am in WLH 2206

Let’s be human, for just a second.

With the folks next to you, share:

- Your name, major, and preferred pronouns
- Your favorite snack
- Why you’re taking this course

Objectives for this morning

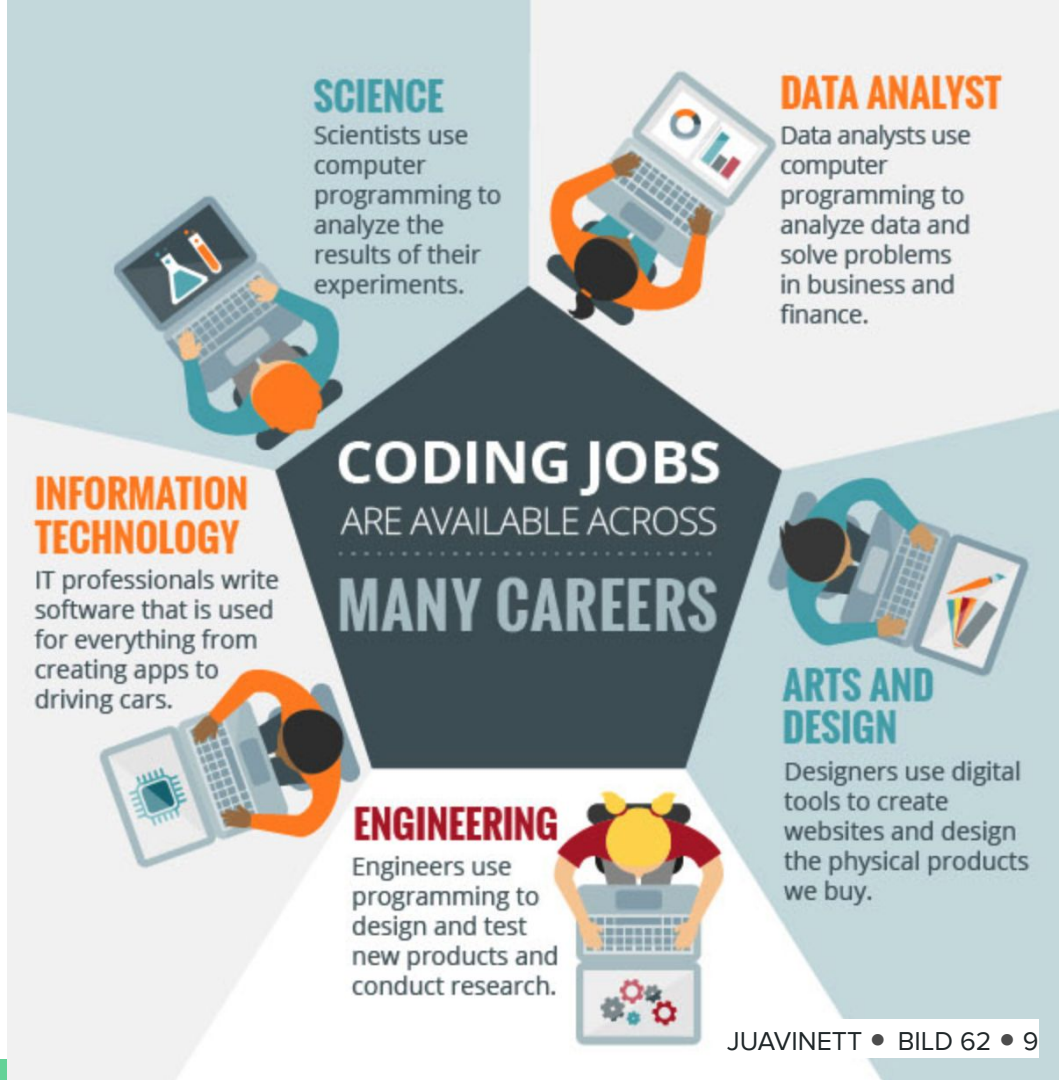
- Introduce the teaching staff, students, and class
- **Motivate learning how to code as a biology student**
- Discuss course logistics, expectations, & tools

What does coding have to do with *biology*?
Why *you, right now*?

Why should I learn how to code?

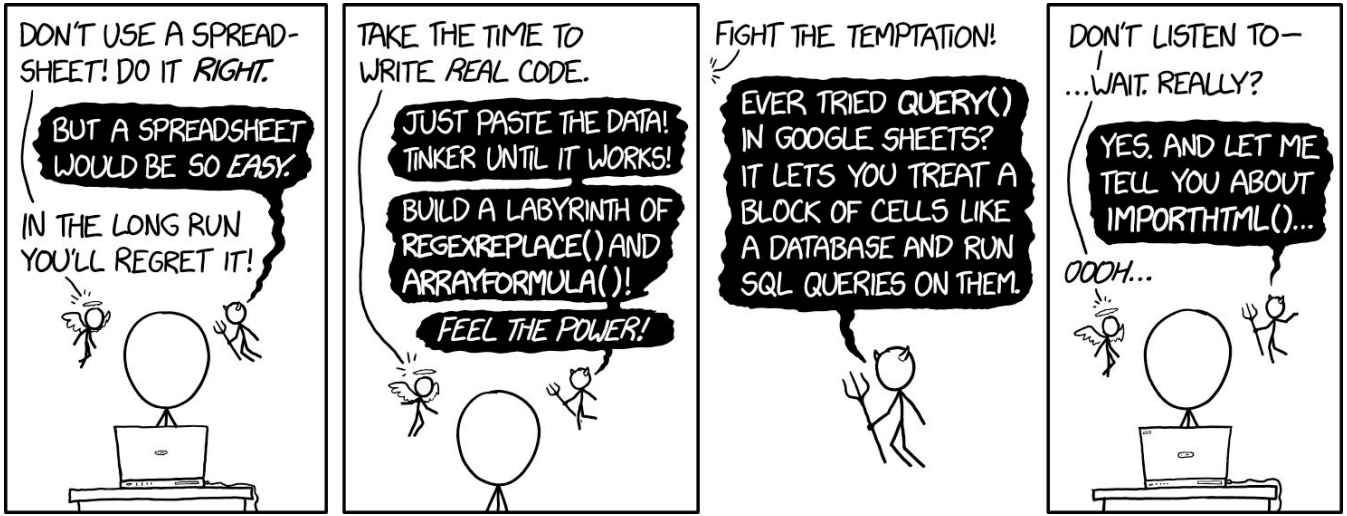
- Coding is useful for:
  - Data acquisition (controlling hardware, image acquisition, etc)
  - Data analysis & visualization
  - Computational modeling
- Beyond research, there are more and more jobs for software engineers, and they pay well

(see report by Burning Glass: https://www.burning-glass.com/research-project/coding-skills/)



Excel can only handle datasets with ~1 million rows, and ~16,000 columns — many datasets in biology are much larger than this!

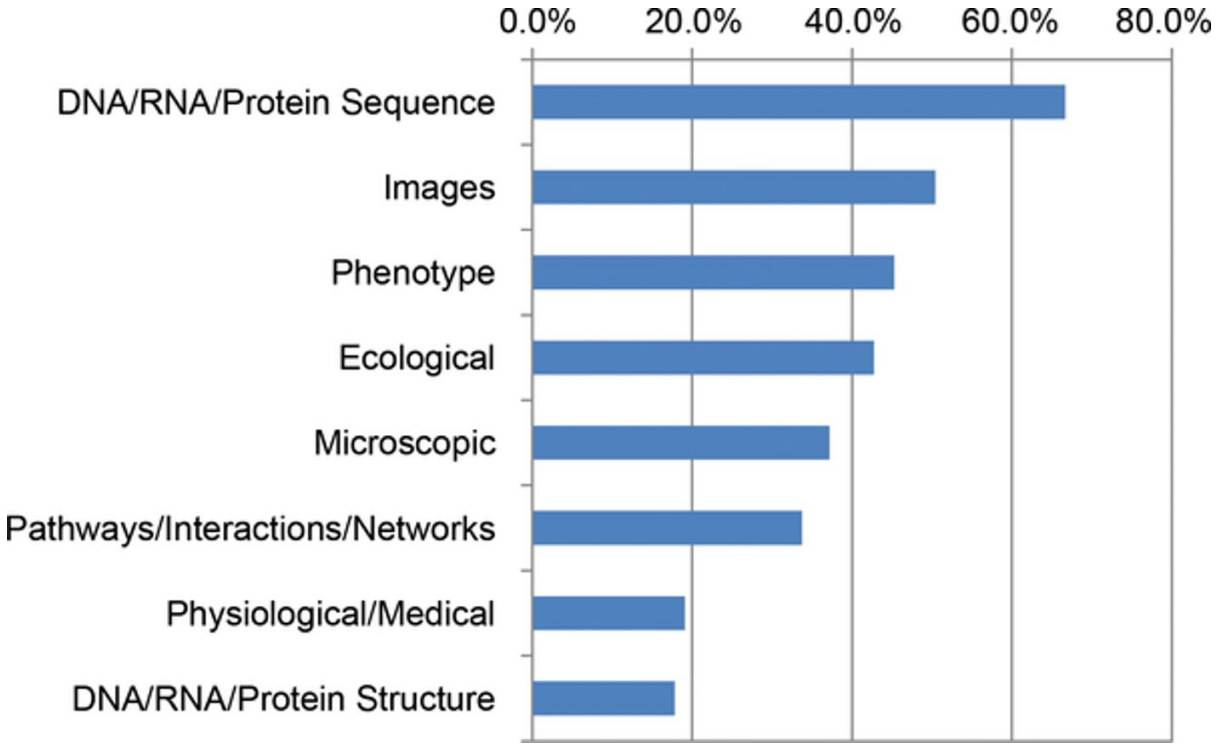
You can automate analyses in Excel, but this is quite limited.



https://xkcd.com/2180/

There are also specialized biological data analysis software programs, but often these are limited in how much they can be customized.

Code is *infinitely* customizable.

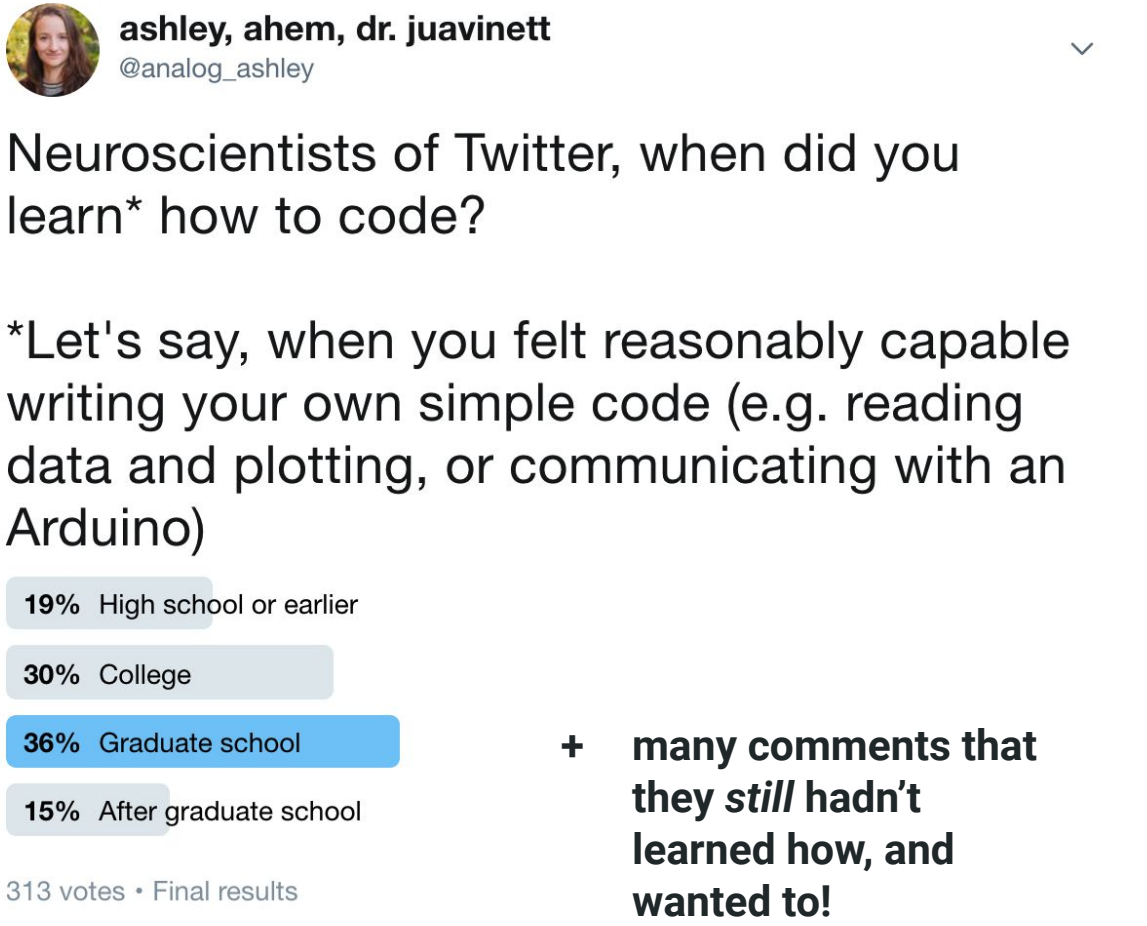


Major data types used by National Science Foundation (NSF) Biological Sciences Directorate (BIO) principal investigators (PIs).

Barone L, Williams J, Micklos D (2017) Unmet needs for analyzing biological big data: A survey of 704 NSF principal investigators. PLOS Computational Biology https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005755

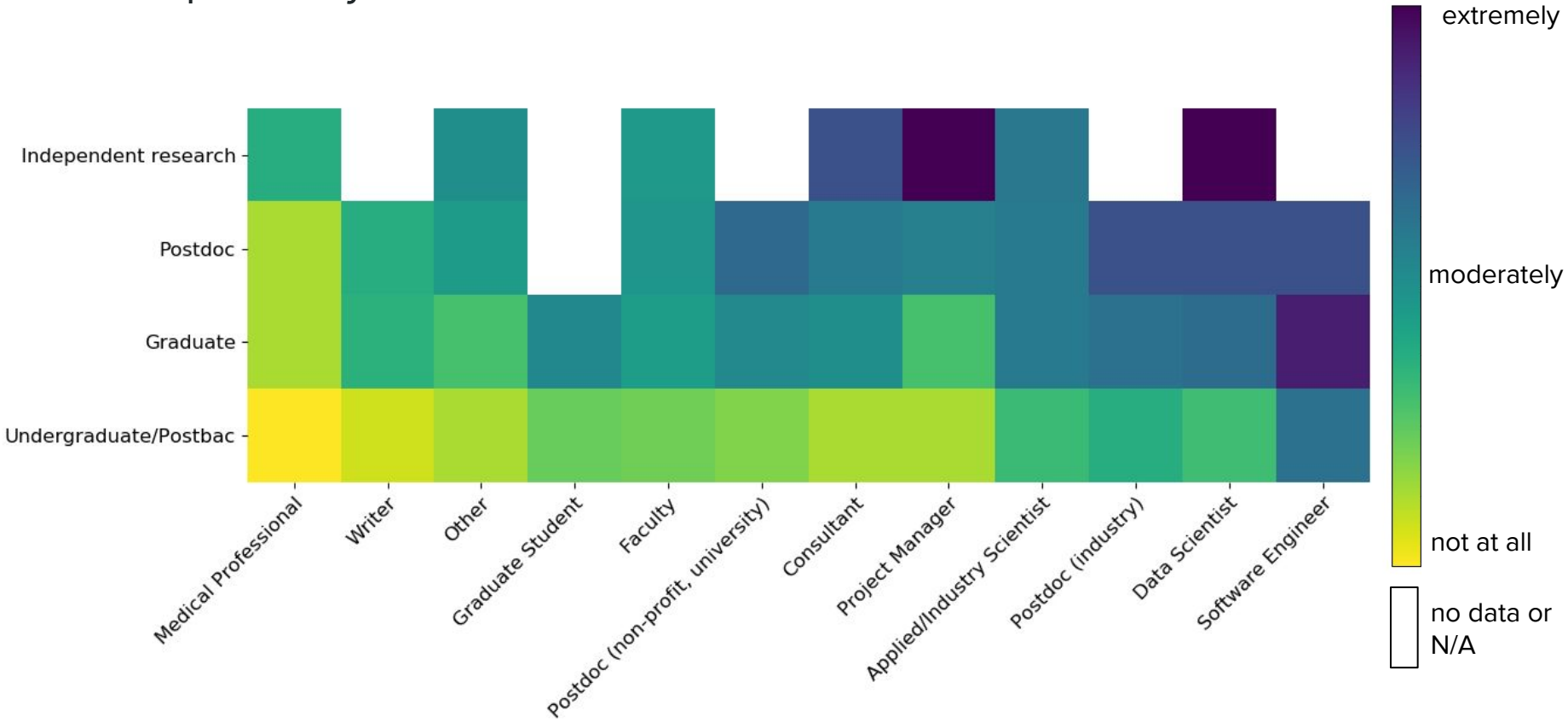
By taking this class, you're ahead of the game!

Many researchers learn to code really informally, and relatively late in their careers



+ many comments that they *still* hadn't learned how, and wanted to!

How comfortable did/do you feel working with code at this point in your career?



Objectives for this morning

- Introduce the teaching staff, students, and class
- Motivate learning how to code as a biology student
- **Discuss course logistics, expectations, & tools**

First step: let's drop our ideas of what it means to be a ***coder***.

Programming, like learning a language, *takes time*.

MASSIVE SCIENCE | Menu | Search | Account

Your language brain matters more for learning programming than your math brain

New research contradicts long held assumptions about coding

Amy Nippert
Neuroscience
University of Minnesota

May 12, 2020 | 2 peer comments

Christina Morillo on Wikimedia Commons.

https://massivesci.com/articles/programming-math-language-python-women-in-science/, summarizes this article: https://www.nature.com/articles/s41598-020-60661-8

Previous studies have shown that math and logic problems seem to rely mainly on the multiple demand regions in the left hemisphere, while tasks that involve spatial navigation activate the right hemisphere more than the left. The MIT team found that reading computer code appears to activate both the left and right sides of the multiple demand network, and ScratchJr activated the right side slightly more than the left. This finding goes against the hypothesis that math and coding rely on the same brain mechanisms.

https://news.mit.edu/2020/brain-reading-computer-code-1215
about this study: https://elifesciences.org/articles/58906

What will help you succeed in this course?

Things that predict success:

- How successful you *think* you'll be
- Completing assignments on time
- Asking questions when you have them
- Attending discussion sections & office hours

Things that ***do not*** predict success:

- Gender
- Age
- Personality
- Math ability

29A @StuxnetStudios · 14h

New programming student:

"I'm not very good at this. When I type out the code, I have to fix lots of errors. And I have to look up how to do most of it."

Instructor:

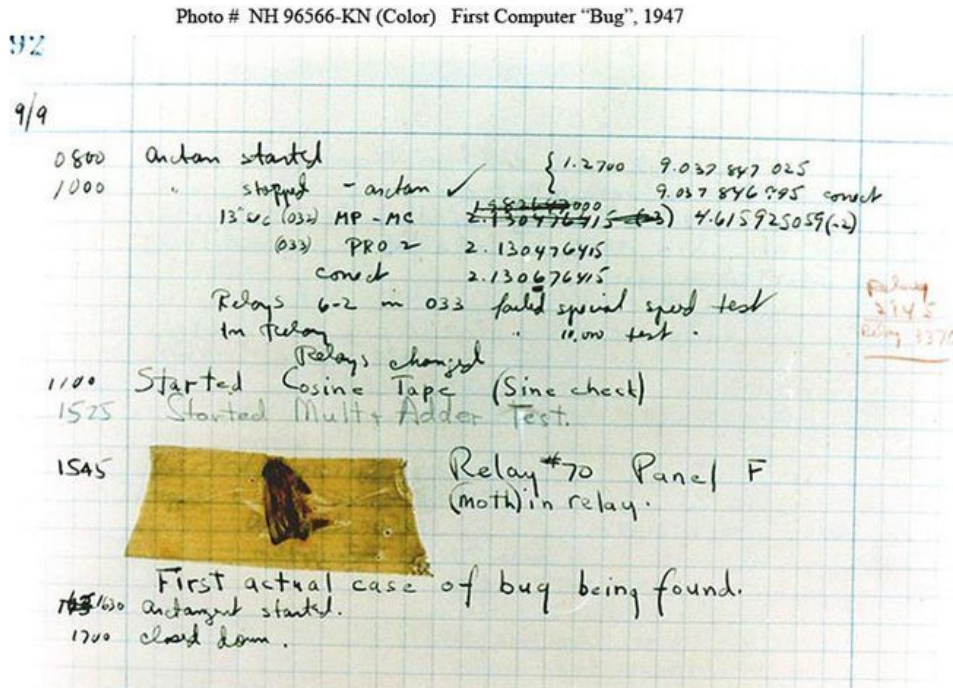
"You're doing it right."

29 275 1.4K

Historical sidenote: why is it called a **bug**?

In 1947, computer scientist & legend **Grace Hopper** found a *literal bug* in their computer, causing it to produce many errors.

Interview with Grace Hopper: https://www.youtube.com/watch?v=QA33wW5LaNY



https://www.nationalgeographic.org/thisday/sep9/worlds-first-computer-bug/

What is programming, anyway?

- Programming is the way humans communicate with computers
  - It's a language!
- The instructions we give the computer are taken **literally** and **sequentially**.

What is programming, anyway?

- Programming is the way humans communicate with computers
  - It's a language!
- The instructions we give the computer are taken **literally** and **sequentially**

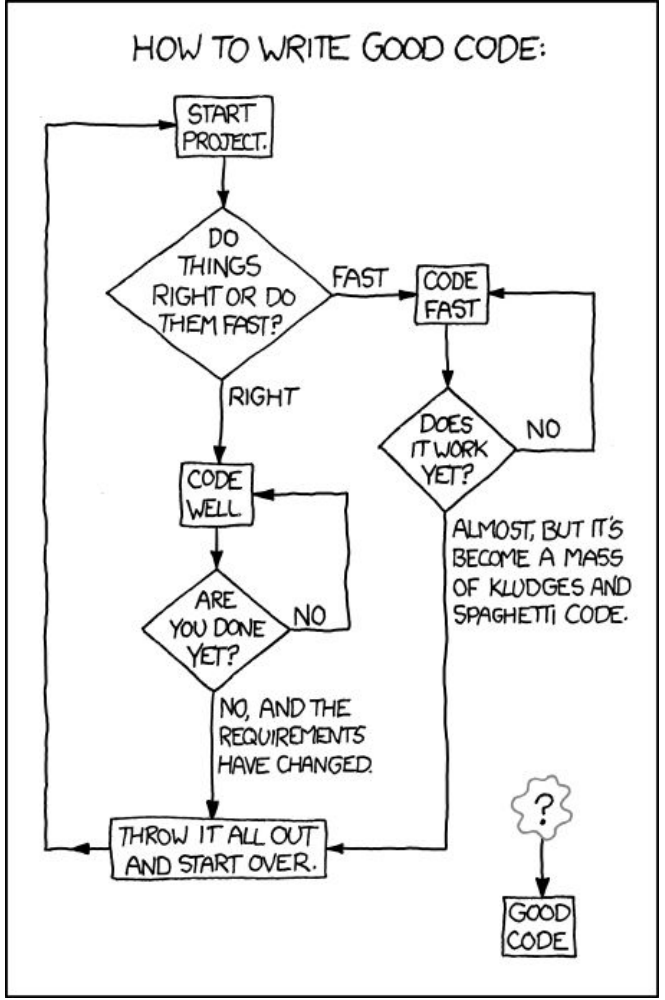
Capitalization matters:
`print() ≠ Print()`

```
b = a * 2
a = 2
```

computer: what is a?

The path to writing good, efficient code

1. Make it **work**
2. Make it **right**
3. Make it **fast**



XKCD, https://xkcd.com/844/

The path to writing good, efficient code

1. Make it **work**
2. Make it **right**
3. Make it **fast**

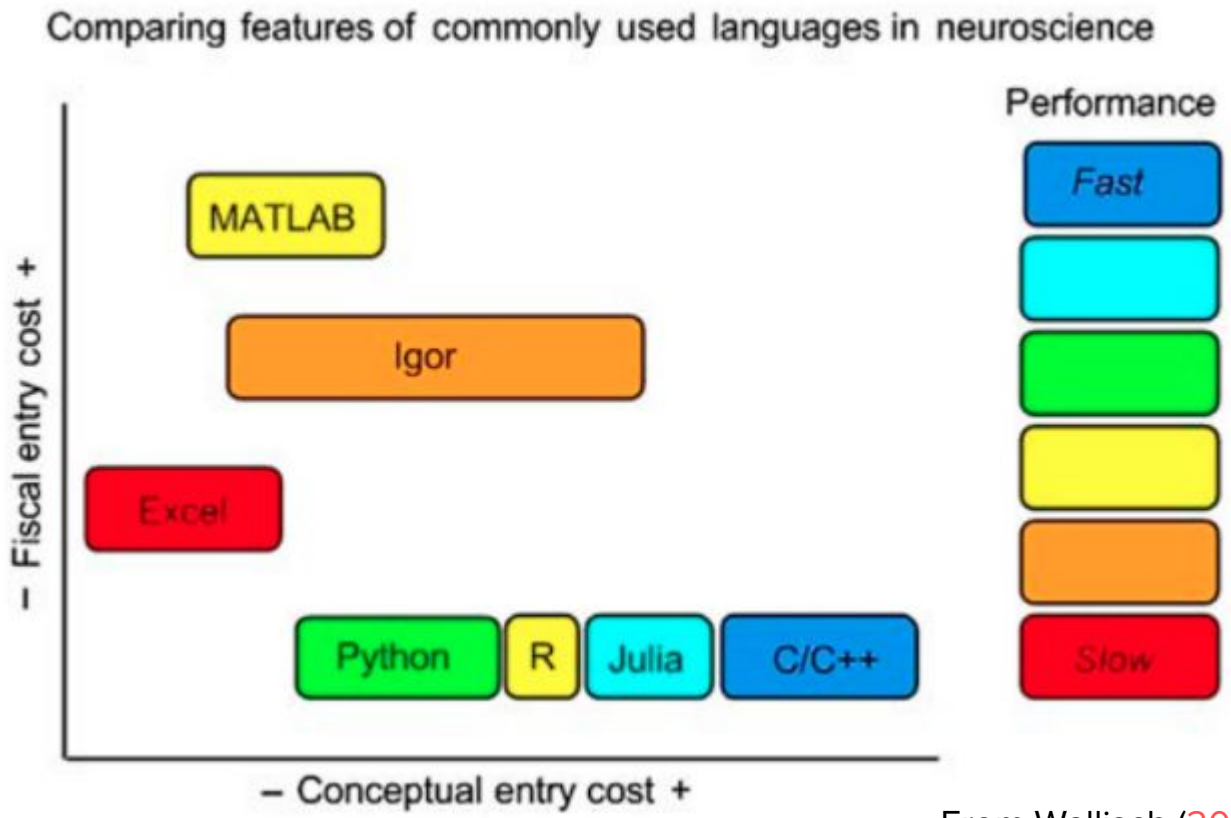
Our goal is to get to this step

If you ultimately became a back-end programmer, you'd care about step 3.

For most problems data scientists face, step 3 isn't paramount.

Considerations for choosing a programming language

- Fiscal & conceptual entry
- Usage in particular field or profession



From Wallisch (2017)

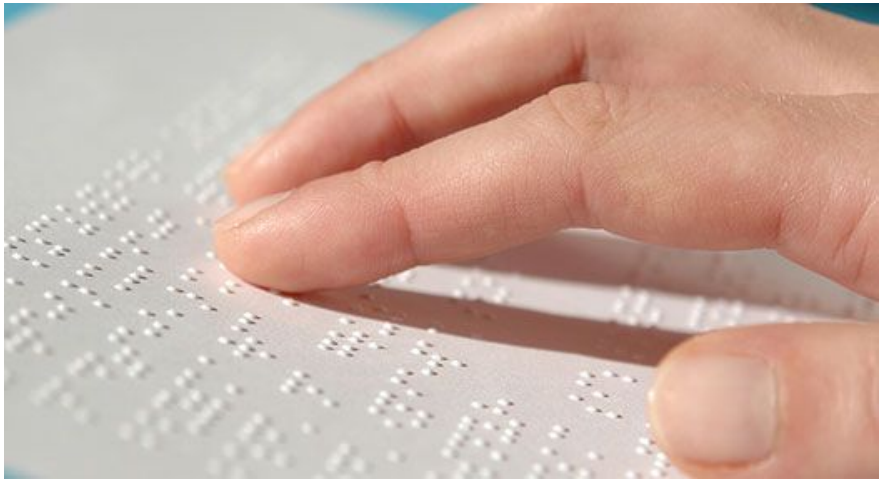
All coding languages eventually need to talk to the computer in binary:

`01001000 01100101 01101100 01101100 01101111 00100001`

(hello)

Learn How To Write Your Name In Binary Code

There are many types of binary code, beyond computers



Braille
https://www.afb.org/blindness-and-low-vision/braille/what-braille

A ●−	J ●−−−	S ●●●
B −●●●	K −●−	T −
C −●−●	L ●−●●	U ●●−
D −●●	M −−	V ●●●−
E ●	N −●	W ●−−
F ●●−●	O −−−	X −●●−
G −−●	P ●−−●	Y −●−−
H ●●●●	Q −−●−	Z −−●●
I ●●	R ●−●	

Morse code
https://www.discoveryworld.org/about/blog/discover_at_home/morse-code/

In this class, we'll use Python

- Programming language, development led by Python Software Foundation (www.python.org)
- Uses concise structure & wording similar to human language
- An **interpreted** language — it doesn't speak *directly* to the computer
- Can be used for many purposes, from web programming, to creating games, to analyzing & visualizing data
  - Extension: '.py'
- We'll also work in **Jupyter Notebooks**
  - Extension '.ipynb'

python™

Course logistics

Course Objectives

- Read and run basic Python programs, recognizing the structures used (i.e. variables, conditionals, loops, functions) and explaining how they work
- Manipulate and create objects in Python, including data structures and classes
- Write, edit, and execute Python code in Jupyter Notebooks as well as the command line
- Visualize and run hypothesis-testing on simple datasets in Python
- Implement common algorithms for analyzing biological data (e.g., time series, images) and determine when such computations are appropriate

Grading breakdown

In-class work & participation (10%)

Assignments (50%)

Midterm (15%)

Final project (25%)

Notes:

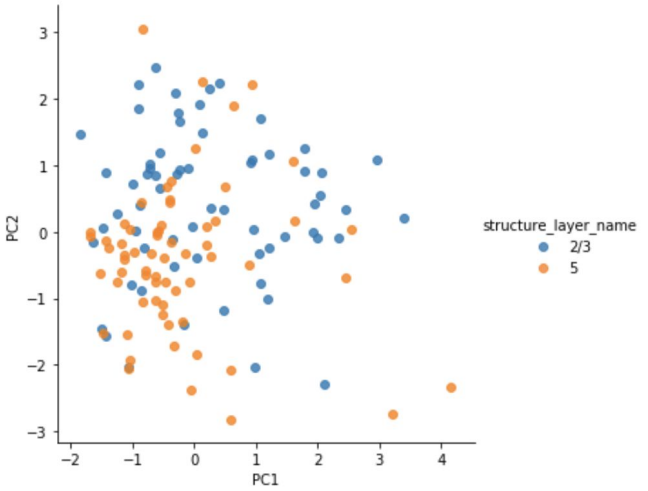
- Assignments & project components lose 10% each day they are late.

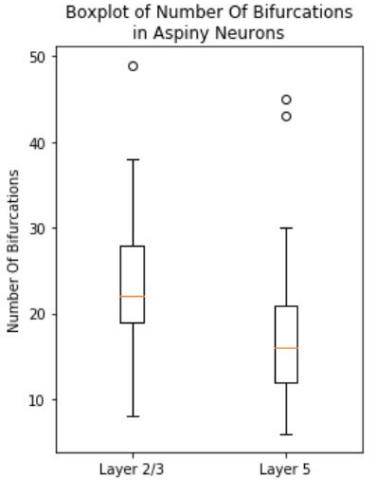
Assignments

- Due **every Monday at 5 pm**
- Worth 2.5-10% each
- Completed individually
- Programmatically graded (via Datahub/NBGrader)
- This week, we'll walk through how to submit these.

Project, groups of 2-3

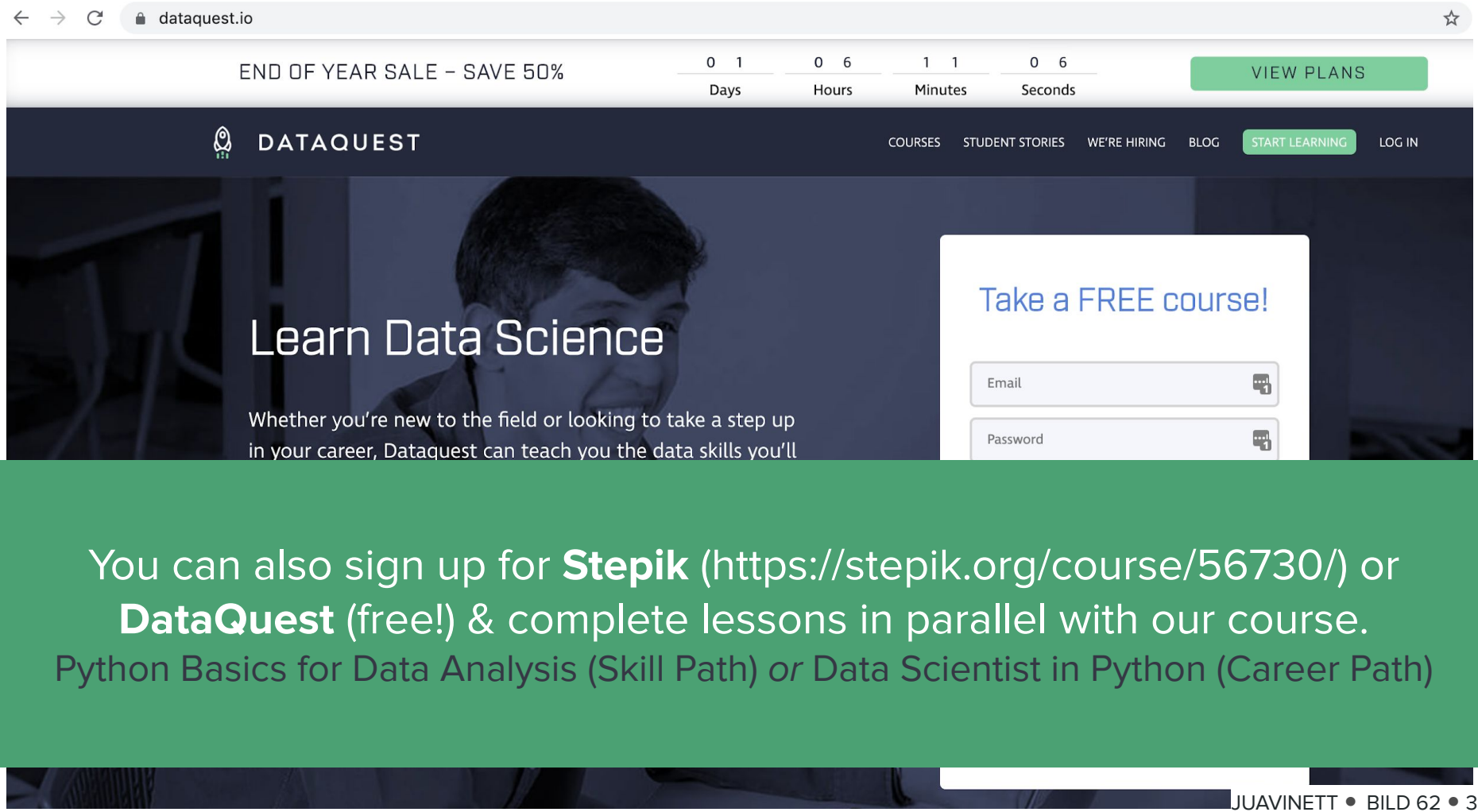
- Includes the project proposal, code, and deliverables.
- Your final project will take some sort of raw biological data and analyze it to draw conclusions.
- We will discuss possibilities for your project as we move through the course.





Organization of content in this course

- **Lectures:** Information that I present in class; many PDF, some via Jupyter Notebooks, or a mix of the two
- **Materials**: Jupyter Notebooks that we will manipulate in class, and that will be useful sandboxes for you
- **Resources**: Additional resources that can help provide more background information to supplement your learning
    - See syllabus as well as links at the end of lectures



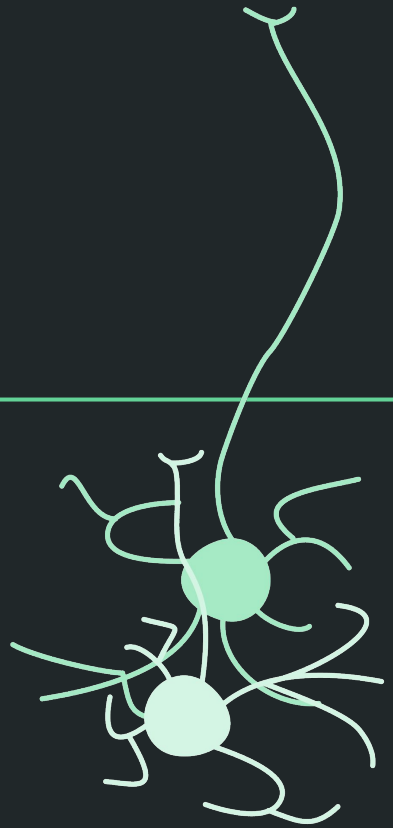
You can also sign up for **Stepik** (https://stepik.org/course/56730/) or **DataQuest** (free!) & complete lessons in parallel with our course.
Python Basics for Data Analysis (Skill Path) *or* Data Scientist in Python (Career Path)

Office hours

Why should you come to office hours?

- You have clarifying questions about the course or its content
- You have concerns about the course and your progress
- You'd like to talk about career paths in biology or neuroscience

Tools for this class



Course Tools

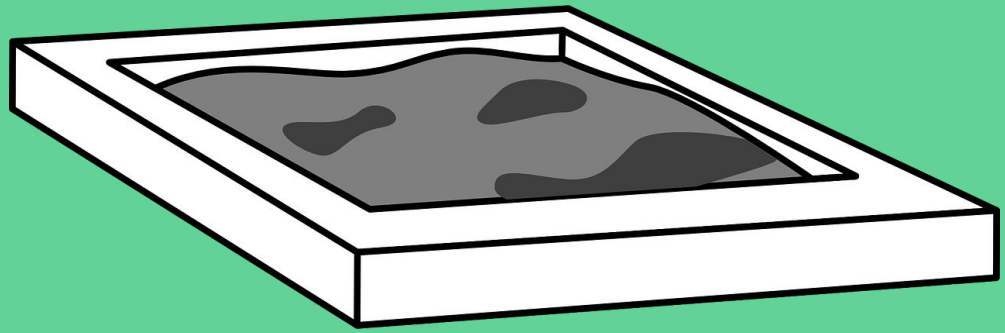
Submitting non-coding assignments & managing grades

Sharing private course content

Coding exercises & assignments

Sharing public course materials
https://github.com/BILD62

Interacting with course materials

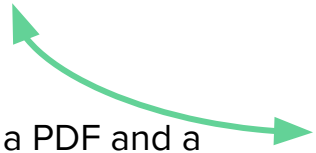


You can find all of our course materials on either Canvas or the course GitHub: http://www.github.com/BILD62

Lectures

In other words, PDF slides shown during class.

Hosted on GitHub.



If I use both a PDF and a Jupyter Notebook during lecture, these numbers will match

Materials

Jupyter Notebooks hosted at http://www.github.com/BILD62/Materials.git

You can pull these locally or to DataHub, or look at them online via GitHub or Binder

launch binder

Assignments

Submitted through **Assignments** tab

Answers to weekly assignments will be posted at http://www.github.com/BILD62/Assignments.git

💫 THE MAGIC LINK FOR THIS COURSE:

https://datahub.ucsd.edu/hub/user-redirect/git-sync?repo=https://github.com/BILD62/Materials

💫 THE MAGIC LINK FOR THIS COURSE:

Sync with your datahub:

https://datahub.ucsd.edu/hub/user-redirect/git-sync?repo=https://github.com/BILD62/Materials

Where our course content lives

To clone Materials to DataHub:

1. Click on the magic link: https://datahub.ucsd.edu/hub/user-redirect/git-sync?repo=https://github.com/BILD62/Materials
2. Log in to DataHub as prompted.
3. You'll be in the Materials folder now!
4. If you want, save your own copy by adding your initials to the end of the file name. **DO NOT DO THIS FOR ASSIGNMENTS!**
5. Next time you click the link, you'll have a fresh copy, plus your copy.

To interact with Jupyter Notebooks on your computer

OPTIONAL

1. Install Anaconda with Python 3.7 for your operating system.
2. If you're using Windows, download git.
3. In Terminal (Mac) or the Anaconda Prompt (Windows), clone the repository by running the following command:

   `git clone http://www.github.com/BILD62/Materials.git`
4. Open Jupyter Notebook. There are two ways to open:
   - In Terminal (Mac) or the Anaconda Prompt (Windows), type **`jupyter notebook`**
   - Open Anaconda Navigator and launch jupyter notebook
5. On the Jupyter landing page, navigate to the notebook and open it.
   - It will open in a browser but is *not* using an internet connection.

This is a new class —
thanks for your patience!

We'll also be trying to learn about how this is going with some ***education research***.

University of California, San Diego
Consent to Act as a Research Subject
Investigating the Impact of Pedagogical Choices on University Student Learning and Engagement

Who is conducting the study, why you have been asked to participate, how you were selected, and what is the approximate number of participants in the study? Gabriele Wienhausen, Director of the Teaching and Learning Commons, together with her education research colleagues is conducting a research study to find out more about how pedagogical choices affect student learning and experience in the classroom. You have been asked to participate in this study because you are a student in a class that is being studied or used as a control. There will be approximately 500,000 participants in this study.

Complete both pre- and post-surveys on Qualtrics, gain 1% extra credit

Why is this study being done? The purpose of this study is to create knowledge that has the potential to improve the learning and educational experience of students at UC San Diego and beyond.

Before next class...

- Take the entry survey https://forms.gle/Hh89AdinCVHqnJJaA
- Take the computing attitudes survey https://ucsd.co1.qualtrics.com/jfe/form/SV_2uuTU5drhQI2JLw
- Access Canvas (canvas.ucsd.edu) & DataHub (datahub.ucsd.edu)
- (Optional) Sign up for Stepik and/or DataQuest
- (Optional) Install Python 3.7 (via the Anaconda distribution) on your computer (https://www.anaconda.com/distribution/)

You only *really* need access to the DataHub, but having the ability to use Python & Jupyter Notebooks on your local computer *may* be useful!