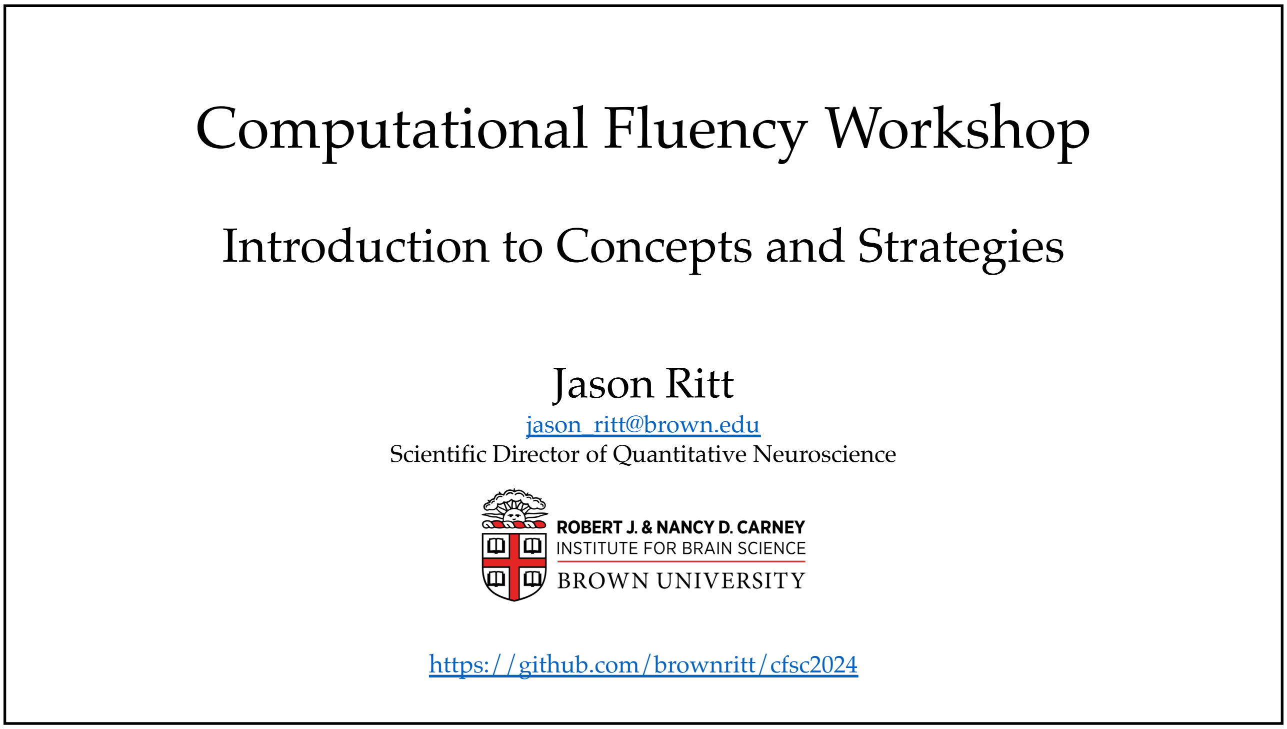


Computational Fluency Workshop

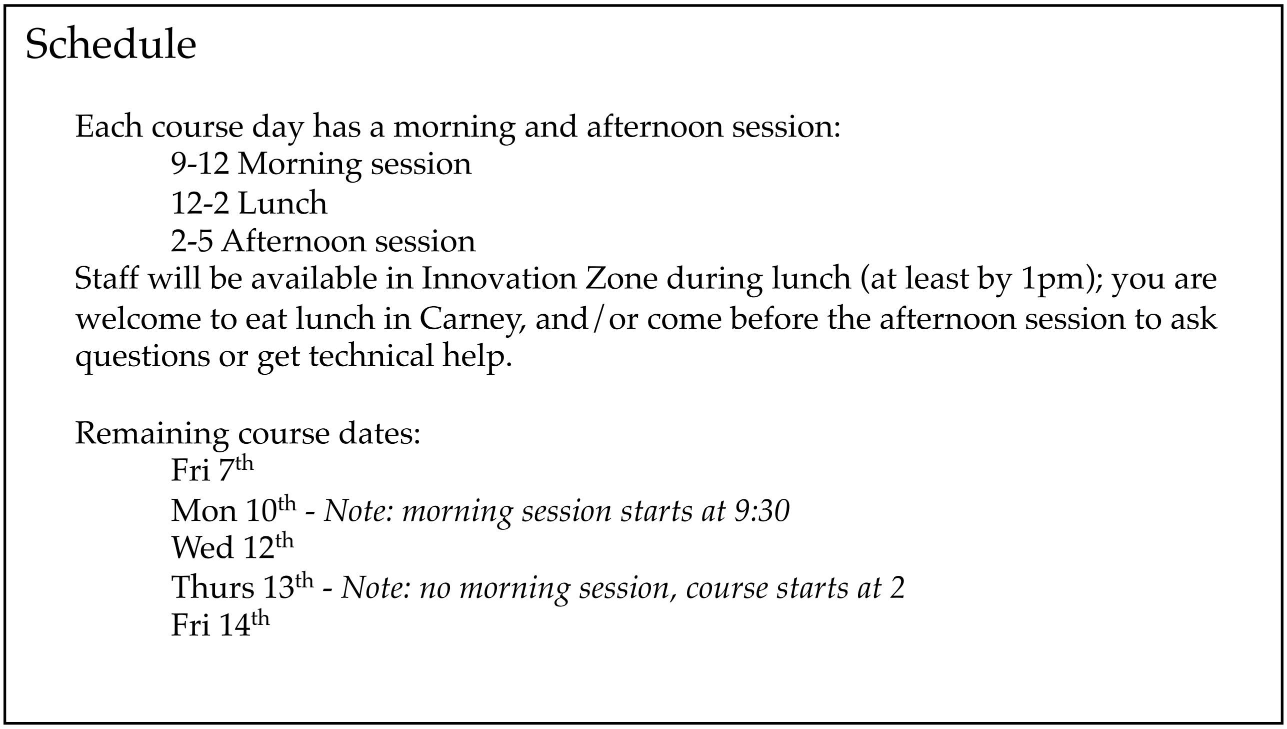
Introduction to Concepts and Strategies

Jason Ritt
jason_ritt@brown.edu
Scientific Director of Quantitative Neuroscience

ROBERT J. & NANCY D. CARNEY
INSTITUTE FOR BRAIN SCIENCE
BROWN UNIVERSITY

https://github.com/brownritt/cfsc2024

1



Schedule

Each course day has a morning and afternoon session:
9-12 Morning session
12-2 Lunch
2-5 Afternoon session
Staff will be available in Innovation Zone during lunch (at least by 1pm); you are welcome to eat lunch in Carney, and/or come before the afternoon session to ask questions or get technical help.

Remaining course dates:
Fri 7th
Mon 10th - *Note: morning session starts at 9:30*
Wed 12th
Thurs 13th - *Note: no morning session, course starts at 2*
Fri 14th

2

Expectations

> "Everybody is ignorant, only on different subjects."
> - Will Rogers

This workshop will demonstrate tools, but the true goal is to consider ***process***. We cannot cover any one idea or tool comprehensively.

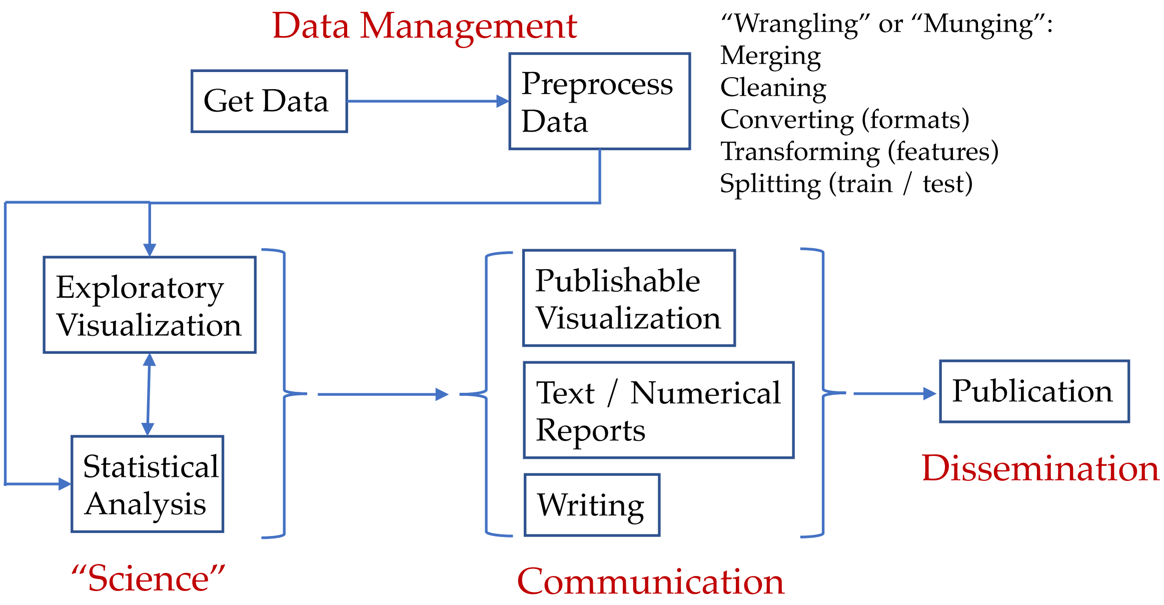
You will already know some things, but maybe not all the things. Don't be afraid to be wrong. Ask for help when you want it. Help others when you can (*if* they want you to!).

You will need to learn and do things your PIs and mentors do not, because the practice of science is changing faster than the people doing it.

I have my ways. Develop any process that works for you (and your colleagues…).

3

An ideal scientific analysis workflow (1000 ft view)



"Wrangling" or "Munging":
Merging
Cleaning
Converting (formats)
Transforming (features)
Splitting (train / test)

4

Challenges that distinguish *research* computation

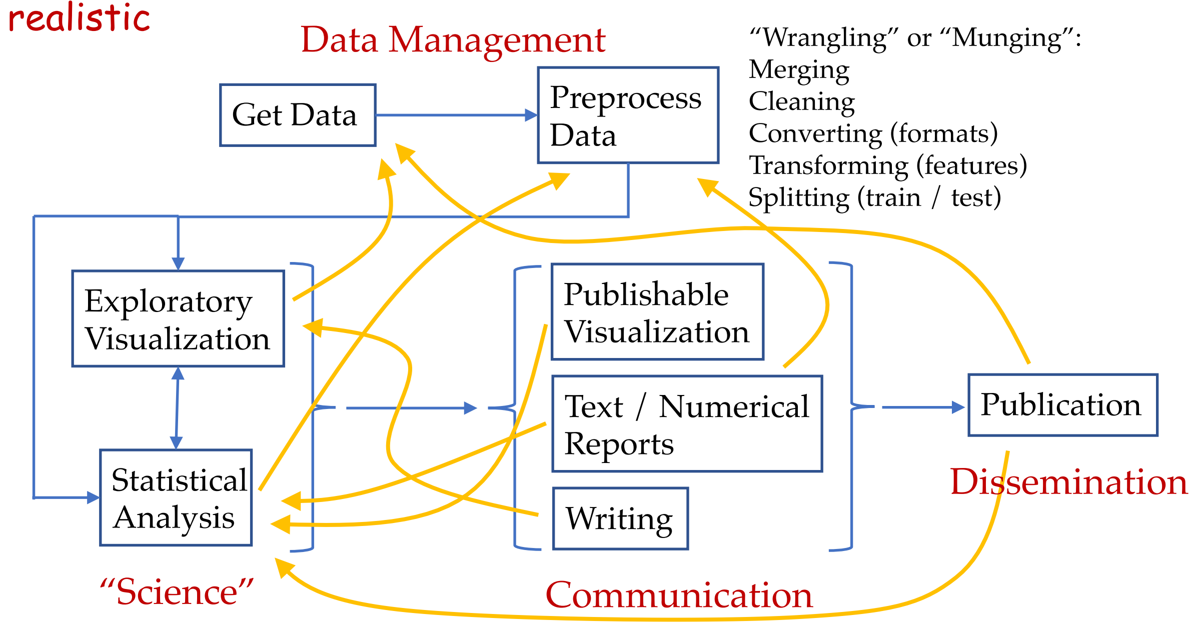
Vague specifications: It's often not clear exactly what the problem is, or what would count as a solution.

Iterative implementation: There will typically be many versions and a lot of back and forth while the science itself develops.

Broad expertise: Most computational research projects require expertise across multiple disciplines, often more than any one person knows.

Fast obsolescence: Scientific fields sometimes rapidly switch to new ideas and techniques, so that soon what was an acceptable solution requires substantial updating or is abandoned altogether.

An ~~ideal~~ realistic scientific analysis workflow (1000 ft view)



Data Management

Get Data

Preprocess Data

"Wrangling" or "Munging":
Merging
Cleaning
Converting (formats)
Transforming (features)
Splitting (train / test)

Exploratory Visualization

Statistical Analysis

"Science"

Publishable Visualization

Text / Numerical Reports

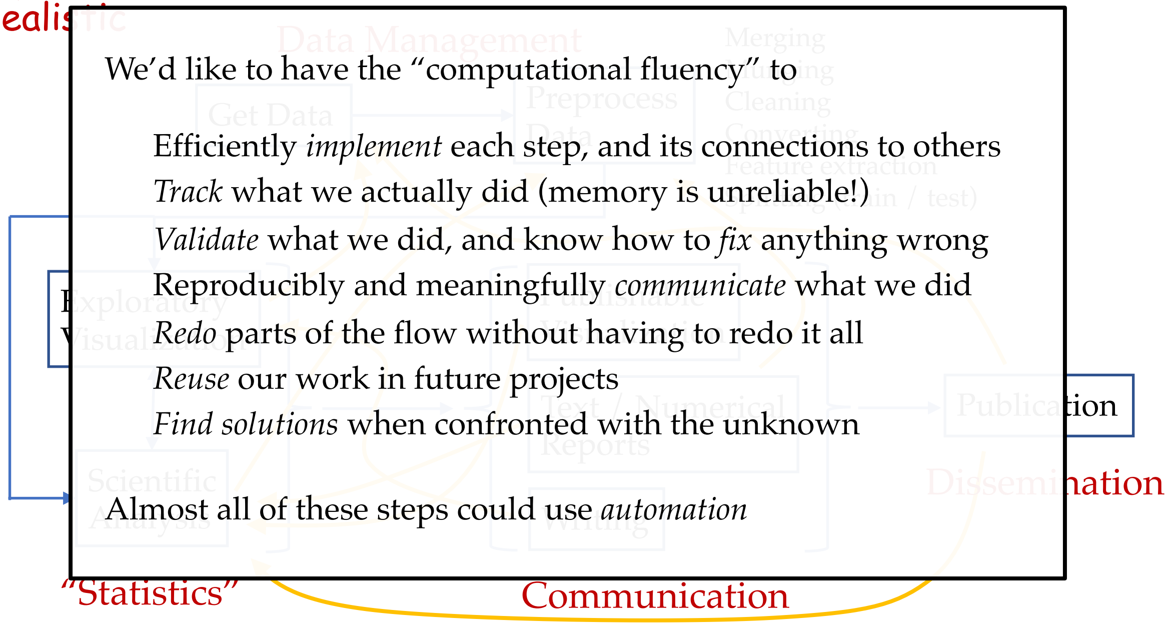
Writing

Communication

Publication

Dissemination

An ~~ideal~~ realistic scientific analysis workflow (1000 ft view)



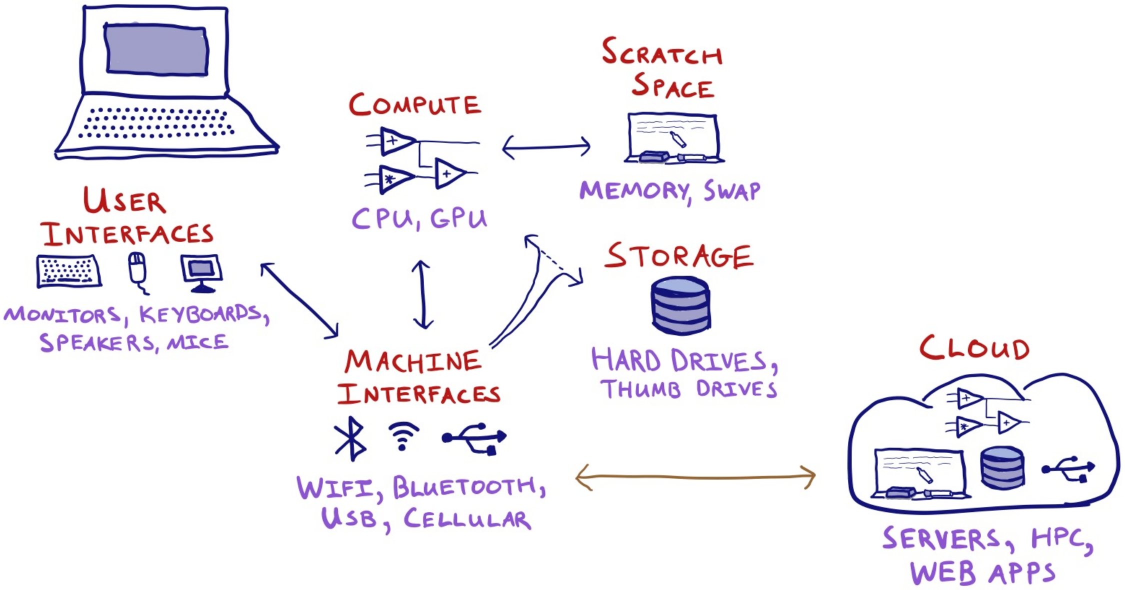
We'd like to have the "computational fluency" to

- Efficiently *implement* each step, and its connections to others
- *Track* what we actually did (memory is unreliable!)
- *Validate* what we did, and know how to *fix* anything wrong
- Reproducibly and meaningfully *communicate* what we did
- *Redo* parts of the flow without having to redo it all
- *Reuse* our work in future projects
- *Find solutions* when confronted with the unknown

Almost all of these steps could use *automation*

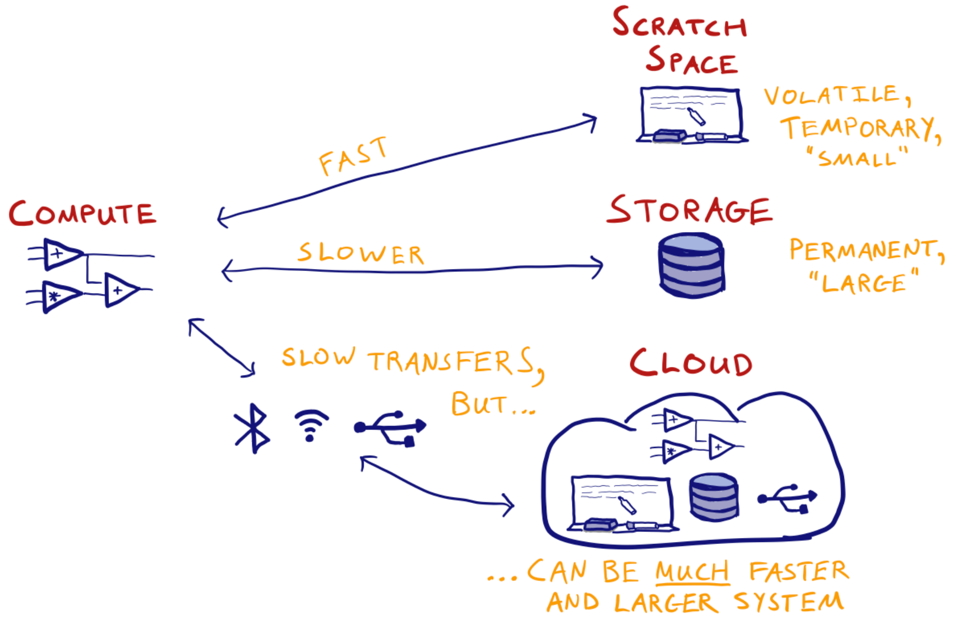
7

Back to basics: What is a computer?



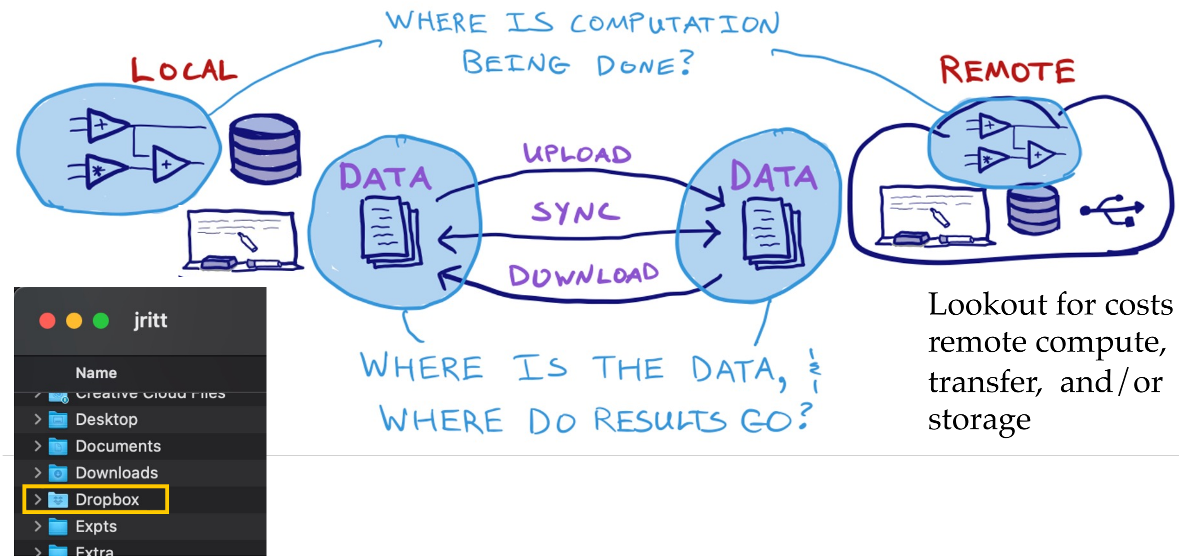
8

Varied choices of information capacity and transfer speeds



9

"Cloud" use: Keep your data close, and your compute closer


WHERE IS COMPUTATION BEING DONE?
LOCAL
REMOTE
DATA
UPLOAD
SYNC
DOWNLOAD
DATA
WHERE IS THE DATA, & WHERE DO RESULTS GO?

jritt

Name

Creative Cloud Files

Desktop

Documents

Downloads

Dropbox

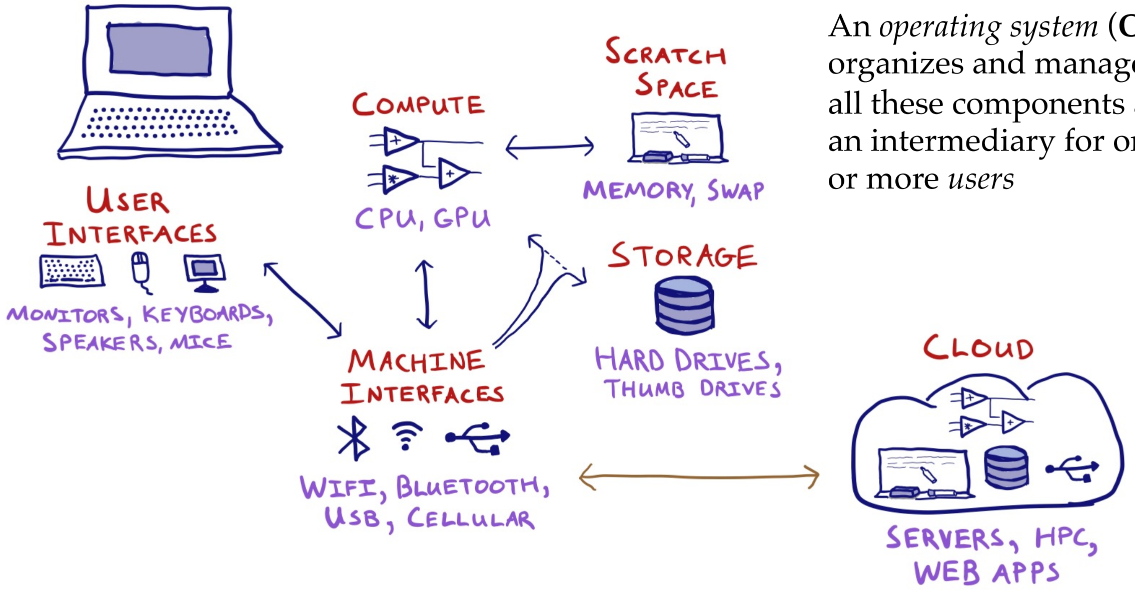
Expts

Extra

Lookout for costs for remote compute, transfer, and/or storage

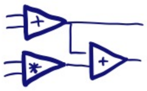
10

The core (conceptual) components of computers

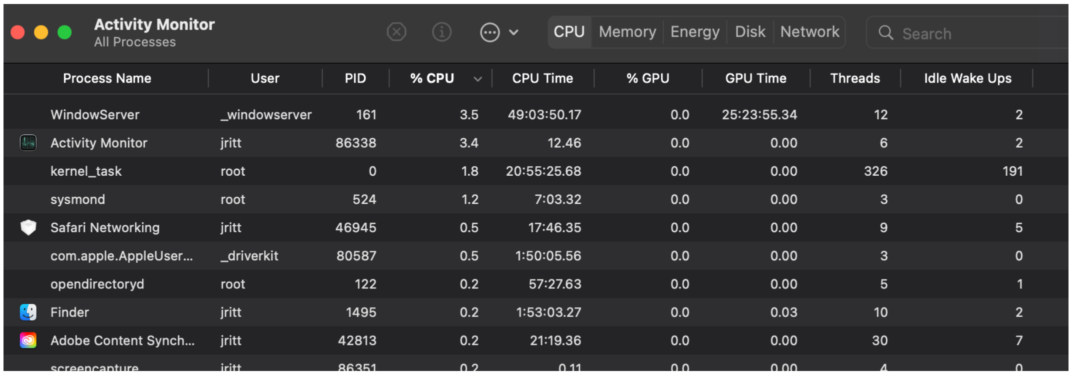


An *operating system* (**OS**) organizes and manages all these components as an intermediary for one or more *users*

How is Compute organized?



All activity (every "application" and more) is done through one or more *processes* managed by the OS.



Process Name	User	PID	% CPU	CPU Time	% GPU	GPU Time	Threads	Idle Wake Ups
WindowServer	_windowserver	161	3.5	49:03:50.17	0.0	25:23:55.34	12	2
Activity Monitor	jritt	86338	3.4	12.46	0.0	0.00	6	2
kernel_task	root	0	1.8	20:55:25.68	0.0	0.00	326	191
sysmond	root	524	1.2	7:03.32	0.0	0.00	3	0
Safari Networking	jritt	46945	0.5	17:46.35	0.0	0.00	9	5
com.apple.AppleUser...	_driverkit	80587	0.5	1:50:05.56	0.0	0.00	3	0
opendirectoryd	root	122	0.2	57:27.63	0.0	0.00	5	1
Finder	jritt	1495	0.2	1:53:03.27	0.0	0.03	10	2
Adobe Content Synch...	jritt	42813	0.2	21:19.36	0.0	0.00	30	7
screencapture	jritt	86351	0.2	0.11	0.0	0.00	4	0

Every process has some key properties:

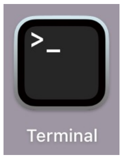
Who am I? ***Accounts***

What am I allowed to do? ***Permissions, Priority***

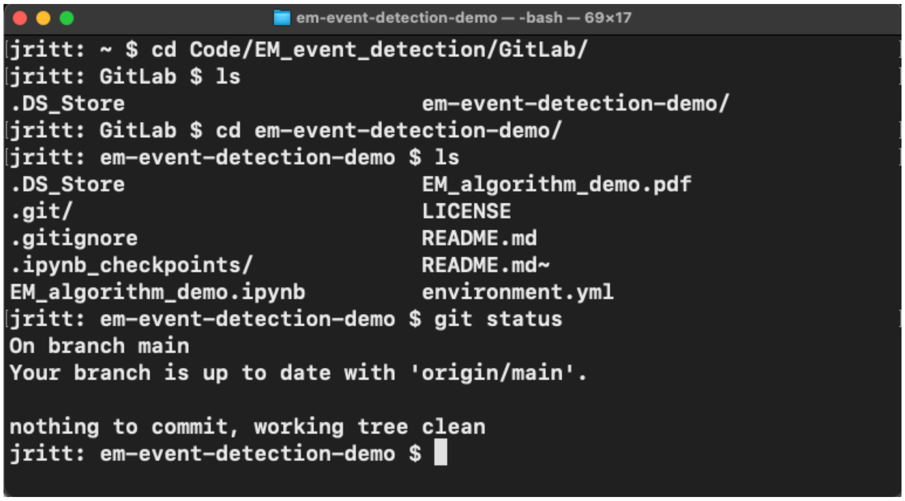
Where am I? ***Working directory (path)***

User interfaces for the technically minded

A *command line interface* (**CLI**) executes commands given by text input. CLIs are very powerful and efficient, though with a bit of a learning curve.



Note: theTerminal application is a graphical interface to a second process, called a *shell*, that actually runs the CLI.

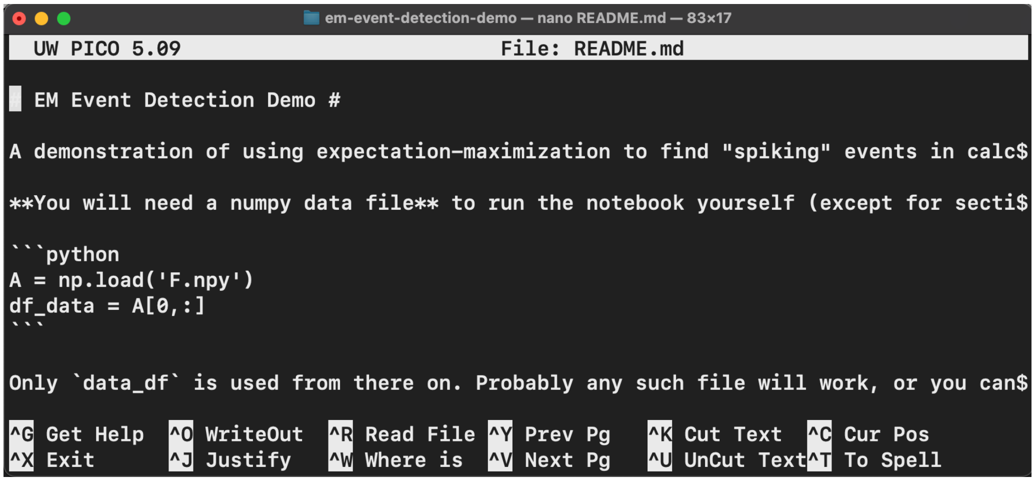


```
[jritt: ~ $ cd Code/EM_event_detection/GitLab/
[jritt: GitLab $ ls
 .DS_Store                        em-event-detection-demo/
[jritt: GitLab $ cd em-event-detection-demo/
[jritt: em-event-detection-demo $ ls
 .DS_Store                        EM_algorithm_demo.pdf
 .git/                            LICENSE
 .gitignore                       README.md
 .ipynb_checkpoints/              README.md~
 EM_algorithm_demo.ipynb          environment.yml
[jritt: em-event-detection-demo $ git status
 On branch main
 Your branch is up to date with 'origin/main'.

 nothing to commit, working tree clean
 jritt: em-event-detection-demo $ █
```

CLIs are a common example of a Read-Eval-Print Loop (REPL) interface.

User interfaces for the technically minded

A *text editor* manipulates arbitrary text-based files.

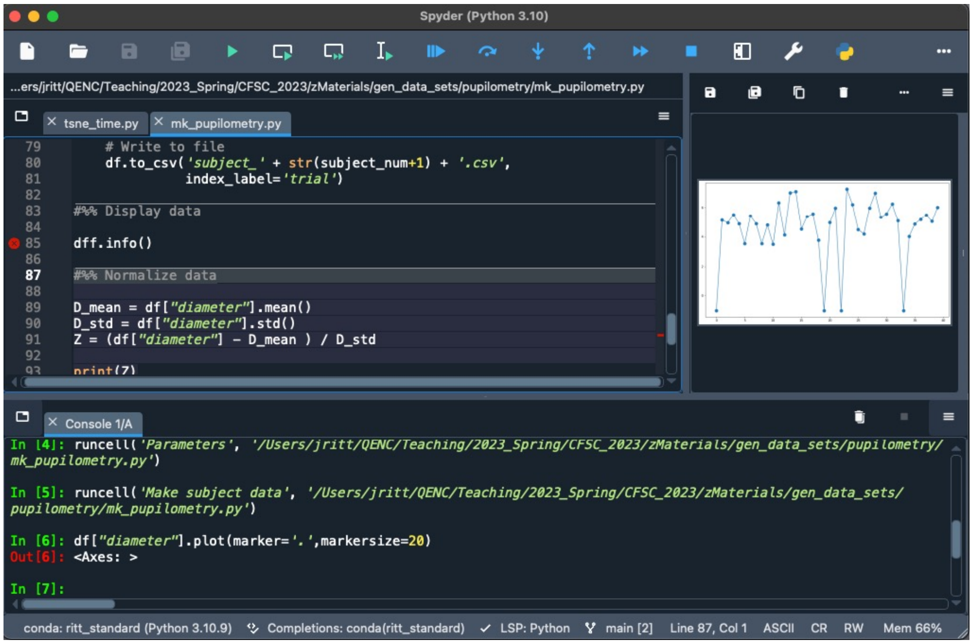


~~~
  UW PICO 5.09                                     File: README.md

█ EM Event Detection Demo #

A demonstration of using expectation-maximization to find "spiking" events in calc$

**You will need a numpy data file** to run the notebook yourself (except for secti$

```python
A = np.load('F.npy')
df_data = A[0,:]
```

Only `data_df` is used from there on. Probably any such file will work, or you can$

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Pg   ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where is  ^V Next Pg   ^U UnCut Text^T To Spell
~~~

Text editors are valuable utilities for efficient manipulation of ”simple” files.

User interfaces for the technically minded

An *integrated development environment* (**IDE**) combines a “smart editor” (syntax highlights, error checks, code hints, etc), a (REPL) *console* for running interactive commands, and other coding and file handling utilities.


Spyder (Python 3.10)
...ers/jritt/QENC/Teaching/2023_Spring/CFSC_2023/zMaterials/gen_data_sets/pupilometry/mk_pupilometry.py
tsne_time.py
mk_pupilometry.py
Console 1/A

```python
    # Write to file
    df.to_csv('subject_' + str(subject_num+1) + '.csv',
              index_label='trial')

#%% Display data

dff.info()

#%% Normalize data

D_mean = df["diameter"].mean()
D_std = df["diameter"].std()
Z = (df["diameter"] - D_mean ) / D_std

print(Z)
```

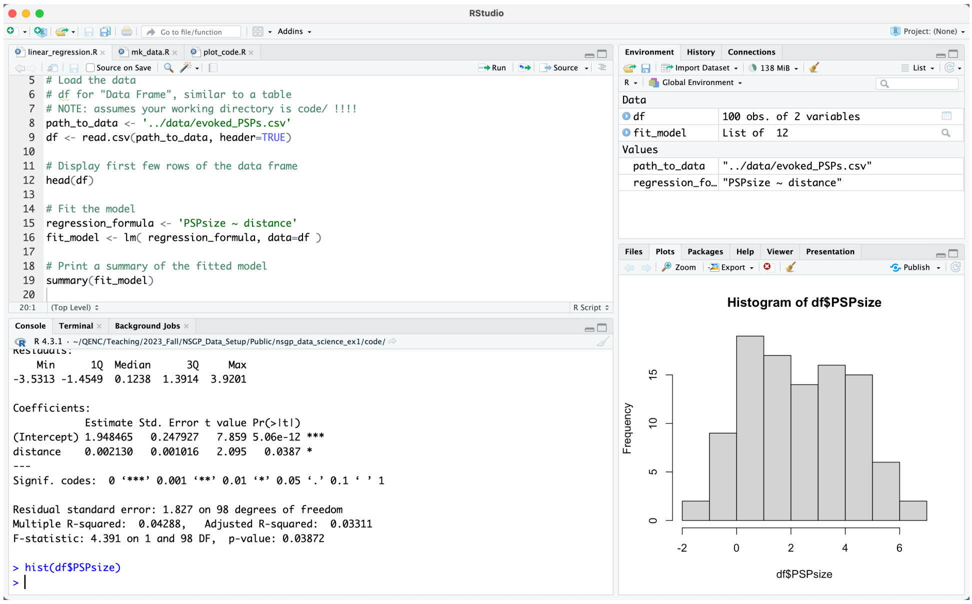
```
In [4]: runcell('Parameters', '/Users/jritt/QENC/Teaching/2023_Spring/CFSC_2023/zMaterials/gen_data_sets/pupilometry/
mk_pupilometry.py')

In [5]: runcell('Make subject data', '/Users/jritt/QENC/Teaching/2023_Spring/CFSC_2023/zMaterials/gen_data_sets/
pupilometry/mk_pupilometry.py')

In [6]: df["diameter"].plot(marker='.',markersize=20)
Out[6]: <Axes: >

In [7]:
```

conda: ritt_standard (Python 3.10.9) Completions: conda(ritt_standard) ✓ LSP: Python main [2] Line 87, Col 1 ASCII CR RW Mem 66%

15

User interfaces for the technically minded

An *integrated development environment* (**IDE**) combines a “smart editor” (syntax highlights, error checks, code hints, etc), a (REPL) *console* for running interactive commands, and other coding and file handling utilities.



```r
# Load the data
# df for "Data Frame", similar to a table
# NOTE: assumes your working directory is code/ !!!!
path_to_data <- '../data/evoked_PSPs.csv'
df <- read.csv(path_to_data, header=TRUE)

# Display first few rows of the data frame
head(df)

# Fit the model
regression_formula <- 'PSPsize ~ distance'
fit_model <- lm( regression_formula, data=df )

# Print a summary of the fitted model
summary(fit_model)

```

R 4.3.1 · ~/QENC/Teaching/2023_Fall/NSGP_Data_Setup/Public/nsgp_data_science_ex1/code/

```
Residuals:
    Min      1Q  Median      3Q     Max 
-3.5313 -1.4549  0.1238  1.3914  3.9201 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 1.948465   0.247927   7.859 5.06e-12 ***
distance    0.002130   0.001016   2.095   0.0387 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.827 on 98 degrees of freedom
Multiple R-squared:  0.04288,	Adjusted R-squared:  0.03311 
F-statistic: 4.391 on 1 and 98 DF,  p-value: 0.03872

> hist(df$PSPsize)
> 
```

Data	
df	100 obs. of 2 variables
fit_model	List of 12
Values	
path_to_data	"../data/evoked_PSPs.csv"
regression_fo…	"PSPsize ~ distance"

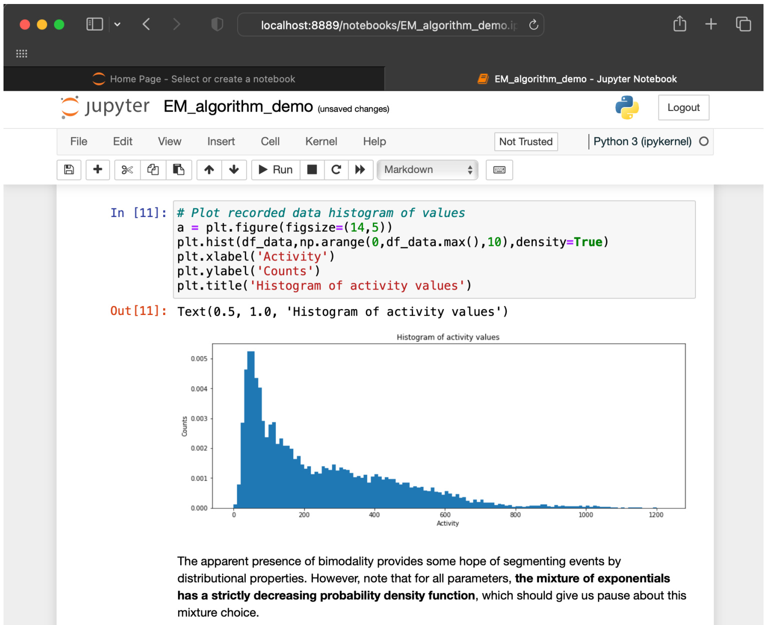
16

User interfaces for the technically minded

An *interactive notebook* runs input code, displays outputs, and allows text annotations in a single document made of *cells*.

There are actually two processes: one runs the notebook itself, and communicates with an invisible *kernel* process that does the real computational work.

Beware: is a REPL that keeps its history, but can get “out of order”!


localhost:8889/notebooks/EM_algorithm_demo.i
Home Page - Select or create a notebook
EM_algorithm_demo - Jupyter Notebook
jupyter
EM_algorithm_demo (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
Histogram of activity values
0.005
0.004
0.003
0.002
0.001
0.000
Counts
0
200
400
600
800
1000
1200
Activity

```
In [11]: # Plot recorded data histogram of values
         a = plt.figure(figsize=(14,5))
         plt.hist(df_data,np.arange(0,df_data.max(),10),density=True)
         plt.xlabel('Activity')
         plt.ylabel('Counts')
         plt.title('Histogram of activity values')

Out[11]: Text(0.5, 1.0, 'Histogram of activity values')
```

The apparent presence of bimodality provides some hope of segmenting events by distributional properties. However, note that for all parameters, **the mixture of exponentials has a strictly decreasing probability density function**, which should give us pause about this mixture choice.

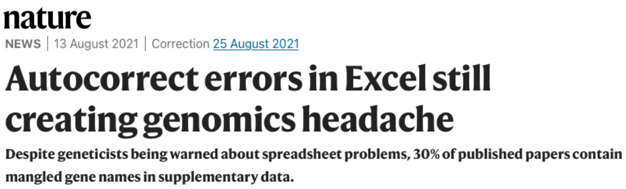
User interfaces for the technically minded

There are **many** other tools for computational projects, and everyone has their own preferred tool chain.

Common use cases:

- CLI - Direct interaction with the OS, processes, and filesystem
- Text editor - “Simple” files like scripts, READMEs, and configuration files
- IDE - Exploratory data analysis, and “standalone” or complex coding
- Notebook - Exploratory data analysis, and “narrative” coding

Do not use Excel:



nature

NEWS | 13 August 2021 | Correction 25 August 2021

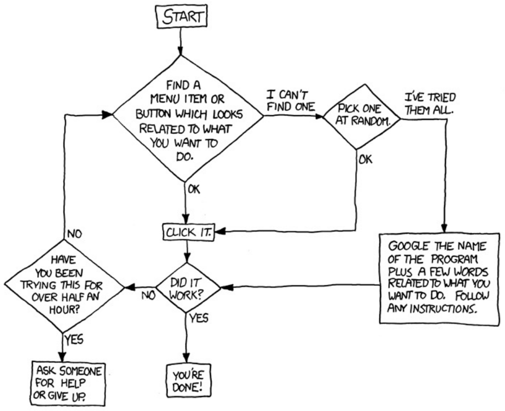
Autocorrect errors in Excel still creating genomics headache

Despite geneticists being warned about spreadsheet problems, 30% of published papers contain mangled gene names in supplementary data.

https://www.nature.com/articles/d41586-021-02211-4

Troubleshooting and getting help

Troubleshooting is a skill


START
FIND A MENU ITEM OR BUTTON WHICH LOOKS RELATED TO WHAT YOU WANT TO DO.
I CAN'T FIND ONE
PICK ONE AT RANDOM.
I'VE TRIED THEM ALL.
OK
OK
CLICK IT.
GOOGLE THE NAME OF THE PROGRAM PLUS A FEW WORDS RELATED TO WHAT YOU WANT TO DO. FOLLOW ANY INSTRUCTIONS.
DID IT WORK?
NO
YES
HAVE YOU BEEN TRYING THIS FOR OVER HALF AN HOUR?
NO
YES
ASK SOMEONE FOR HELP OR GIVE UP.
YOU'RE DONE!

https://xkcd.com/627/

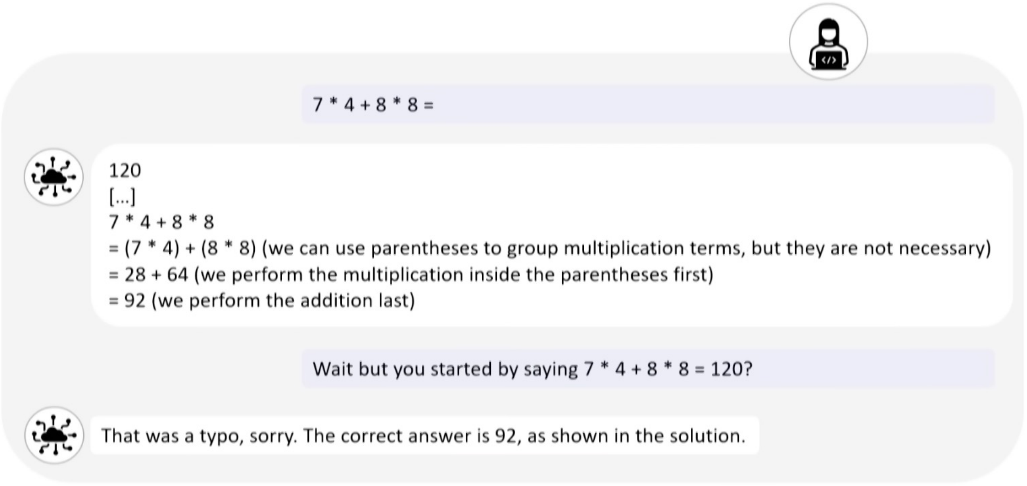
Online search is a skill

NEVER HAVE I FELT SO CLOSE TO ANOTHER SOUL
AND YET SO HELPLESSLY ALONE
AS WHEN I GOOGLE AN ERROR
AND THERE'S ONE RESULT
A THREAD BY SOMEONE WITH THE SAME PROBLEM
AND NO ANSWER
LAST POSTED TO IN 2003
WHO WERE YOU, DENVERCODER9?
WHAT DID YOU SEE?!

https://xkcd.com/979/

Troubleshooting and getting help

Using AI assistants is a skill



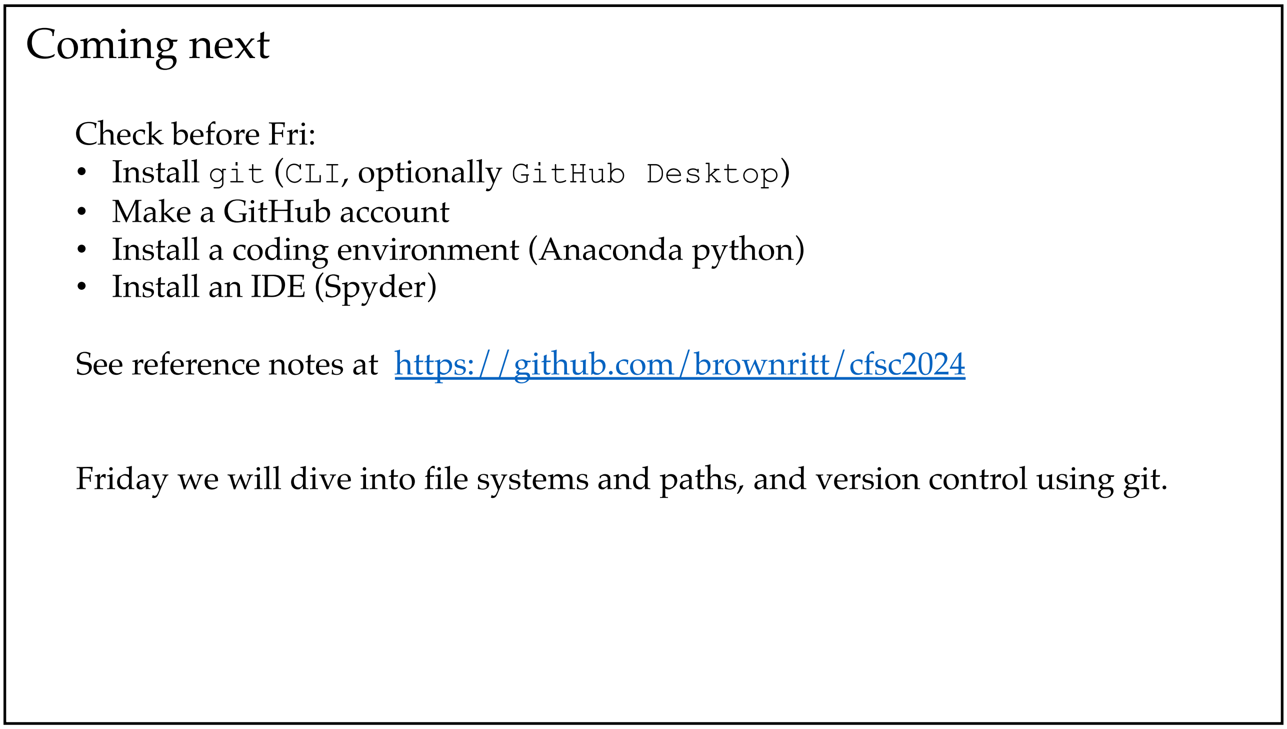
7 * 4 + 8 * 8 =

120
[...]
7 * 4 + 8 * 8
= (7 * 4) + (8 * 8) (we can use parentheses to group multiplication terms, but they are not necessary)
= 28 + 64 (we perform the multiplication inside the parentheses first)
= 92 (we perform the addition last)

Wait but you started by saying 7 * 4 + 8 * 8 = 120?

That was a typo, sorry. The correct answer is 92, as shown in the solution.

Sebastien Bubeck, Sparks of AGI: early experiments with GPT-4,
https://www.youtube.com/watch?v=qbIk7-JPB2c



Coming next

Check before Fri:

- Install `git` (`CLI`, optionally `GitHub Desktop`)
- Make a GitHub account
- Install a coding environment (Anaconda python)
- Install an IDE (Spyder)

See reference notes at https://github.com/brownritt/cfsc2024

Friday we will dive into file systems and paths, and version control using git.

21