Jupyter Notebooks as a Teaching Tool in Neuroscience Courses

Ashley Juavinett, Ph.D.
Assistant Teaching Professor
UC San Diego, Division of Biological Sciences

UCSD

Goals for this workshop

- Provide a general framework on teaching coding to neuroscience students
- Introduce Jupyter Notebooks as a tool for teaching
- Demonstrate the use of the Jupyter Notebooks

While we're warming up, let's prepare for coding together.

https://bit.ly/FUNSVMjupyter

That link will send you to:
https://mybinder.org/v2/gh/ajuavinett/FUN_SVM_2020.git/master?filepath=Tutorial.ipynb

Alternate link to Colaboratory (click on Tutorial.ipynb):
http://colab.research.google.com/github/ajuavinett/FUN_SVM_2020

Where I’m coming from…

Postdoc: More MATLAB

Lafayette College (coding class optional — I didn’t take it)

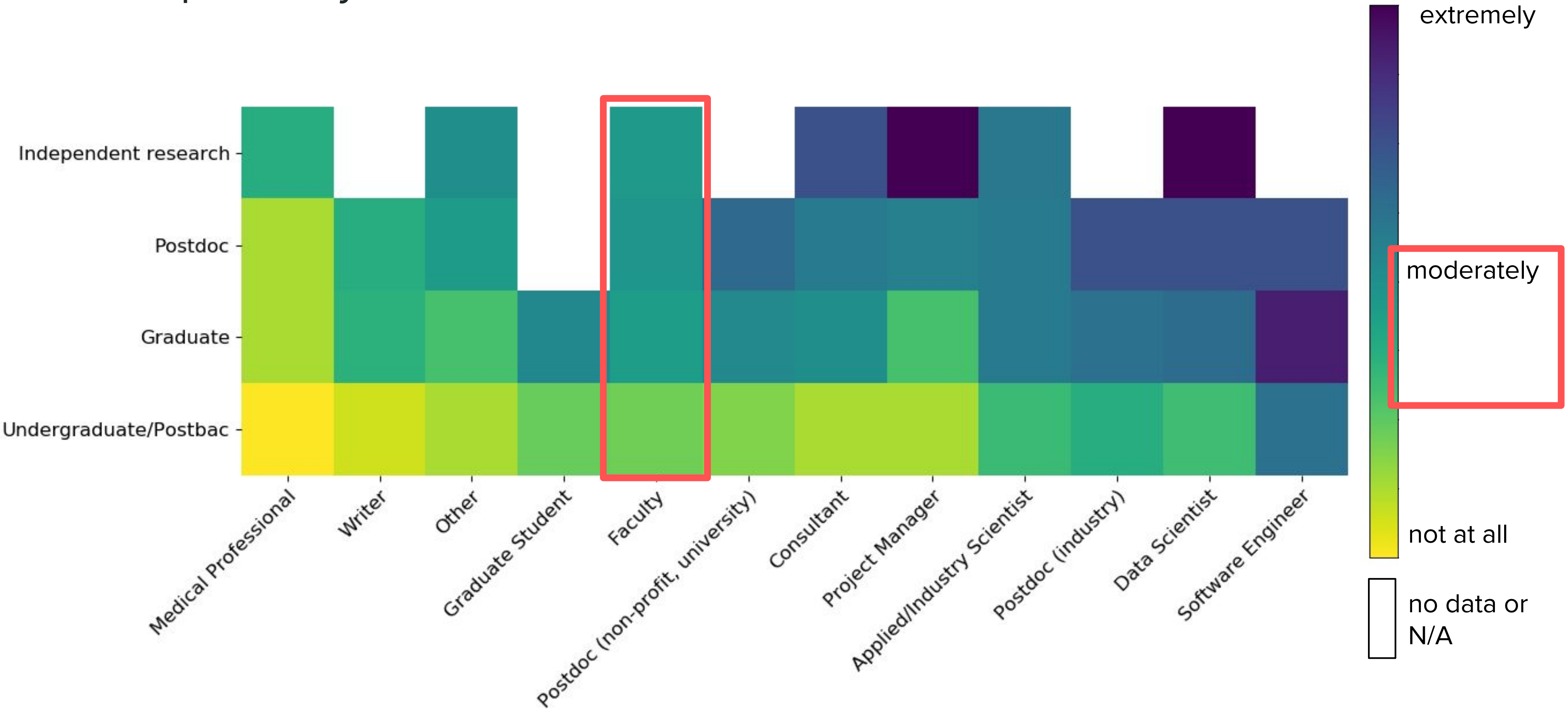
PhD: “learned how to code” in MATLAB, dabbled in Python

Faculty: When I learned Python 😬

I’ve *never* taken a formal coding class!

Even if you haven’t, you can teach students how to code.

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

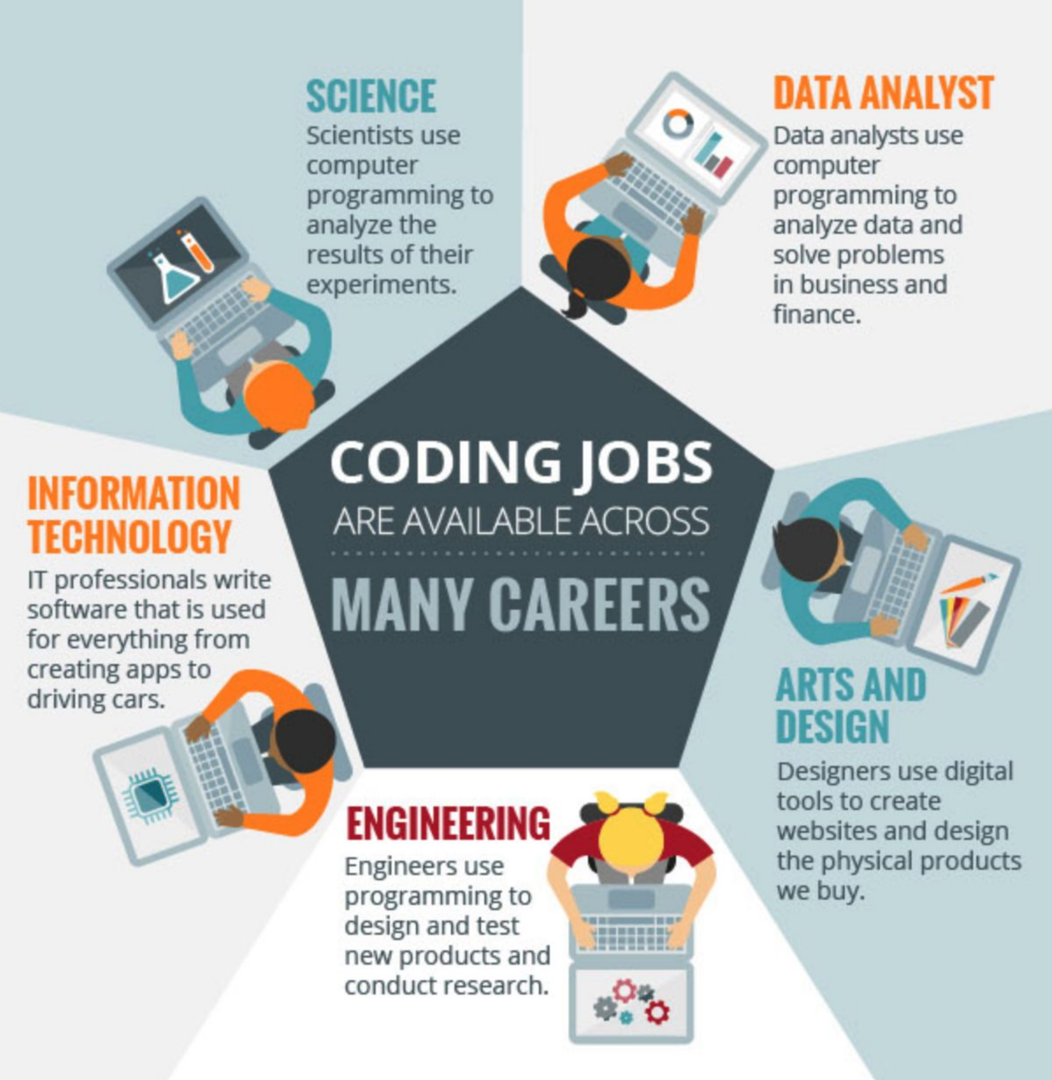


Do you teach or demonstrate programming in your courses?

Respond with yes or no

Why do neuroscience students need to 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)



CODING JOBS ARE AVAILABLE ACROSS **MANY CAREERS**

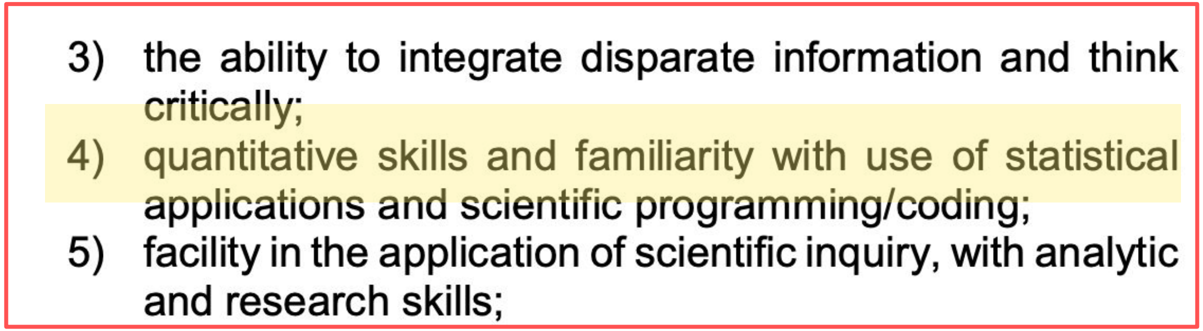
SCIENCE
Scientists use computer programming to analyze the results of their experiments.

DATA ANALYST
Data analysts use computer programming to analyze data and solve problems in business and finance.

ARTS AND DESIGN
Designers use digital tools to create websites and design the physical products we buy.

ENGINEERING
Engineers use programming to design and test new products and conduct research.

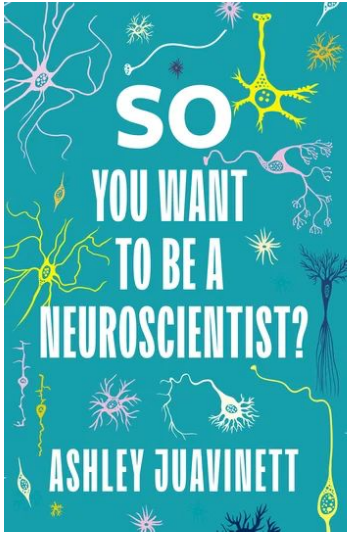
INFORMATION TECHNOLOGY
IT professionals write software that is used for everything from creating apps to driving cars.



3) the ability to integrate disparate information and think critically;
4) quantitative skills and familiarity with use of statistical applications and scientific programming/coding;
5) facility in the application of scientific inquiry, with analytic and research skills;

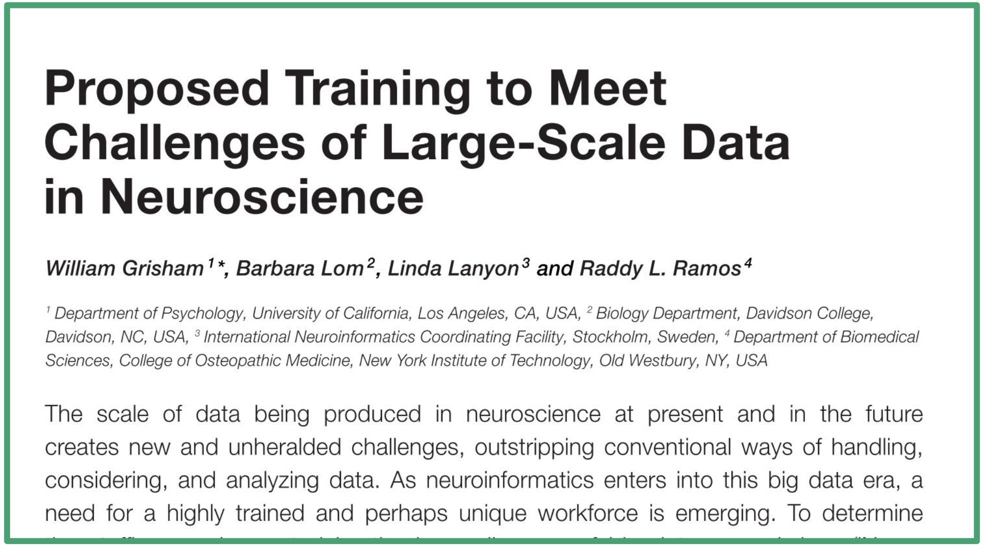
“The New Blueprints: Undergraduate Neuroscience Education in the Twenty-First Century” (Wierterlak et al., *JUNE, 2018*)

(Grisham et al., *Frontiers in Neuroinformatics*, 2016)



See also Akil et al., “Neuroscience Training for the 21st Century,” *Neuron* (2016)

And forthcoming *So you want to be a neuroscientist?* (Columbia University Press)



Proposed Training to Meet Challenges of Large-Scale Data in Neuroscience

William Grisham [1], Barbara Lom [2], Linda Lanyon [3] and Raddy L. Ramos [4]*

[1] Department of Psychology, University of California, Los Angeles, CA, USA, [2] Biology Department, Davidson College, Davidson, NC, USA, [3] International Neuroinformatics Coordinating Facility, Stockholm, Sweden, [4] Department of Biomedical Sciences, College of Osteopathic Medicine, New York Institute of Technology, Old Westbury, NY, USA

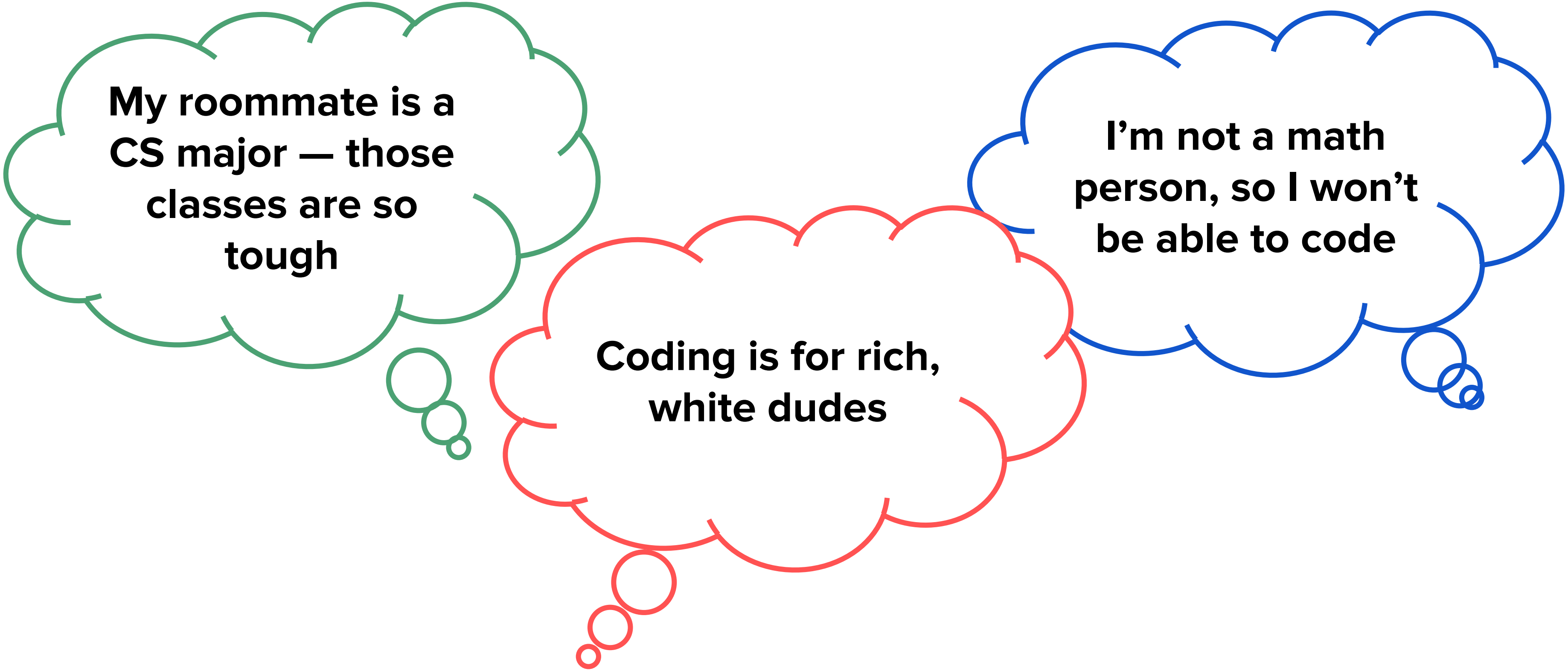
The scale of data being produced in neuroscience at present and in the future creates new and unheralded challenges, outstripping conventional ways of handling, considering, and analyzing data. As neuroinformatics enters into this big data era, a need for a highly trained and perhaps unique workforce is emerging. To determine

Yet, few neuroscience programs require coding, and only some offer it as an elective

- In a study of undergraduate neuroscience majors, only **six** of 118 institutions required coding, and **nine** included coding as an elective (Pinard-Welyczko et al., *JUNE*, 2017)
  - In a smaller study of 22 programs, 28% require programming as a coursework, 50% offer it as an elective (SfN, 2016)
- 15% of neuroscience PhD programs require programming, and 55% of including programming as an elective (SfN, 2016)

We can integrate coding into neuroscience courses *(especially remote laboratory courses)*

How students may perceive coding

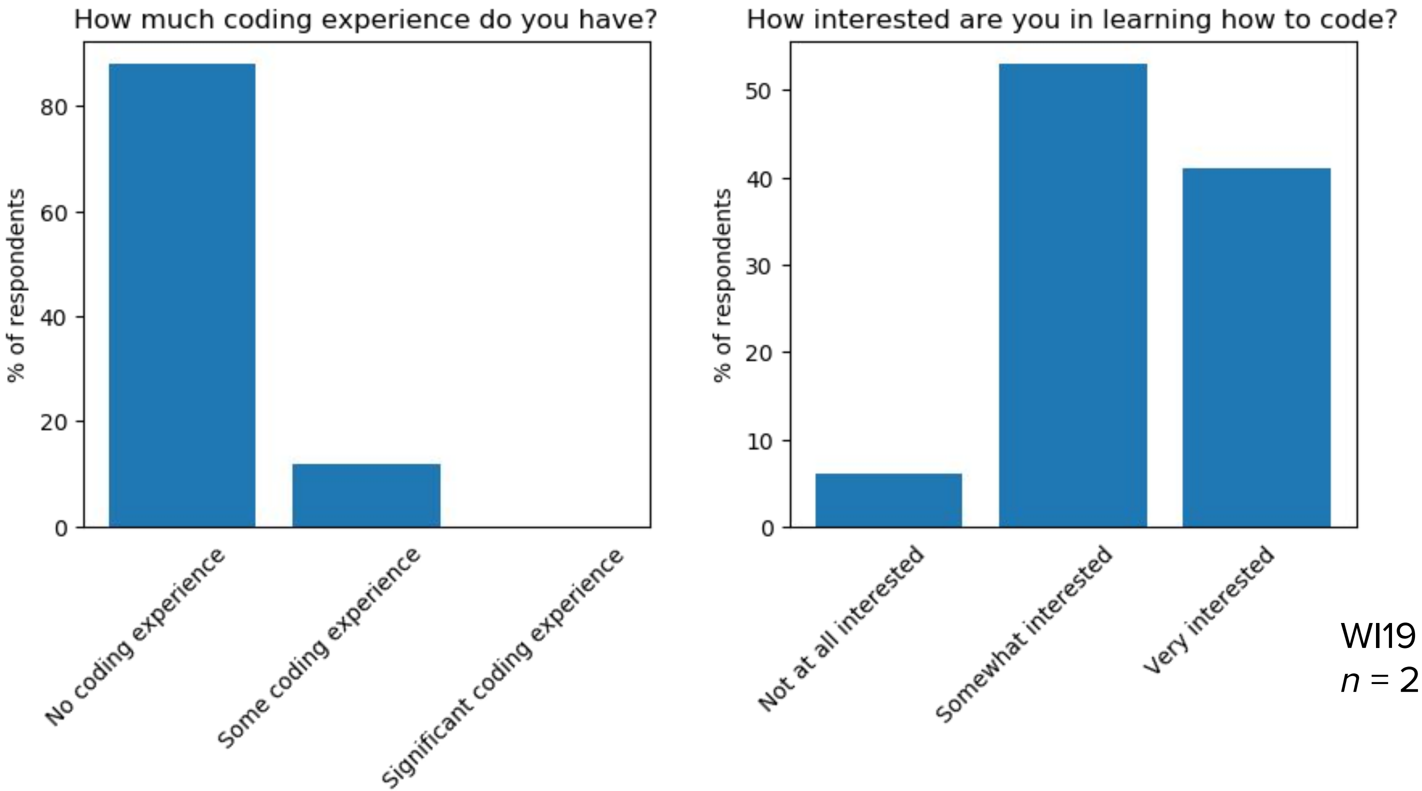


My roommate is a CS major — those classes are so tough

Coding is for rich, white dudes

I'm not a math person, so I won't be able to code

Incoming Neurobiology Lab students at UC San Diego don't have much coding experience, but they are interested in learning!



WI19 students
n = 20

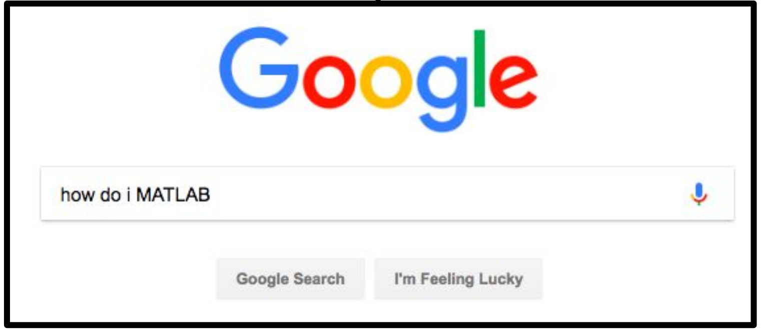
A general framework for teaching coding to neuroscience students

- There is no **programming gene**
- Frame coding as a **skill** — *it's like learning a language*
- Encourage **growth mindsets** around coding
    - Students with growth mindsets perform better over time (even in organic chemistry)
    - 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

A general framework for teaching coding to neuroscience students

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



how do i MATLAB

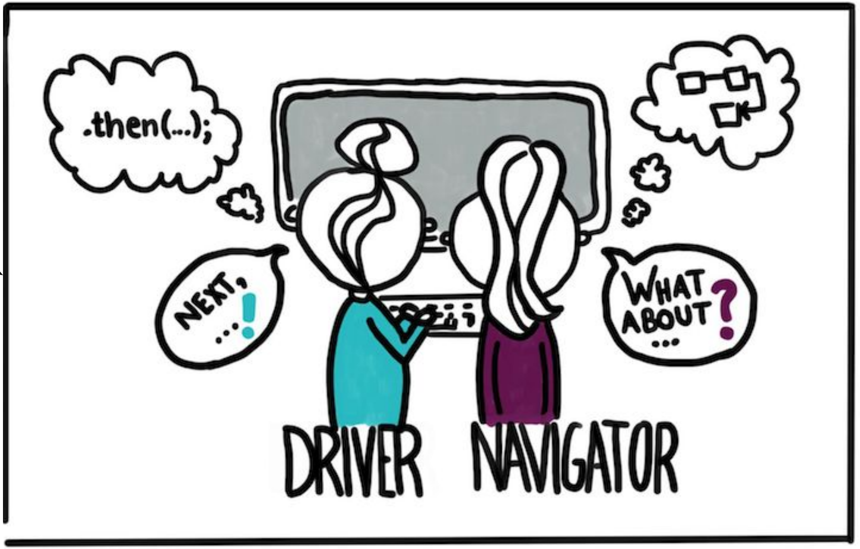


Image Source

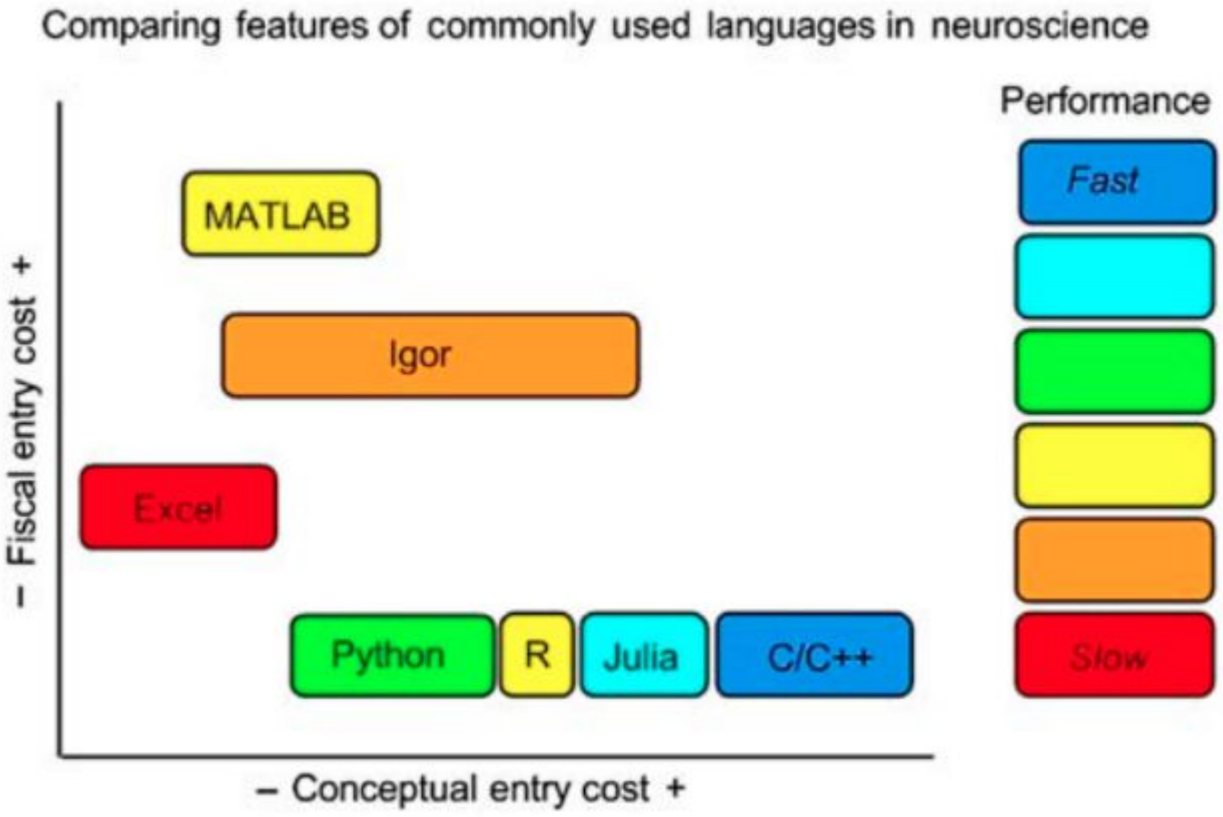
More thoughts on this: Ten quick tips for teaching programming

Goals for this workshop

- Provide a general framework on teaching coding to neuroscience students
- **Introduce Jupyter Notebooks as a tool for teaching**
- Demonstrate the use of the Jupyter Notebooks

Considerations for choosing a programming language

- Fiscal & conceptual entry
- Usage in neuroscience
    - Python, MATLAB, and R



From Wallisch (2017)

Benefits of teaching with Python

- Free, with a *ton* of open access neuroscience code & tutorials
  - Including entire neuroscience courses, e.g., Neuromatch Academy
- Uses concise structure & wording similar to human language
- Works well with free “Notebook” like computing interfaces, such as Jupyter Notebooks, Binder, or Google Colaboratory

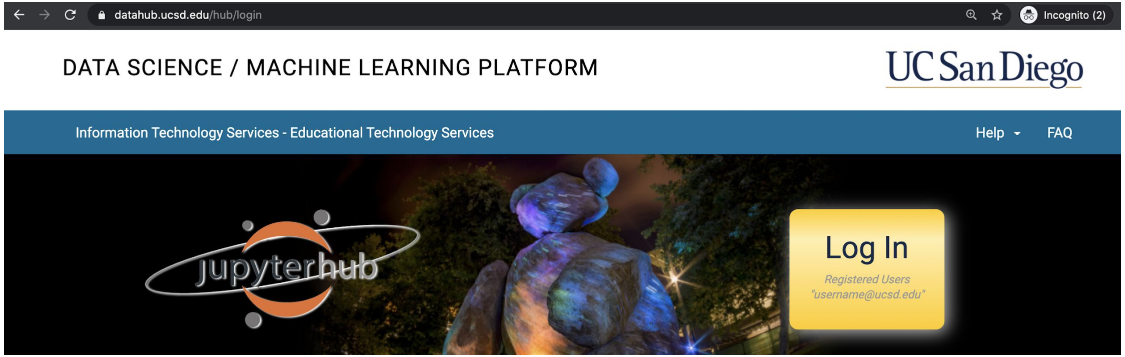
we will use these today

python™ binder

colab

There are several ways to implement Jupyter Notebooks:

- Run notebooks via Binder or Google Colaboratory, as we'll do today
- Set up a JupyterHub for your course or university
- Students download the Anaconda Distribution on their computer
  - Includes Jupyter Notebook, many useful packages and more



datahub.ucsd.edu/hub/login

DATA SCIENCE / MACHINE LEARNING PLATFORM

UC San Diego

Information Technology Services - Educational Technology Services

ANACONDA®

Goals for this workshop

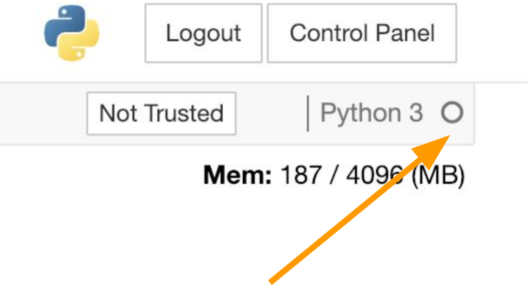
- Provide a general framework on teaching coding to neuroscience students
- Introduce Jupyter Notebooks as a tool for teaching
- **Demonstrate the use of the Jupyter Notebooks for:**
  - Statistics
  - Signal Processing
  - Neural Data Science

About Jupyter Notebooks

- Jupyter is a loose acronym for Julia, Python, and R 😏
- Run in a web browser!
- Shows plots directly in the notebook as you work your way through, performing analyses in real-time
- **Cell**: the main organizational structure of the notebook
  - Use **Shift+Enter** to run a cell (or press Run)
  - You can run cells out of order, and move cells around!
  - Cells can be **code** or **markdown** (descriptive text or images)

Using Jupyter Notebooks (continued)

- Processing-intensive cells can take longer to run, but your code may also get stuck in a cell.
  - Interrupt a stuck cell using Kernel > Interrupt
- **If you change anything in the cell, you need to re-run it.**
- For help:
  - Help > User Interface tour
  - Help > Keyboard Shortcuts



You can tell if the kernel is busy by whether or not the circle next to Python 3 (upper right corner) is filled or not. (filled = busy)

Let’s code!

https://bit.ly/FUNSVMjupyter

That link will send you to:
https://mybinder.org/v2/gh/ajuavinett/FUN_SVM_2020.git/master?filepath=Tutorial.ipynb

Alternate link to Colaboratory (click on Tutorial.ipynb):
http://colab.research.google.com/github/ajuavinett/FUN_SVM_2020