Welcome! To get us started…

Please take the pre-workshop survey:
https://forms.gle/uDjFrXvt1sm1XaEGA

Log into a computer & log into a Google account

Navigate to
http://www.github.com/ajuavinett/Allen2025

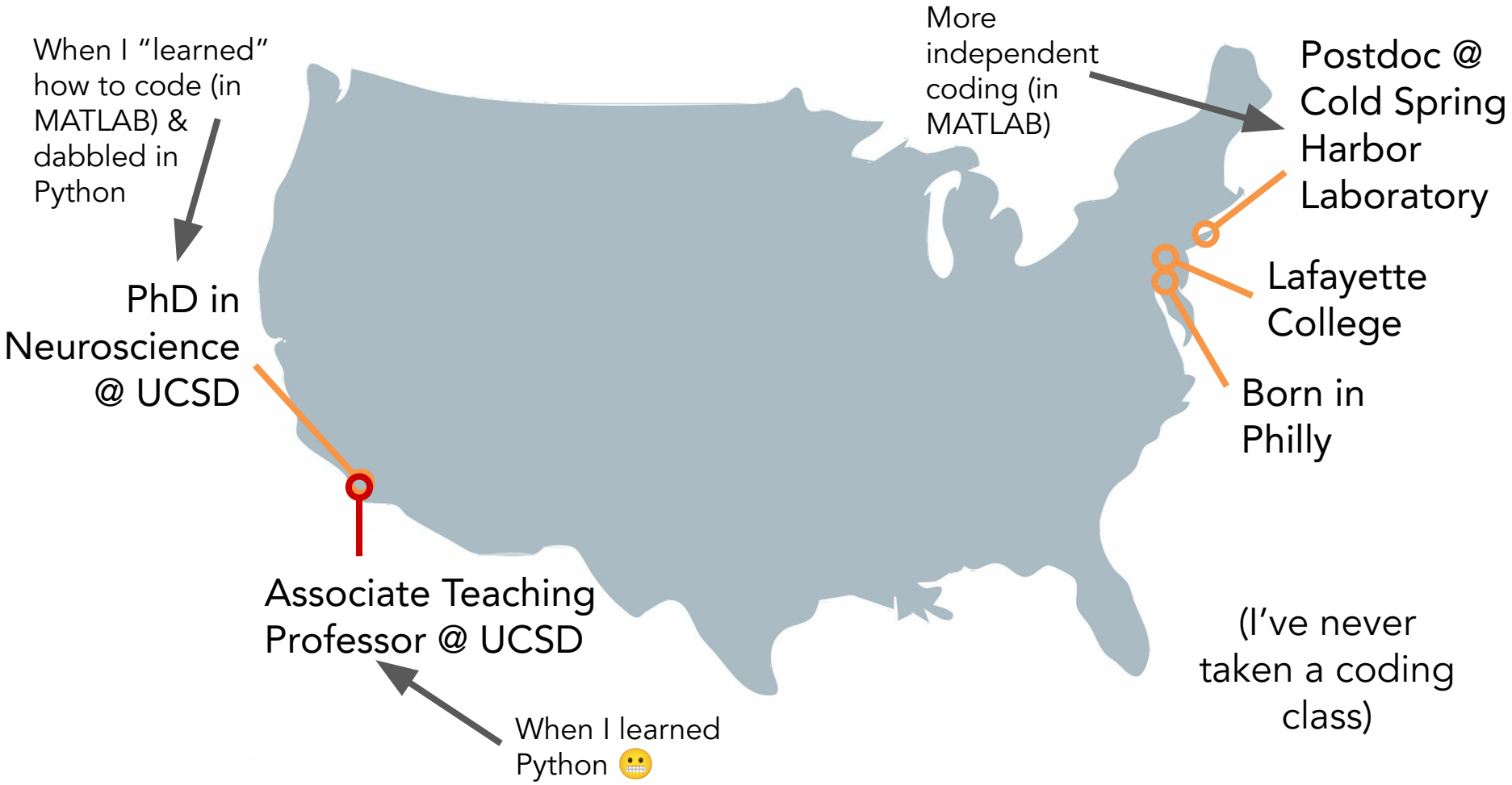
Integrating programming into biology & neuroscience courses

Ashley Juavinett, PhD

@analog_ashley

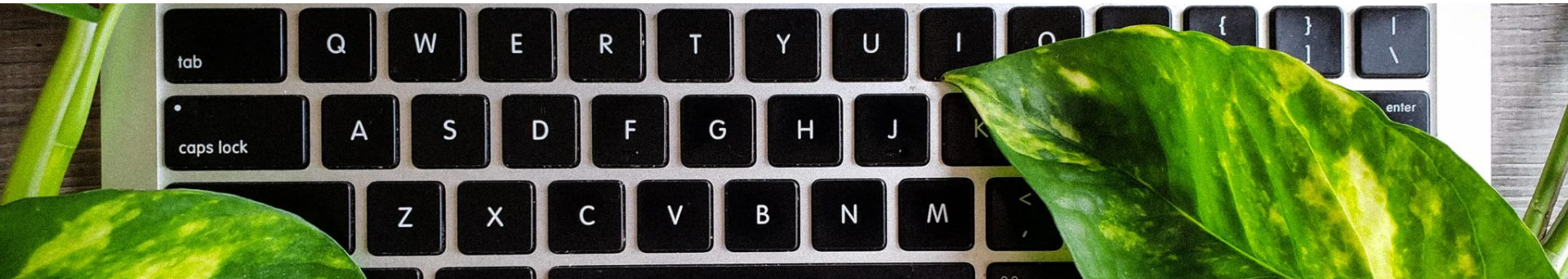
UC San Diego
SCHOOL OF BIOLOGICAL SCIENCES
Neurobiology Department

MSED
Math & Science Education
Doctoral Program


When I "learned" how to code (in MATLAB) & dabbled in Python
PhD in Neuroscience @ UCSD
Associate Teaching Professor @ UCSD
When I learned Python 😬
More independent coding (in MATLAB)
Postdoc @ Cold Spring Harbor Laboratory
Lafayette College
Born in Philly
(I've never taken a coding class)

By the end of this workshop, you'll be able to:

- Identify multiple ways of integrating programming into your course
- Access and use cloud-based computing platforms (e.g. Google Colab)
- Write & edit code in computational notebooks



Workshop plan

Time	**Description**
1:30 – 1:50 pm	**Part I:** Motivation & Framework
1:50 – 2:20 pm	**Part II:** The Tools (choosing a programming language, working with computational notebooks)
2:20 - 3:00 pm	**Part III:** Cell Types Lesson
3:00 - 3:30 pm	**Part IV:** Reflection, Wrap-up & promising practices for teaching coding

PART I

The Motivation

Biological data is getting bigger

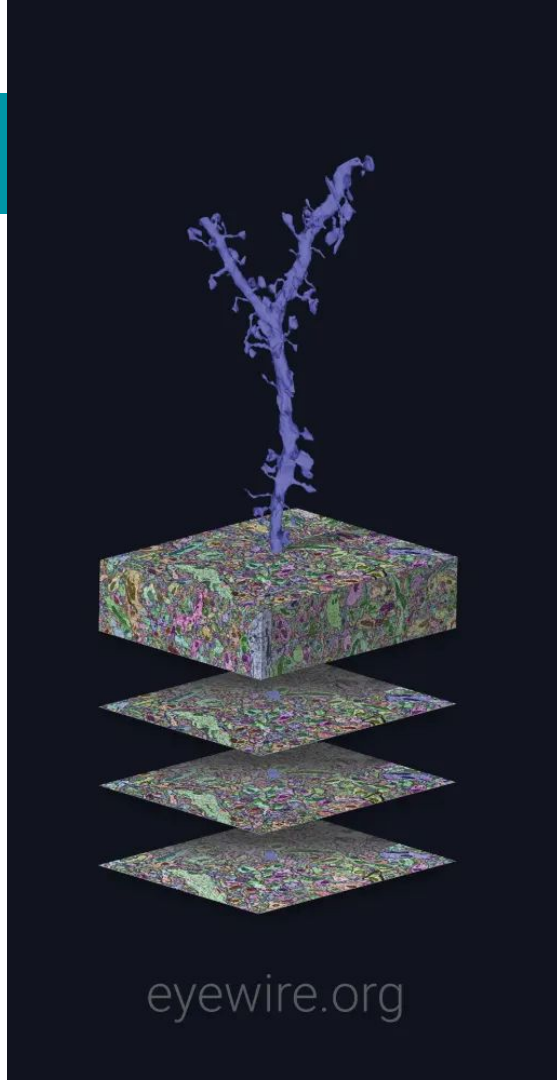
One ~500 mm^3 mouse brain...

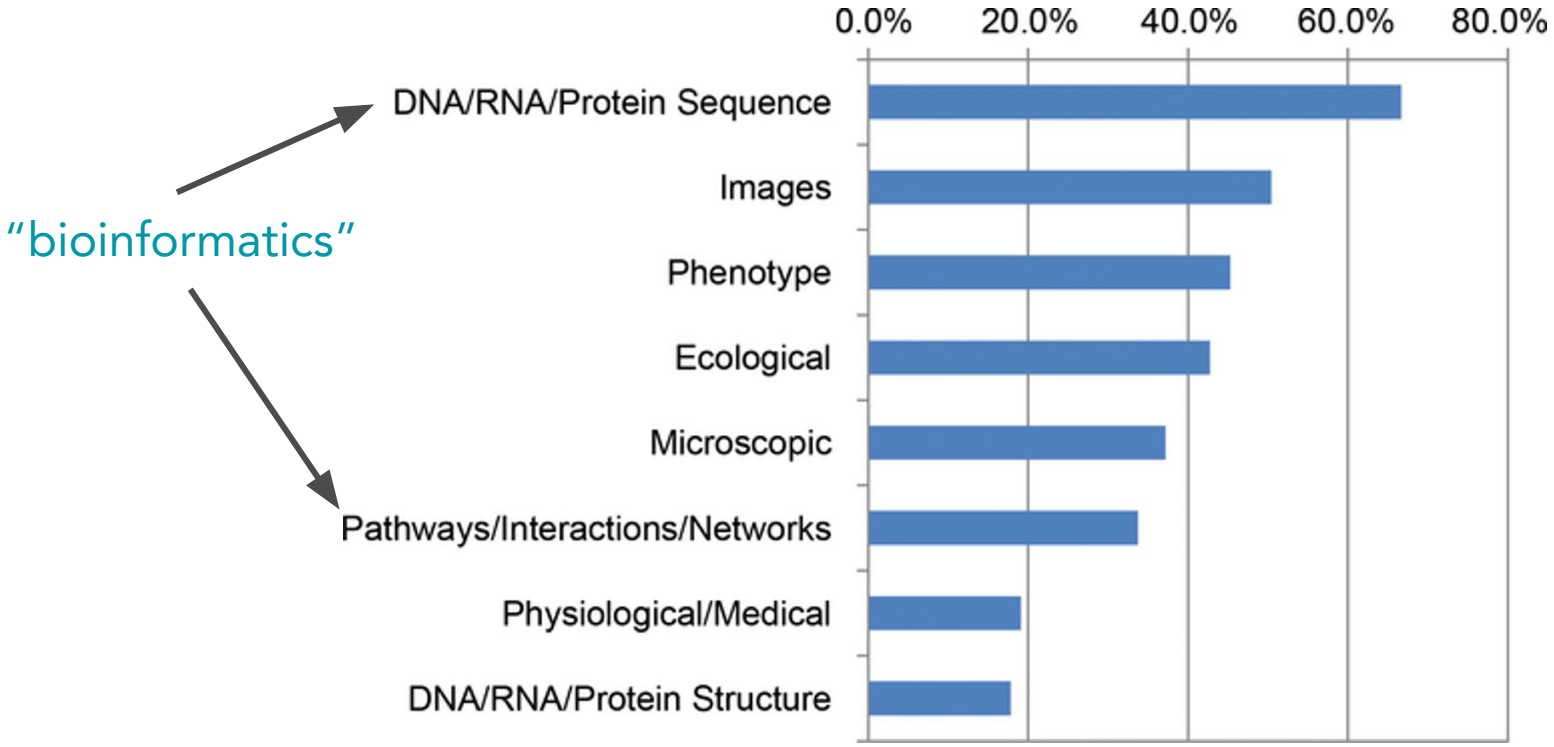
assuming ~1,000 connections for each neuron, the resulting connection matrix contains ~10^{11} entries
≈ a few hundred gigabytes

imaging at 5 nm × 5 nm × 40 nm resolution
≈ 500 petabytes

recording from each pixel for 20 min at 1000 Hz
≈ 500 petabytes

Estimates: Engert, 2014; Image: EyeWire





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

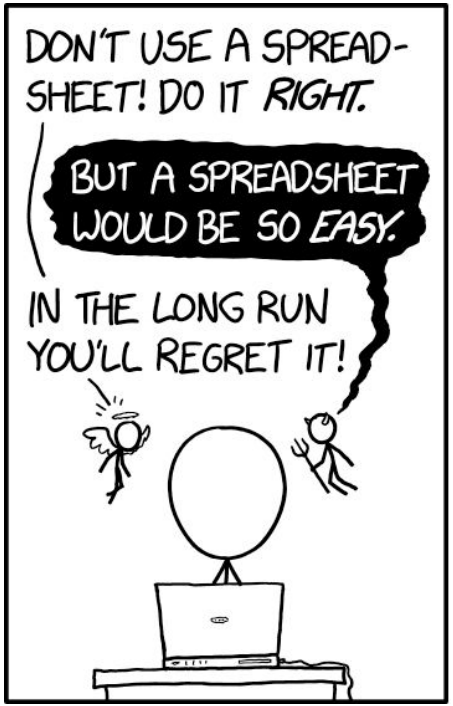
Barone L, Williams J, Micklos D (2017) Unmet needs for analyzing biological big data: A survey of 704 NSF principal investigators. PLOS Computational Biology https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005755

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

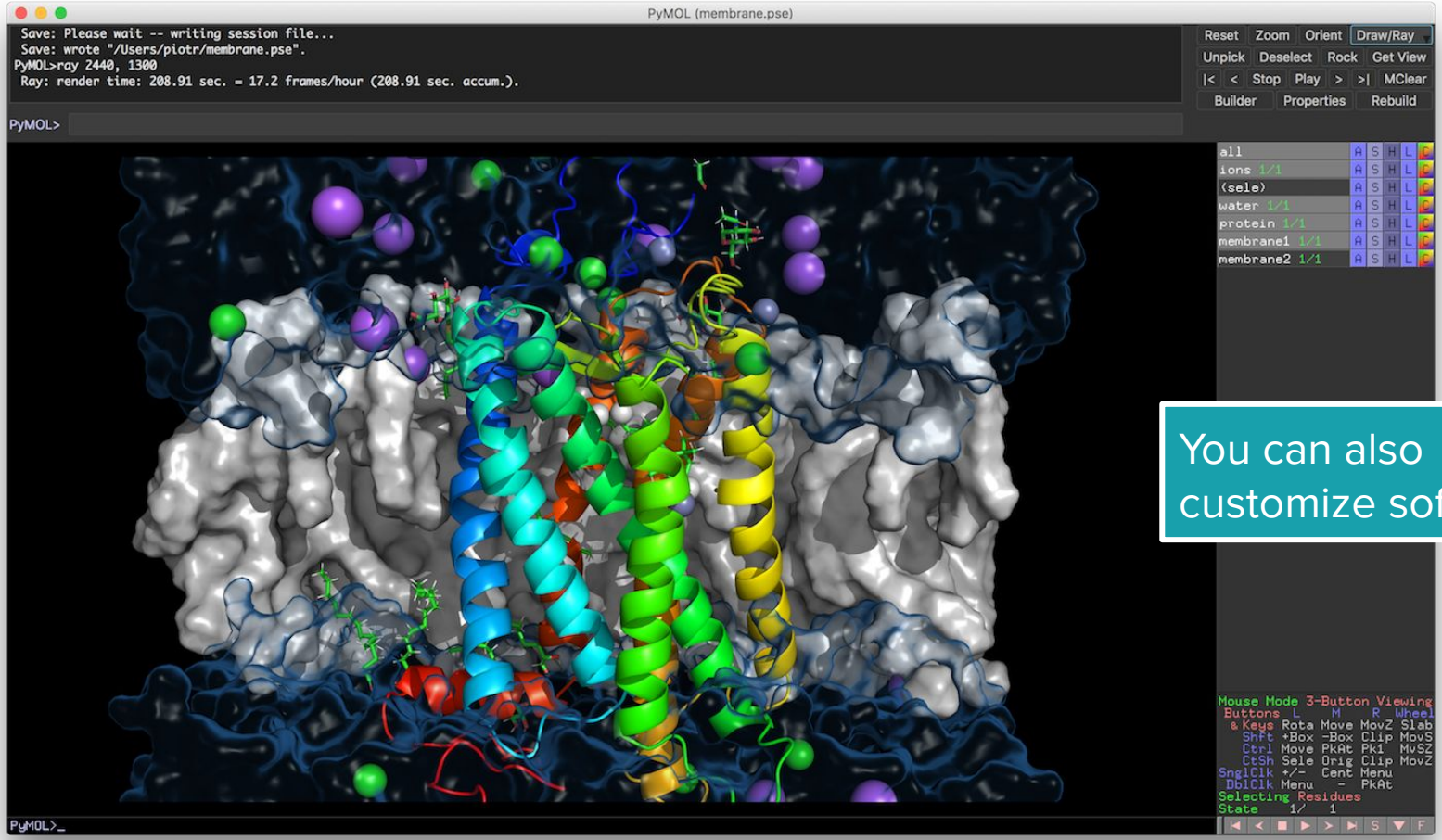
You can automate analyses in Excel, but this is quite limited.

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.



https://xkcd.com/2180/



You can also customize software!

pymol.org

pymol.org/2/

PyMOL by Schrödinger

DOWNLOAD SCREENSHOTS PRODUCTS SUPPORT CONTACT

Open-Source Philosophy

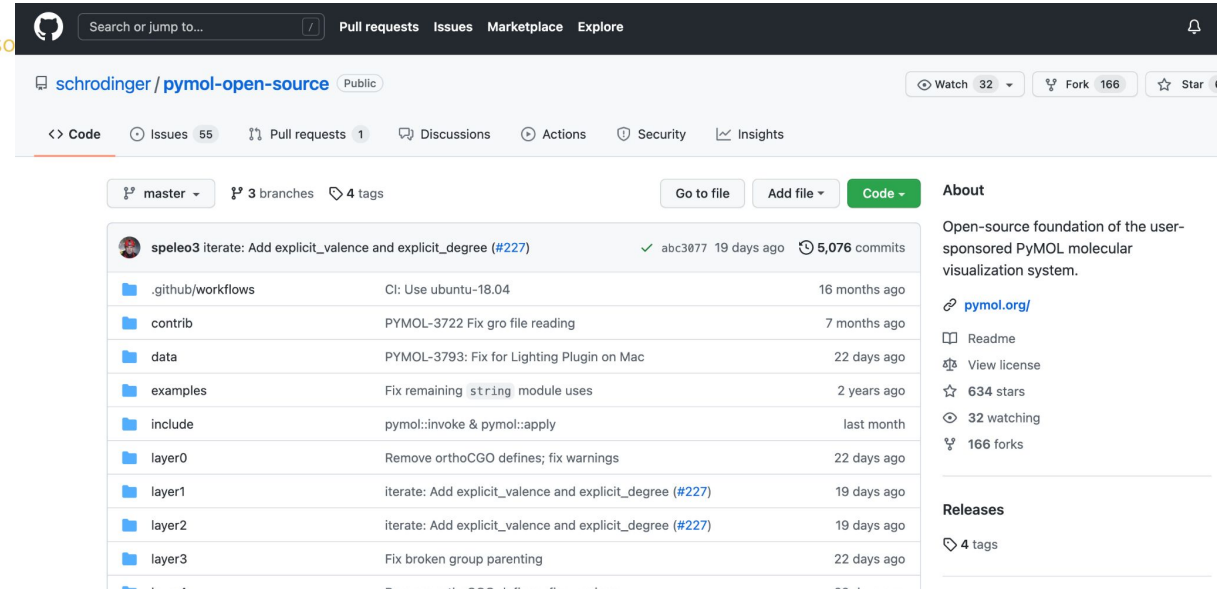
PyMOL is a commercial product, but we make most of its source code freely available under a permissive license. The open source project is maintained by Schrödinger and ultimately funded by everyone who purchases a PyMOL license.

Open source enables open science.
This was the vision of the original PyMOL author Warren L. DeLano.

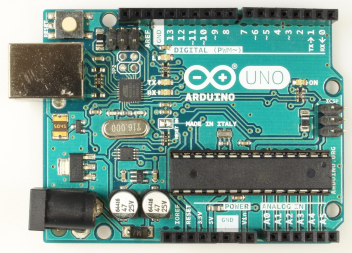
Visit the Open-Source Project

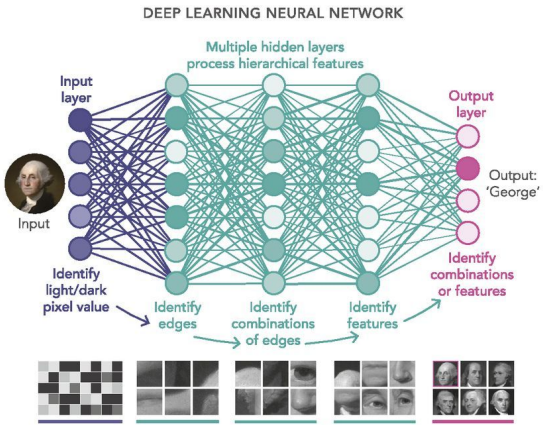
Become a sponso

AND many software packages for biologists can be modified… if you know how to code!



Why do biology students need to learn how to code?





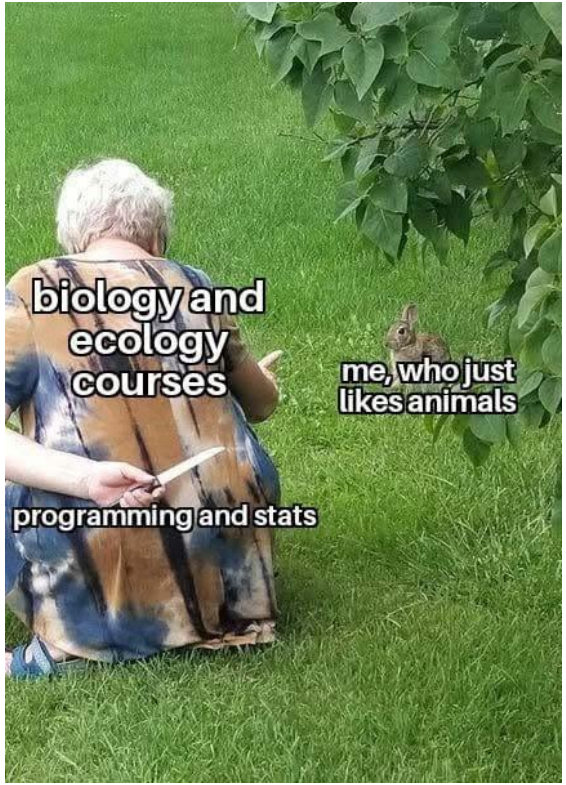
wrangling large datasets

custom data acquisition

statistics, data analysis & visualization

computational modeling

Why do biology students need to learn how to code?



wrangling large datasets

custom data acquisition

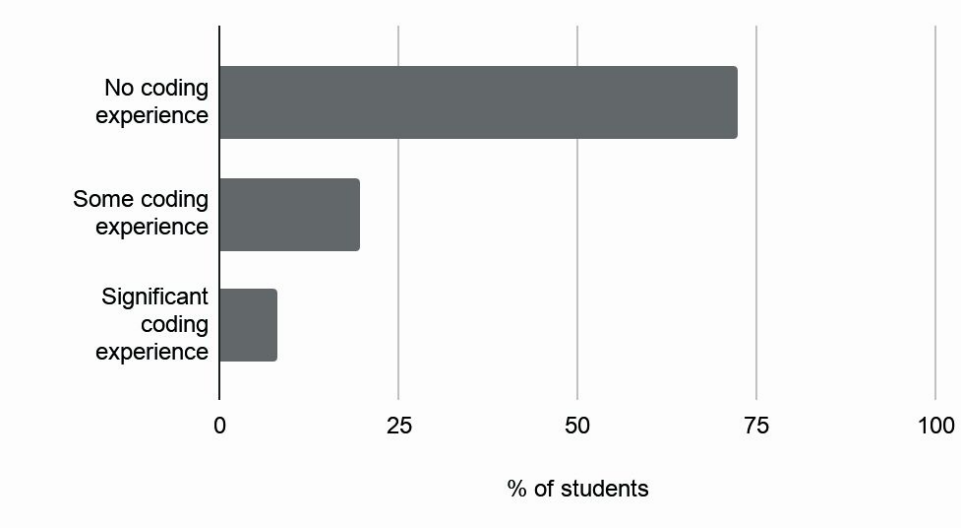
statistics, data analysis & visualization

computational modeling

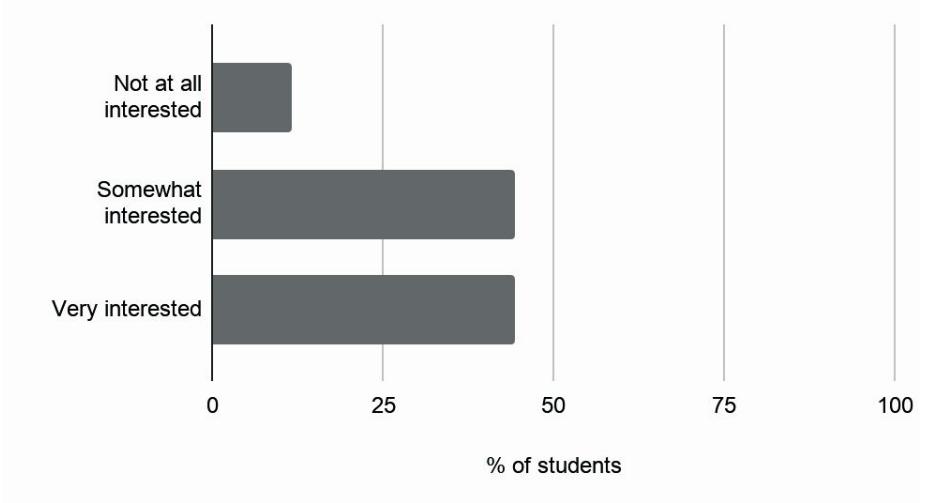

The next generation of biologists needs new skills to adapt to this changing landscape of data.

Students *want* to learn, but have limited opportunities

How much coding experience do you have?



How interested are you in learning how to code?



4 sections of neurobiology lab; 138 students total

See also Zuckerman et al. (2024) Student Motivations and Expectations for an Introductory Programming Course in Biology

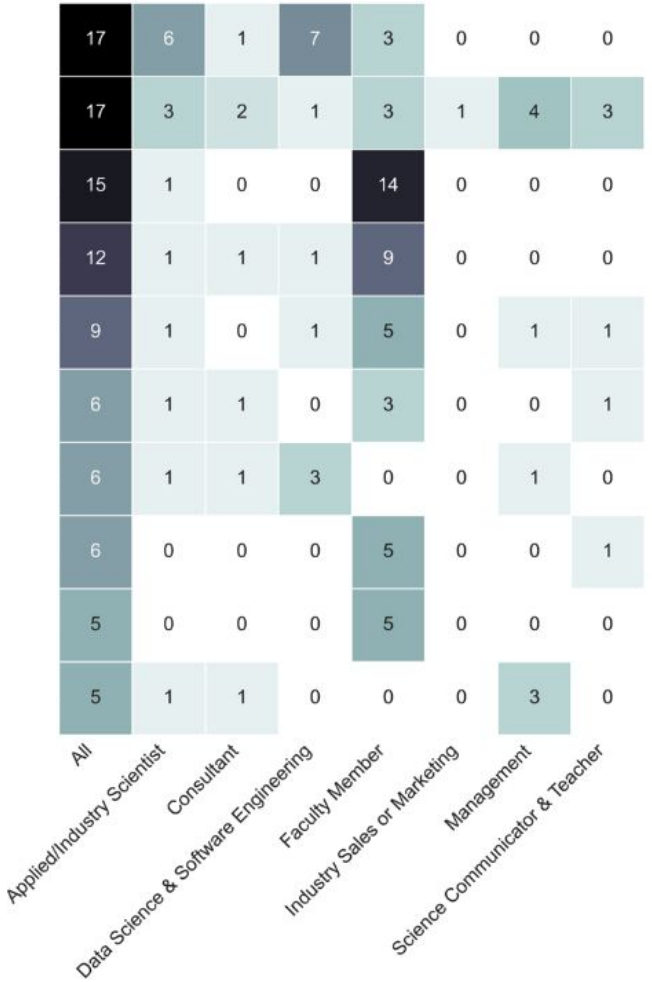
Beyond *research*, programming is a useful skill for individuals with biology & neuroscience degrees

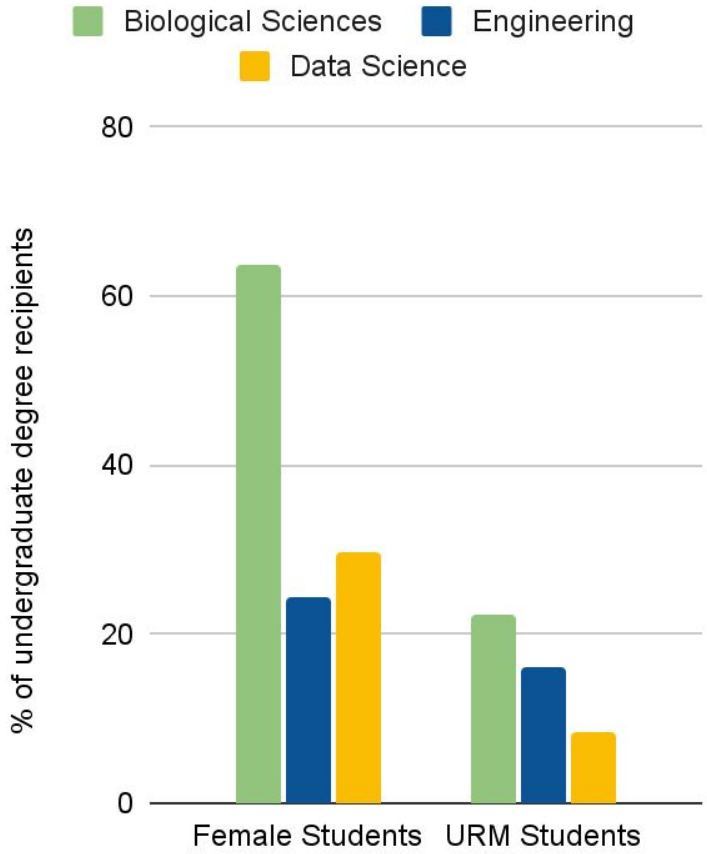
What is one thing that has prepared you for your current position?

(Open ended question in a survey of 125 neuroscience professionals)

Saloni Shah
UCSD Biology Alumna

Shah & Juavinett, *Journal of Undergraduate Neuroscience Education*, 2022



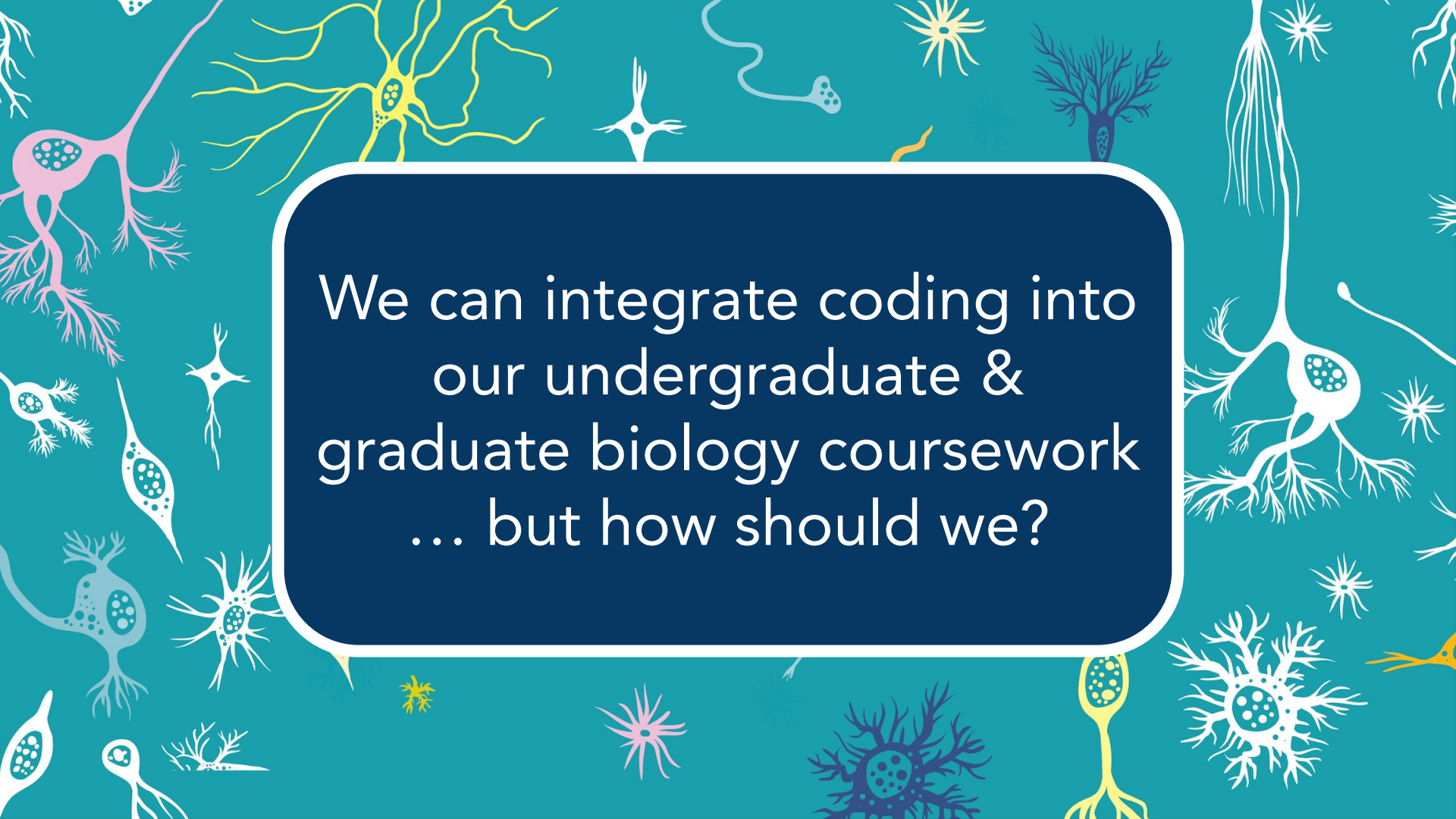


https://ir.ucsd.edu/undergrad/stats-data/ug-degree.html

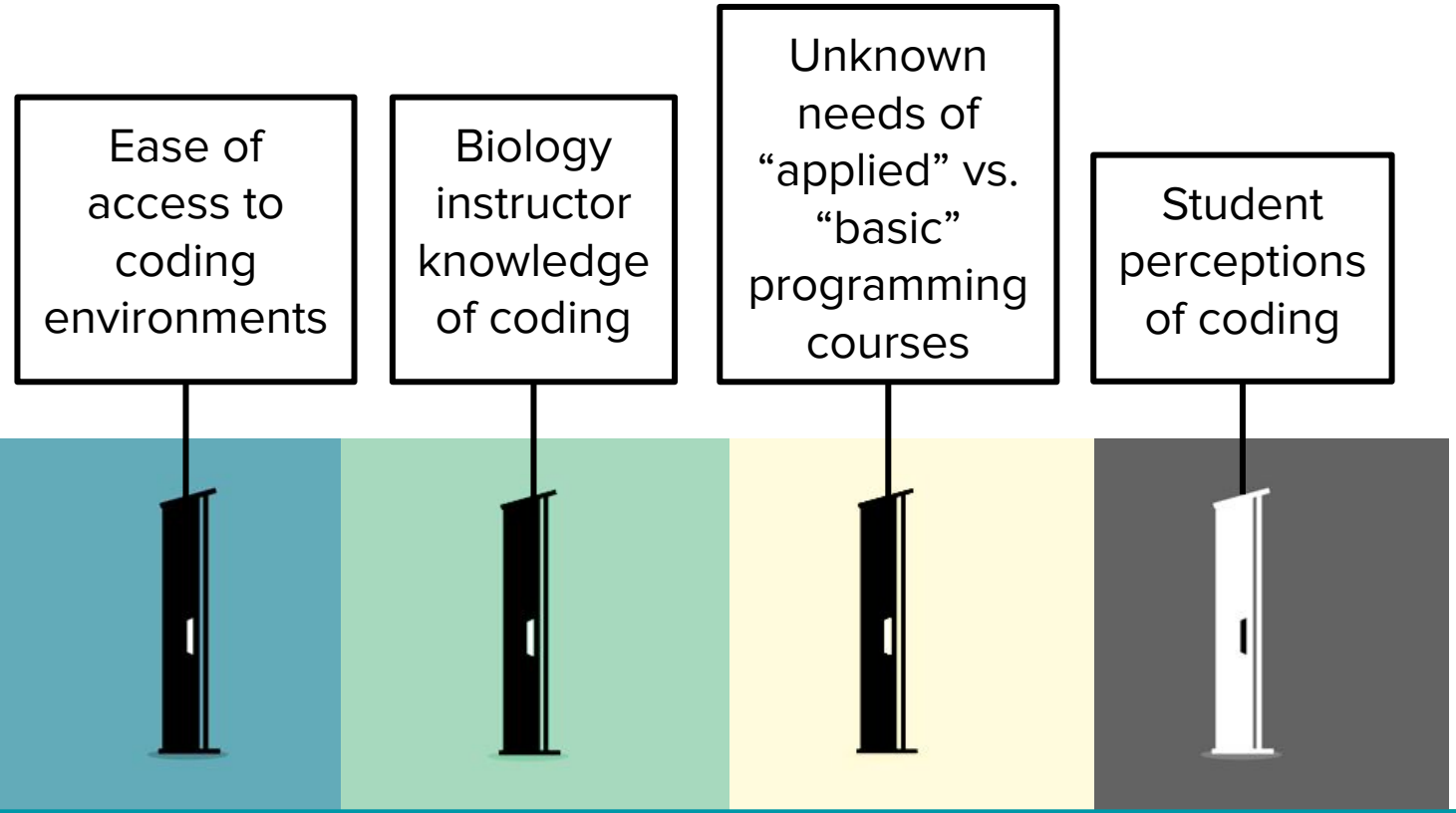
This is also a matter of equity.

More and more jobs require programming experience, and the jobs that do pay more.

(see report by Burning Glass, this US News & World Report article, data in Shah & Juavinett, 2022, and BLS 2024)


We can integrate coding into our undergraduate & graduate biology coursework … but how should we?

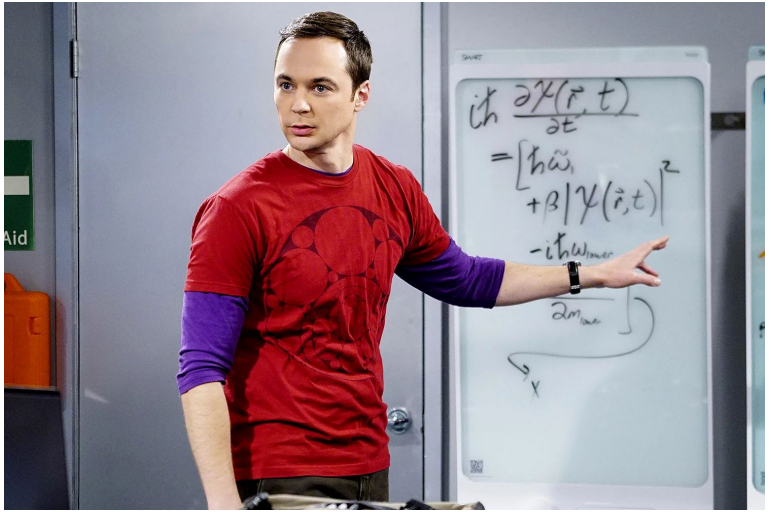
Barriers to bringing coding into our courses



Juavinett, *Neuron,* “The next generation of neuroscientists needs to learn how to code, and we need new ways to teach them” 2022

Student perceptions of coding & computer science (CS)

- **singularly focused** - CS requires an obsession with CS, necessarily at the exclusion of having other interests or meeting personal needs.
- **asocial** - computer scientists do not have social skills and CS requires working in isolation.
- **competitive** - CS courses (and thus potentially the field as a whole) are competitive and CS work is done individually as opposed to in collaboration with others.
- **male** - success in CS requires one to identify as male, or men are innately more talented in CS than women.



Lewis et al., “I Don’t Code All Day”: Fitting in Computer Science When the Stereotypes Don’t Fit (2016)

See also Cheryan et al., 2009, 2011, & 2013; Cvencek et al., 2015; Ensmenger, 2012; Goodenow, 1993; Lee 1998; Margolis & Fisher, 2003; Margolis et al., 2010; Steele, 1997

Neurobiology students start with preconceived notions about coding

I am intimidated because I have some buddies who are computer science majors and share their stresses to me sometimes

I've heard very awful things about computer science courses but it seems interesting especially if the connection to neurobiology is there!

Not very comfortable since I'm not very much a computer person, also **numbers stress me out**

I'm curious about it. I don't know if I'll be a good coder or not.

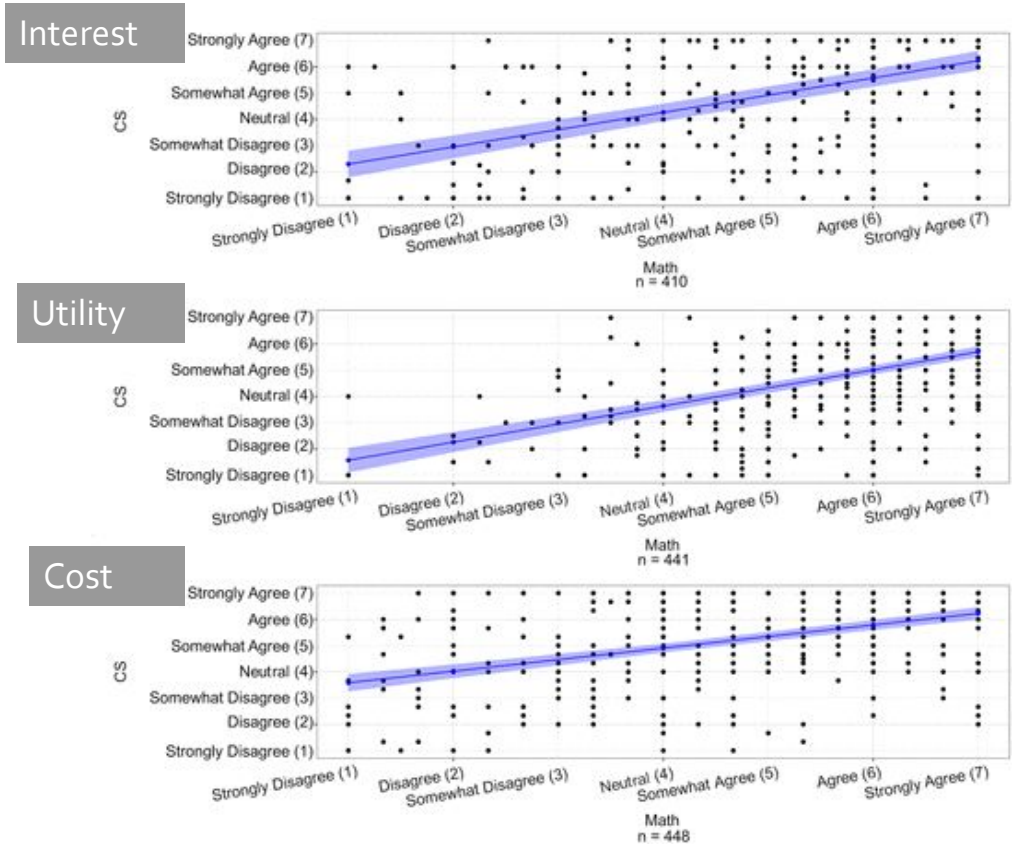
Responses to the question *"How do you feel about learning how to code?"*

Biology students conflate CS with mathematics

In biological contexts, math is more interesting, has more utility, and has less cost than CS

Math & CS task values (interest, utility, and cost) are strongly correlated

(Caughman & Weigel, *CBE Life Sciences,* 2022)



Barriers to bringing coding into our courses

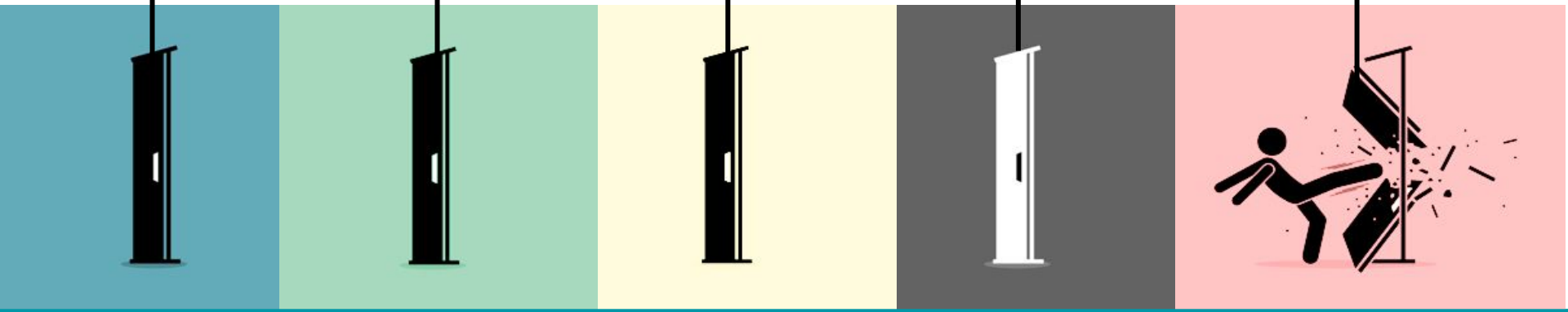
Ease of access to coding environments

Biology instructor knowledge of coding

Unknown needs of “applied” vs. “basic” programming courses

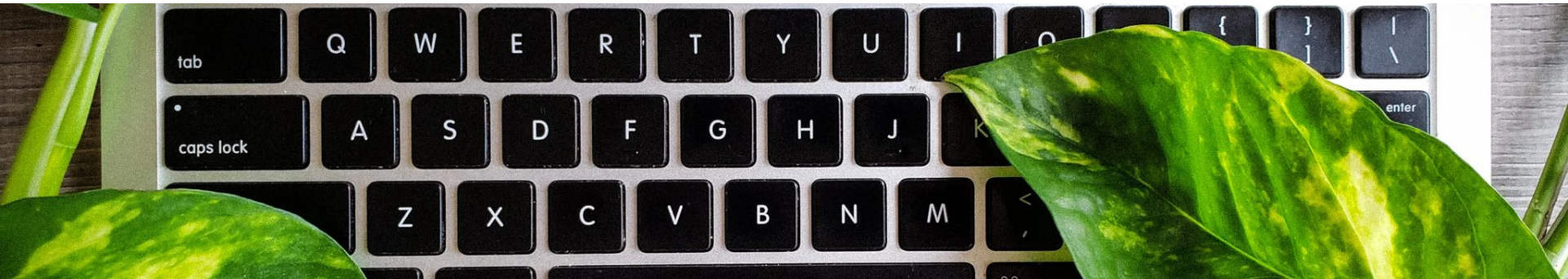
Student perceptions of coding

Accessible, discipline-specific coding resources for all educators



By the end of this workshop, you'll be able to:

- Identify multiple ways of integrating programming into your course
- Access and use cloud-based computing platforms (e.g. Google Colab)
- Write & edit code in computational notebooks



Three possible ways to include coding into the curriculum

Exposure to coding in coursework

- Limited time to devote to coding
- Students *see* utility and *edit* code but do not *write* their own code
- Using code may even be optional

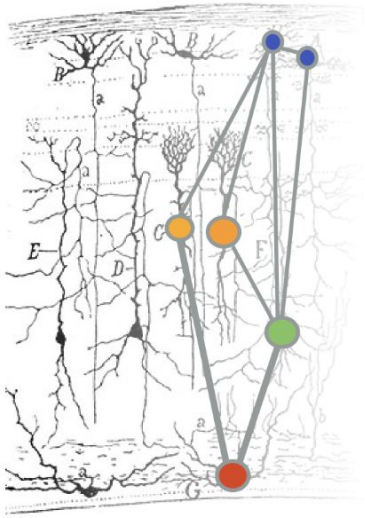
Less independent coding
Lower instructor proficiency

Integration of coding in coursework

- 1-2 hours dedicated to contextualized, introductory programming lessons
- Students edit code and *begin* to write their own
- Lessons exist in standalone modules
- Students can ask their own questions of data
- E.g., Griffith et al. 2024, Juavinett 2020

More independent coding
Higher instructor proficiency

Juavinett, *JUNE*, 2024



BIPN 145 Digital Manual

- Working with LabChart Files
- **Visualizing Data in Python**
  - Introduction to Jupyter Notebooks
  - **Plot scatter**
  - Fit a curved line
  - Plot dot plot
  - Analyze EEG data

Notebooks for a neurobiology lab class

Hosted & available at bipn145.github.io

Import data they've recorded from our physiology equipment

Learn how to use computational notebooks

Use their measured values to understand correlation & linear regression

Fit curves to data we've collected

Plot a prism-style plot & implement two-sample statistics

... ask me how I built this JupyterBook!

Notebooks are designed to expose students to coding, with straightforward tasks to complete

> **Task**: Choose two different Cre lines to compare, and assign them to the variables below by replacing the `...`. The value of your variable needs to be a **string** -- in other words, it should have quotes around it. The cell will print how many cells are in your datasets. If you have less than 10 cells, consider choosing a different Cre-line.

> You can find information on the different cre-lines that are available in this glossary or on the Allen Institute's website.

> **Note**: Be sure that you are using the *entire* name of the Cre line -- that means *everything* within the single quotes above.

In [ ]:
```
cre_line_1 = ...
cre_line_2 = ...

cre_line_1_df = mouse_ephys_df[mouse_ephys_df['transgenic_line']==cre_line_1]
cre_line_2_df = mouse_ephys_df[mouse_ephys_df['transgenic_line']==cre_line_2]

print(cre_line_1 + ' has ' + str(len(cre_line_1_df)) + ' cells')
print(cre_line_2 + ' has ' + str(len(cre_line_2_df)) + ' cells')
```

Three possible ways to include coding into the curriculum

Exposure to coding in coursework

- Limited time to devote to coding
- Students *see* utility and *edit* code but do not *write* their own code
- Using code may even be optional

Contextualized (discipline-based) coding classes

- An entire class dedicated to teaching coding in the context of neuroscience
- Students design & write their own code
- May include novel research projects

Less independent coding
Lower instructor proficiency

Integration of coding in coursework

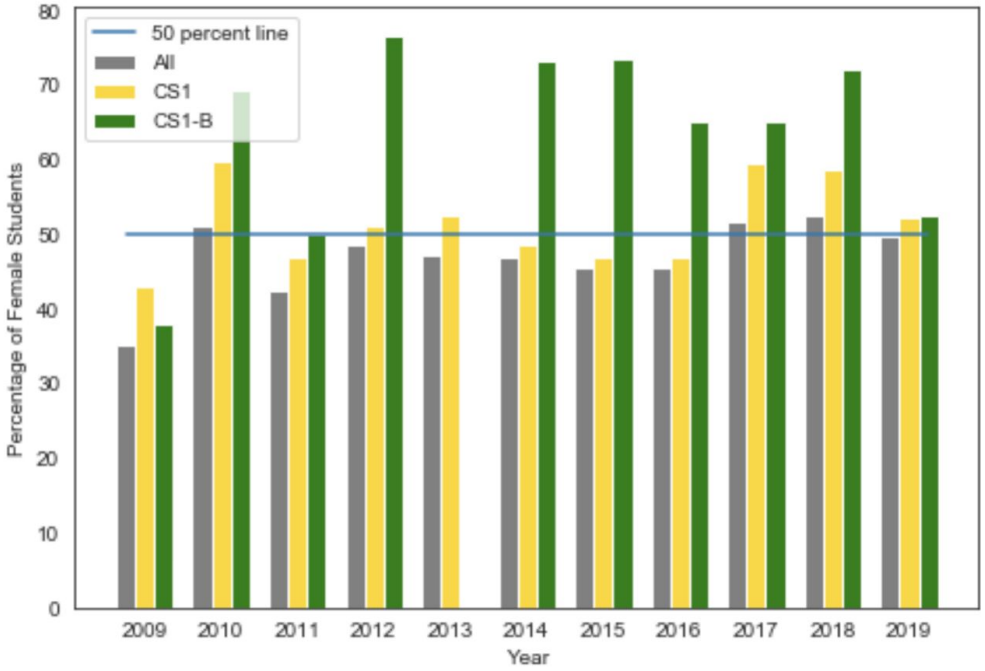
- 1-2 hours dedicated to contextualized, introductory programming lessons
- Students edit code and *begin* to write their own
- Lessons exist in standalone modules
- Students can ask their own questions of data

More independent coding
Higher instructor proficiency

Juavinett, *JUNE*, 2024

Contextualized coding invites in more female students

Figure 2: Female Students: % in CS1 and CS1-B, per year

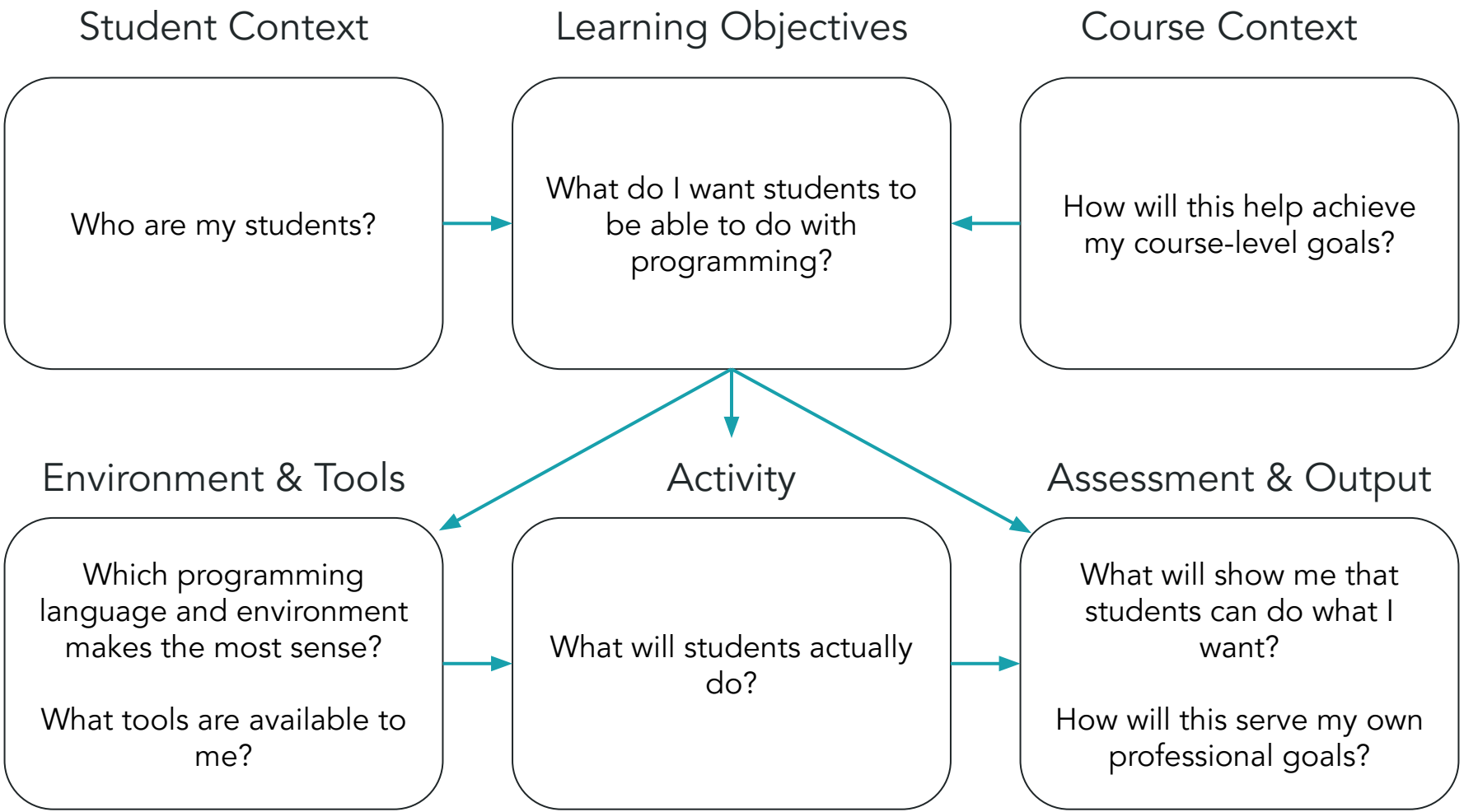


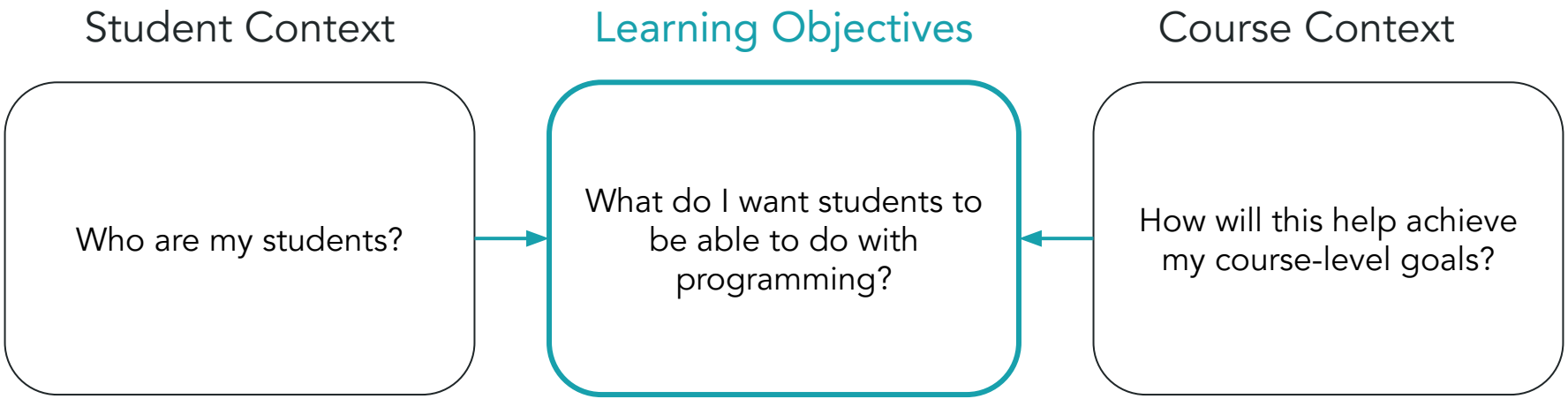
learn how to code while looking at

> genes
> cells
> brains

Introduction to Python for Biologists (BILD 62)
Spring 2022, T/Th@2pm

Dodds et al., *SIGCSE* 2021
"A Biology-based CS1: Results and Reflections, Ten Years in"
(results from this study less clear for first-generation or URM students)


Student Context
Learning Objectives
Course Context
Who are my students?
What do I want students to be able to do with programming?
How will this help achieve my course-level goals?
Environment & Tools
Activity
Assessment & Output
Which programming language and environment makes the most sense?
What tools are available to me?
What will students actually do?
What will show me that students can do what I want?
How will this serve my own professional goals?


Student Context
Learning Objectives
Course Context
Who are my students?
What do I want students to be able to do with programming?
How will this help achieve my course-level goals?

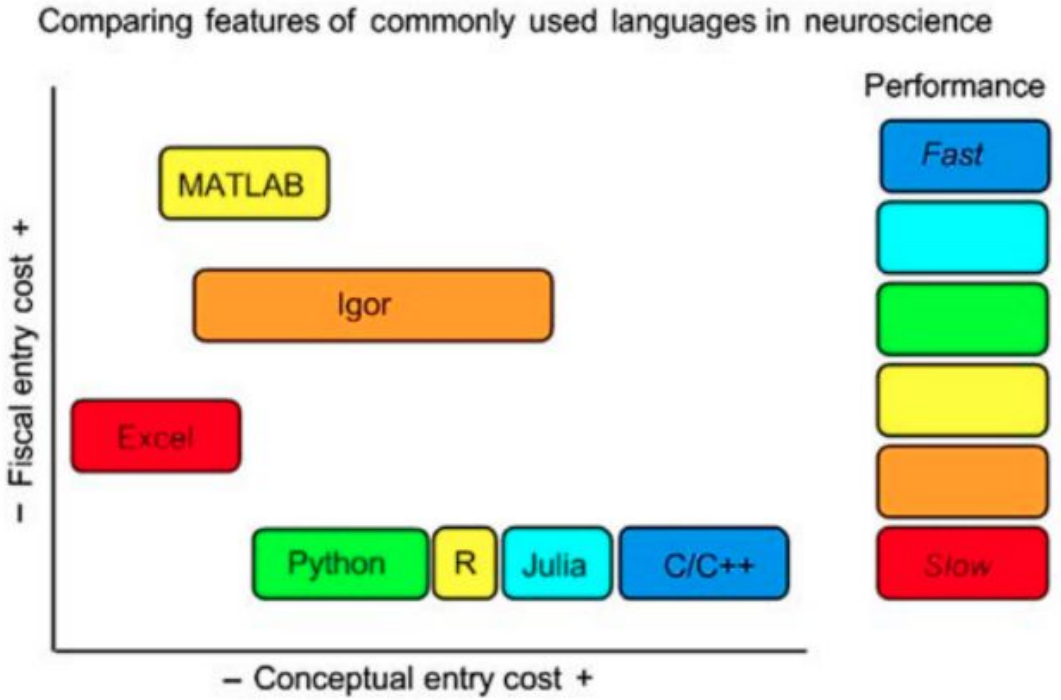
PART II

The Tools

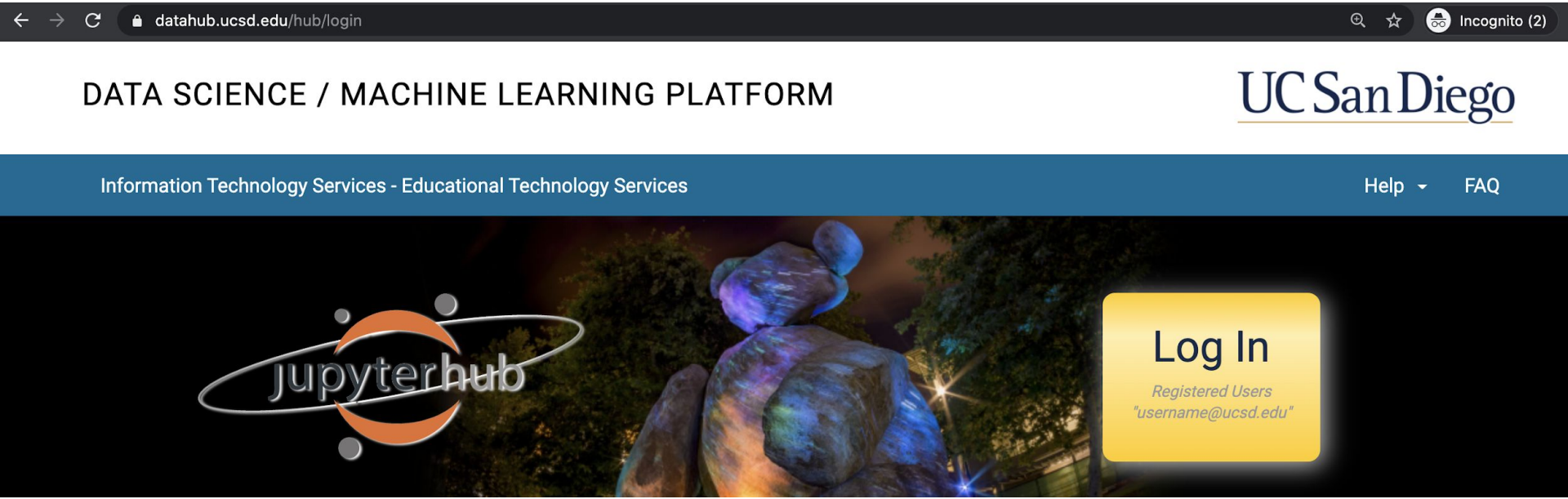
Considerations for choosing a programming language

- Fiscal & conceptual entry (see right)
- Usage in your field
- Ease of use
- *My recommendation:* Python or R

Shiny apps!
(Lauren Bucker's talk)



From Wallisch (2017)



UCSD has a data science online platform designed for exactly this, but Google Colab works well for educators at institutions without a Jupyter Hub

Considerations when choosing tools & a teaching approach

- How much do you want students to write code *de novo* versus editing provided code
  - Carpentries Model: live coding on a blank page

https://carpentries.org/

- Do you want students to be dependent on (sometimes expensive) school or course-provided software?
- Do students have their own laptops (or do they have iPads)?
- Do students have access to a computer with a decent internet connection?

Should you use a notebook environment?

Pros

- Easy accessible to anyone with an internet connection (via Colab)
- You can set up notebook template for students with markdown documentation
- Scientists use them!

Cons

- No “workspace” for variables
- Non-linear work is less intuitive
- Students may perceive it as inauthentic (Zuckerman & Juavinett, 2024)

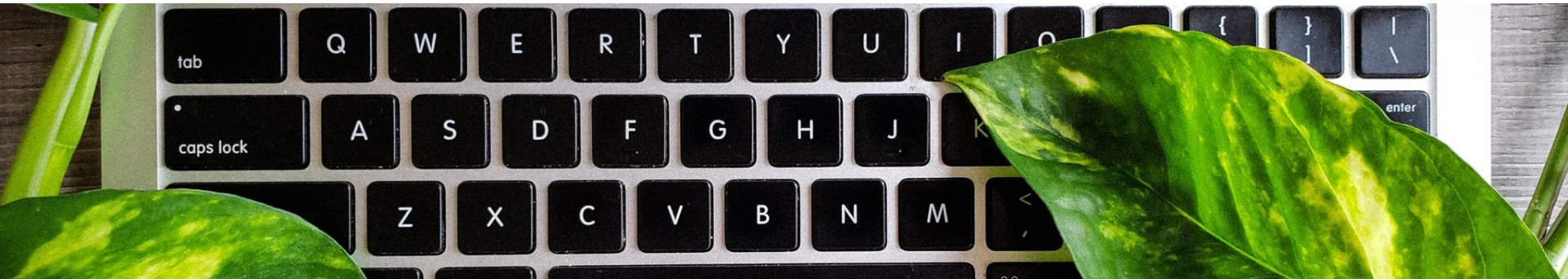
Tools for today

Google Colab, a cloud-based computing *notebook*

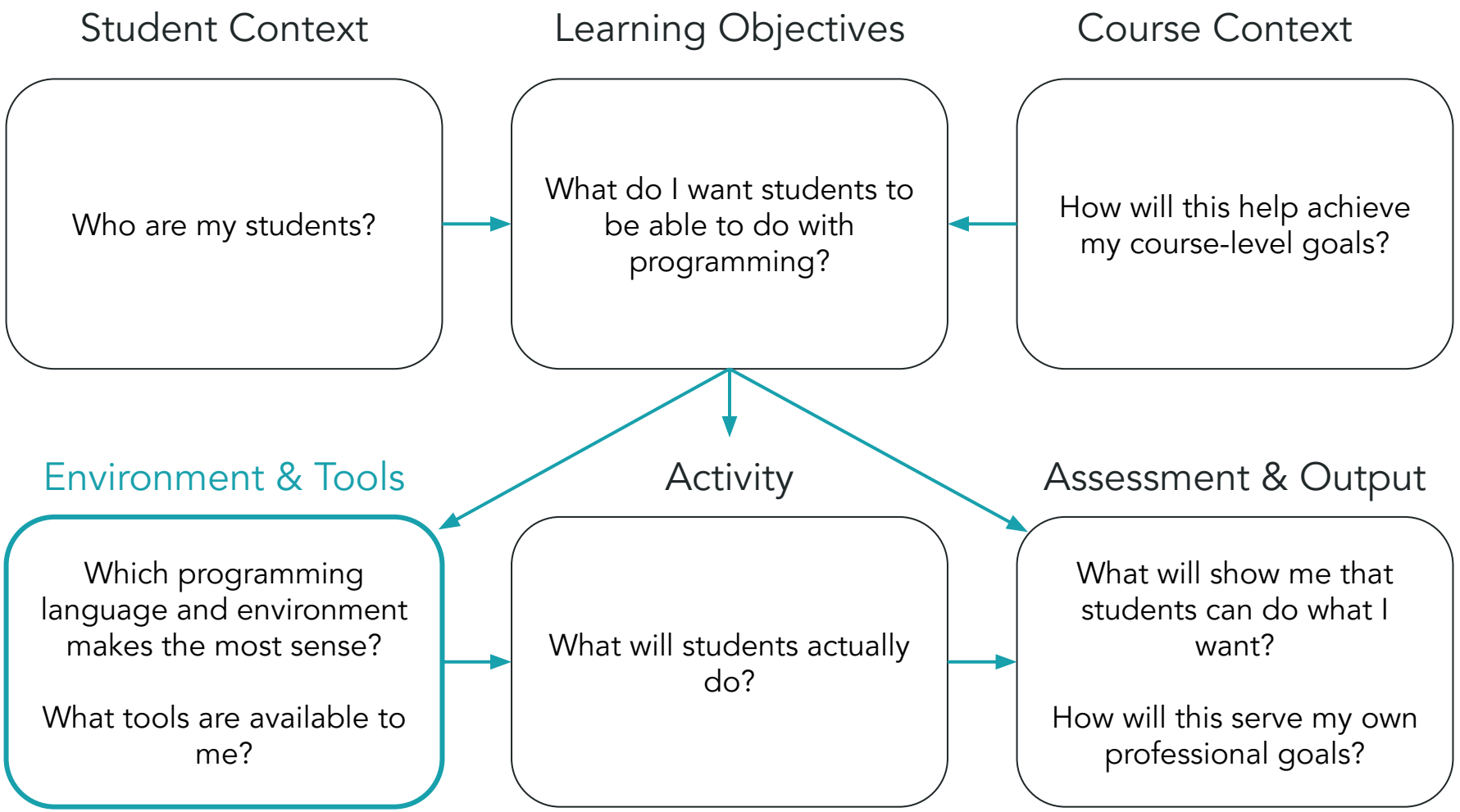
- Very similar to Jupyter notebooks, with small differences.
- Conveniently saves & launches from Google Drive or with a button in GitHub

By the end of this workshop, you'll be able to:

- Identify multiple ways of integrating programming into your course
- Access and use cloud-based computing platforms (e.g. Google Colab)
- Write & edit code in computational notebooks



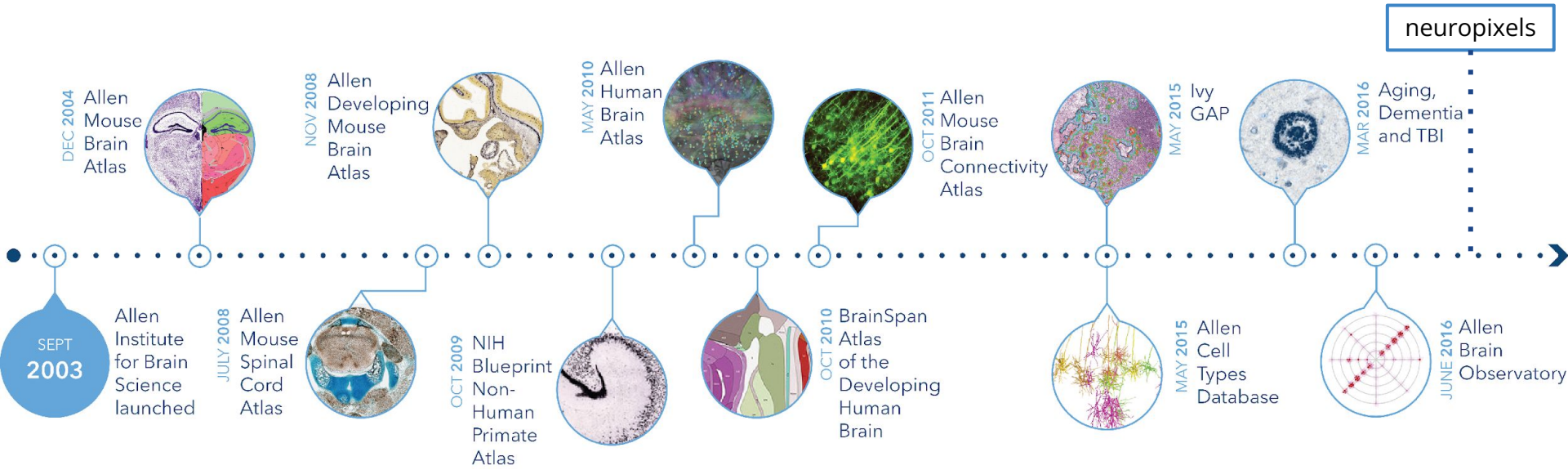

Let’s open up our first notebook from
http://www.github.com/ajuavinett/Allen2025


Student Context
Learning Objectives
Course Context
Who are my students?
What do I want students to be able to do with programming?
How will this help achieve my course-level goals?
Environment & Tools
Activity
Assessment & Output
Which programming language and environment makes the most sense?
What tools are available to me?
What will students actually do?
What will show me that students can do what I want?
How will this serve my own professional goals?

PART III

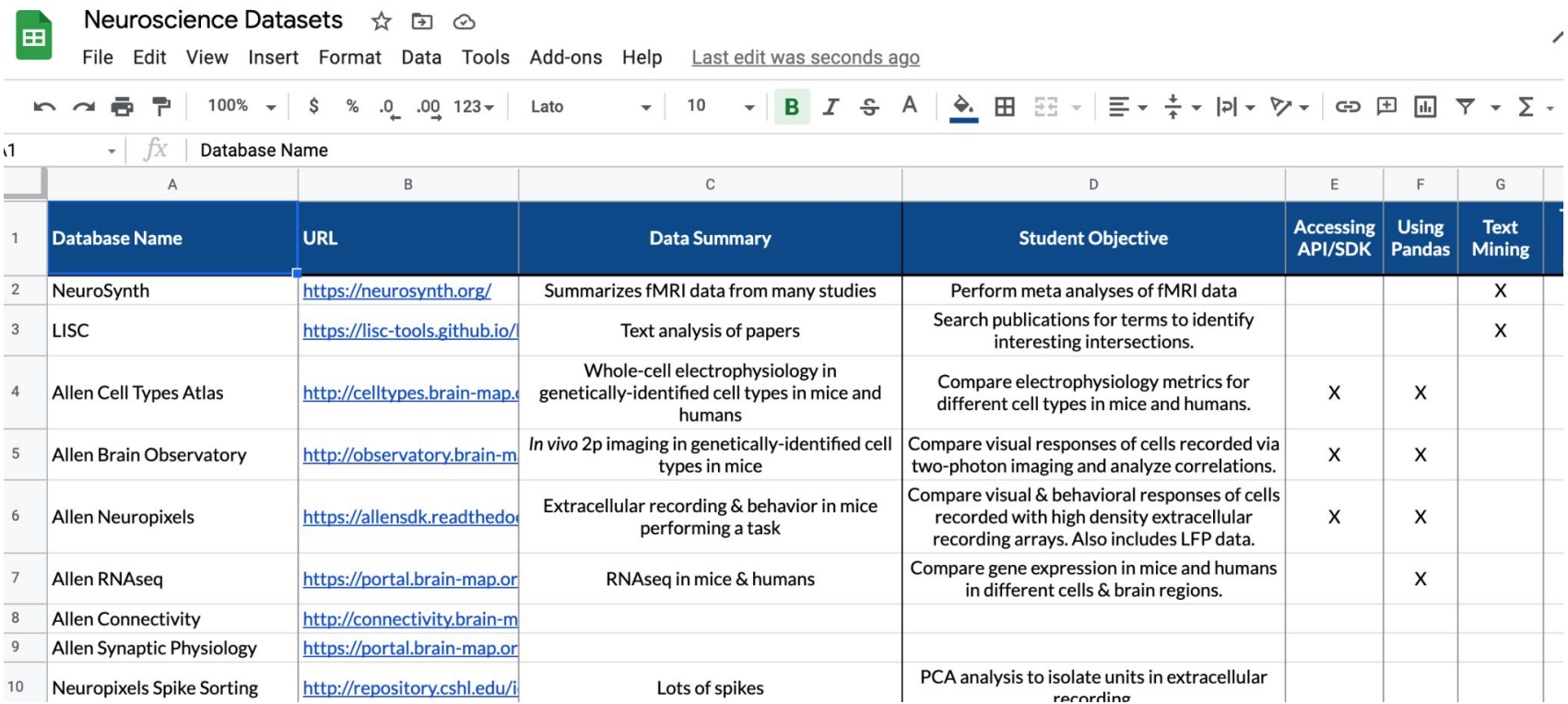
Cell Types Lesson

Datasets are increasingly being shared online



NEURODATA WITHOUT BORDERS

DANDI
dandiarchive.org



Neuroscience Datasets

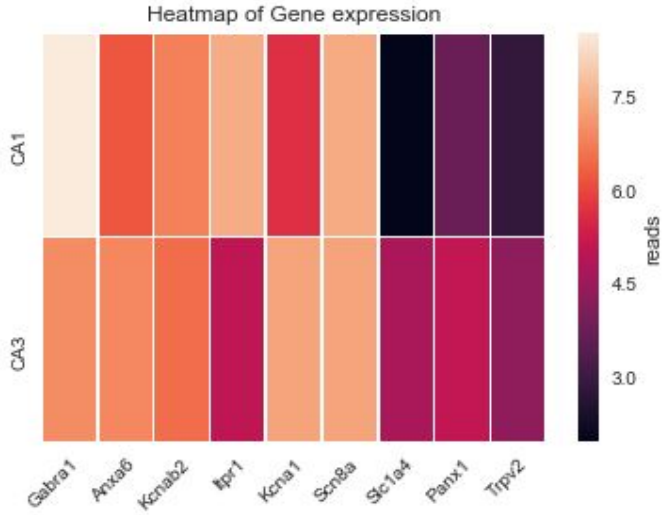
File Edit View Insert Format Data Tools Add-ons Help Last edit was seconds ago

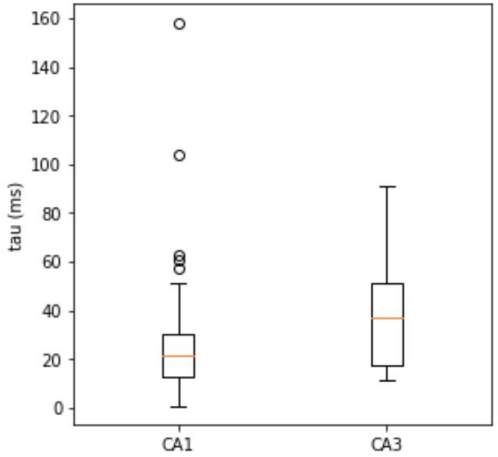
Database Name

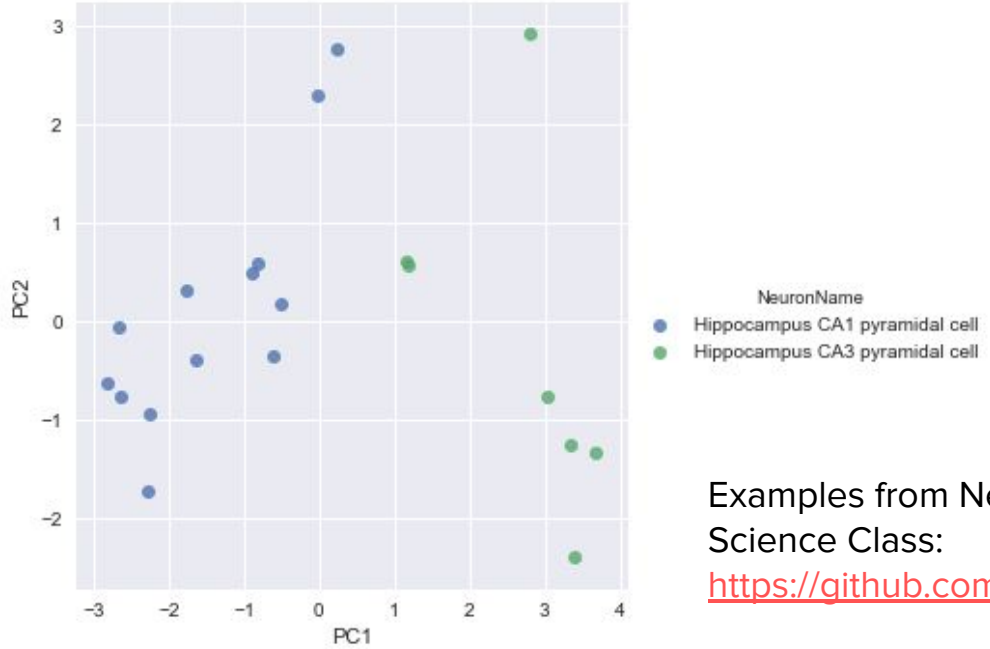
	A	B	C	D	E	F	G
1	**Database Name**	**URL**	**Data Summary**	**Student Objective**	**Accessing API/SDK**	**Using Pandas**	**Text Mining**
2	NeuroSynth	https://neurosynth.org/	Summarizes fMRI data from many studies	Perform meta analyses of fMRI data			X
3	LISC	https://lisc-tools.github.io/l	Text analysis of papers	Search publications for terms to identify interesting intersections.			X
4	Allen Cell Types Atlas	http://celltypes.brain-map.(	Whole-cell electrophysiology in genetically-identified cell types in mice and humans	Compare electrophysiology metrics for different cell types in mice and humans.	X	X	
5	Allen Brain Observatory	http://observatory.brain-m	*In vivo* 2p imaging in genetically-identified cell types in mice	Compare visual responses of cells recorded via two-photon imaging and analyze correlations.	X	X	
6	Allen Neuropixels	https://allensdk.readthedo(	Extracellular recording & behavior in mice performing a task	Compare visual & behavioral responses of cells recorded with high density extracellular recording arrays. Also includes LFP data.	X	X	
7	Allen RNAseq	https://portal.brain-map.or	RNAseq in mice & humans	Compare gene expression in mice and humans in different cells & brain regions.		X	
8	Allen Connectivity	http://connectivity.brain-m					
9	Allen Synaptic Physiology	https://portal.brain-map.or					
10	Neuropixels Spike Sorting	http://repository.cshl.edu/i	Lots of spikes	PCA analysis to isolate units in extracellular recording			

Open Neuroscience Datasets https://bit.ly/openneurodatasets

Students perform *real*, novel research





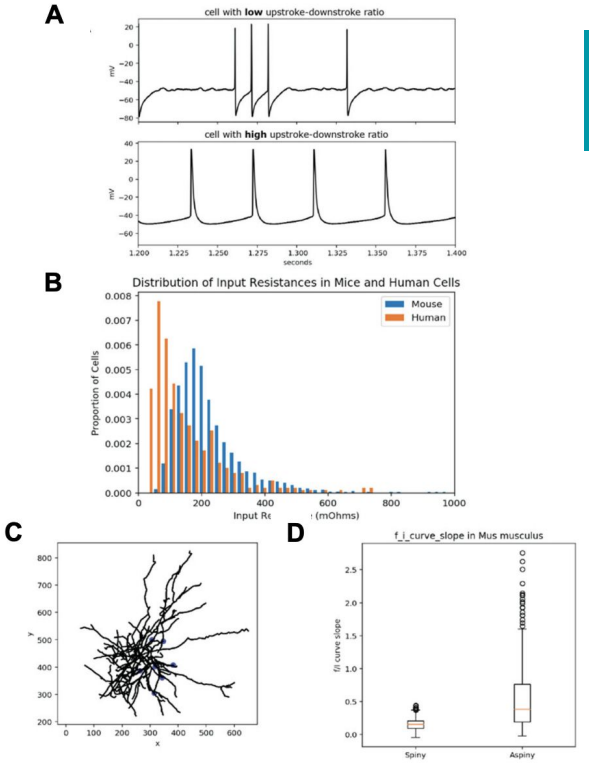


Examples from Neural Data Science Class: https://github.com/BIPN162

Cell Types Lesson Learning Objectives

- Relate the diverse intrinsic electrophysiological features of neurons to their structure and function
- Compare electrophysiological characteristics of neurons in humans and mice
- Practice using Jupyter Notebooks to run and edit Python code
- Develop a sense of belonging and self-efficacy in coding and neuroscience research

More information & resources:
https://sites.google.com/ucsd.edu/neuroedu/cell-types



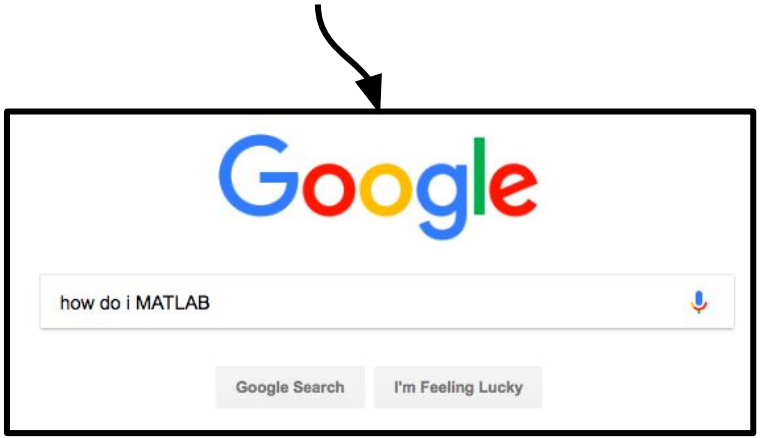
Juavinett, "Learning to code while analyzing an open access dataset," *JUNE*, 2020

PART IV

Wrap-Up & Promising Practices

A general framework for teaching coding to neuroscience students

- Live code in front of students
- Encourage pair programming
- Normalize errors and searching for answers



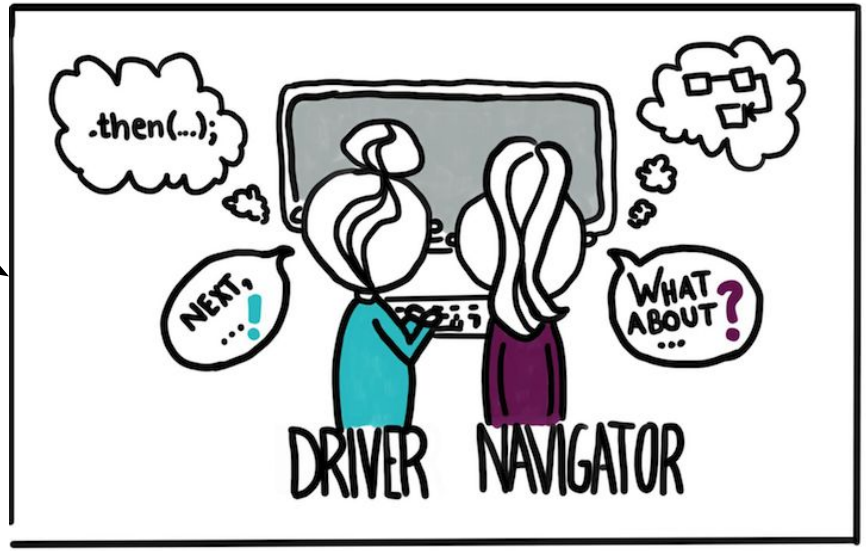


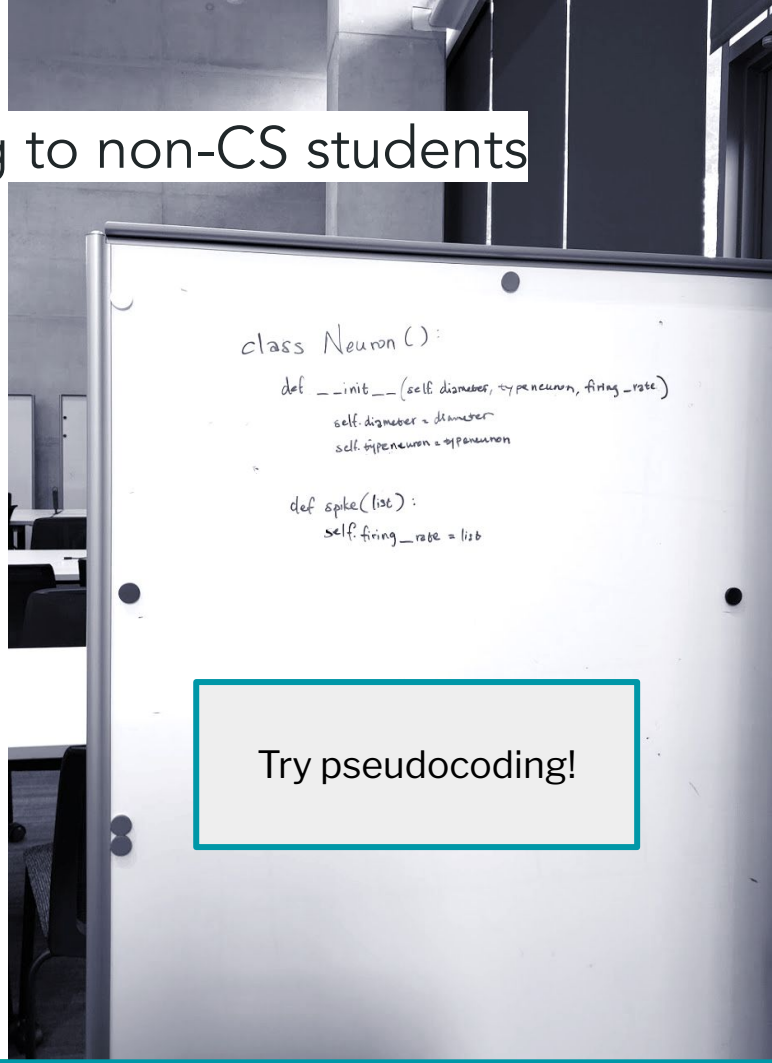
Image Source

More thoughts on this: Ten quick tips for teaching programming

A general framework for teaching coding to non-CS students

- Frame coding as a **skill** — *it's like learning a language*
- Encourage **growth mindsets** around coding
  - Students with growth mindsets perform better over time
  - Beliefs about whether computational abilities are fixed or malleable impact: sense of belonging, how we respond to difficulties, and ultimately, achievement

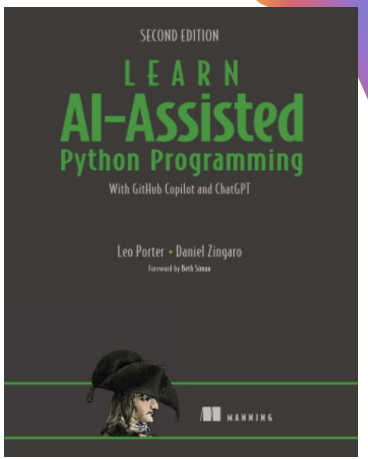
More thoughts on this: Medium | There is No Such Thing as a Computational Person



Oh yeah, what about AI assistants?

You (and your students) can and *should* use it to learn!

(…how you integrate it into your class depends on your learning objectives!)



Learn AI-Assisted Python Programming (Second Edition)

by Leo Porter and Daniel Zingaro

Extensions of what you've seen here

- Other sample notebooks
- This JUNE publication by Ho et al. (2021)
- Your work?

The Journal of Undergraduate Neuroscience Education (JUNE), Fall 2021, 20(1):A100-A110

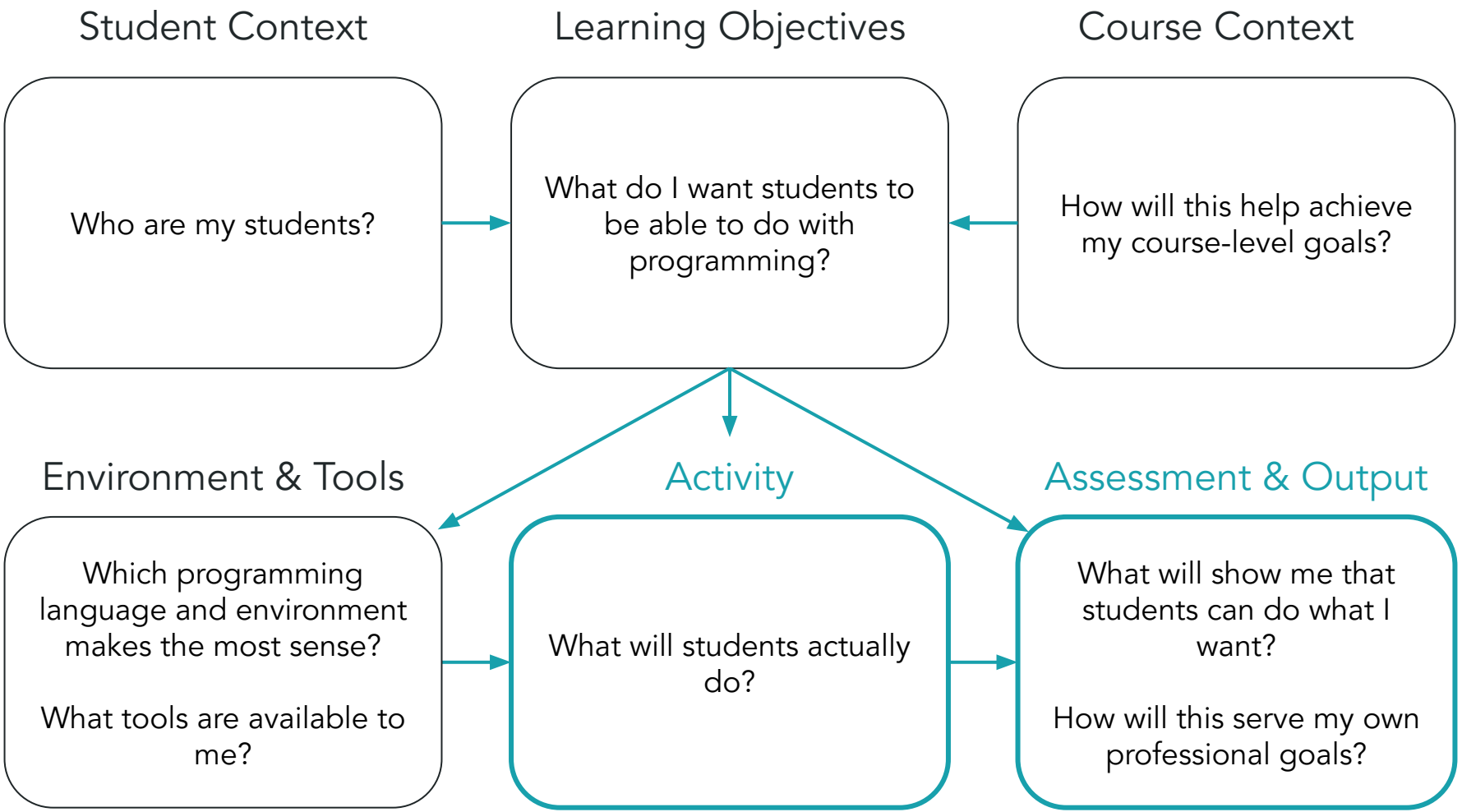
ARTICLE

Pandemic Teaching: Using the Allen Cell Types Database for Final Semester Projects in an Undergraduate Neurophysiology Lab Course

Yi-Yun Ho[1], Andrea Roeser[1], Gwenda Law[2], and Bruce R. Johnson[1]

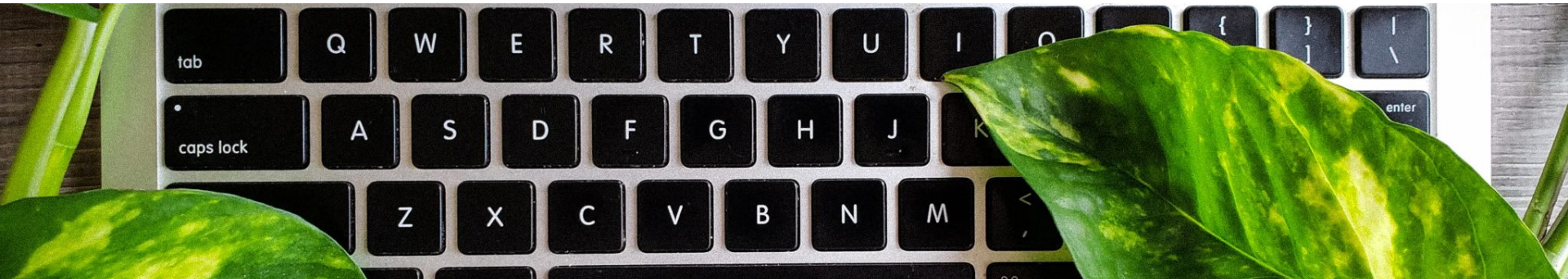
[1]Department of Neurobiology and Behavior, Cornell University, Ithaca, NY 14853; [2]Department of Biomedical Engineering, Cornell University, Ithaca, NY 14853.

We designed a final semester research project that allowed students to apply the electrophysiological concepts they learned in a lab course to propose and answer experimental questions without access to laboratory equipment. We created the activity based on lesson plans from Ashley Juavinett and the Allen Institute for Brain Science (AIBS) Allen SDK online examples. An interactive graphic interface final semester project allowed students to ask real-world medical and scientific questions from "start to end". Through this project, students developed skills to navigate an extensive online database and gained experience with coding-based data analysis. They chose neuronal populations from human and mouse brains to compare passive properties and neuronal excitability


Student Context
Learning Objectives
Course Context
Who are my students?
What do I want students to be able to do with programming?
How will this help achieve my course-level goals?
Environment & Tools
Activity
Assessment & Output
Which programming language and environment makes the most sense?
What tools are available to me?
What will students actually do?
What will show me that students can do what I want?
How will this serve my own professional goals?

By the end of this workshop, you'll be able to:

- Identify multiple ways of integrating programming into your course
- Access and use cloud-based computing platforms (e.g. Google Colab)
- Write & edit code in computational notebooks



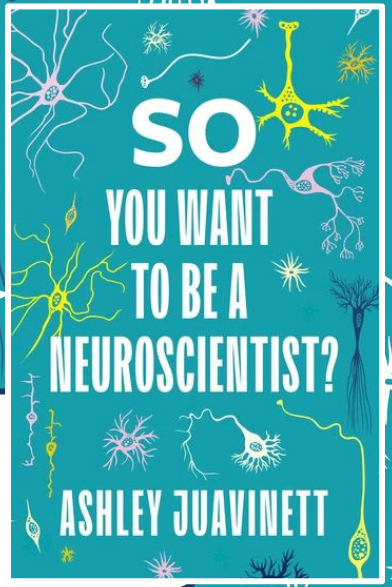
Post Workshop Survey
https://forms.gle/oeZLjdSzxY9xGwoY6

THE KAVLI FOUNDATION

Support for online neural data science textbook: nwb4edu.github.io

Victor Magdaleno-Garcia, mentee & research intern

ajuavine@ucsd.edu
@analog_ashley



ALLEN INSTITUTE for BRAIN SCIENCE

Support for using & sharing datasets

Austin Zuckerman, PhD
Research on computing attitudes in biology students