

CEE 690-02

ENVIRONMENTAL SPATIAL  
DATA ANALYSIS (ESDA)

Lecture I

# Introduction

- Instructor: Nathaniel Chaney (Nate)
- TA: Reza Momenifar

# Motivation

There are big environmental challenges that we want to take on...

**Here are some examples from Hydrology**

# Floods



Where will it flood during a hurricane?

How can we mitigate the impacts of floods?

What is the risk that my house is flooded?

# Agriculture

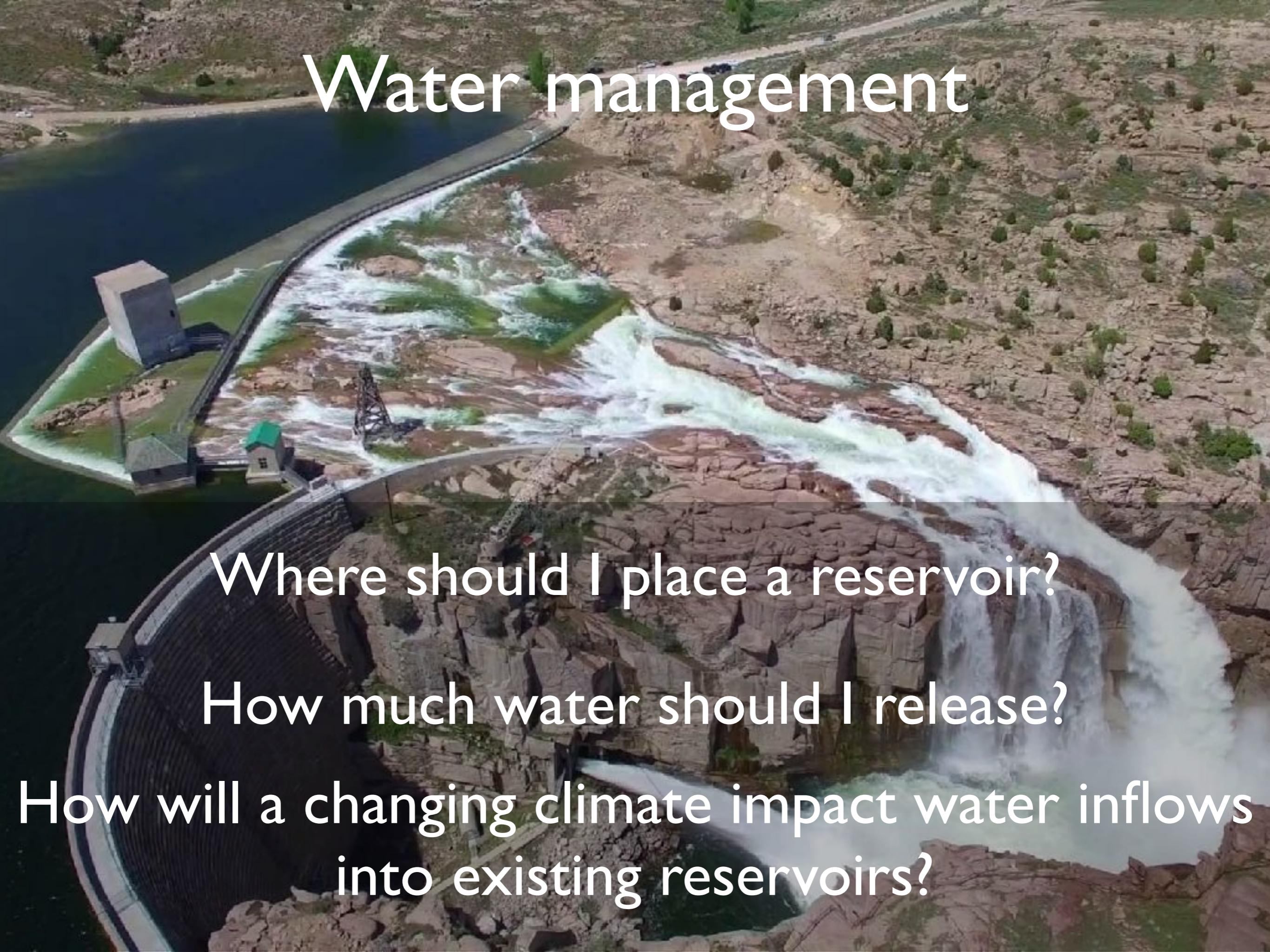


When should I irrigate my crops? And how much?

Where should the irrigation water come from?

What type of irrigation system should I use?

# Water management

An aerial photograph of a large concrete dam with a curved top. Water flows over the dam's crest, creating white rapids. A long, narrow溢流道 (spillway) extends from the right side of the dam into a rocky valley. The reservoir behind the dam is dark blue. The surrounding terrain is dry and rocky, with sparse green vegetation.

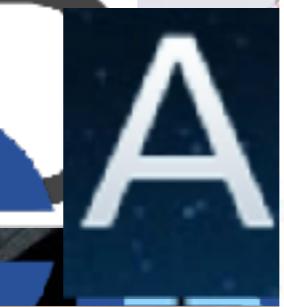
Where should I place a reservoir?

How much water should I release?

How will a changing climate impact water inflows  
into existing reservoirs?

How do we address those  
questions?

**Data, data, and more data**



rest

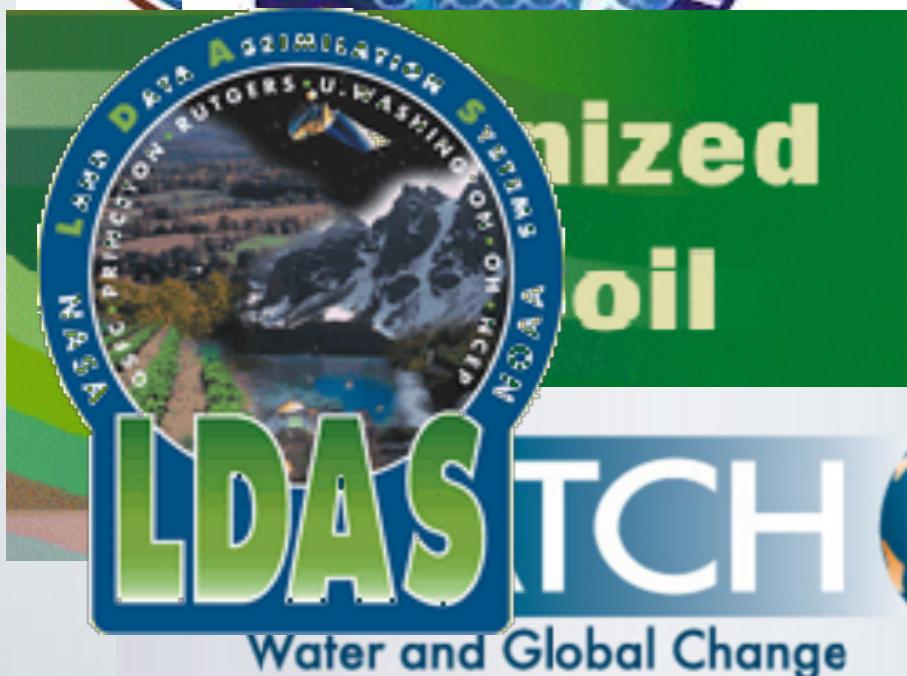


TANDEM

onitor

A-40

# “Age of Data”



THE LONG TERM  
ECOLOGICAL  
RESEARCH  
NETWORK



Source: Proctor, J. et al. (2009) Geospatial data mining for success in the long-term ecological research program. *Geospatial and Temporal Analysis*, 1, 1–20. doi:10.1080/19412569.2009.9628200

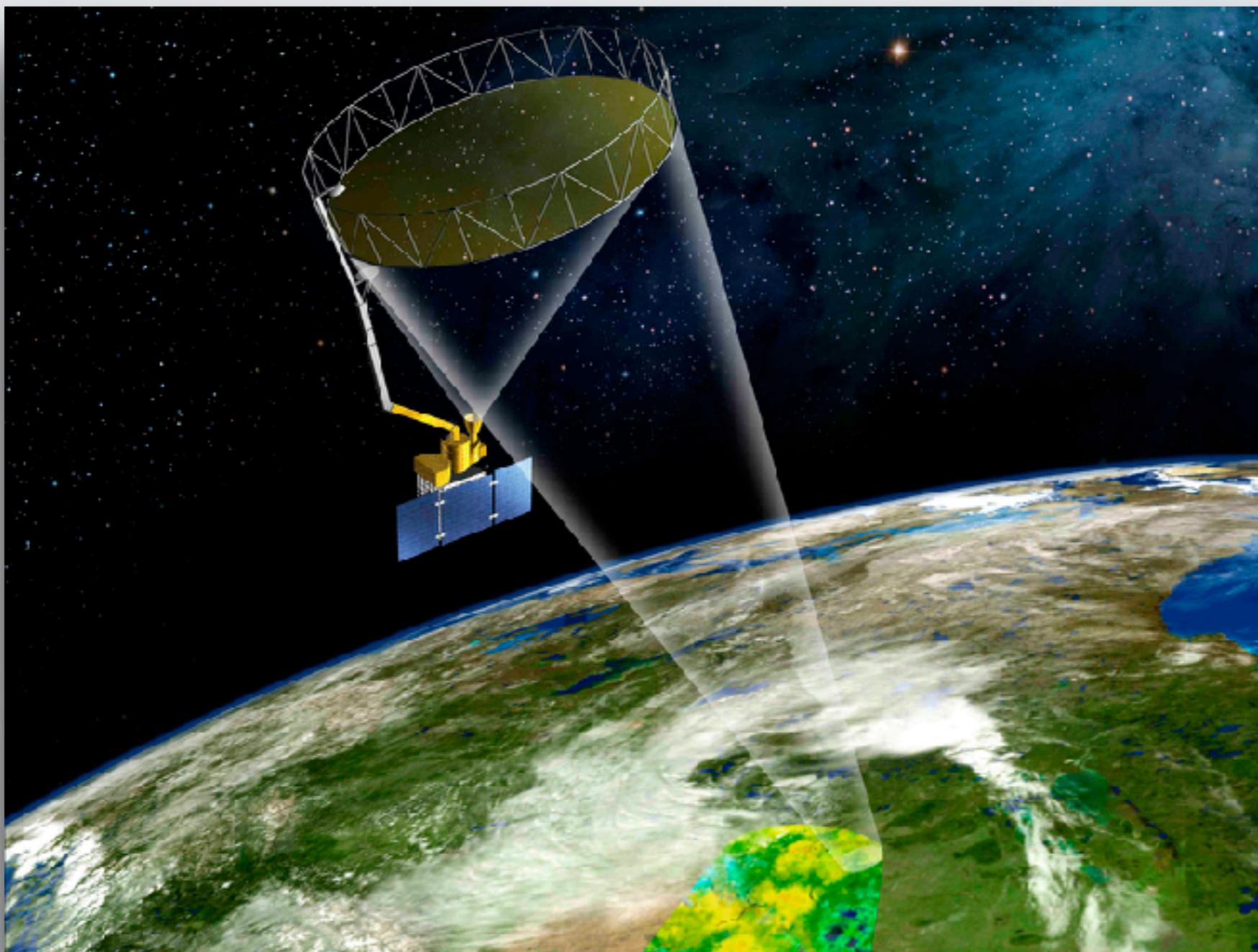
Legend:  
Region 1: 1.000 - 1.500  
Region 2: 1.501 - 2.000  
Region 3: 2.001 - 2.500  
Region 4: 2.501 - 3.000  
Region 5: 3.001 - 3.500  
Region 6: 3.501 - 4.000  
Region 7: 4.001 - 4.500  
Region 8: 4.501 - 5.000  
Region 9: 5.001 - 5.500  
Region 10: 5.501 - 6.000  
Region 11: 6.001 - 6.500  
Region 12: 6.501 - 7.000  
Region 13: 7.001 - 7.500  
Region 14: 7.501 - 8.000  
Region 15: 8.001 - 8.500  
Region 16: 8.501 - 9.000  
Region 17: 9.001 - 9.500  
Region 18: 9.501 - 10.000  
Region 19: 10.001 - 10.500  
Region 20: 10.501 - 11.000  
Region 21: 11.001 - 11.500  
Region 22: 11.501 - 12.000  
Region 23: 12.001 - 12.500  
Region 24: 12.501 - 13.000  
Region 25: 13.001 - 13.500  
Region 26: 13.501 - 14.000  
Region 27: 14.001 - 14.500  
Region 28: 14.501 - 15.000  
Region 29: 15.001 - 15.500  
Region 30: 15.501 - 16.000  
Region 31: 16.001 - 16.500  
Region 32: 16.501 - 17.000  
Region 33: 17.001 - 17.500  
Region 34: 17.501 - 18.000  
Region 35: 18.001 - 18.500  
Region 36: 18.501 - 19.000  
Region 37: 19.001 - 19.500  
Region 38: 19.501 - 20.000  
Region 39: 20.001 - 20.500  
Region 40: 20.501 - 21.000  
Region 41: 21.001 - 21.500  
Region 42: 21.501 - 22.000  
Region 43: 22.001 - 22.500  
Region 44: 22.501 - 23.000  
Region 45: 23.001 - 23.500  
Region 46: 23.501 - 24.000  
Region 47: 24.001 - 24.500  
Region 48: 24.501 - 25.000  
Region 49: 25.001 - 25.500  
Region 50: 25.501 - 26.000  
Region 51: 26.001 - 26.500  
Region 52: 26.501 - 27.000  
Region 53: 27.001 - 27.500  
Region 54: 27.501 - 28.000  
Region 55: 28.001 - 28.500  
Region 56: 28.501 - 29.000  
Region 57: 29.001 - 29.500  
Region 58: 29.501 - 30.000  
Region 59: 30.001 - 30.500  
Region 60: 30.501 - 31.000  
Region 61: 31.001 - 31.500  
Region 62: 31.501 - 32.000  
Region 63: 32.001 - 32.500  
Region 64: 32.501 - 33.000  
Region 65: 33.001 - 33.500  
Region 66: 33.501 - 34.000  
Region 67: 34.001 - 34.500  
Region 68: 34.501 - 35.000  
Region 69: 35.001 - 35.500  
Region 70: 35.501 - 36.000  
Region 71: 36.001 - 36.500  
Region 72: 36.501 - 37.000  
Region 73: 37.001 - 37.500  
Region 74: 37.501 - 38.000  
Region 75: 38.001 - 38.500  
Region 76: 38.501 - 39.000  
Region 77: 39.001 - 39.500  
Region 78: 39.501 - 40.000  
Region 79: 40.001 - 40.500  
Region 80: 40.501 - 41.000  
Region 81: 41.001 - 41.500  
Region 82: 41.501 - 42.000  
Region 83: 42.001 - 42.500  
Region 84: 42.501 - 43.000  
Region 85: 43.001 - 43.500  
Region 86: 43.501 - 44.000  
Region 87: 44.001 - 44.500  
Region 88: 44.501 - 45.000  
Region 89: 45.001 - 45.500  
Region 90: 45.501 - 46.000  
Region 91: 46.001 - 46.500  
Region 92: 46.501 - 47.000  
Region 93: 47.001 - 47.500  
Region 94: 47.501 - 48.000  
Region 95: 48.001 - 48.500  
Region 96: 48.501 - 49.000  
Region 97: 49.001 - 49.500  
Region 98: 49.501 - 50.000  
Region 99: 50.001 - 50.500  
Region 100: 50.501 - 51.000  
Region 101: 51.001 - 51.500  
Region 102: 51.501 - 52.000  
Region 103: 52.001 - 52.500  
Region 104: 52.501 - 53.000  
Region 105: 53.001 - 53.500  
Region 106: 53.501 - 54.000  
Region 107: 54.001 - 54.500  
Region 108: 54.501 - 55.000  
Region 109: 55.001 - 55.500  
Region 110: 55.501 - 56.000  
Region 111: 56.001 - 56.500  
Region 112: 56.501 - 57.000  
Region 113: 57.001 - 57.500  
Region 114: 57.501 - 58.000  
Region 115: 58.001 - 58.500  
Region 116: 58.501 - 59.000  
Region 117: 59.001 - 59.500  
Region 118: 59.501 - 60.000  
Region 119: 60.001 - 60.500  
Region 120: 60.501 - 61.000  
Region 121: 61.001 - 61.500  
Region 122: 61.501 - 62.000  
Region 123: 62.001 - 62.500  
Region 124: 62.501 - 63.000  
Region 125: 63.001 - 63.500  
Region 126: 63.501 - 64.000  
Region 127: 64.001 - 64.500  
Region 128: 64.501 - 65.000  
Region 129: 65.001 - 65.500  
Region 130: 65.501 - 66.000  
Region 131: 66.001 - 66.500  
Region 132: 66.501 - 67.000  
Region 133: 67.001 - 67.500  
Region 134: 67.501 - 68.000  
Region 135: 68.001 - 68.500  
Region 136: 68.501 - 69.000  
Region 137: 69.001 - 69.500  
Region 138: 69.501 - 70.000  
Region 139: 70.001 - 70.500  
Region 140: 70.501 - 71.000  
Region 141: 71.001 - 71.500  
Region 142: 71.501 - 72.000  
Region 143: 72.001 - 72.500  
Region 144: 72.501 - 73.000  
Region 145: 73.001 - 73.500  
Region 146: 73.501 - 74.000  
Region 147: 74.001 - 74.500  
Region 148: 74.501 - 75.000  
Region 149: 75.001 - 75.500  
Region 150: 75.501 - 76.000  
Region 151: 76.001 - 76.500  
Region 152: 76.501 - 77.000  
Region 153: 77.001 - 77.500  
Region 154: 77.501 - 78.000  
Region 155: 78.001 - 78.500  
Region 156: 78.501 - 79.000  
Region 157: 79.001 - 79.500  
Region 158: 79.501 - 80.000  
Region 159: 80.001 - 80.500  
Region 160: 80.501 - 81.000  
Region 161: 81.001 - 81.500  
Region 162: 81.501 - 82.000  
Region 163: 82.001 - 82.500  
Region 164: 82.501 - 83.000  
Region 165: 83.001 - 83.500  
Region 166: 83.501 - 84.000  
Region 167: 84.001 - 84.500  
Region 168: 84.501 - 85.000  
Region 169: 85.001 - 85.500  
Region 170: 85.501 - 86.000  
Region 171: 86.001 - 86.500  
Region 172: 86.501 - 87.000  
Region 173: 87.001 - 87.500  
Region 174: 87.501 - 88.000  
Region 175: 88.001 - 88.500  
Region 176: 88.501 - 89.000  
Region 177: 89.001 - 89.500  
Region 178: 89.501 - 90.000  
Region 179: 90.001 - 90.500  
Region 180: 90.501 - 91.000  
Region 181: 91.001 - 91.500  
Region 182: 91.501 - 92.000  
Region 183: 92.001 - 92.500  
Region 184: 92.501 - 93.000  
Region 185: 93.001 - 93.500  
Region 186: 93.501 - 94.000  
Region 187: 94.001 - 94.500  
Region 188: 94.501 - 95.000  
Region 189: 95.001 - 95.500  
Region 190: 95.501 - 96.000  
Region 191: 96.001 - 96.500  
Region 192: 96.501 - 97.000  
Region 193: 97.001 - 97.500  
Region 194: 97.501 - 98.000  
Region 195: 98.001 - 98.500  
Region 196: 98.501 - 99.000  
Region 197: 99.001 - 99.500  
Region 198: 99.501 - 100.000

# What Petabytes of data feels like

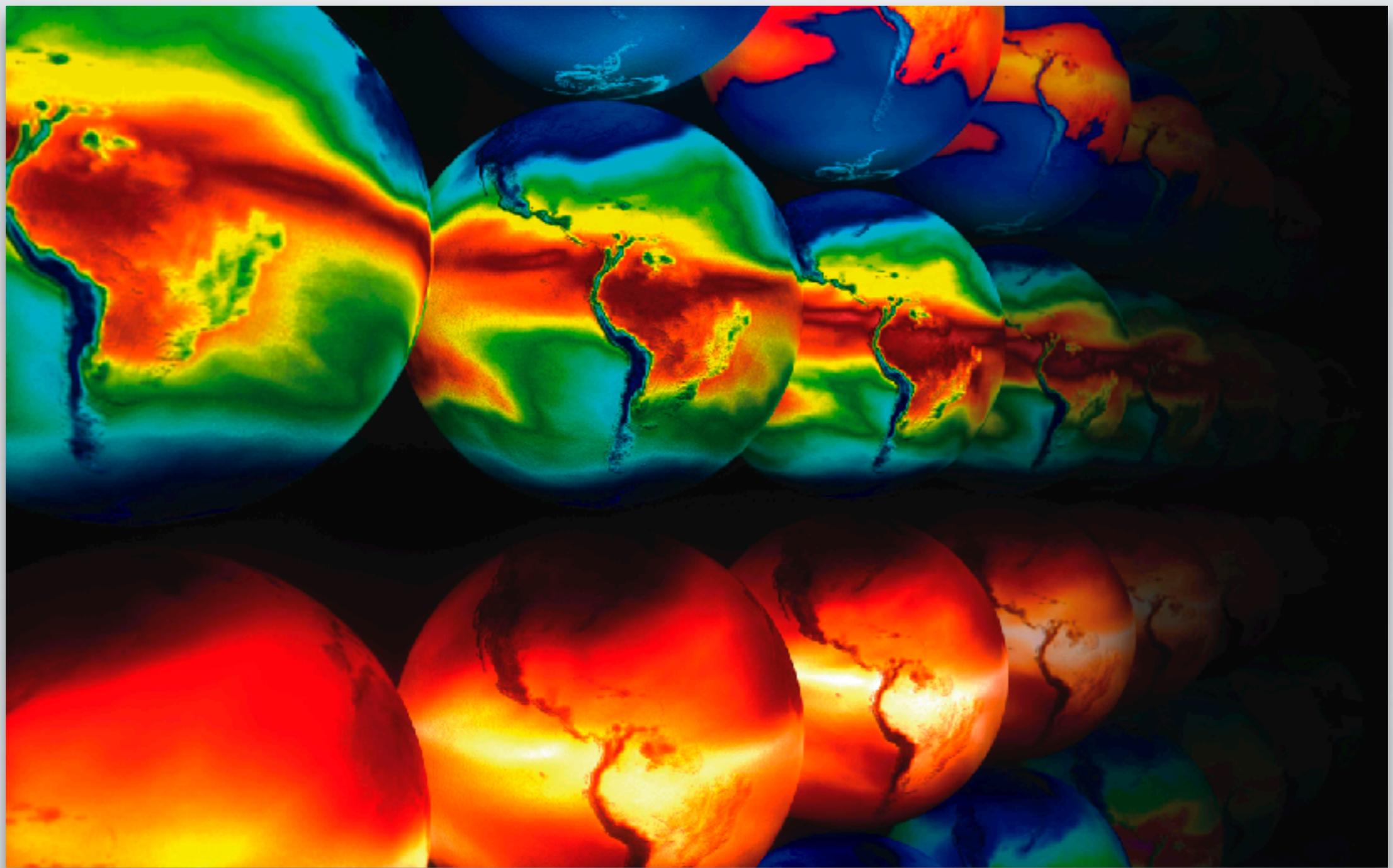


And the problem is even much  
worse with spatial data

# Example: Satellite remote sensing



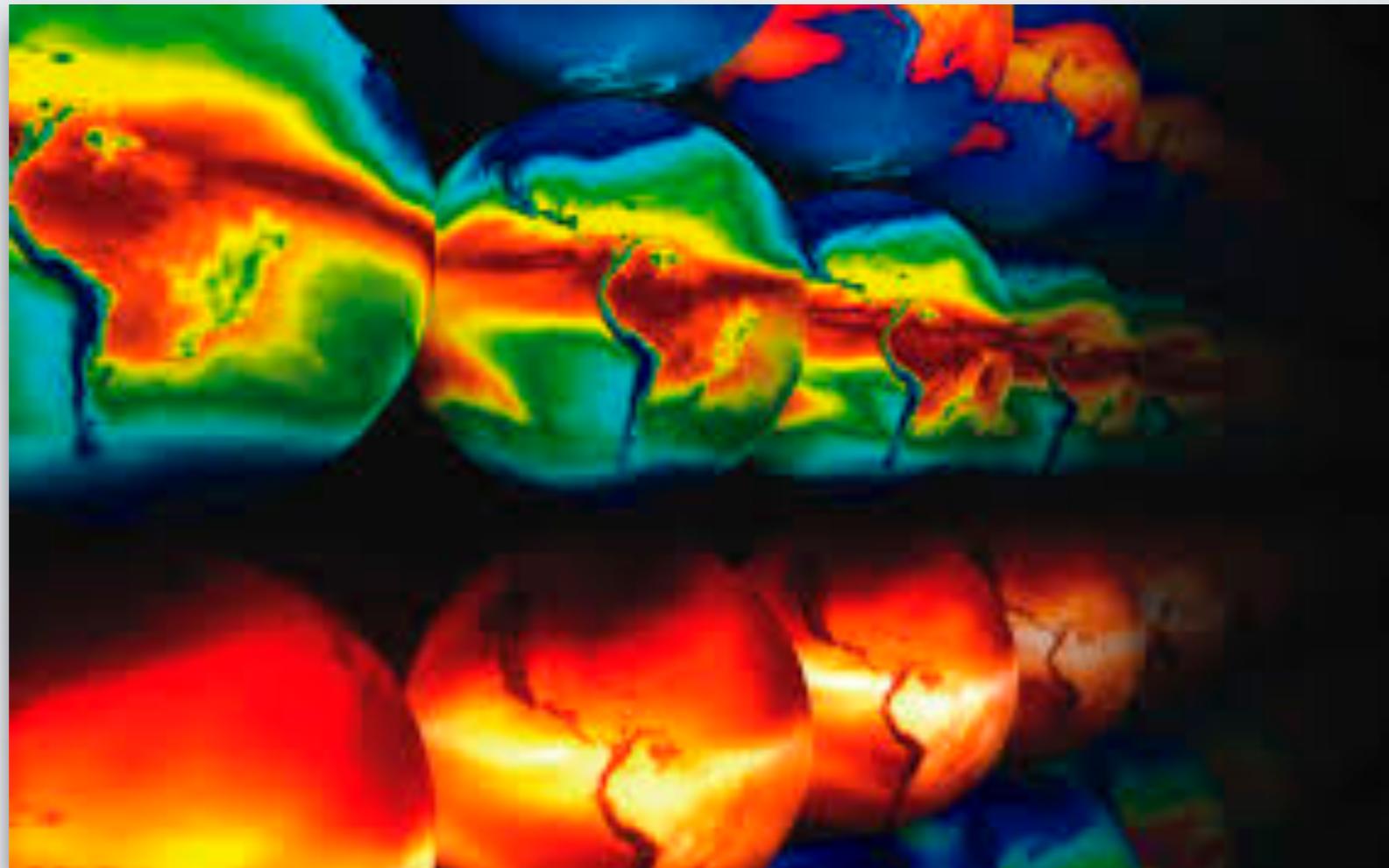
# Example: Climate model output



This course aims to give you the  
tools and knowledge on how to  
tackle these data

What topics will we cover?

# Topics: Analyze spatial environmental data



How can we analyze terabytes/petabytes of climate model output and/or satellite data?

# Topics: Map projections



Robinson

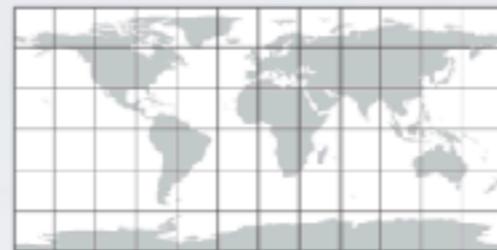


Plate Carrée



Winkel Tripel



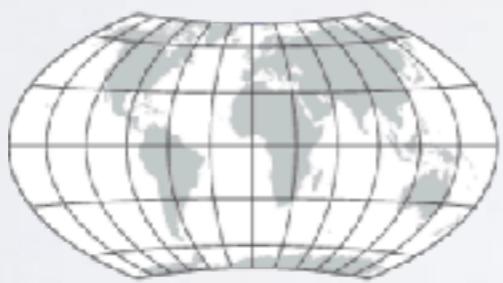
Eckert IV



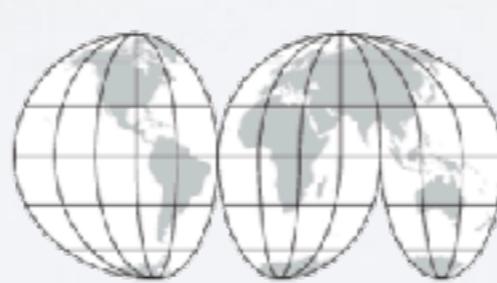
Mollweide



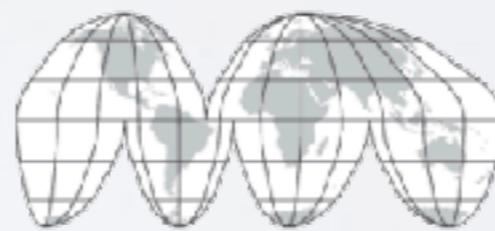
Mercator



Wagner VII



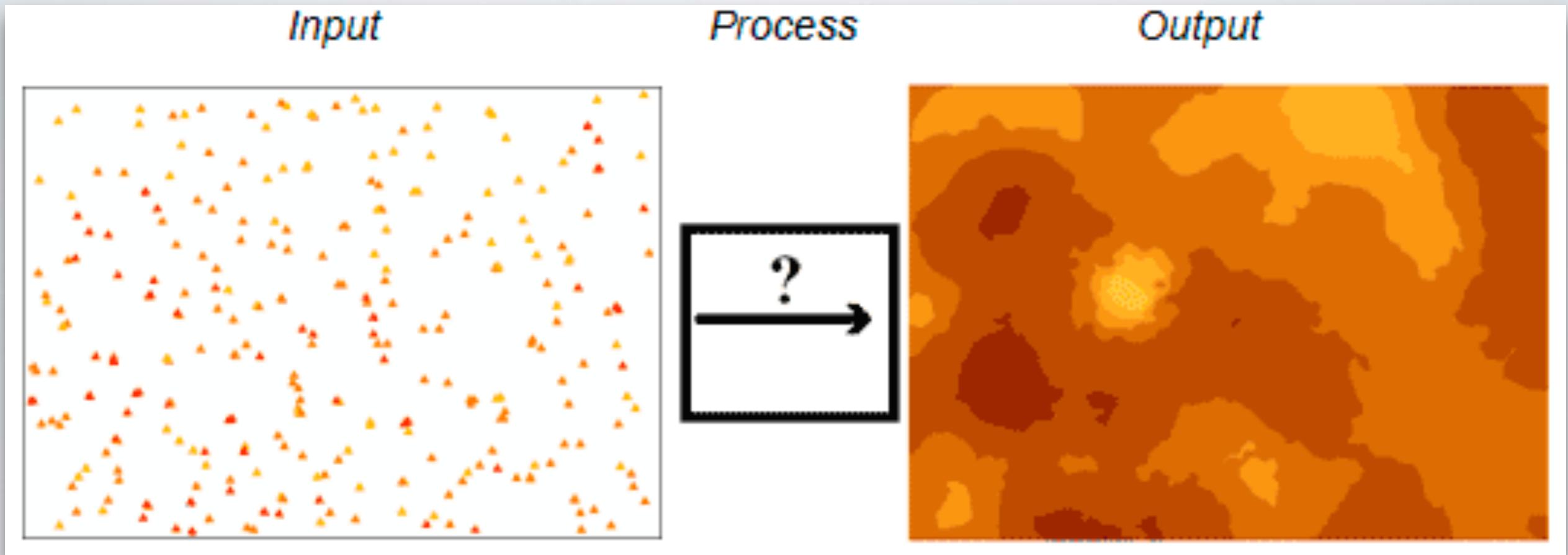
Interrupted Mollweide



Goode Homolosine

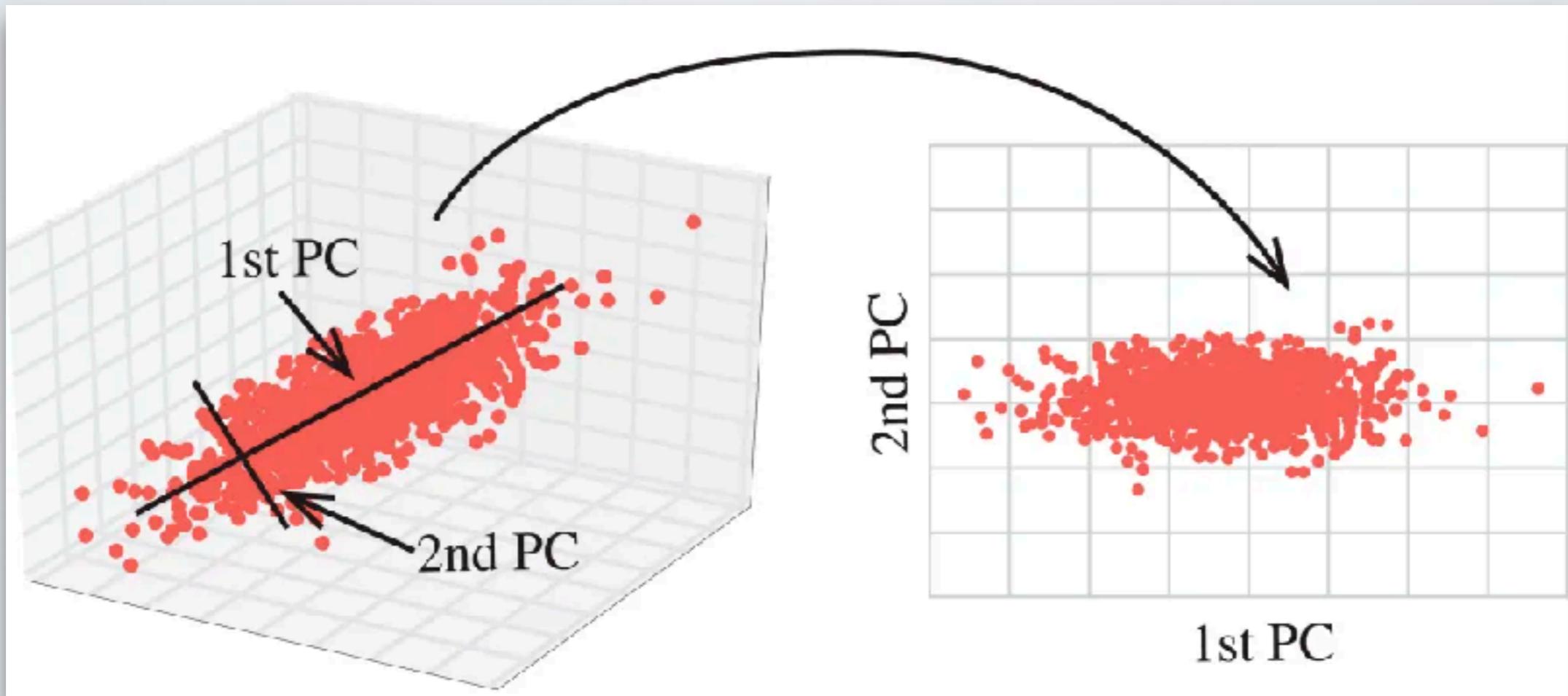
How can we transform a map to a different projection?

# Topics: Spatial interpolation/ prediction



How do you create a spatial map from point data?

# Topics: Reduce dimensions

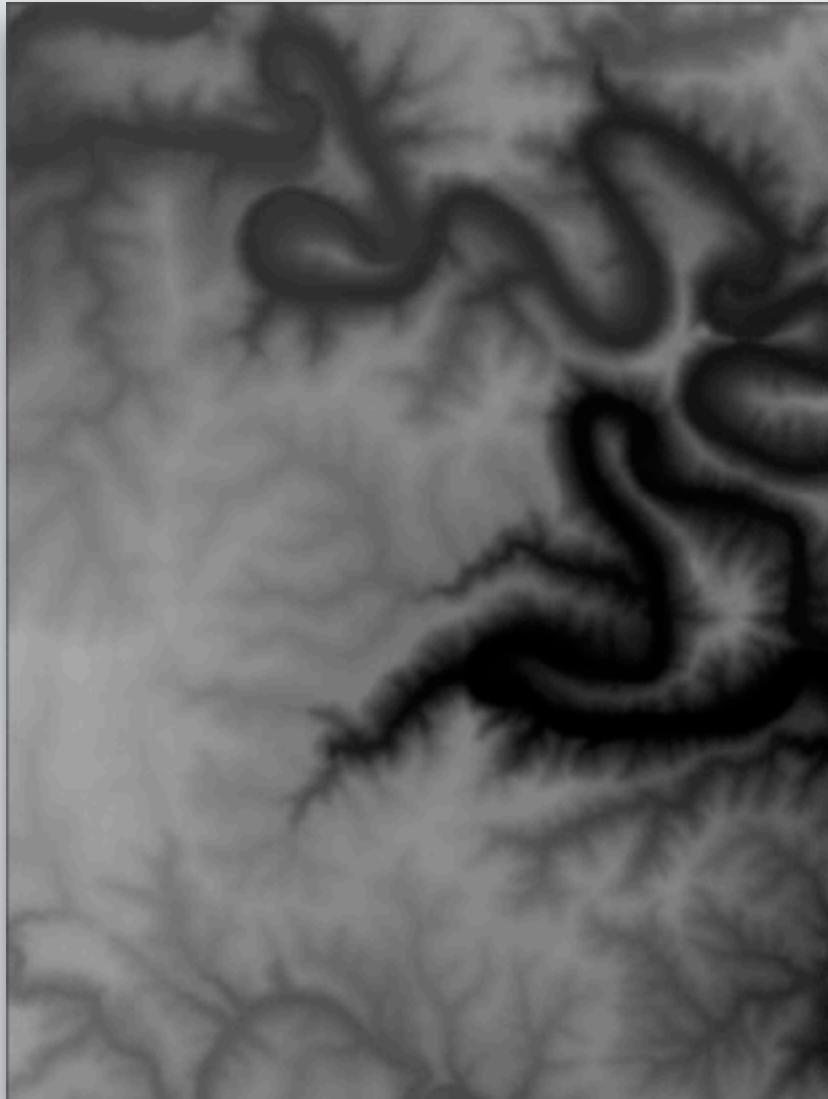


How do we compress data?

What are the most important features in the data?

# Topics: Terrain analysis

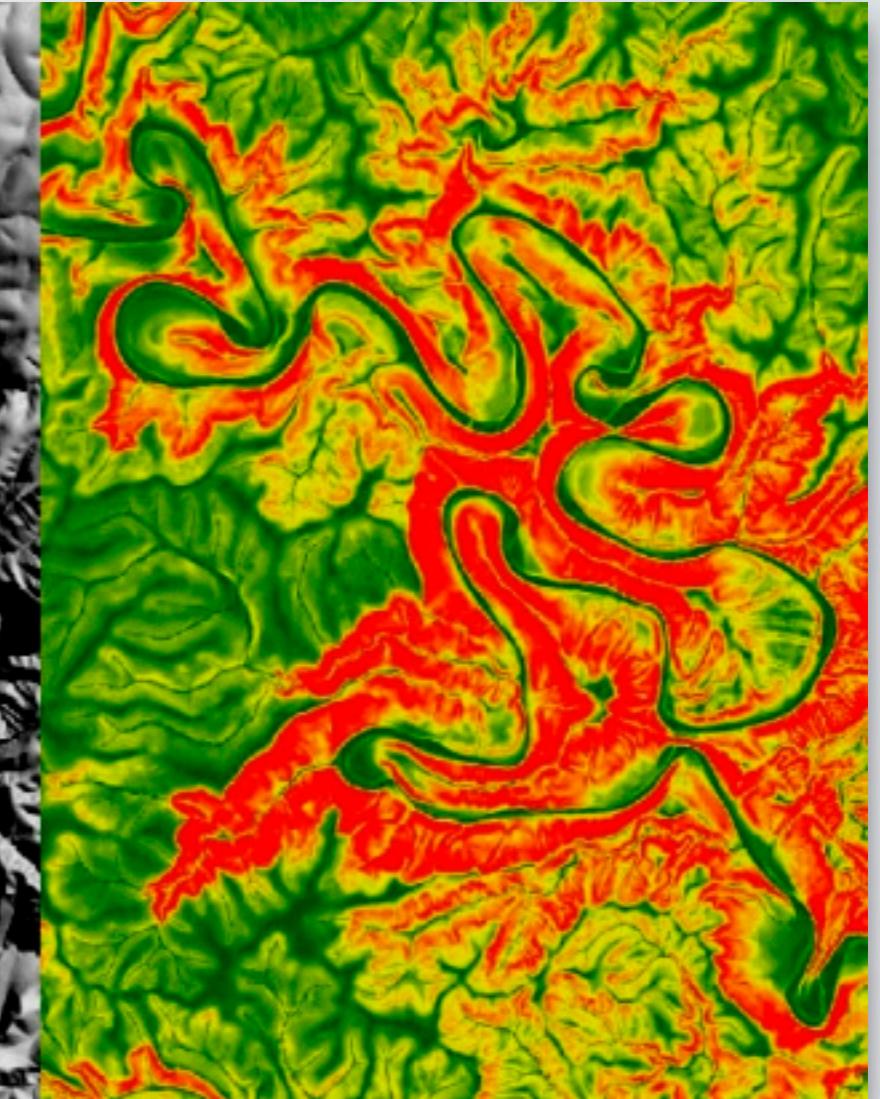
Elevation



Hillshading



Slope



What can we learn from elevation data?

And more...

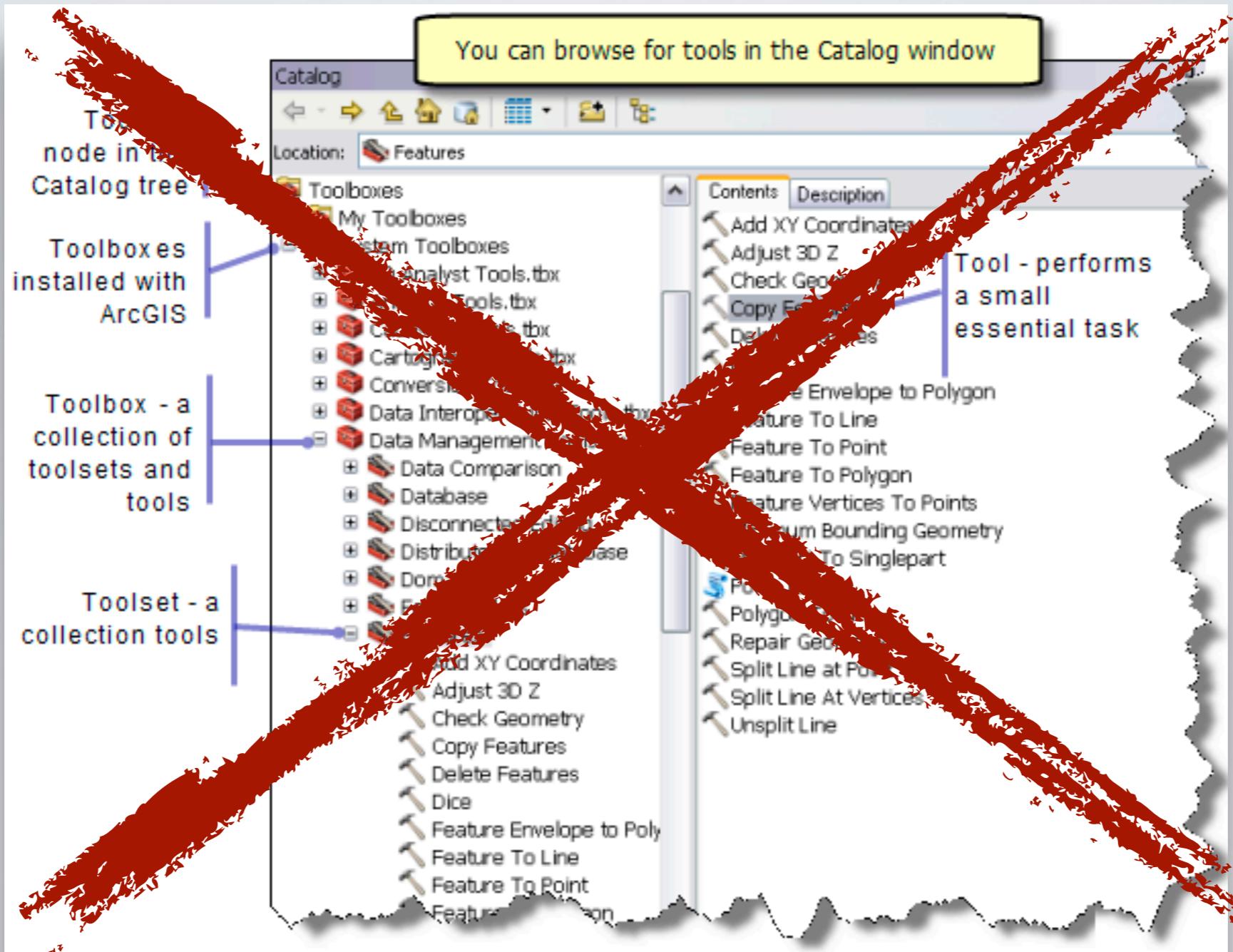
But what tools will we use to  
learn how to create and analyze  
these large datasets?

Let's start off with what we will  
NOT be using this semester

This is NOT an ArcGIS course



# This is NOT a “click the button” class



If that was what you were expecting then  
this may not be the class for you

Then what will we use?

Programming, programming, and  
more programming

If you are terrified of programming... Don't  
be! It is worth the time investment.

However, if you are not willing to dedicate  
the time and just want to “pass by” then  
this class is probably not a good fit.

Which programming languages  
will we use?

# Primary language: Python



# Python level: Intermediate

# Secondary language: R



# R level: Beginner

# Secondary language: Bash



Bash level: Beginner\*

# Don't be afraid

We are going to spend a third of the class in learning how to program in Python (and some R and Bash).

It will get hard but it will be additive so one can take this class without a background in these programming languages

# Syllabus

## Instructor

Professor Nathaniel W. Chaney (Nate)

Email: [nathaniel.chaney@duke.edu](mailto:nathaniel.chaney@duke.edu)

Office: FCIEMAS 2463

Office hours: By appointment on Friday mornings ([calendly.com/nathaniel-chaney](https://calendly.com/nathaniel-chaney))

## TA

Reza Momenifar

Email: [mohammadreza.momenifar@duke.edu](mailto:mohammadreza.momenifar@duke.edu)

Office: FCIEMAS 2421

Office hours: Mondays 8:00-10:00 AM

# Course Logistics: General Info

- Grading:
  - Homework: 40%
  - Final Project: 40%
  - Participation: 20%

Textbooks?

There are none.

Welcome to the wild, wild west!

# Homework

- Completed assignments will be submitted via each student's private GitHub repository
- Each assignment must be submitted before class on the day listed on the schedule.
- The homework with the lowest grade will be dropped
- Late homeworks will not be accepted
- Solutions to the homeworks will be provided via Sakai

We are here to help. That being said,  
Google will answer almost all questions  
much faster than we will (Use it!).



# Participation I

- Discussion/questions - Avoid the one-way lecture approach relies on your involvement
- Bring you laptop to class to follow along with the interactive lectures



# Participation II:

## Assignment review

- Beginning of next lecture after homework is due: Presentation to review the solutions from the previous homework
- Groups of students will be selected to lead these presentations (we will do the first one next week)
- Number of presenters will depend on the length and complexity of the homework

## Final Project

The final project will be done in groups of 3-4 students and will involve the following components:

- Initial Proposal (October 10th via email)
  - 1 page, Single-spaced, 12 point font size, 1 inch margin
  - Contains: Title, motivation, and sketch of proposed methodology
- Proposal presentations (November 5th)
  - 10 minute informal oral presentation
  - Open discussion/feedback in class
- Revised Proposal (November 10th via email)
  - 3 pages, Single-spaced, 12 point font size, 1 inch margin
  - Contains: Title, introduction, objectives, Methodology, and a timeline of tasks
- Oral presentation (December 3rd and 5th in class)
  - 20 minute oral presentation, 5 minutes for questions
  - Everyone needs to be present for each presentation

# Tentative schedule I

Date	Topic	New Software	Assignments	Presentation
08/27	Introduction	Jupyter/GitHub/Bash		
08/29	Python overview	Python		
09/03	Multi-dimensional arrays I	NumPy	HW #1 due	
09/05	Visualizing data I	Matplotlib/Seaborn		Reza and Nate
09/10	Raster data	Rasterio	HW #2 due	
09/12	Data storage	Pickle/H5py/NetCDF/Tiff		TBD
09/17	Multi-dimensional arrays II	Pandas/Xarray	HW #3 due	
09/19	Map projections I	GDAL		TBD
09/24	Map projections II		HW #4 due	
09/26	Visualizing data II	Cartopy		TBD

# Tentative schedule II

10/01	Spatial statistics I	Scipy	HW #5 due	
10/03	Spatial statistics II			TBD
10/08	Fall Break			
10/10	R overview	R/Rpy2	<b>Initial proposal due</b>	
10/15	Spatial statistics III	Gstat (R)	HW #6 due	
10/17	Semivariogram			TBD
10/22	Kriging			
10/24	Conditional Simulation			
10/29	Spatial prediction I	Scikit-learn		
10/31	Spatial prediction II			

# Tentative schedule III

11/05	<b>Proposal presentations</b>			
11/07	Dimensionality Reduction I		<b>Revised proposal due</b>	
11/12	Dimensionality Reduction II		<b>HW #7 due</b>	
11/14	Terrain Analysis I	<b>TBD</b>		<b>TBD</b>
11/19	Terrain Analysis II			
11/21	Vector data	<b>Shapely</b>		
11/26	Speed up code	<b>Numba/Mpi4py/Dask</b>		
11/28	Thanksgiving Break			
12/03	<b>Oral Presentations</b>			
12/05	<b>Oral Presentations</b>			
12/16	<b>Written report due</b>		<b>HW #8 due</b>	

# Docker container

All work in this class will be done online on your own personal Docker container

The screenshot shows a web browser window with the URL [vm-manage.oit.duke.edu](https://vm-manage.oit.duke.edu) in the address bar. The page has a dark blue header with the Duke University logo and the text "Duke | VM Manage". On the right side of the header are links for "Home", "Docker", and "Logout". The main content area is white. On the left, there's a section titled "About VM-Manage" with text about the service and a link to the Virtual Computing Manager. On the right, there's a section titled "Docker containers" with text about reserving personal computer environments and a link to access them. At the bottom, there's a footer with the Duke University logo, copyright information, and links to Accessibility, Style Guide, About Duke.edu, Contact Us, and social media icons.

About VM-Manage

VM-Manage is a reservation service for compute environments such as RStudio, Eclipse, Jupyter, Spyder+Anaconda Python and other course-oriented Docker containers.

For virtual machine reservations visit the new **Virtual Computing Manager** service at: <https://vcm.duke.edu>

Docker containers

Students and instructors can reserve personal computer environments running applications such as RStudio, the Eclipse IDE, Jupyter Notebooks, Matlab, Octave, Mathematica, and others for a semester.

[Click here](#) to access your personal Docker containers.

**Duke**  
UNIVERSITY

© 2012-2016 Duke University, Durham, NC 27708 | [Accessibility](#) | [Style Guide](#) | [About Duke.edu](#) | [Contact Us](#) |

# Look for CEE690-02

## Docker Containers

Welcome back Nathaniel W Chaney!

Your reserved application containers:

### KerasTensorflow - Jupyter with Keras and Tensorflow-GPU

Your KerasTensorflow reservation has expired - container reservations are valid for one semester.

[Click here to renew your reservation](#)

### MATLAB - mathematics and matrix manipulations

Your MATLAB reservation has expired - container reservations are valid for one semester.

[Click here to renew your reservation](#)

### CEE690 - Environmental Spatial Data Analysis

[Click here to log in](#) to your CEE690 environment

If you are unable to connect to your CEE690 environment you can [request a restart](#).

### Spark - PySpark notebooks for Apache Spark cluster

Your Spark reservation has expired - container reservations are valid for one semester.

[Click here to renew your reservation](#)

# Look for CEE690-02

## Docker Containers

Welcome back Nathaniel W Chaney!

Your reserved application containers:

### KerasTensorflow - Jupyter with Keras and Tensorflow-GPU

Your KerasTensorflow reservation has expired - container reservations are valid for one semester.

[Click here to renew your reservation](#)

### MATLAB - mathematics and matrix manipulations

Your MATLAB reservation has expired - container reservations are valid for one semester.

[Click here to renew your reservation](#)

### CEE690 - Environmental Spatial Data Analysis

[Click here to log in](#) to your CEE690 environment

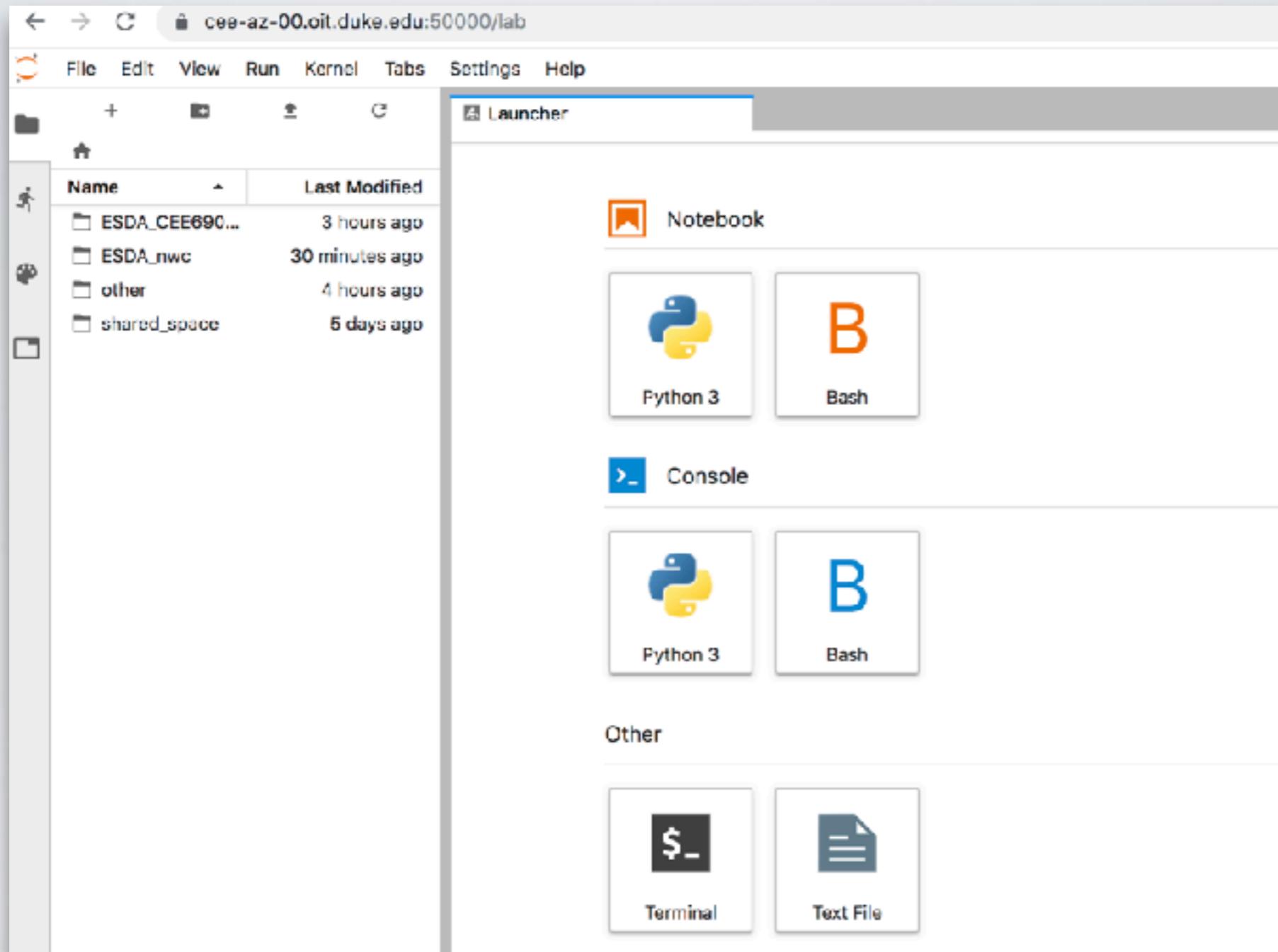
If you are unable to connect to your CEE690 environment you can [request a restart](#).

### Spark - PySpark notebooks for Apache Spark cluster

Your Spark reservation has expired - container reservations are valid for one semester.

[Click here to renew your reservation](#)

# Jupyter Lab

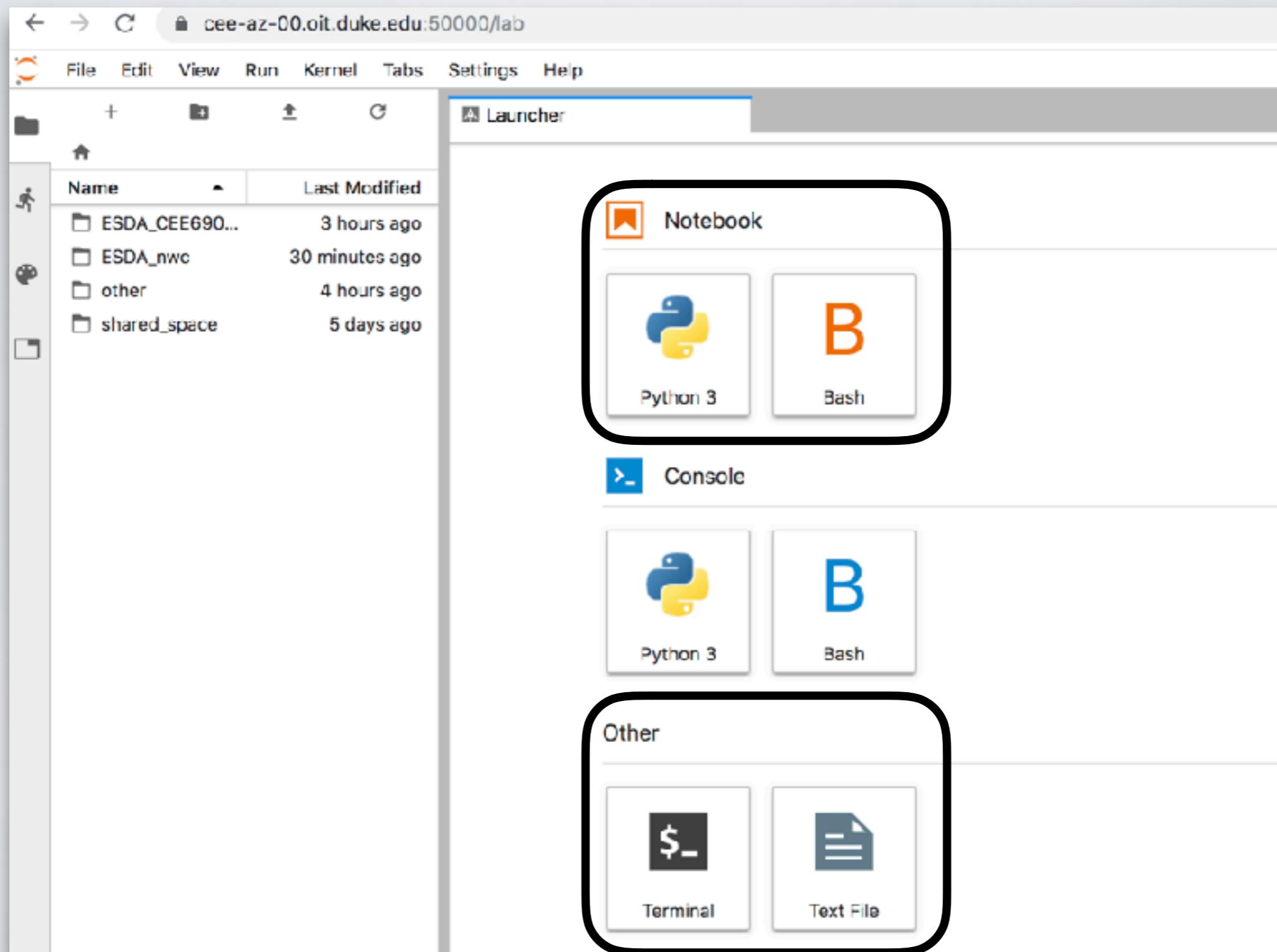


<https://jupyterlab.readthedocs.io/en/stable/>

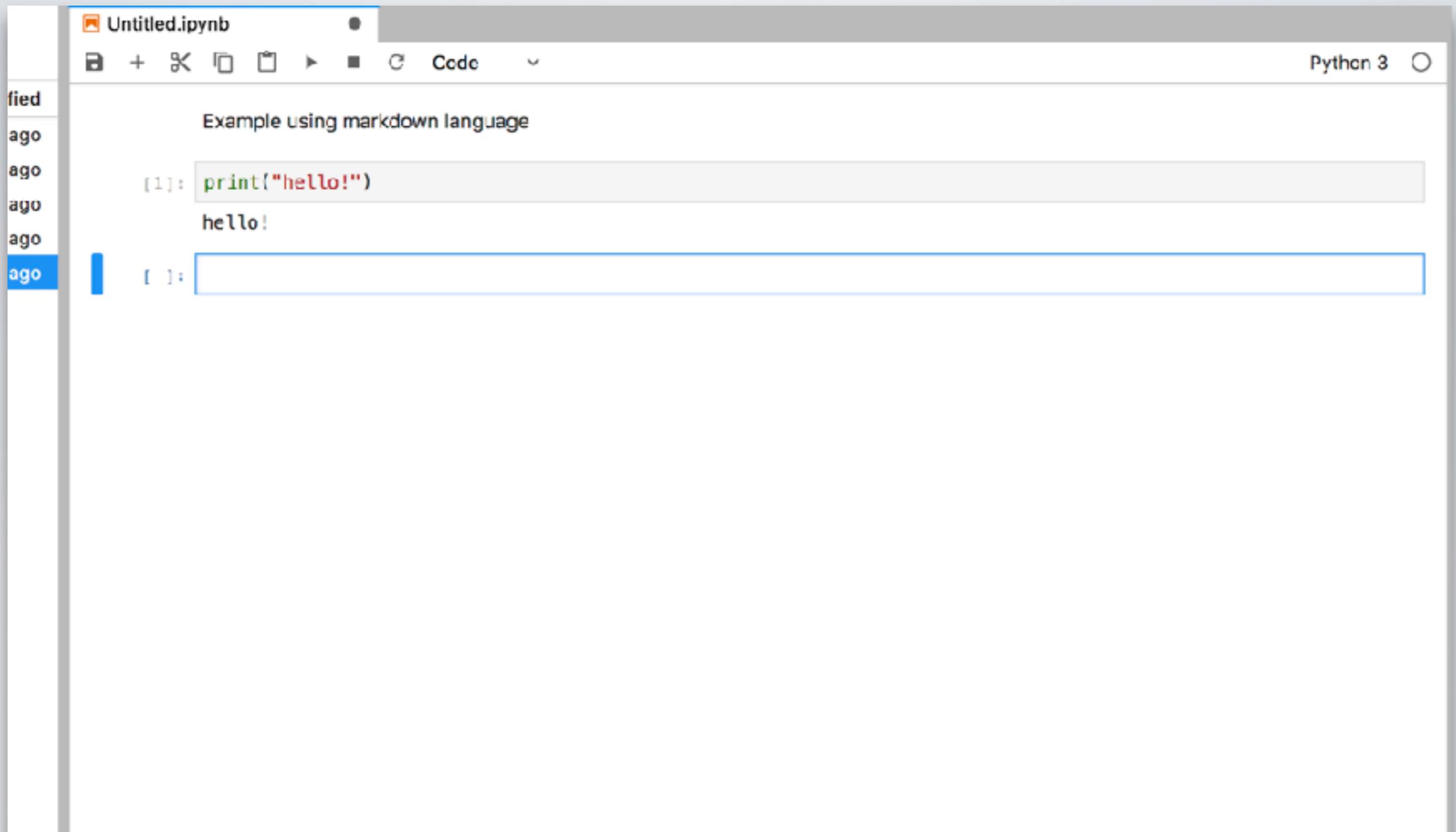
# A way to understand this...

- The Docker container is your own computer (virtual machine)
- Jupyter lab is a way to interact with that computer.
- This is the easiest way to do this. There are certainly more complex ways (and more powerful) to connect to a virtual computer but for this course we will only use Jupyter lab (and Jupyter notebooks).

# Components in Jupyter Lab



# Jupyter Notebook



A screenshot of a Jupyter Notebook interface. The title bar shows "Untitled.ipynb". The toolbar includes icons for file operations like new, open, save, and run, along with a "Code" dropdown and a "Python 3" kernel selector. The main area displays a cell containing a markdown example and a code cell with an output. The code cell contains the Python command `print("hello!")`, which has been executed and printed the word "hello!". A new cell is currently being edited, indicated by a blue selection bar.

```
Example using markdown language
[1]: print("hello!")
hello!
[ ]:
```

[https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what\\_is\\_jupyter.html](https://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/what_is_jupyter.html)

# Jupyter environment (older view)



You can do the same thing with the older view as with Jupyter lab. Jupyter lab is just more convenient and “prettier”. We will need to use the older view many times though since Jupyter Lab doesn't have some functionality that the older view does.

# Terminal



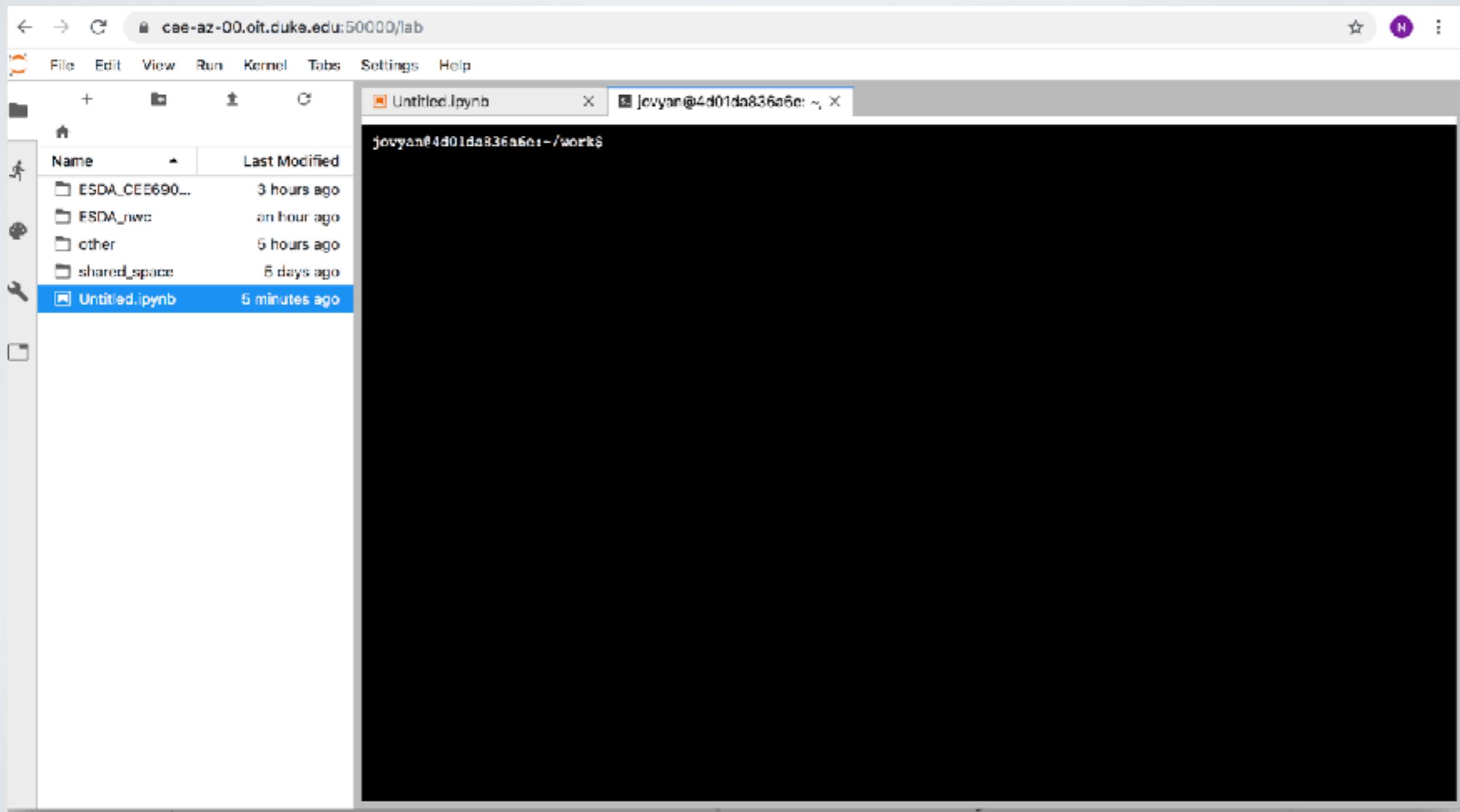
Remains one of the best ways to interact more directly with the computer (flexibility!)

# Linux Ubuntu: The Docker container's operating system under the hood



**So you will actually be using Ubuntu throughout the semester (you just won't really even notice)**

# Terminal in Jupyter lab (i.e., Ubuntu terminal)



# Interacting with the terminal: Bash (Bourne Again SHell)



# Tutorials to learn Bash

<https://linuxconfig.org/bash-scripting-tutorial-for-beginners>

<https://www.codecademy.com/learn/learn-the-command-line>

<https://devhints.io/bash>

<https://hackernoon.com/top-10-bash-file-system-commands-you-can't-live-without-4cd937bd7df1>

# Top bash commands

- 1) **ls** - Lists the folder and file names in the current working directory.
- 2) **cd** - Change Directory to the path specified, for example cd projects
- 3) **mkdir** - Make directories with this command mkdir my\_folder
- 4) **mv** - Moves files and folders. The first argument is the file you want to move, and the second is the location to move it to.
- 5) **cp** - Copies files and folders cp my\_file ./projects . The flag -r recursively copies subfolders and files.
- 6) **rm** - Removes files and folders rm my\_folder . Using -r will again recursively delete subfolders, -f force deletes, and -rf for a recursive force delete.

You will be doing all of your work on your Docker container. But how will you access the assignments and how will you submit them?

Let's talk about version control...

# GitHub: Git on the web



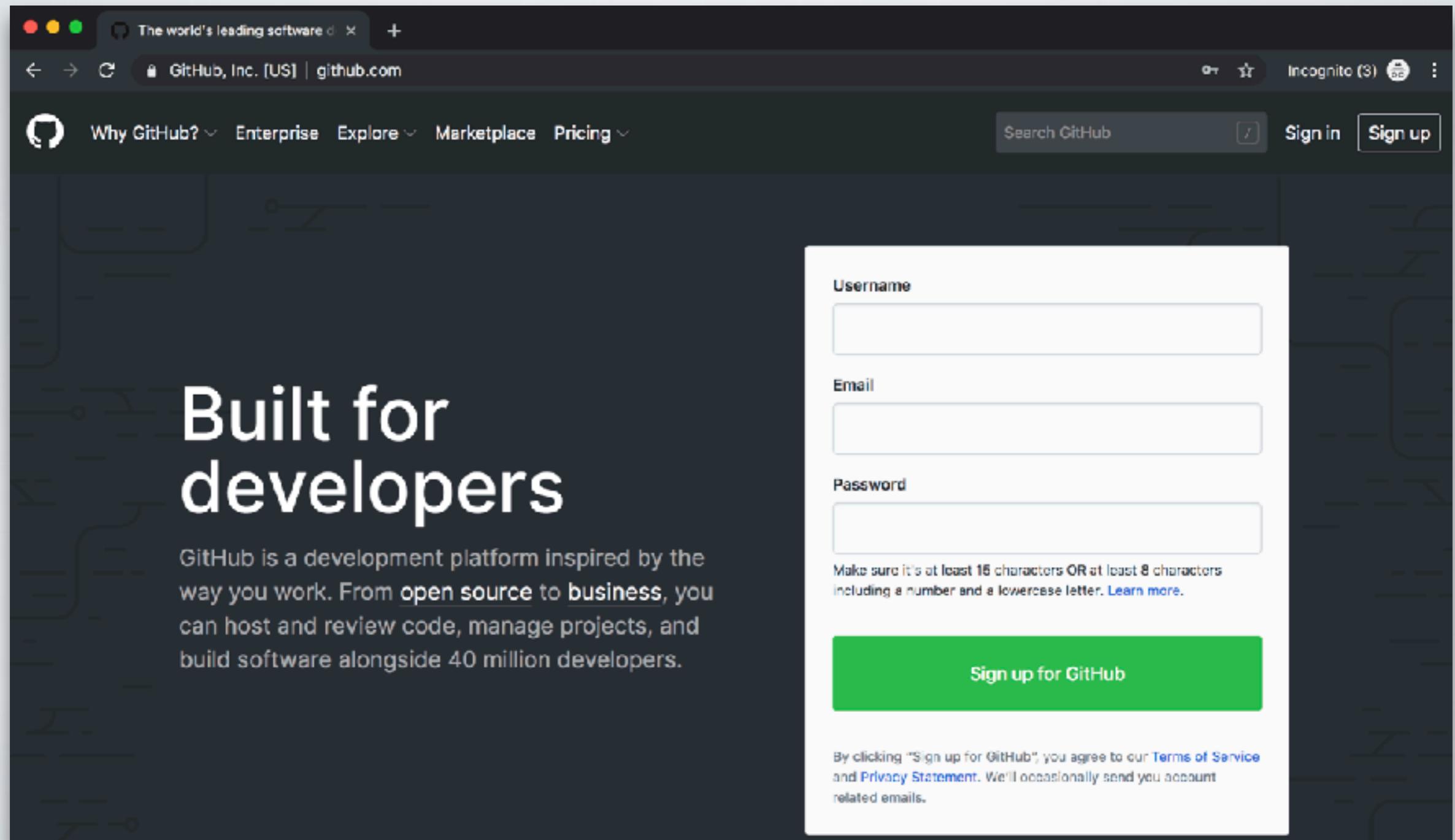
[https://www.youtube.com/watch?](https://www.youtube.com/watch?v=w3jLJU7DT5E)  
[v=w3jLJU7DT5E](https://www.youtube.com/watch?v=w3jLJU7DT5E)

# Version control: Git

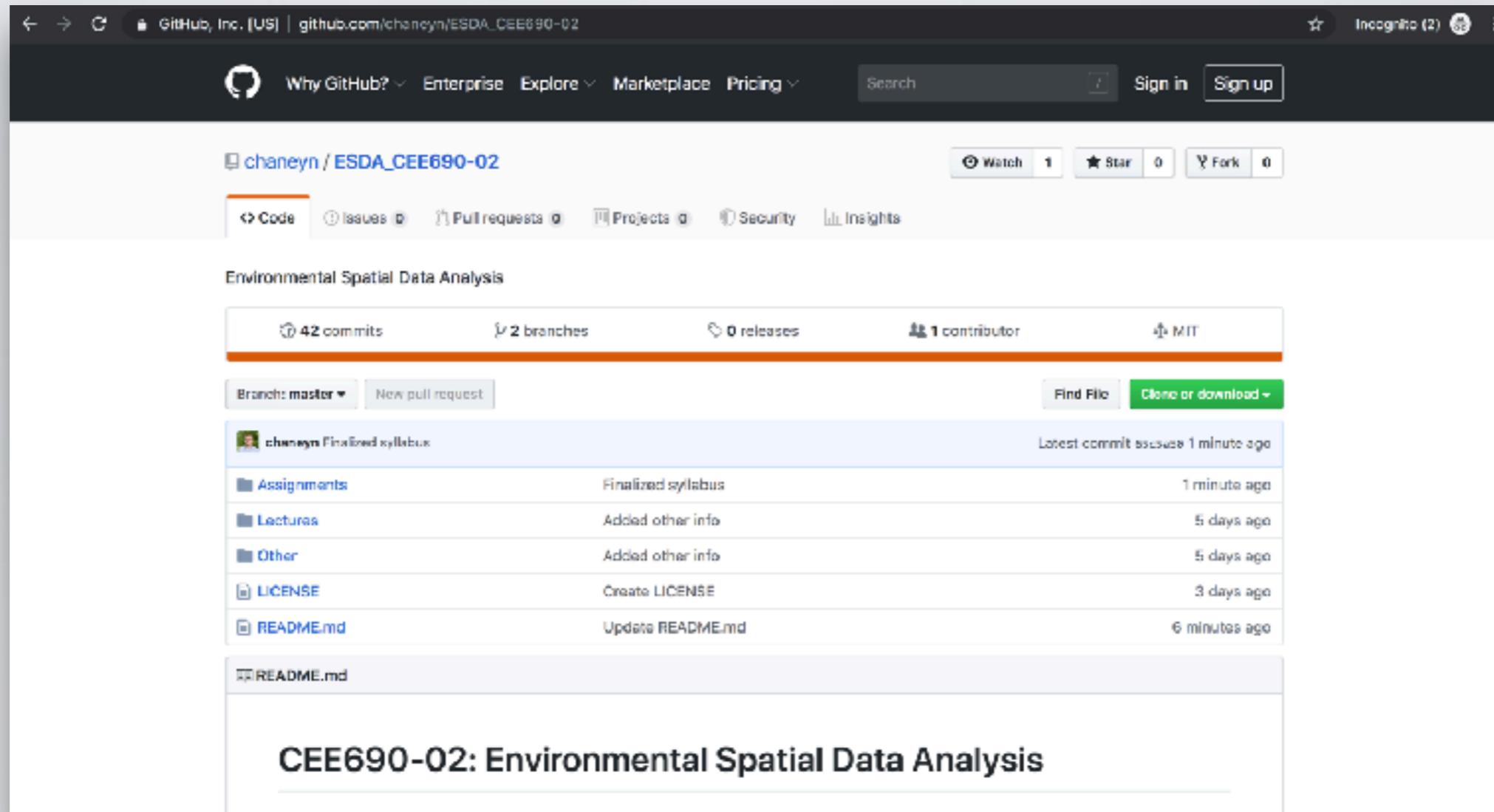


<https://vimeo.com/41027679>

# Create a personal account on GitHub



# The primary class website for this course is on GitHub



[https://github.com/chaneyn/ESDA\\_CEE690-02](https://github.com/chaneyn/ESDA_CEE690-02)

# All lectures, assignments, and materials will be on the class' GitHub repository

The screenshot shows a GitHub repository page for 'chaneyn / ESDA\_CEE690-02'. The repository name is 'Environmental Spatial Data Analysis'. Key statistics displayed include 42 commits, 2 branches, 0 releases, 1 contributor, and an MIT license. The commit history lists several files: 'Finalized syllabus', 'Assignments', 'Lectures', 'Other', 'LICENSE', and 'README.md'. The most recent commit was made 1 minute ago. A large banner at the bottom of the repository page reads 'CEE690-02: Environmental Spatial Data Analysis'.

File	Commit Message	Time Ago
Finalized syllabus	Finalized syllabus	1 minute ago
Assignments	Added other info	6 days ago
Lectures	Added other info	5 days ago
Other	Create LICENSE	3 days ago
LICENSE	Update README.md	6 minutes ago
README.md		

\*Solutions to assignments will be posted on Sakai

# The class syllabus is on the GitHub repository

A screenshot of a GitHub repository page for 'CEE690-02: Environmental Spatial Data Analysis'. The repository has three recent commits:

- Added other info (5 days ago)
- Create LICENSE (3 days ago)
- Update README.md (7 minutes ago)

The README.md file is open, displaying the following content:

## CEE690-02: Environmental Spatial Data Analysis

---

### Fall 2019

---

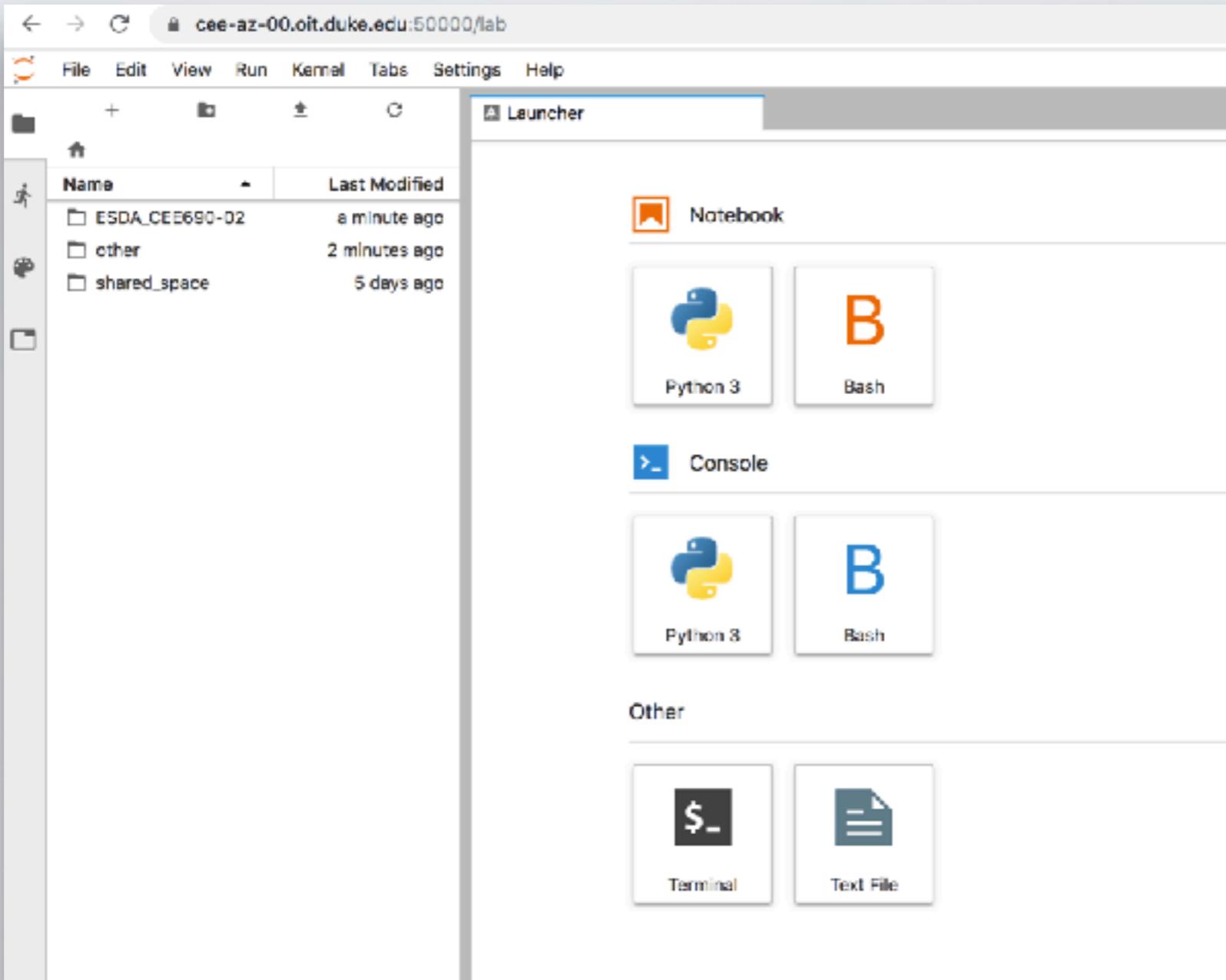
#### Course Information

Lectures are on Tuesdays and Thursdays from 3:05 PM - 4:20 PM in Teer 106. The course website is on GitHub (<https://github.com/chaneyn/CEE690-02>). Class announcements will be made via Sakai (CEE690.02.F19).

#### Instructor

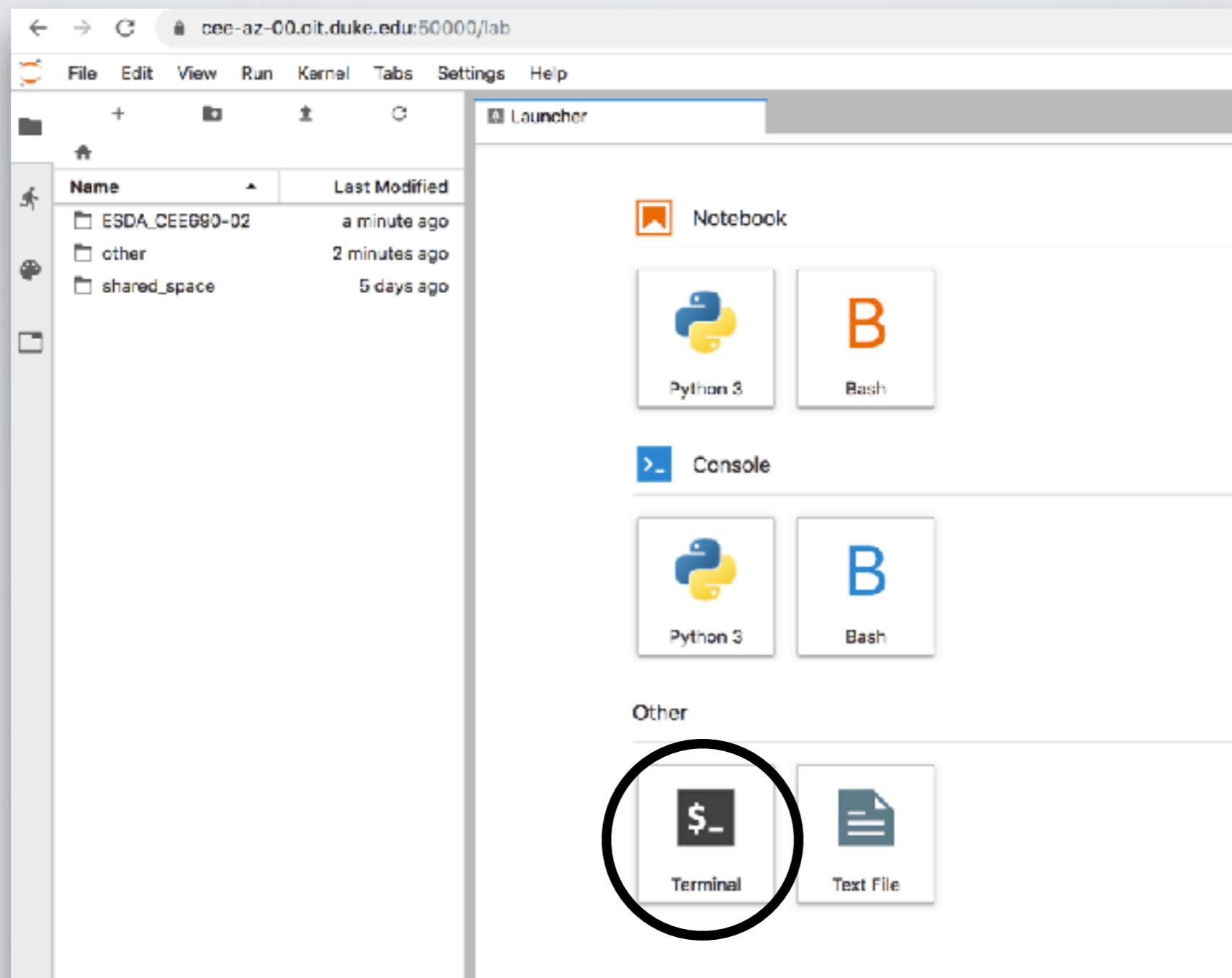
Professor Nathaniel W. Chaney (Nate)  
Email: [nathaniel.chaney@duke.edu](mailto:nathaniel.chaney@duke.edu)  
Office: FCIEMAS 2463  
Office hours: By appointment on Friday mornings ([calendly.com/nathaniel-chaney/](https://calendly.com/nathaniel-chaney/))

# Getting started: Clone the course's GitHub repository



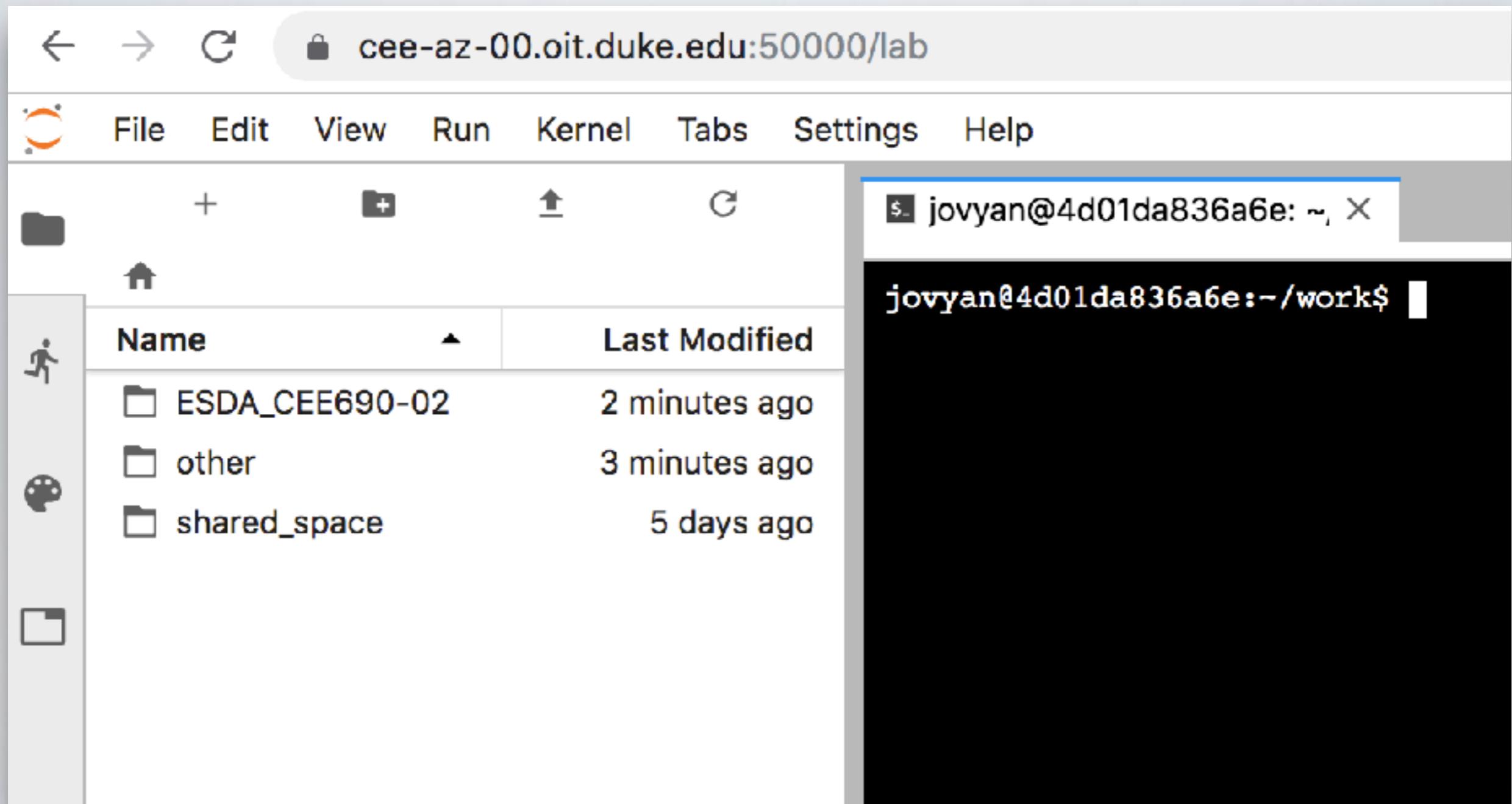
Navigate to your Docker container instance

# Open a terminal !

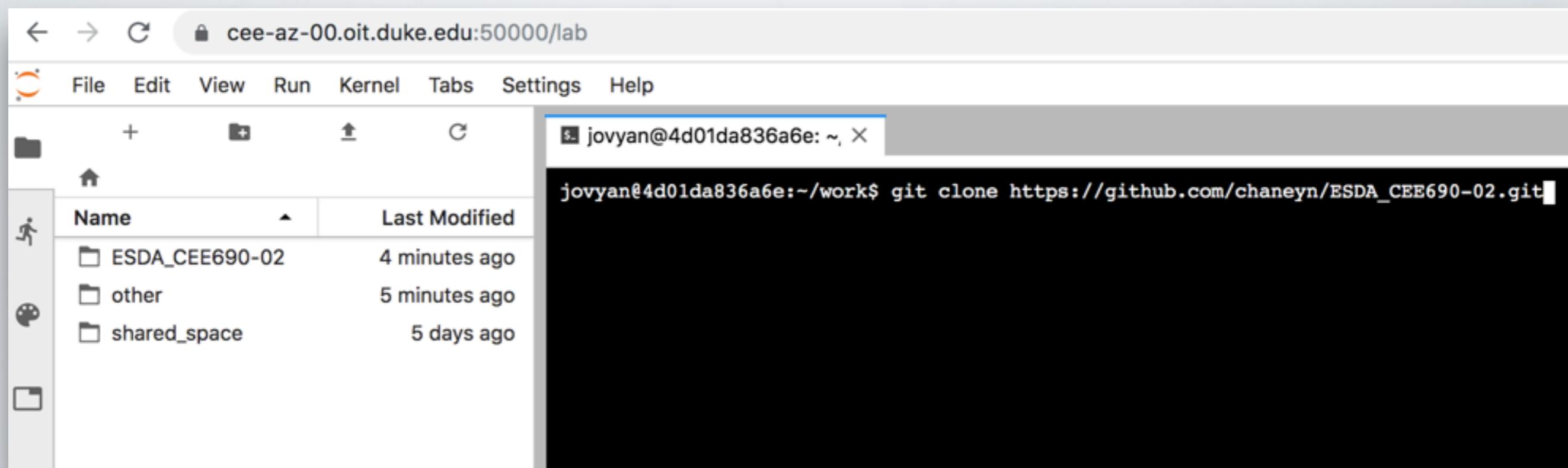


Press the terminal button

# Open a terminal II



# Write the following git command

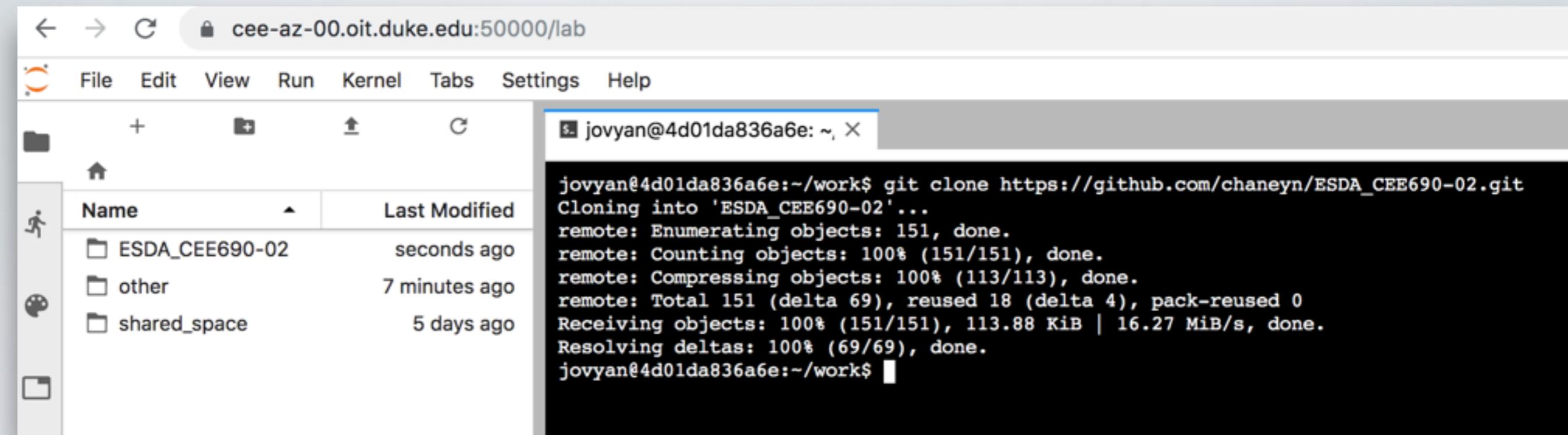


A screenshot of a terminal window titled "jovyan@4d01da836a6e: ~, X". The window shows a command being typed: "jovyan@4d01da836a6e:~/work\$ git clone https://github.com/chaneyn/ESDA\_CEE690-02.git". The terminal is part of a desktop environment with a file manager sidebar on the left.

Name	Last Modified
ESDA_CEE690-02	4 minutes ago
other	5 minutes ago
shared_space	5 days ago

Then press enter

# You will see something like this



The screenshot shows a Jupyter Notebook interface. The top bar includes navigation icons, a URL field with "cee-az-00.oit.duke.edu:50000/lab", and a menu bar with File, Edit, View, Run, Kernel, Tabs, Settings, and Help. On the left is a sidebar with icons for Home, User, Profile, and Help. Below the menu is a file browser with a list of files:

Name	Last Modified
ESDA_CEE690-02	seconds ago
other	7 minutes ago
shared_space	5 days ago

The main area contains a terminal window with the following output:

```
jovyan@4d01da836a6e:~/work$ git clone https://github.com/chaneyn/ESDA_CEE690-02.git
Cloning into 'ESDA_CEE690-02'...
remote: Enumerating objects: 151, done.
remote: Counting objects: 100% (151/151), done.
remote: Compressing objects: 100% (113/113), done.
remote: Total 151 (delta 69), reused 18 (delta 4), pack-reused 0
Receiving objects: 100% (151/151), 113.88 KiB | 16.27 MiB/s, done.
Resolving deltas: 100% (69/69), done.
jovyan@4d01da836a6e:~/work$
```

Congratulations you have cloned your first repository!

# Explore the cloned directory

The screenshot shows a Jupyter Notebook interface. The top navigation bar includes back, forward, and refresh buttons, followed by the URL "cee-az-00.oit.duke.edu:50000/lab". Below the URL is a menu bar with File, Edit, View, Run, Kernel, Tabs, Settings, and Help. To the left is a sidebar with icons for user profile, course information, and file operations. The main area has two tabs: "jovyan@4d01da836a6e: ~" and "README.md". The "README.md" tab is active, displaying a Markdown file with course details.

```
# CEE690-02: Environmental Spatial Data Analysis
## Fall 2019

## Course Information
Lectures are on Tuesdays and Thursdays from 3:05 PM – 4:20 PM (https://github.com/chaneyn/CEE690-02). Class announcements will be posted here.

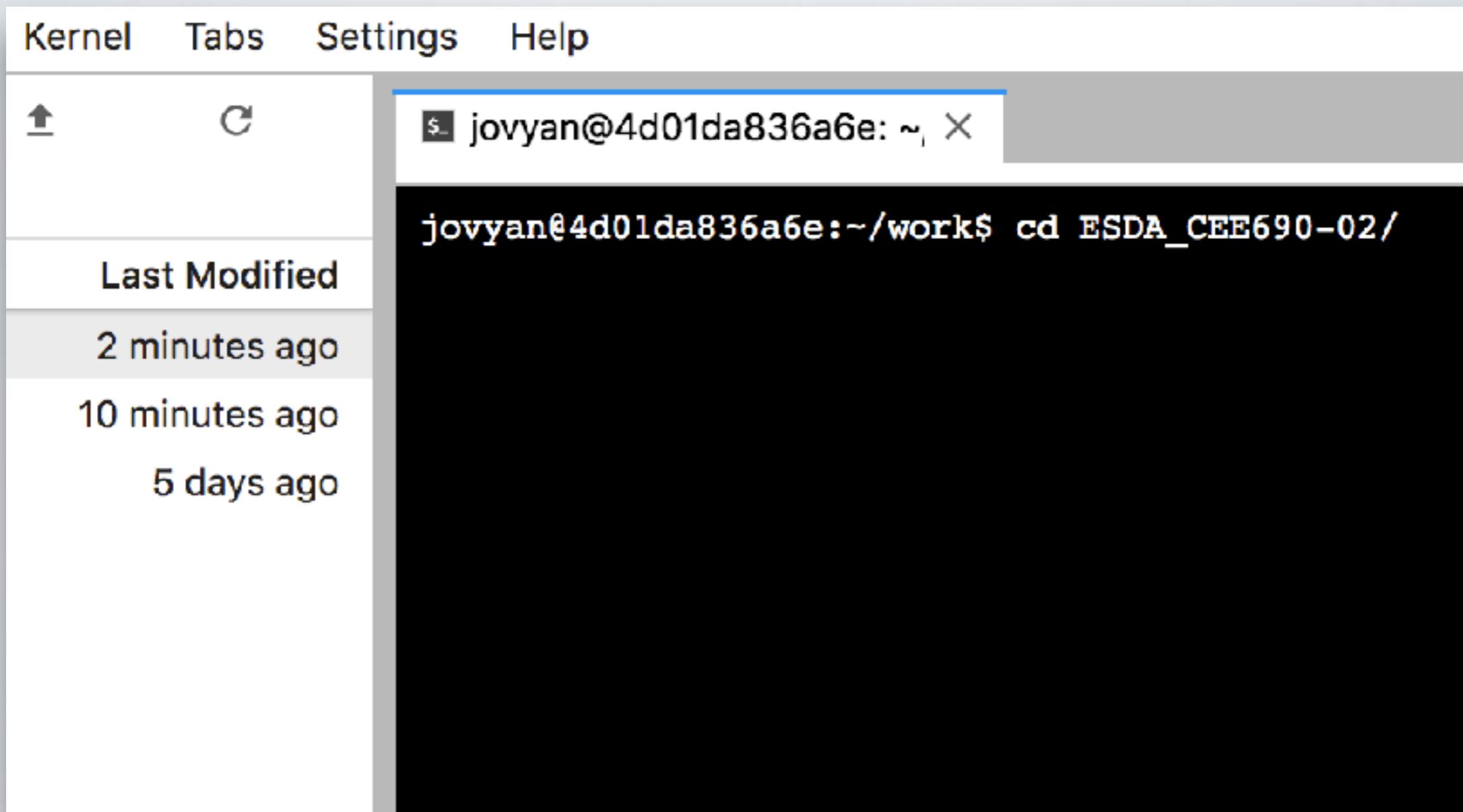
## Instructor
Professor Nathaniel W. Chaney (Nate)
Email: nathaniel.chaney@duke.edu
Office: FCIEMAS 2463
Office hours: By appointment on Friday mornings (calendly.co/nathanielchaney)

## TA
Reza Momenifar
Email: mohammadreza.momenifar@duke.edu
Office: FCIEMAS 2421
Office hours: Mondays 8:00-10:00 AM

## Course Description
This course provides an introduction on how to analyze and visualize spatial data using Python and R. The topics that will be covered include an overview of spatial data structures, conditional simulation, terrain analysis, dimensionality reduction, and machine learning. Python and R will be introduced and used to explore the listed topics.

## Prerequisites
```

# Change directory at the terminal



A screenshot of a terminal window. The window has a menu bar with "Kernel", "Tabs", "Settings", and "Help". Below the menu is a toolbar with an "Up" arrow icon and a "C" icon. A sidebar on the left lists file or directory entries with "Last Modified" headers: "2 minutes ago", "10 minutes ago", and "5 days ago". The main terminal area shows a command line with the prompt "\$ jovyen@4d01da836a6e: ~, X". Below the prompt, the command "jovyen@4d01da836a6e:~/work\$ cd ESDA\_CEE690-02/" is typed. The terminal background is black, and the text is white.

Then press enter

# Git pull

The screenshot shows a terminal window with a light gray header bar containing a small circular icon and the text "jovyan@4d01da836a6e: ~, X". The main body of the terminal is black and contains white text. It shows a command-line session:

```
jovyan@4d01da836a6e:~/work$ cd ESDA_CEE690-02/
jovyan@4d01da836a6e:~/work/ESDA_CEE690-02$ git pull
Already up to date.
jovyan@4d01da836a6e:~/work/ESDA_CEE690-02$
```

To the left of the terminal window, there is a vertical sidebar with a light gray background. It has a header "C" and four items listed under "Last Modified":

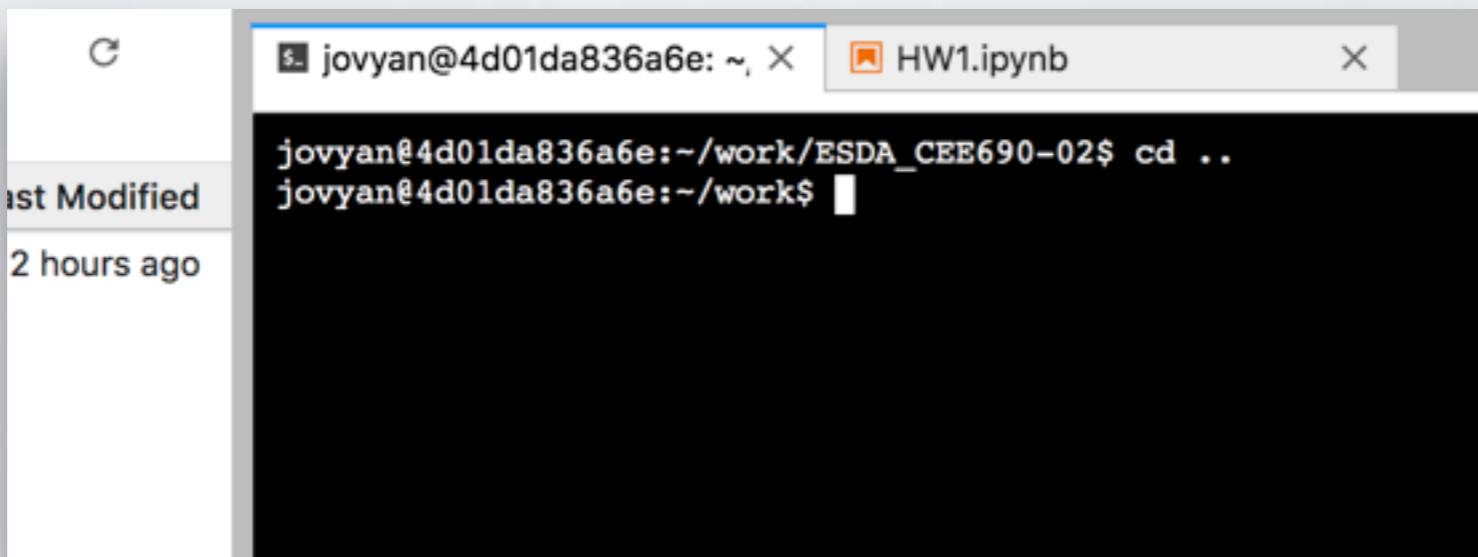
- 3 minutes ago
- 1 minutes ago
- 5 days ago

Press enter after writing “git pull”

This is how you will have update the class repository on your system (i.e., as lectures and assignments are added online)

# Make a GitHub repository to turn in your assignments

**Return to the previous directory**



A screenshot of a terminal window titled "jovyan@4d01da836a6e: ~, X". The window shows the command "jovyan@4d01da836a6e:~/work/ESDA\_CEE690-02\$ cd .." being typed. The terminal is running in a Jupyter Notebook environment, as indicated by the ".ipynb" icon in the title bar. The background of the slide features a blurred version of this terminal window.

**Create a directory (use your own  
initials)**

```
jovyan@4d01da836a6e:~/work$ mkdir ESDA_nwc
```

# Enter directory

```
jovyan@4d01da836a6e:~/work$ cd ESDA_nwc/
```

# Create git repository

```
jovyan@4d01da836a6e:~/work/ESDA_nwc$ git init  
Initialized empty Git repository in /home/jovyan/work/ESDA_nwc/.git/  
jovyan@4d01da836a6e:~/work/ESDA_nwc$
```

# Copy over assignment number 1

```
e:~/work/ESDA_nwc$ cp ../ESDA_CEE690-02/Assignments/HW1.ipynb .
```

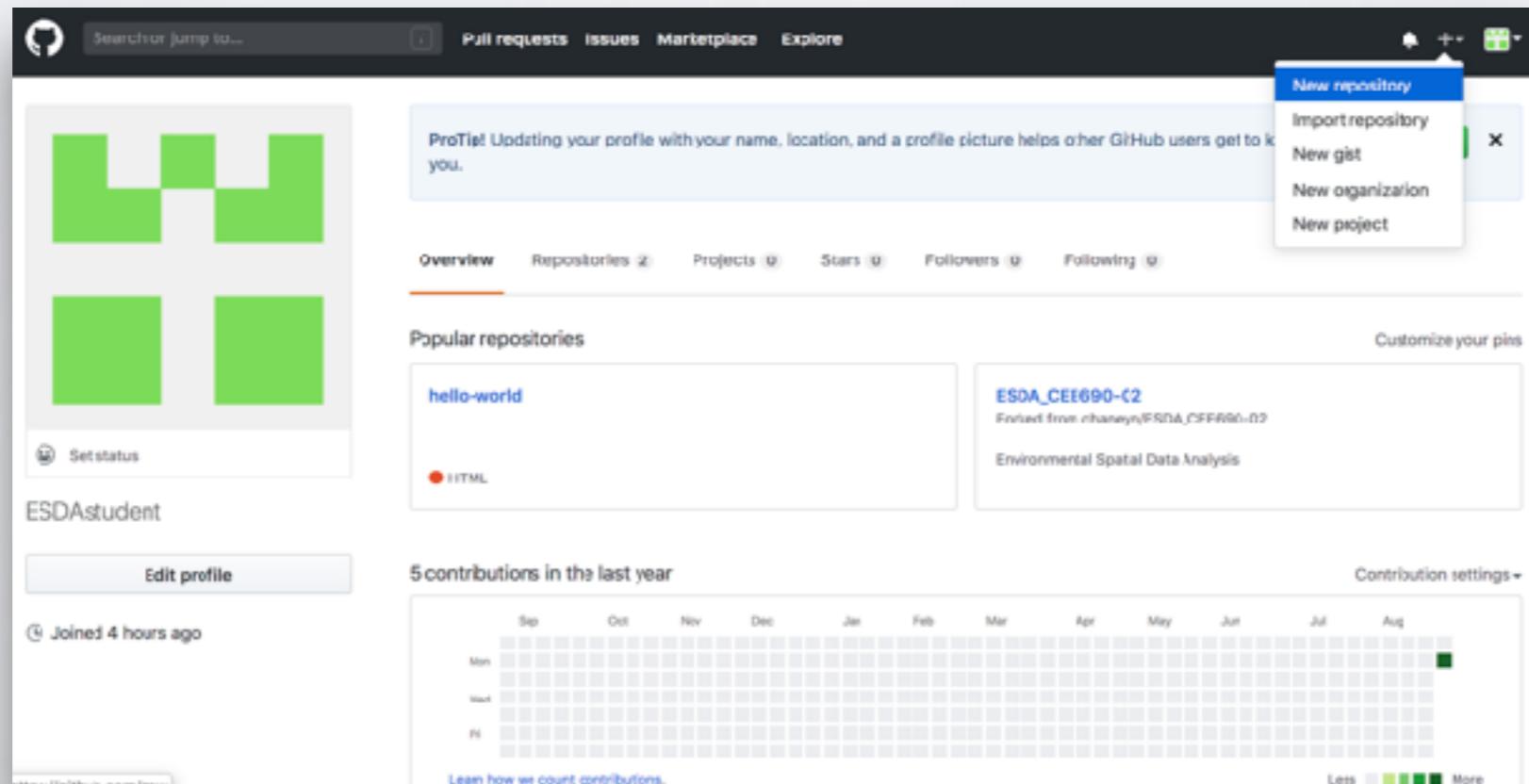
# Add assignment to your repository

```
:~/work/ESDA_nwc$ git add HW1.ipynb
```

# Commit changes to repository

```
jovyan@4d01da836a6e:~/work/ESDA_nwc$ git commit -am "Added HW1 file"
[master (root-commit) 7f144f0] Added HW1 file
 1 file changed, 34 insertions(+)
 create mode 100644 HW1.ipynb
```

Make a corresponding repository on your GitHub account



# Match names

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

### Repository template

Start your repository with a template repository's contents.

No template ▾

Owner

 ESDAstudent ▾

Repository name \*

/ ESDA\_nwc 

Great repository names are short and memorable. Need inspiration? How about [reimagined-octo-guide](#)?

# Make private (unless you want everyone to see your completed assignments!)

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

### Repository template

Start your repository with a template repository's contents.

No template ▾

Owner

Repository name \*

 ESDAstudent ▾

/

ESDA\_nwc



Great repository names are short and memorable. Need inspiration? How about [reimagined-octo-guide](#)?

# Make private (unless you want everyone to see your completed assignments!)



**Public**

Anyone can see this repository. You choose who can commit.



**Private**

You choose who can see and commit to this repository.

# Create the repository!

Skip this step if you're importing an existing repository.

**Initialize this repository with a README**

This will let you immediately clone the repository to your computer.

Add .gitignore: **None** ▾

Add a license: **None** ▾



**Create repository**

# Attach both your local and online repositories (replace the link with yours)

...or push an existing repository from the command line

```
git remote add origin https://github.com/ESDAstudent/ESDA_nwc.git  
git push -u origin master
```



```
jovyan@4d01da836a6e:~/work/ESDA_nwc$ git push -u origin master  
Username for 'https://github.com': ESDAstudent  
Password for 'https://ESDAstudent@github.com':  
Counting objects: 3, done.  
Delta compression using up to 8 threads.  
Compressing objects: 100% (2/2), done.  
Writing objects: 100% (3/3), 489 bytes | 489.00 KiB/s, done.  
Total 3 (delta 0), reused 0 (delta 0)  
To https://github.com/ESDAstudent/ESDA_nwc.git  
 * [new branch]      master -> master  
Branch 'master' set up to track remote branch 'master' from 'origin'.  
jovyan@4d01da836a6e:~/work/ESDA_nwc$ █
```

# Now they are attached to each other! And the online version has been updated.

The screenshot shows a GitHub repository page for the user 'ESDAstudent' with the repository name 'ESDA\_nwc'. The repository is marked as 'Private'. At the top right, there are buttons for 'Unwatch' (with 1 watch), 'Star' (0 stars), and 'Fork' (0 forks). Below the header, there are navigation links for 'Code', 'Issues 0', 'Pull requests 0', 'Projects 0', 'Security', 'Insights', and 'Settings'. A note below the links states 'No description, website, or topics provided.' with an 'Edit' button. There is also a 'Manage topics' link. Key statistics are displayed: 1 commit, 1 branch, and 0 releases. Below these stats, there are buttons for 'Branch: master ▾', 'New pull request', 'Create new file', 'Upload files', 'Find File', and a green 'Clone or download ▾' button. A commit history is shown, with the latest commit by 'chaneyn' adding 'HW1.ipynb' 8 minutes ago. A blue call-to-action bar at the bottom encourages users to 'Add a README with an overview of your project.' and features a green 'Add a README' button.

ESDAstudent / ESDA\_nwc Private

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

1 commit 1 branch 0 releases

Branch: master ▾ New pull request Create new file Upload files Find File Clone or download ▾

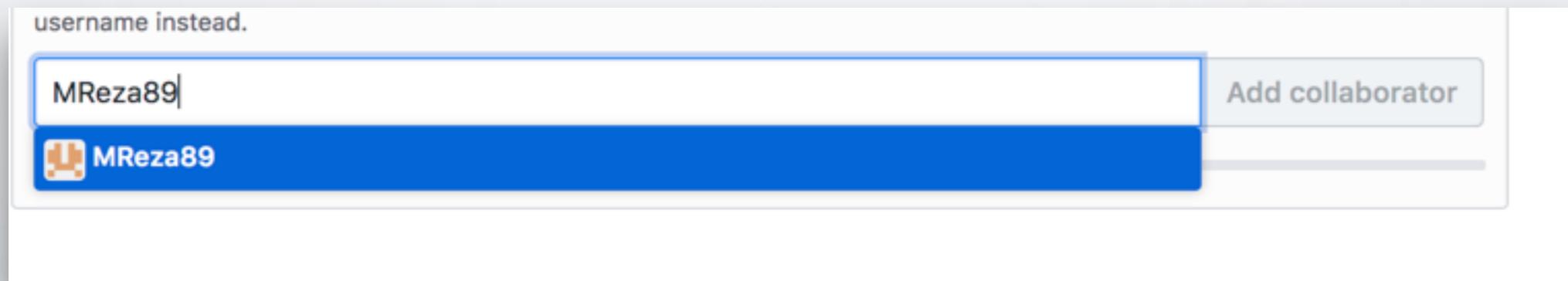
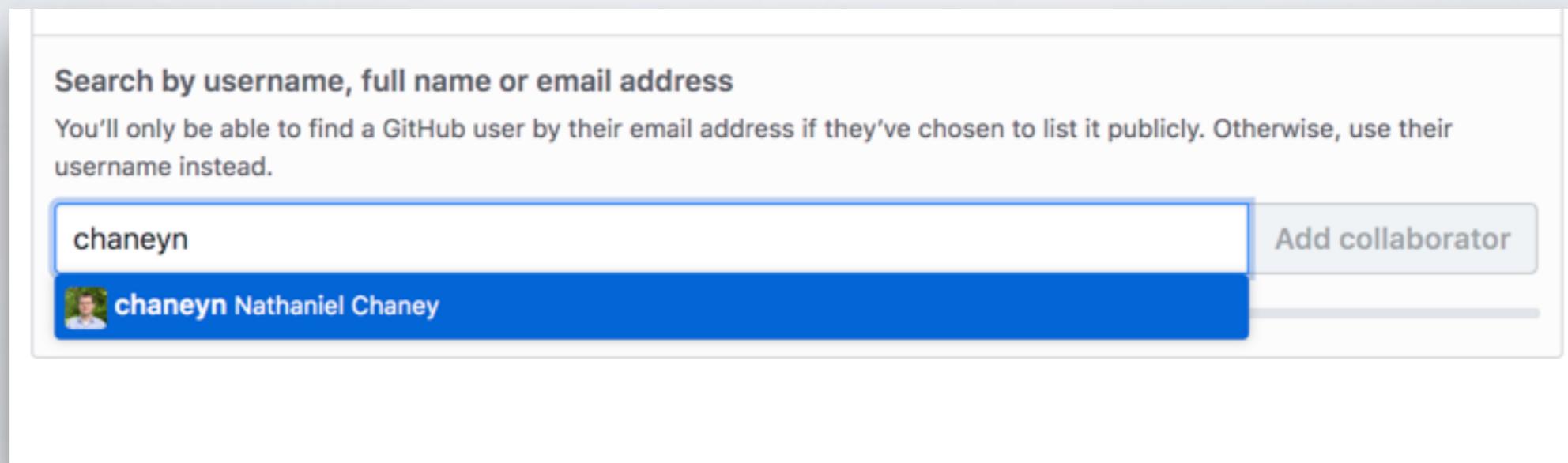
chaneyn Added HW1 file Latest commit 7f144f0 8 minutes ago

HW1.ipynb Added HW1 file 8 minutes ago

Add a README with an overview of your project. Add a README

# Last thing to do is to share your private repository with Reza and Nate

On your repository go to “Settings” then “Collaborators”



With every new assignment:

- Copy the assignment from the class repository
- Add it to your private repository (git add)
- Commit the changes (git commit)
- Complete the assignment
- Commit the changes (git commit)
- Push the changes to the online repository (git push)

But why so complicated? Why not just send  
the completed assignment via email?

Because using version control is critical nowadays to doing most research and industry data science. Forcing you to use version control throughout the course will ensure that you learn how to use it.

**Assignments submitted any other way will NOT  
be accepted.**

Week 1 homework: Get your Docker container up and running, clone the class repository, create and link your private repository, and solve some introductory Python exercises.

**This will be hard for many, but if we get this out of the way at the beginning we will avoid a lot of headaches moving forward.**