While you wait

1. Connect to TAMU VPN and Login to Grace
   ssh <username>@grace.tamu.edu

2. Go to your scratch directory
   cd $SCRATCH

3. Clone the notebook repository from github
   git clone https://github.com/abishekg7/python_geos.git

   (OR)

Copy notebooks from Grace scratch
   cp -r /scratch/training/python_geos/notebooks .

Python Tools for Geosciences – HPRC Short Course – Spring 2022
Python Tools for Geosciences

Spring 2022 HPRC Short Course
Apr 1, 2022

Abishek Gopal
Assistant Research Scientist
iHESP, Texas A&M Oceanography
Texas A&M High Performance Research Computing
Expectations for this course

• Get an overview of some recent Python libraries designed to support geoscientific analysis

• Learn about the data structures in xarray, how to load and visualize netCDF files, and some basic operations

• Explore other geoscience packages built on top of xarray

• Intended to be a starting point for switching your workflow to Python
Helpful HPRC resources

- Grace quick start guide
  - [https://hprc.tamu.edu/wiki/Grace:QuickStart](https://hprc.tamu.edu/wiki/Grace:QuickStart)

- Introduction to HPRC – Short course
  - [https://hprc.tamu.edu/training/intro_hprc.html](https://hprc.tamu.edu/training/intro_hprc.html)

- Submit tickets to help@hprc.tamu.edu
Upcoming relevant HPRC short courses

- **Apr 8: Introduction to Fortran**
  - **Instructor:** Abishek Gopal
  - **Time:** Friday, Apr 8, 10:00AM - 12:30PM

- **Apr 8: Introduction to Perl**
  - **Instructor:** Keith Jackson
  - **Time:** Friday, Apr 8, 1:30PM - 4:00PM

- **Apr 8: Introduction to Code Parallelization using MPI and OpenMP**
  - **Instructor:** Marinus Pennings
  - Opens Friday, Apr 8 on Google Classroom

[https://hprc.tamu.edu/training/index.html](https://hprc.tamu.edu/training/index.html)
Acknowledgements

• Course materials adapted from detailed xarray, xgcm and Siphon tutorial notebooks
  – https://github.com/xarray-contrib/xarray-tutorial
  – https://gallery.pangeo.io/repos/xgcm/xgcm-examples/

• The HPRC team supporting the short course operations

• Sanjiv R., Steve Y., Fred C., Dapeng Li (iHESP)

• Kristen Thyng (previously: TAMU, now: Axiom Data Science)
Course outline

- Intro to the Pangeo stack
- xarray data structures
- Reading and writing netCDF files

- Plotting with matplotlib and cartopy
- Spatial operations in xgcm
- Data access using Siphon
Launching a JupyterLab notebook from Grace portal

1. Go to https://portal.hprc.tamu.edu/

2. Interactive Apps -> JupyterLab
This app will launch a JupyterLab server on the Grace cluster.

Optional Environment to be activated
/scratch/training/python_geos/conda/envs/training

Enter the name of the environment to be activated. (Optional)

Number of hours
3

Number of cores:
4

Specify the number of cores [1–48] allocated on a node from the Grace cluster.

Total memory (GB)
30

Requested total memory (2 - 360GB)

Enter environment path
/scratch/training/python_geos/conda/envs/training

Request 3 cores/ 30 GB for 3 hours

Hit Launch

* The JupyterLab session data for this session can be accessed under the data root
Connect to JupyterLab session

```
# Navigate to the scratch directory
cd $SCRATCH

# Clone the Python geos repository
!git clone https://github.com/abishekg7/python_geos.git

# Copy the notebooks directory
!cp -r /scratch/training/python_geos/notebooks .
```
Check if the virtualenv works correctly

1. Data structures in xarray
2. Reading and visualizing climate data
3. Computation with Xarray
4. Lazy loading and chunking with Dask arrays
5. Introduction to Dask

Exercise: Print Hello, world!

Each notebook will have exercises for you to solve. You'll be given a blank or partially completed cell, followed by a hidden cell with a solution. For example.

Print the text "Hello, world!".

```python
print('Hello world')
```

Hello world

Run the cell below. Let's make sure the virtual environment is loaded correctly

```python
import xarray
import cartopy
```
In case of session not starting or virtualenv issues

Email output.log to help@hprc.tamu.edu
Course outline

- Intro to the Pangeo stack
- xarray data structures
- Reading and writing netCDF files

- Plotting with matplotlib and cartopy
- Spatial operations in xgcm
- Data access using Siphon
Current/last generation of post-processing tools

- Mature tools/languages for working with moderate resolution datasets
- Often optimized to do specific tasks really well/fast.
- Not designed with high-resolution datasets in mind.
Python geo-scientific software stack

Credit: Ryan Abernathey. Inspired by Jake VanderPlas PyCon 2019
• Builds on NumPy by applying metadata such as dimensions, coordinates, data variables and attributes to raw NumPy arrays.
• Inherits Pandas functionality
• xarray.Dataset is an in-memory representation of the netCDF file format
• xarray works seamlessly with the dask library to enable parallel computations more easily
Apply operations over named dimensions

Select values by label or logical conditions, instead of integer location

Easily use the **split-apply-combine** paradigm with groupby

Keep track of arbitrary metadata in the form of a Python dictionary
A comprehensive library for creating static, animated, and interactive visualizations in Python.

Cartopy adds understanding of map projections to matplotlib plots.

https://matplotlib.org/gallery/

https://scitools.org.uk/cartopy/docs/latest/gallery/index.html
Short break!
(15 minutes)

We will resume at 2:45 Central
Course outline

- Intro to the Pangeo stack
- xarray data structures
- Reading and writing netCDF files

- Plotting with matplotlib and cartopy
- Spatial operations in xgcm
- Data access using Siphon
• xarray doesn’t implicitly understand GCM grids
• xgcm wraps xarray to add an understanding of grid topology
• Implements spatial derivative operators
• Understands only C-grids for now, but other works are in progress
• Grid-aware vertical interpolation

https://xgcm.readthedocs.io/en/latest/grids.html
A collection of Python utilities for downloading data from remote data services

https://matplotlib.org/gallery/

https://scitools.org.uk/cartopy/docs/latest/gallery/index.html
Some great Python modules to go along with xarray and dask!
pop-tools  

Tools to support analysis of POP2-CESM model solutions with xarray

- Wraps xgcm to provide support for POP2 grids.
- Inherits spatial derivative operators from xgcm
- Support for POP2 region masks

xroms

- Wraps xgcm to provide ROMS-specific grid manipulations and functions of interest to oceanographers.
- Developed by Kristen Thyng, Rob Hetland, et al. at TAMU
- Wraps cf-xarray to generalize coordinate and dimension calling.
- Wraps xcmocean to automatically choose colormaps for plotting!

https://github.com/kthyng/xroms
GeoCAT

A collection of Python utilities related to NCL

https://matplotlib.org/gallery/
A collection of Python tools for reading, visualizing, and performing calculations with weather data.

https://unidata.github.io/python-training/gallery/gallery-home/
Key Takeaways

• The Pangeo framework rethinks how we analyze large datasets
  – Resusable software design can help avoid re-writing analysis scripts that
    has already been developed by community
  – In its developmental stages, and will take a few more years to reach the
    depth/breadth of existing geoscience tools
  – For newer analysis tools development, consider using Pangeo

• NCO, CDO, Ferret, etc are still extremely handy for specific tasks
Additional Python resources

• Previously offered HPRC short courses
  – Introduction to Python
    • https://hprc.tamu.edu/training/intro_python.html
  – Introduction to Scientific Python
    • https://hprc.tamu.edu/training/intro_scientific_python.html
  – Introduction to Python for MATLAB users
    • https://hprc.tamu.edu/training/python_matlab.html

• NumPy for MATLAB users (Quick reference)
  – http://mathesaurus.sourceforge.net/matlab-numpy.html
Additional resources

• Official Documentation
  – xarray docs
  – xgcm docs

• Ask for help:
  – Use the python-xarray on StackOverflow
  – GitHub Issues for bug reports and feature requests
  – Pangeo forums http://discourse.pangeo.io/
Questions?
cd $SCRATCH
# Load Anaconda
ml Anaconda3/2020.07

# Create the virtual environment
conda create python=3.7 -n training -c conda-forge

# Activate the virtual environment
source activate training

# Install packages into the virtual environment
conda install -c conda-forge cartopy matplotlib xarray
conda install -c conda-forge xgcm dask esmpy

# Deactivate the environment
source deactivate
Pangeo

https://pangeo.io/architecture.html
<table>
<thead>
<tr>
<th>Build Your Own Pangeo</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Storage Formats</strong></td>
</tr>
<tr>
<td><strong>ND-Arrays</strong></td>
</tr>
<tr>
<td><strong>Data Models</strong></td>
</tr>
<tr>
<td><strong>Processing Mode</strong></td>
</tr>
<tr>
<td><strong>Compute Platform</strong></td>
</tr>
</tbody>
</table>

Launching a JupyterLab notebook from Grace portal

1. Go to [https://portal.hprc.tamu.edu/](https://portal.hprc.tamu.edu/)

2. Interactive Apps -> JupyterLab Geoscience
JupyterLab - Geoscience

This app will launch a JupyterLab server on the Grace cluster for the Python Tools for Geosciences short course.

Module

Anaconda3/5.3.0

Anaconda/3-x.x.x.x and Anaconda3 use Python3

Optional Environment to be activated

/scratch/training/python_geos/conda/envs/training

Enter the name of the environment to be activated.

Account

This field is optional.

Email

This field is optional.

I would like to receive an email when the session starts

Launch

* The JupyterLab - Geoscience session data for this session can be accessed under the data root directory.

Check environment path

/scratch/training/python_geos/conda/envs/training

Hit Launch