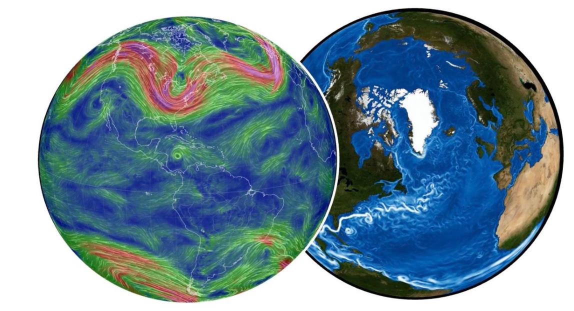
Teaching Climate Change through Data Analysis and Open Science

Rebecca L. Beadling[1] and Students in *Observing & Modeling Climate Change*

[1] Temple University, Department of Earth and Environmental Science, Philadelphia, PA

rebecca.beadling@temple.edu



Abstract

A new course, *Observing and Modeling Climate Change*, is underway in Temple University's (TU) Earth and Environmental Science Department (EES). In *Observing and Modeling Climate Change*, students gain a foundational understanding of anthropogenic climate change and explore the evidence directly through analysis and visualization of real-world observational datasets. After investigating observational evidence, we build an understanding of climate models, the experiments performed including climate projections, and how to access, analyze, and visualize publicly available model output. Along the way, students gain experience in open-source tools used to analyze and visualize observational datasets and climate model output. All course content and assignments are centered around the use of Jupyter Notebooks and all assignments are turned in and delivered via GitHub, with the course ran using virtual machines deployed in Microsoft's Azure Lab. Students first gain the foundations of the Unix Command Line Interface, Python, and version control using Git and GitHub. The course is designed to move from analyzing 1-dimensional observational datasets through 4D time-evolving ocean and atmospheric fields from climate models. During the final 4 weeks, students apply their knowledge and skills to a student-led project that investigates an aspect of our changing climate, culminating in a final presentation.

Course Infrastructure

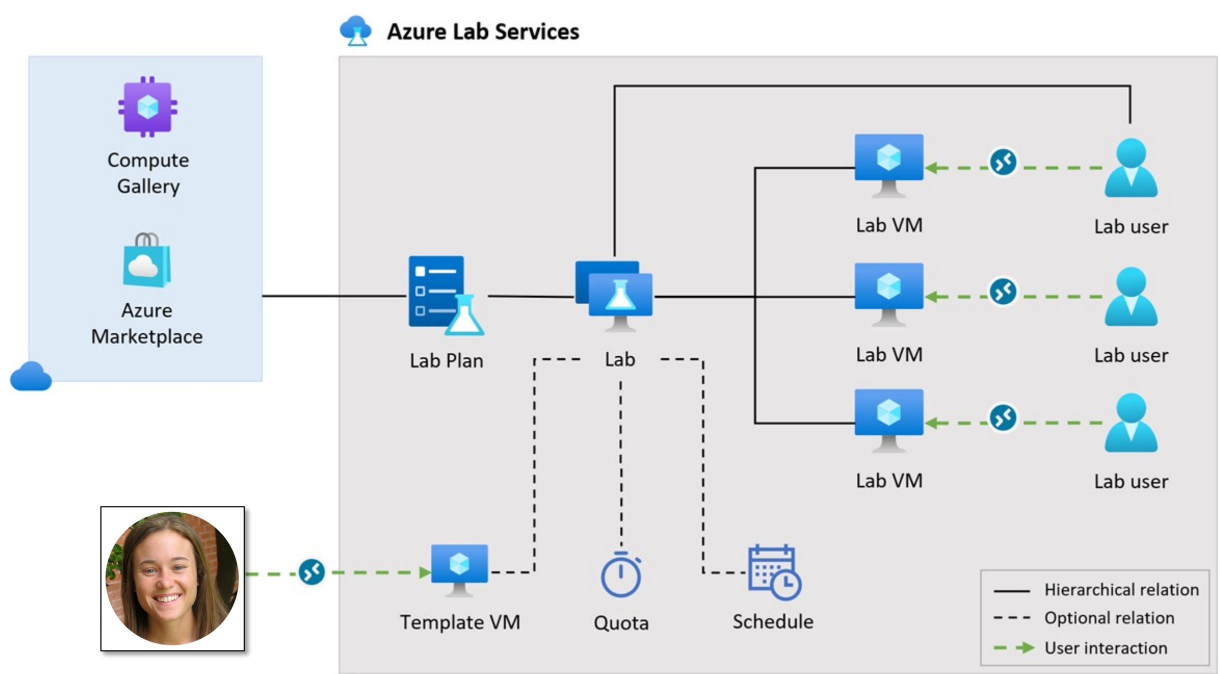


Fig 1. Microsoft Azure Lab Virtual Machine (VM) infrastructure for the course.

Ubuntu Server 20.04 LTS
Large | 8 cores | 16GB RAM | 128GB Standard SSD
$0.70 per hour | Private IP address: 10.0.0.4

The **Microsoft Azure** cloud computing platform's **Azure Lab Services (Fig.1)** is used for all computing and data storage needs, allowing classroom labs to be set up in the cloud easily and quickly. A **Unix Ubuntu** template virtual machine (VM) is created, customized, replicated, and assigned to each student in course. Students log into their unique VM using Secure Shell (SSH) protocol from the command line.

Students are provided with a course guide created and hosted on **HackMD**, a platform for collaborating on open-source projects. The guide covers logging into the Azure VM, open-source / freely-available tutorials and resources to be used through the semester, getting set up with a GitHub account, and other relevant topics.

A **GitHub** organization is created by the instructor and all data and Jupyter Notebooks are delivered via GitHub. The course is designed in "units" which are GitHub **template repositories** that contain all the files necessary to complete accompanying assignments. Students create a GitHub account and create a repro from the provided template and clone it onto their VM. SSH authentication is set up via SSH key generation so that students can pull and push seamlessly from their VM to their remote GitHub repo. The organization maintains ownership of their Git repos to grade assignments & troubleshoot.

Developing the climate analysis tool-kit

The course consists of 23 students: 8 graduate and 15 undergraduate. Prior programming experience is diverse across the students, ranging from zero programming experience, an introductory data science course, to graduate students who are using the skills and tools covered in the course in their own research. However, even amongst the graduate students, many are in their first-year and with very limited programming experience. To address this challenge and increase accessibility, the first three weeks of the course are spent getting spun-up on the tools we will be using throughout the semester and completing assignments that build an intuition for command line navigation, the use of Git and GitHub for version control, navigating Jupyter Notebooks, and the foundations of Python.

1

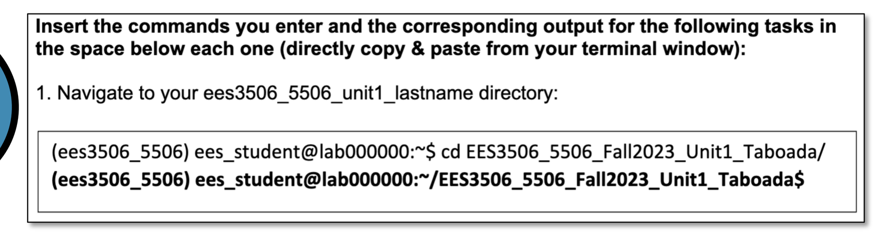


Fig 2. Example from Unix assignment.

Key skills for working with with large climate datasets, climate models, and climate model output include being able to log into remote machines, efficiently download data from remote archives, and create, manipulate, and run programming code. All the above requires the ability to work at the **command line**.

Lessons designed around **Software Carpentry's Unix Shell tutorial** walk students through the basics of command line navigation and challenge them to apply their understanding to navigating their own VM (**Fig. 2**)

2

Git and **GitHub** are open-source tools used to facilitate version control, collaboration and project management, and the open sharing of code, results, and data **enhancing reproducible science**. Students work through exercises working with Git from the command line and cloning from and pushing to their remote GitHub repos (**Fig 3**).

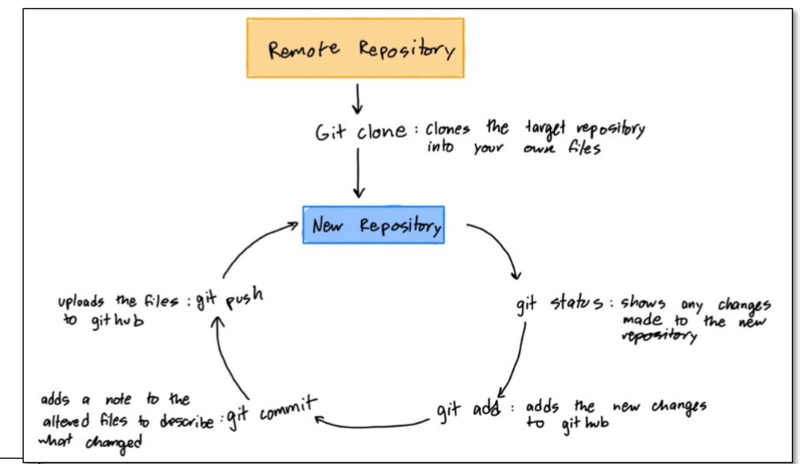


Fig 3. Student drawing of the connections and usage of key Git commands

3

Students learn how to use **Jupyter Notebooks** for interactive development, documentation, and Python code execution. Best practices for creating notebooks that can be widely used by the scientific community are emphasized.

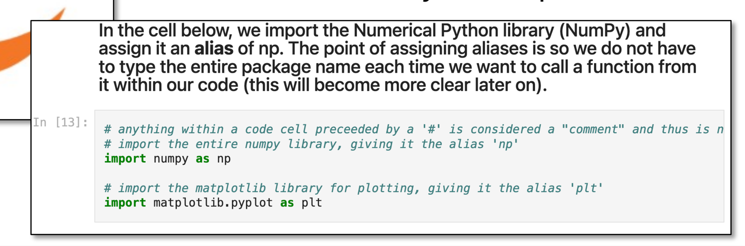


Fig 4. Example of Jupyter Notebook from unit 2 with descriptive Markdown

4

Python Foundations:

- modules, packages, & libraries
- arithmetic & logic.
- data types
- Indexing
- control flow
- functions
- **Numpy** arrays
- **pandas**
- Object-oriented plotting with **Matplotlib**
- **Xarray**
- **momlevel**

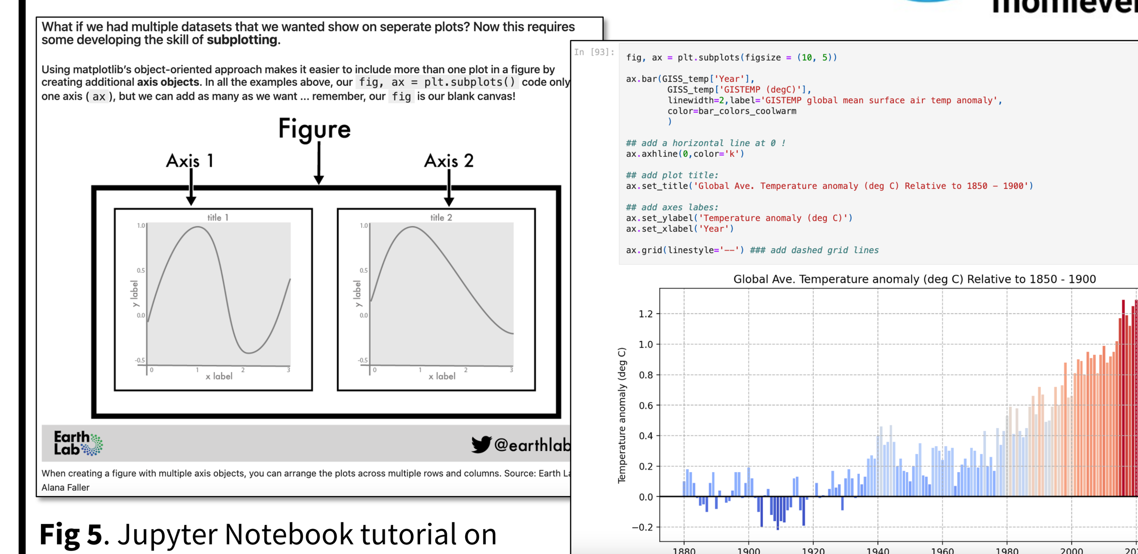
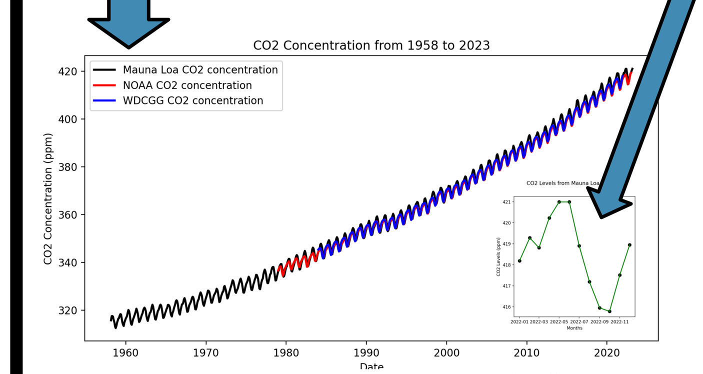


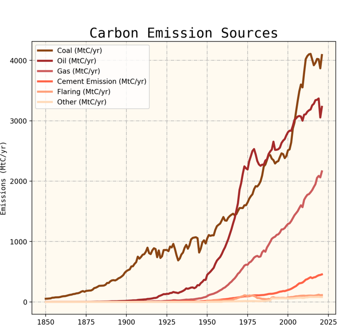
Fig 5. Jupyter Notebook tutorial on object-oriented plotting using matplotlib.

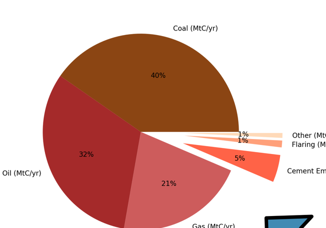
1D: *investigating controls on Earth's average surface temperature*

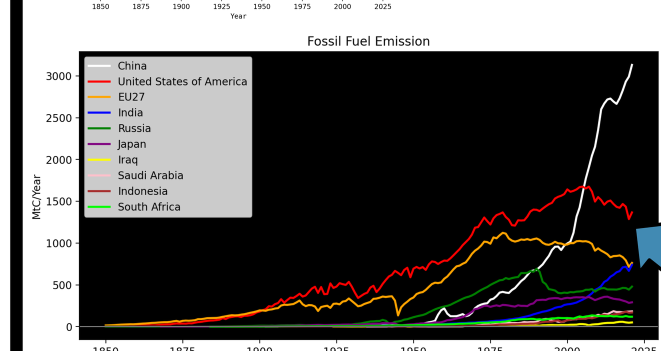
Observed atmospheric CO_2 concentrations (Jesse Marshall, General Science with Teaching)



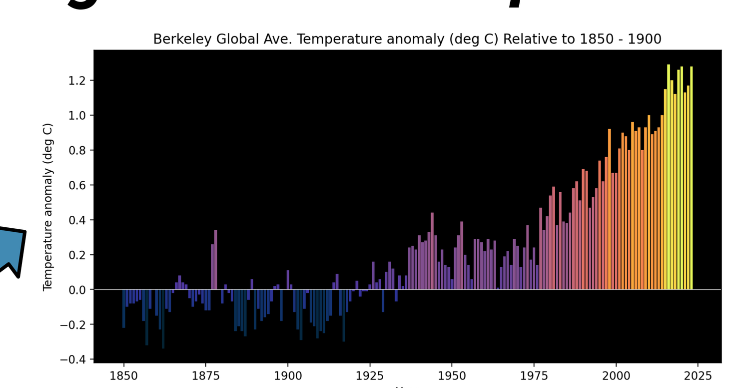
Annual Climatology of CO2 as measured at Mauna Loa (Christian Kelly, EES Climate)



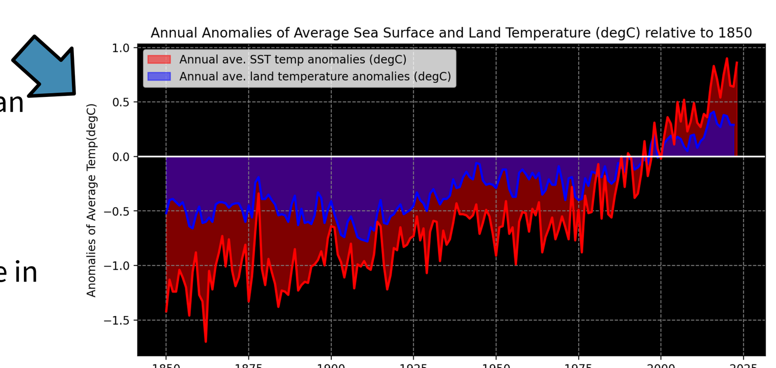




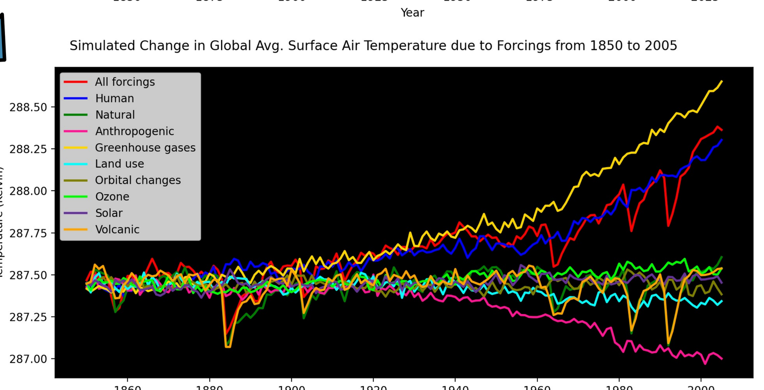
Observed global average surface air temperature anomalies relative to 1850-1900 average (Bertie Miller, MS Geology)



Observed global average surface air temperature anomalies relative to 1850-1900 average over land vs. ocean (Anna Coomans, EES Climate)



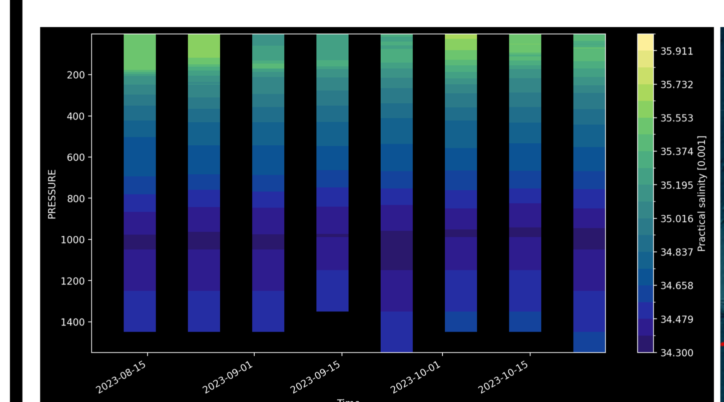
Simulated change in global average surface air temperature in response to individual climate forcings from NASA GISS-E2 (Hunter Barbieri, EES Climate)

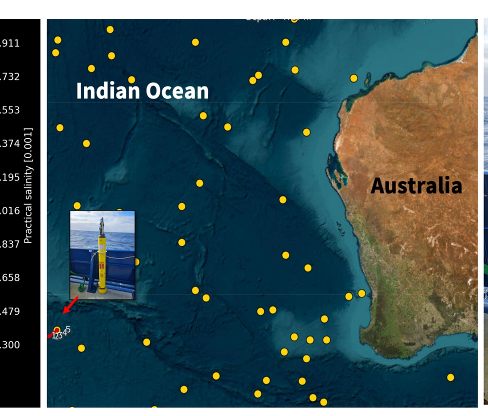


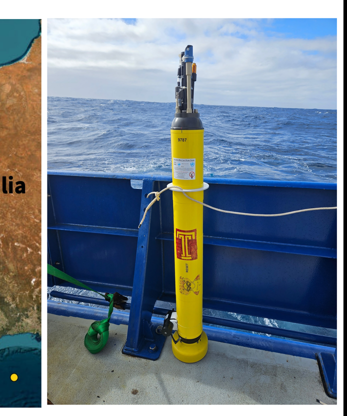
Carbon emissions by type from the Global Carbon Budget (James Milward, MS Environmental Geoscience)

Carbon emissions by country from the Global Carbon Budget (Isaac Pointer, General Science with Teaching)

4D: *Interior Ocean Structure*



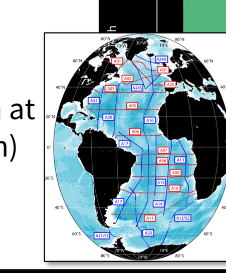


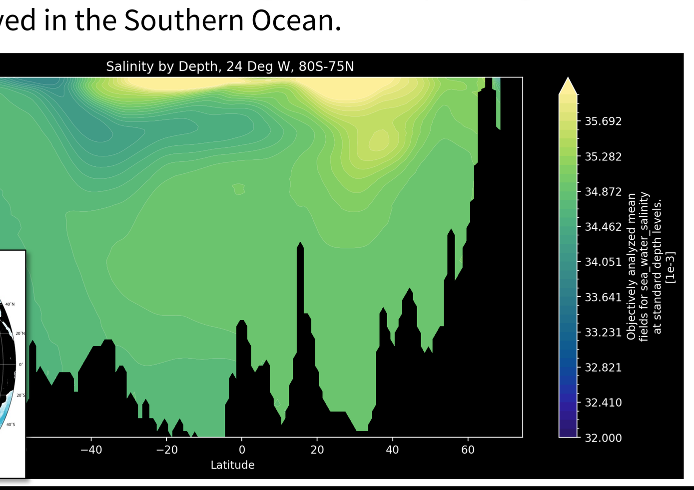


After learning about the ocean's role in the climate system, students analyzed observations from "Hooter the Owl", a Southern Ocean Carbon and Climate Observations and Modeling (SOCCOM) profiling float adopted by Temple University that was recently deployed in the Southern Ocean.

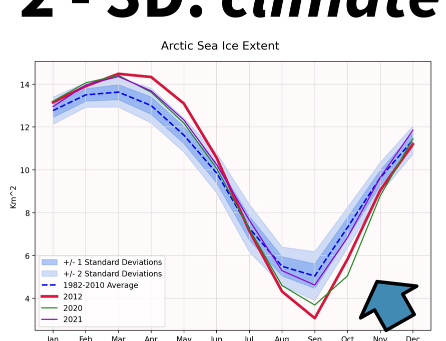
Students investigated interior ocean properties (temperature, salinity, oxygen, and nitrate) and ocean circulation pathways using the World Ocean Atlas 18 Product (WOA18)

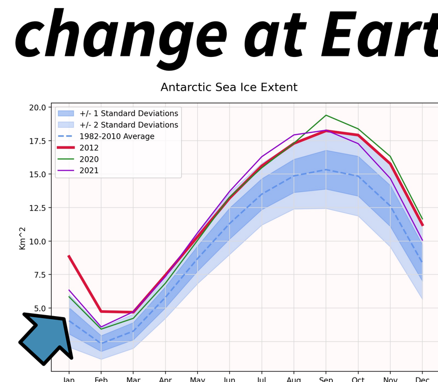
Observed salinity in the Atlantic ocean at 24°W (A16 repeat hydrographic section) (Will Ellinger, EES Climate)



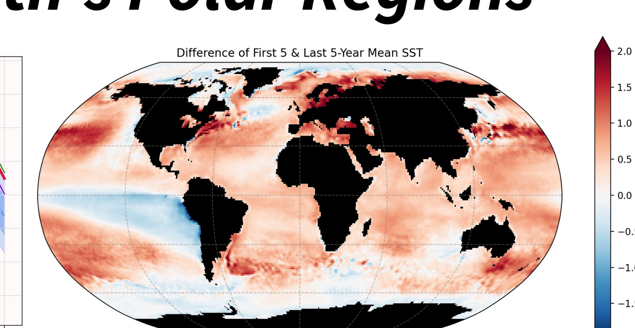


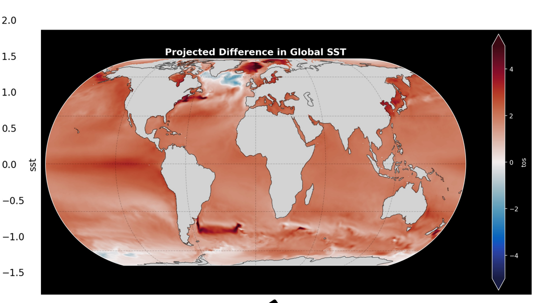
2 - 3D: *climate change at Earth's Polar Regions*





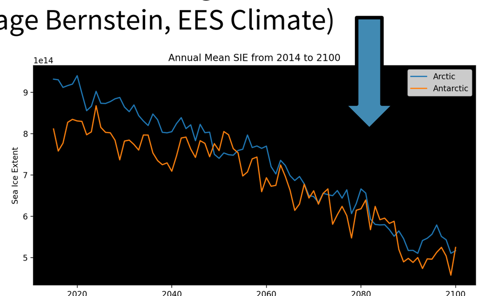
Observed Arctic and Antarctic sea ice extent (Emma Holtzman, EES Climate)

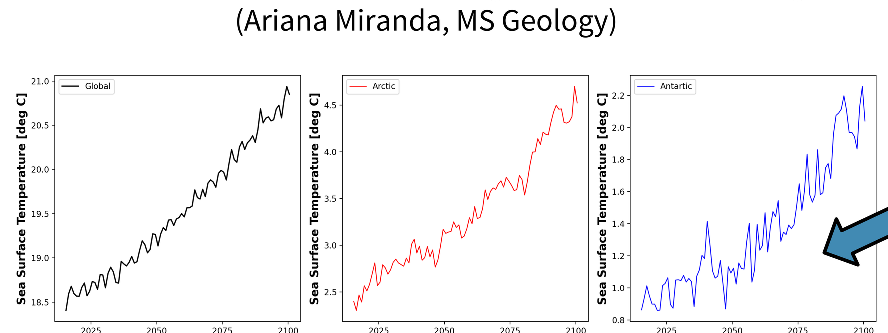




Observed change in sea surface temperatures (2017 to 2022 average – 1993 to 1998 average) (Ariana Miranda, MS Geology)

Projected change in sea ice extent under the the SSP5-8.5 high emissions scenario (Gage Bernstein, EES Climate)





Projected change in sea surface temperatures under the SSP5-8.5 high-emissions scenario from GFDL-ESM4 model output stored on **Google Cloud** (left: Caroline Merheb, PhD Geosciences; top: Sofia Taboada, EES Geochemistry)

Successes

- Providing students with real-world skill sets that are directly applicable to climate science and climate modeling.
- Exposing students to open-source tools and resources.
- Promoting collaboration and reproducible science.
- Learning through hands-on analysis rather than traditional lecture-based delivery.

Challenges

- Wide diversity of computational & climate science backgrounds.
- Time management in a project-oriented course.
- Delivery of all content via GitHub limits dataset sizes (conversion to Zarr format required for large global fields).
- Grading Jupyter Notebooks.
- Managing Azure Lab VM quotas.

ASK THE STUDENTS STANDING HERE WHAT THEY THINK!!