

Developing educational resources for cloud -based remote sensing data with xarray

Emma Marshall
SIParCS intern

July 26, 2022



Who am I?

PhD student

Snow & Ice Research Lab

Geography Department

University of Utah

Research interests:

- Mountain glaciers in High Mountain Asia
- Lake-terminating glaciers
- Remote sensing -derived glacier surface velocity data



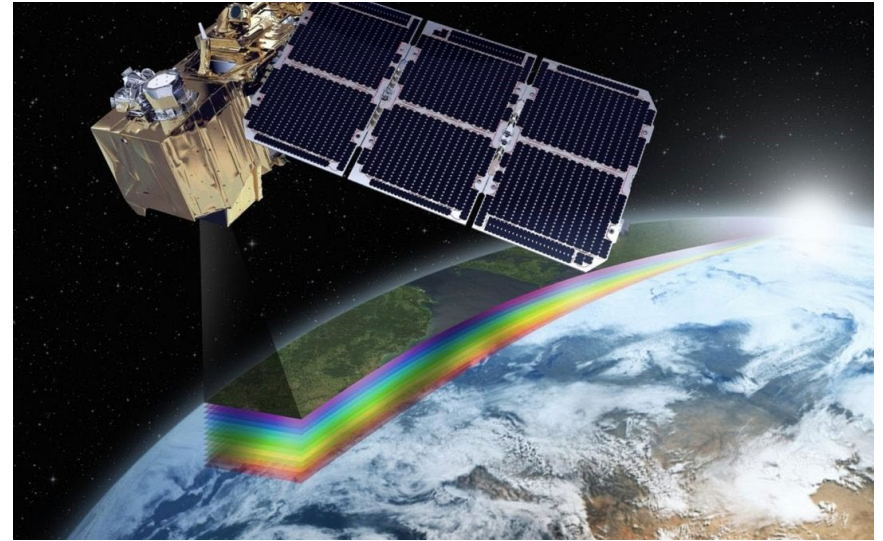


Background

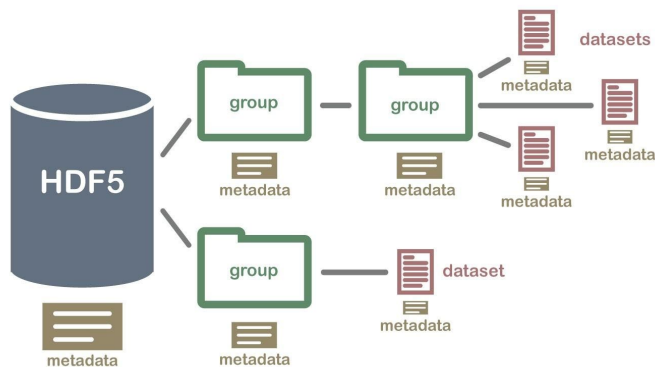
- I. Satellite remote sensing data
- II. Xarray for multi -dimensional gridded datasets
- III. Transition to open, cloud -based science

Satellite remote sensing data

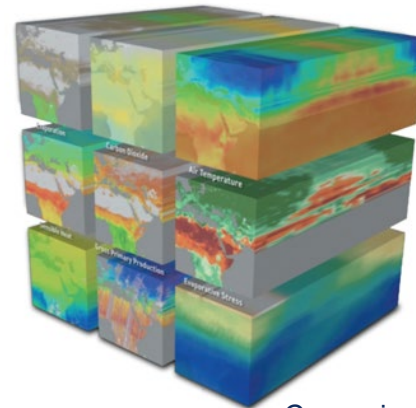
- Remote sensing datasets are large, complex; increasing volume of available data
- Multi-variate, multi-dimensional, metadata



Cervest

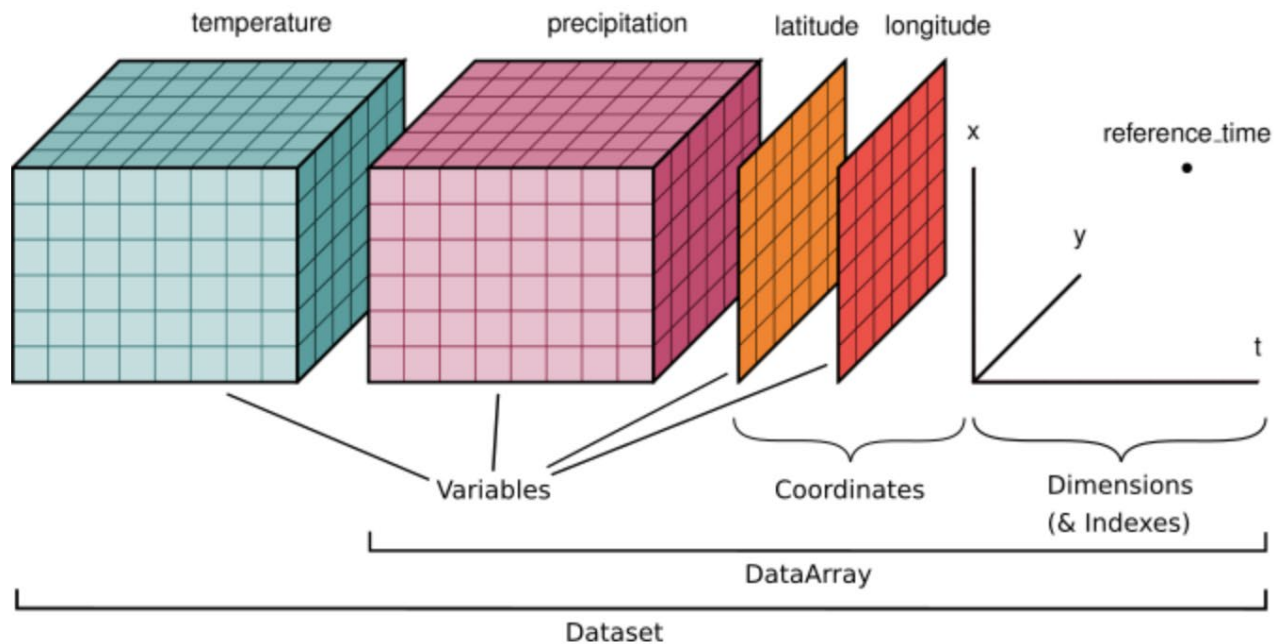


NEON



Copernicus

Xarray



“[Xarray](#) is an open source project and Python package that makes working with labelled multi-dimensional arrays simple, efficient, and fun”

Xarray has functionality for organizing, analyzing raster data, and backend integration with cloud-optimized data types

Transition to cloud -based, open science

Accessing, manipulating data is a common bottleneck in remote sensing research workflows

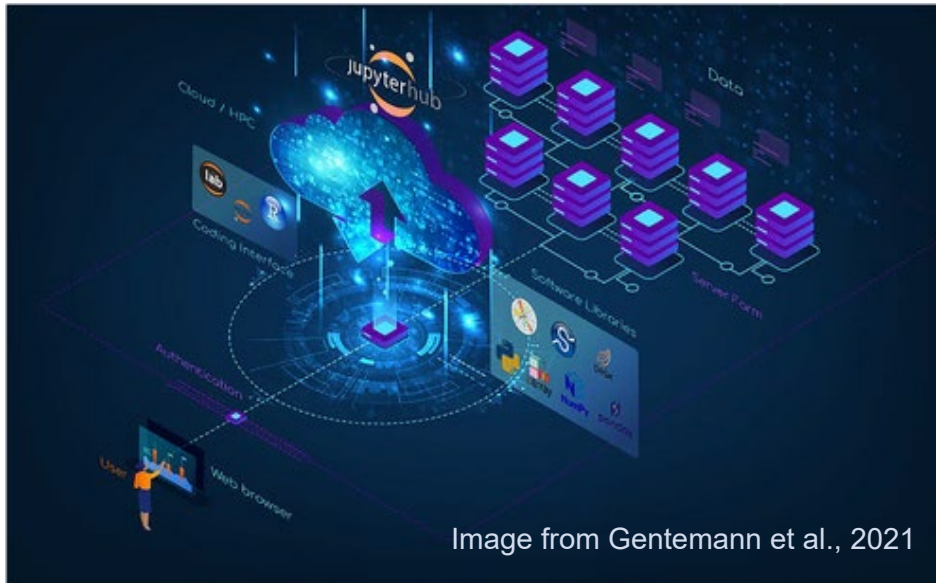


Image from Gentemann et al., 2021

“... [the transition to open science] ... is a product of practices, norms, and community behavior around that technology... it is important to deliberately design a new community infrastructure ...” Gentemann et al., 2021

Cloud computing resources can democratize scientific participation, reduce computational barriers to entry

Objectives

- I. Gain experience working with cloud -hosted data, parallelized workflows and xarray
- II. Contribute educational resources related to xarray, remote sensing data and cloud - computing resources to the open -source scientific community

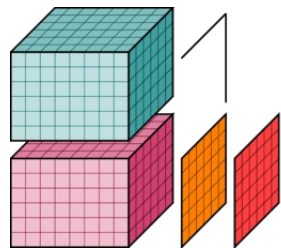
Cloud computing and software resources

Cloud platforms

- Amazon Web Services
- Microsoft Planetary Computer
- Pangeo Jupyter hub
- Alaska Satellite Facility On Demand processing

Open source software

- xarray
- dask



xarray



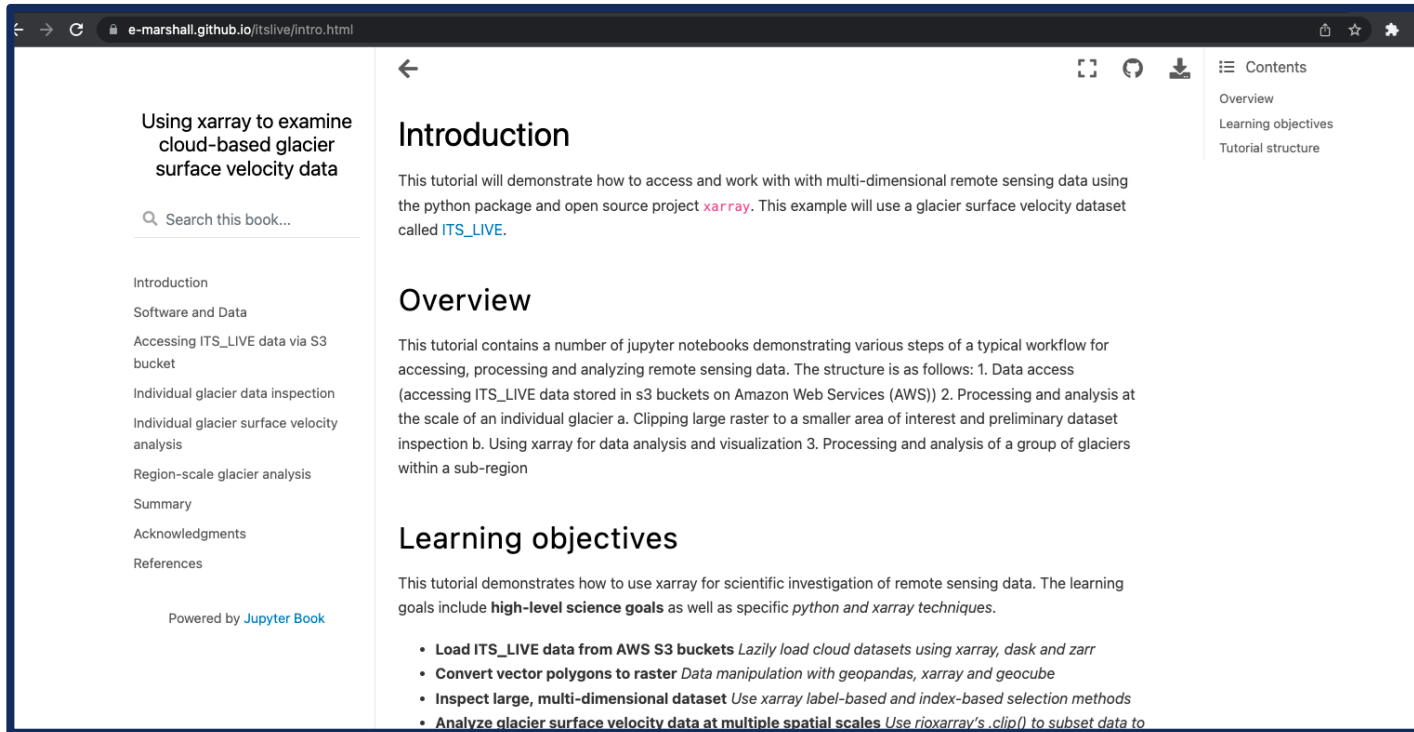


Educational resources

- I. Jupyter book tutorials
- II. Other open source contributions

Jupyter book 1:

ITS_LIVE glacier velocity data hosted in AWS S3 buckets



The screenshot shows a web browser displaying a Jupyter Book page. The address bar shows the URL `e-marshall.github.io/itslive/intro.html`. The page content is as follows:

Using xarray to examine cloud-based glacier surface velocity data

Search this book...

- Introduction
- Software and Data
- Accessing ITS_LIVE data via S3 bucket
- Individual glacier data inspection
- Individual glacier surface velocity analysis
- Region-scale glacier analysis
- Summary
- Acknowledgments
- References

Powered by [Jupyter Book](#)

Introduction

This tutorial will demonstrate how to access and work with with multi-dimensional remote sensing data using the python package and open source project `xarray`. This example will use a glacier surface velocity dataset called `ITS_LIVE`.

Overview

This tutorial contains a number of jupyter notebooks demonstrating various steps of a typical workflow for accessing, processing and analyzing remote sensing data. The structure is as follows: 1. Data access (accessing ITS_LIVE data stored in s3 buckets on Amazon Web Services (AWS)) 2. Processing and analysis at the scale of an individual glacier a. Clipping large raster to a smaller area of interest and preliminary dataset inspection b. Using xarray for data analysis and visualization 3. Processing and analysis of a group of glaciers within a sub-region

Learning objectives

This tutorial demonstrates how to use xarray for scientific investigation of remote sensing data. The learning goals include **high-level science goals** as well as specific *python and xarray techniques*.

- **Load ITS_LIVE data from AWS S3 buckets** Lazily load cloud datasets using `xarray`, `dask` and `zarr`
- **Convert vector polygons to raster** Data manipulation with `geopandas`, `xarray` and `geocube`
- **Inspect large, multi-dimensional dataset** Use `xarray` label-based and index-based selection methods
- **Analyze glacier surface velocity data at multiple spatial scales** Use `rioxarray's clip()` to subset data to

<https://e-marshall.github.io/itslive/intro.html>

Jupyter book 2:

Sentinel1 Radiometric Terrain Corrected backscatter data

jupyter

My sample book

Search this book

Introduction

Accessing Sentinel1 RTC data from Planetary Computer

Exploring S1 GRD image with sentinel

```
#scatter plot
lake2_rtc.where(lake2_rtc.pass_dir == 'descending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'blue', marker='x', linewidth=0)
lake2_rtc.where(lake2_rtc.pass_dir == 'ascending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'blue', marker='o', linewidth=0, alpha = 0.6)

lake1_rtc.where(lake2_rtc.pass_dir == 'descending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'red', marker = 'x', linewidth=0)
lake1_rtc.where(lake2_rtc.pass_dir == 'ascending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'red', marker = 'o', linewidth = 0, alpha = 0.6)

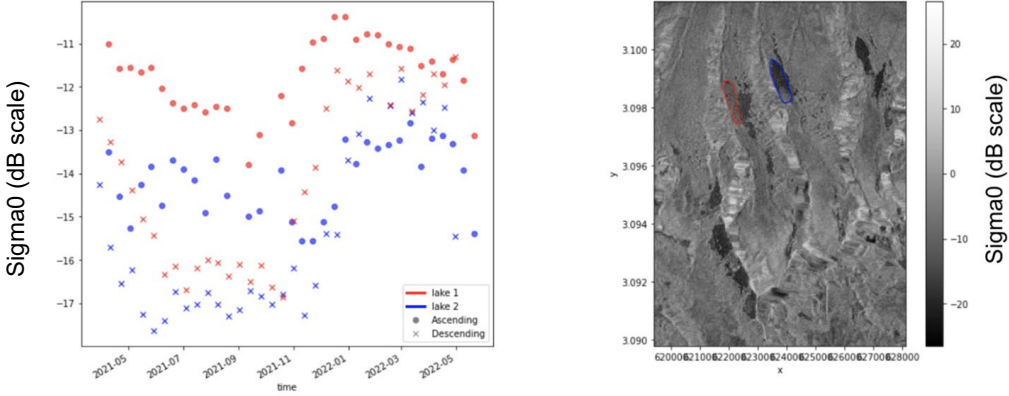
#backscatter image
sigma_n_vv.sel(time='2022-04-30').plot(ax=axes[1], cmap=plt.cm.Greys_r)
lakes_prj.plot(edgecolor=lakes_prj['color'], facecolor='none', ax=axes[1])

legend_elements = [Line2D([0], [0], color = 'r', lw = 3, label = 'lake 1'),
                    Line2D([0], [0], color = 'b', lw = 3, label = 'lake 2'),
                    Line2D([0],[0], color = 'grey', lw = 0, marker = 'o', label = 'Ascending'),
                    Line2D([0],[0], color = 'grey', lw = 0, marker = 'x', label = 'Descending')]

axes[0].legend(handles = legend_elements, loc = 'lower right')

axes[0].set_title('VV backscatter over proglacial lakes 2021-2022')
axes[1].set_title('4-30-2022 RTC image, glacial lakes outlined')
```

Temporal variability in backscatter over two proglacial lakes, Bhutan, Himalaya



The left plot shows the temporal variability of backscatter (Sigma0) in dB scale for two lakes (lake 1 and lake 2) from May 2021 to May 2022. Lake 1 is represented by red markers (circles for ascending, crosses for descending) and lake 2 by blue markers. The right plot shows a grayscale image of the lakes with outlines, with a color bar indicating Sigma0 (dB scale) from -20 to 20.

Sentinel1 RTC data from Planetary Computer

Accessing Sentinel1 RTC data from Planetary Computer

Exploring S1 GRD image with sentinel

Scale to decibel

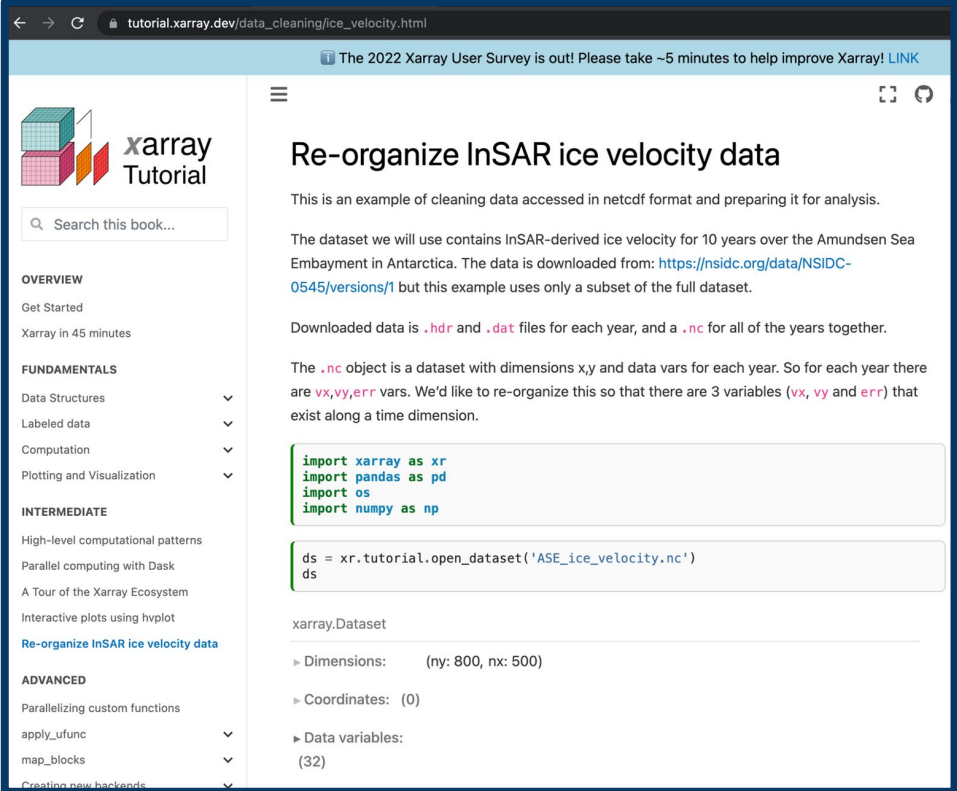
Planetary Computer RTC image and ASF RTC images

Glacial lake outlines

Individual scenes

Other open source contributions

- Data cleaning example for xarray tutorial
- Added examples to xarray codebase
- Co-presented at xarray tutorial during SciPy 2022 in Austin, TX (July 2022)



The screenshot shows a web browser displaying a tutorial page titled "Re-organize InSAR ice velocity data". The page is part of the xarray tutorial series, as indicated by the logo and navigation menu on the left. The main content area contains text explaining the dataset and the goal of re-organizing it. Below the text, there are two code blocks: the first shows the necessary imports for xarray, pandas, os, and numpy; the second shows the code to open the dataset. The output of the code is displayed below, showing the dimensions and coordinates of the dataset.

```
import xarray as xr
import pandas as pd
import os
import numpy as np

ds = xr.tutorial.open_dataset('ASE_ice_velocity.nc')
ds
```

xarray.Dataset
Dimensions: (ny: 800, nx: 500)
Coordinates: (0)
Data variables: (32)

https://tutorial.xarray.dev/data_cleaning/ice_velocity.html

What did I learn?

- Collaborative workflows
- Vectorized and xarray - native programming
- Parallelized workflows
- Cloud computing
- Tutorial design + construction
- Introduction to the open source scientific software community

What's next?

- Finish incorporating new chapters in the Sentinel1 book
- Share work in open science session at Fall conference
- Pangeo / xarray blog posts
- Continue to contribute to open source packages !

Thank you:

Deepak Cherian (NCAR)

Scott Henderson (University of Washington)

Jessica Scheick (University of New Hampshire)

Kevin Paul (NCAR)

Virginia Do, Jerry Cycone, everyone at CISL and across NCAR, and especially all of the NCAR/UCAR interns

References

1. Gentemann, C. L., Holdgraf, C., Abernathey, R., Crichton, D., Colliander, J., Kearns, E. J., et al. (2021). Science storms the cloud. *AGU Advances* , 2, e2020AV000354. <https://doi.org/10.1029/2020AV000354>
2. Stern C, Abernathey R, Hamman J, Wegener R, Lepore C, Harkins S and Merose A (2022) Pangeo Forge: Crowdsourcing Analysis -Ready, Cloud Optimized Data Production. *Front. Clim.* 3:782909. doi: 10.3389/fclim.2021.782909
3. NASA turns to the cloud for help with next -generation Earth missions. *Sea Level News*. October 13, 2021. Sea level change: observations from space. NASA. <https://sealevel.nasa.gov/news/226/nasa-turns-to-the-cloud-for-help-with-next-generation-earth-missions/>

Images

Satellite Imagery

Alaska Satellite Facility Vertex portal. Scenes:

S1B_IW_GRDH_1SSV_20170213T000216_20170213T000241_004274_0076A5_CA26

S1A_IW_GRDH_1SDV_20220720T155707_20220720T155734_044186_054625_C18A

S1A_IW_GRDH_1SSV_20170511T155632_20170511T155659_016536_01B692_3800

S1A_IW_GRDH_1SSV_20170530T154815_20170530T154839_016813_01BF22_D7E9

Other images

https://cervest.earth/news/remote_sensing_of_planet_earth_part_1_introduction_to_satellite_imagery

https://www.copernicus.eu/en/news/news/observer_data_cubes_enabling_and_facilitating_earth_observation_applications

https://www.neonscience.org/resources/learning_hub/tutorials/about_hdf5

<https://docs.xarray.dev/en/stable/>

An aerial grayscale topographic map of a mountainous region. A prominent river valley runs horizontally across the center. A semi-transparent white rounded rectangle is overlaid on the map, containing the text "Questions?".

Questions?







Jupyter book tutorials

Sentinel 1
synthetic
aperture radar
radiometrically
terrain corrected
backscatter time
series

jupyter {book} Accessing Sentinel1 RTC data from Planetary

My sample book

Contents

Accessing Sentinel1 RTC data from Planetary Computer

Inspect STAC metadata

```
#scatter plot
lake2_rtc.where(lake2_rtc.pass_dir == 'descending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'blue', marker='x', linewidth=0)
lake1_rtc.where(lake2_rtc.pass_dir == 'ascending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'red', marker='o', linewidth=0, alpha = 0.6)

lake1_rtc.where(lake2_rtc.pass_dir == 'descending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'red', marker = 'x', linewidth=0)
lake1_rtc.where(lake2_rtc.pass_dir == 'ascending', drop=True).mean(dim=['x','y']).plot(ax=axes[0], color = 'red', marker = 'o', linewidth = 0, alpha = 0.6)

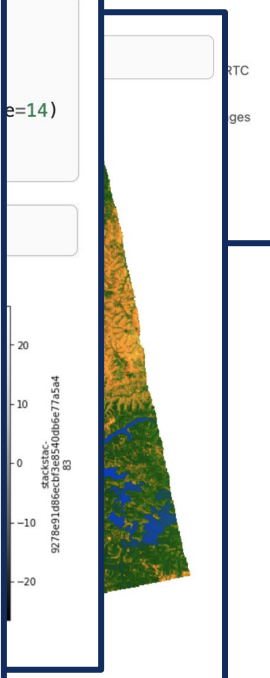
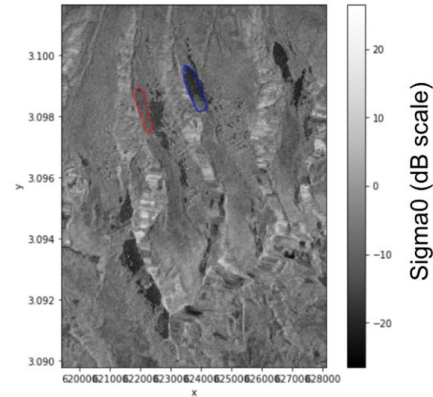
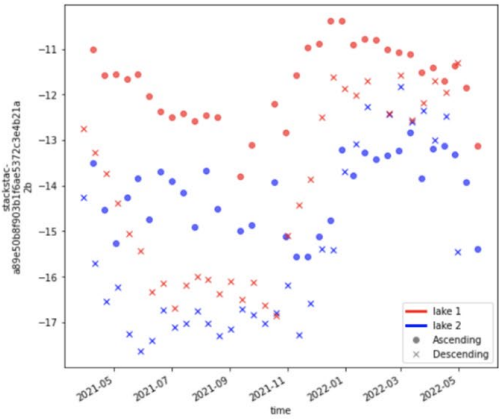
#backscatter image
sigma_n_vv.sel(time='2022-04-30').plot(ax=axes[1], cmap=plt.cm.Greys_r)
lakes_prj.plot(edgecolor=Lakes_prj['color'], facecolor='none', ax=axes[1])

legend_elements = [Line2D([0], [0], color = 'r', lw = 3, label = 'lake 1'),
                    Line2D([0], [0], color = 'b', lw = 3, label = 'lake 2'),
                    Line2D([0],[0], color = 'grey', lw = 0, marker = 'o', label = 'Ascending'),
                    Line2D([0],[0], color = 'grey', lw = 0, marker = 'x', label = 'Descending')]

axes[0].legend(handles = legend_elements, loc = 'lower right')

axes[0].set_title('VV backscatter over proglacial lakes 2021-2022')
axes[1].set_title('4-30-2022 RTC image, glacial lakes outlined')
```

Temporal variability in backscatter over two proglacial lakes, Bhutan, Himalaya



Count Tasks 284 Chunks 869

Type float64 numpy.ndarray

CPU t Wall

xarray.D (time: 7

STAC metadata

What did I learn?

- Collaborative workflows
- Vectorized and 'xarray -native' programming
- Parallelized workflows
- Cloud computing
- Tutorial design + construction
- Introduction to the open source and open science community



**Thank you to my internship mentors this summer
for their guidance, advice and support:**

Deepak Cherian (NCAR)

Scott Henderson (University of Washington)

Jessica Scheick (University of New Hampshire)

Kevin Paul (NCAR)

Thank you to

Virginia Do and the SIParCS and NESSI programs,
all UCAR interns!



What's next?

- Finish incorporating new chapters in the Sentinel1 book
- Share work in open science session at Fall conference
- Pangeo / xarray blog posts
- Continue to contribute to open source packages !



Other open source contributions

- data cleaning example for xarray tutorial
- added examples to xarray codebase
- assisted at xarray tutorial during SciPy 2022 in Austin, TX (July 2022)



What's next?

- Finish incorporating new chapters in the Sentinel1 book
- Share work in an Open Science session at AGU
- Present Sentinel1 jupyter book at NISAR science conference, August 2022, Pasadena, CA
- Publish jupyter books in the journal of open source education?
- Pangeo / xarray blog posts
- Continue to contribute to open source packages !

be more general, trim down

Jupyter book tutorials

Glacier velocity data hosted in AWS S3 buckets

executable book with binder

- cloud hosted data and execution
- open science!

The screenshot shows a web browser displaying a Jupyter Book tutorial. The browser address bar shows the URL `e-marshall.github.io/itslive/intro.html`. The page title is "Using xarray to examine cloud-based glacier data". The main content area is titled "Extracting and visualizing data" and contains the following text:

```
rgi_itslive.explore()
```

and work with with multi-dimensional remote sensing data using `xarray`. This example will use a glacier surface velocity dataset

The page includes a table of contents on the right side with the following items:

- Read in vector data
- Taking a look at a
- ty time
- ed analysis
- ason
- cting and
- izing data
- ngle point

The main content area is divided into sections:

- Introduction**
- Software and Data**
- Accessing ITS_LIVE data via S3 bucket**
- Individual glacier data inspection**
- Individual glacier surface velocity analysis**
- Region-scale glacier analysis**
- Summary**
- Acknowledgments**
- References**

The page is powered by [Jupyter Book](#).

Overview

This tutorial contains a number of jupyter notebooks demonstrating various steps of a typical workflow for accessing, processing and analyzing remote sensing data. The structure is as follows: 1. Data access (accessing ITS_LIVE data stored in s3 buckets on Amazon Web Services (AWS)) 2. Processing and analysis at the scale of an individual glacier a. Clipping large raster to a smaller area of interest and preliminary dataset inspection b. Using xarray for data analysis and visualization 3. Processing and analysis of a group of glaciers within a sub-region

Learning objectives

This tutorial demonstrates how to use xarray for scientific investigation of remote sensing data. The learning goals include **high-level science goals** as well as specific *python and xarray techniques*.

- **Load ITS_LIVE data from AWS S3 buckets** Lazily load cloud datasets using `xarray`, `dask` and `zarr`
- **Convert vector polygons to raster** Data manipulation with `geopandas`, `xarray` and `geocube`
- **Inspect large, multi-dimensional dataset** Use `xarray` label-based and index-based selection methods
- **Analyze glacier surface velocity data at multiple spatial scales** Use `rioxarray`'s `.clip()` to subset data to