Long-form tutorial:

Streaming tutorial in NSIDC Tutorials:
https://github.com/nsidc/NSIDC-Data-Tutorials/blob/main/notebooks/streaming/earthaccess_streaming_webinar.ipynb

earthaccess documentation and Github:

Earthaccess documentation:
Documentation: https://earthaccess.readthedocs.io/en/stable/

Documentation specific to streaming access:
https://earthaccess.readthedocs.io/en/stable/user/access/

Contributor guide:
https://earthaccess.readthedocs.io/en/stable/contributor/

Where to ask questions: https://earthaccess.zulipchat.com/

Github location: https://github.com/earthaccess-dev/earthaccess

- Open a GitHub issue to report a bug, unexpected behavior, a stumbling block in our documentation, or any other problem.
  https://github.com/earthaccess-dev/earthaccess/issues
- Open a Q&A GitHub Discussion to ask questions, share insights, or connect with others about Earth data access (broadly)
  https://github.com/earthaccess-dev/earthaccess/discussions/new?category=q-a
- Open an Ideas GitHub Discussion to suggest new features or improvements, and find collaborators and coordinate the work
  https://github.com/earthaccess-dev/earthaccess/discussions/new?category=ideas
- Join a Collaboration Café to meet other users/developers/maintainers and get help on, or give help to, anything related to earthaccess. Community calendar - earthaccess

Q&A:

Do these earth data also have like Landsat data with the different bands like Red, Green NIR, etc.?
MERRA-2 is an atmospheric reanalysis. This is similar to a weather forecast model. The difference between a reanalysis and weather forecast model is that the model used in the reanalysis is fixed and the model runs are made over a long period. For MERRA-2 this is 1980 to present. The variables are a mix of upper air and surface variables, including temperature,

humidity, winds and pressure. The dataset I used in the demonstration contains variables commonly used in weather forecasts, for example 2 m air temperature, 500 hPa geopotential height, 10 m winds. An overview of MERRA-2 can be found at https://gmao.gsfc.nasa.gov/gmao-products/merra-2/. Detailed information about the MERRA-2 data product highlighted in this webinar can be found here https://disc.gsfc.nasa.gov/datasets/M2T1NXSLV_5.12.4/summary

If we want longer date then will it be 2012-01-10, 2012-12-13 etc? How to download longer date?

You can select the start and end time of the temporal range within which you are searching for data. It helps to familiarize yourself with the data set first to understand the time period over which data were/are being collected. Details on how to use the earthaccess Python library to search for files by time range can be found here https://earthaccess.readthedocs.io/en/stable/user/search/#search-for-datasets-or-data-by-time-range

what does the 5th statement do?

I think this is what you are asking about:
`round(sum([g.size() for g in results]), 2)`.
We are using this to calculate the total size of the files we want to access. The `[g.size() for g in results]` is called a *list comprehension*, basically a loop, that gets the size of each file returned in `results`. This creates a list of file sizes in Megabytes that we add up using `sum`. We use `round` to truncate the total size to two decimal places.
You can find a more detailed version of the tutorial presented during the webinar here https://github.com/nsidc/NSIDC-Data-Tutorials/blob/main/notebooks/streaming/earthaccess_streaming_webinar.ipynb

Is MERRA-2 free to access?

Yes, MERRA-2 data, like all NASA Earthdata, are freely available to all. You do need to register for an Earthdata Login in order to access data. You can do that here https://urs.earthdata.nasa.gov/

On which computer is Dask running?

You get to decide where your compute is. It might be your laptop or work station, or an EC2 instance within AWS us-west-2. Reading through the full tutorial here may be helpful https://github.com/nsidc/NSIDC-Data-Tutorials/blob/main/notebooks/streaming/earthaccess_streaming_webinar.ipynb

What does dim=[] do?

`dim=[]` is a keyword argument to `mean` that indicates which dimension of the dataset you want to calculate the mean over. In the demonstration, I calculated the mean over the time dimension. You could also calculate a spatial average over latitude and longitude by setting `dim=['lat', 'lon']`.

See the xarray documentation for more information https://docs.xarray.dev/en/stable/generated/xarray.DataArray.mean.html.

Is all of MERRA2 on AWS us-west-2? What about other NASA sources
Yes, all of MERRA-2 data are in AWS us-west-2, the location of the NASA Earthdata Cloud archive.

what about netcdf?
Yes, netcdf files are “streamable”.

Can the obtained temperatures be put over a map, for example using basemap?
You can use `basemap` to add state borders and a graticule. I prefer `cartopy` but use whatever you are most familiar with. `Xarray` has a nice example of plotting data on a map: https://docs.xarray.dev/en/stable/user-guide/plotting.html#maps. You could also save the mean temperature field to a file and bring it into GIS systems.

what does aws us-west-2 mean ? like you have to be within the west coast?
us-west-2 is the AWS region in which the data are stored (in S3 buckets). You do NOT need to be within the cloud or within a specific geographic region to access the data. You can still download or stream from out of region. Download continues to be free for all users. If you are able to access or set up an EC2 instance within AWS us-west-2, your compute will be next to the data, and streaming will be fast and inexpensive, compared to if you stream out of region. You can access AWS compute through the internet or terminal, so you do not need to physically be in that geographic region.

Does streaming also is useful, when we need constant changeable data such as weather data or atmospheric pressure data?

A:Yes, it can be useful when we have data that updates often, we don’t need to download and keep all the files locally. For exploration purposes streaming data allows us to get the data faster but it depends on the data format as well.

Given the same parameters (e.g. dates, variables) will streaming provide the same unique SHA? That is if we want to ensure a downloaded piece is documented in a reproducible way

Great question! The SHA can be inspected upfront, the files in cloud storage (in particular in AWS S3) have metadata that includes the SHA, “etag” but since we are not getting the full file we don’t have a way to verify the partial stream as SHAs are computed on the totality of the file.

SHA (Secure Hashing Algorithms) are a set of functions that create fixed-length string from any input. In the case of data from S3, they are used to ensure data integrity.

What analyses beyond plotting are supported by streaming?
In the demonstration we calculated the mean temperature and plotted it. But you can do whatever kinds of analysis you want to do; Principle Components Analysis, wavelet analysis. Tools like `xarray` offer some of these analysis methods. For others, you may have to use `scipy`. For other novel forms of analysis you might have to write your own code.

What is the best way for users to get access to us-west-2 to work in-region?

For in-region compute the easiest path is to partner with academic and research institutions that offer ready-to-use platforms like Jupyter hubs.
You can set up an EC2 instance within AWS us-west-2 on your own. An example can be found here:
https://nsidc.org/data/user-resources/help-center/accessing-nasa-earthdata-cloud-jupyterlab-ec2.
Note that depending on the compute and storage resources you choose for your instance, you may incur charges.

In agricultural remote sensing, we often combine climate data with multispectral imagery. Can `earthaccess` stream multiple datasets—say, MERRA-2 alongside Sentinel optical data—into a single aligned dataset on the fly for multimodal analysis?
Yes you can stream multiple datasets. However, you may be limited by bandwidth and memory. My suggestion would be to break your analysis into steps. Use streaming to access and process climate data in one step, and then save the processed data locally. Then access and process the Sentinel data in another step, and save that data. Then you are ready to analyze the two datasets.

What are some areas you would like to get contribution on?

See https://github.com/earthaccess-dev/earthaccess/issues for a list of issues. Look for a *good first issue* tag if you want a (hopefully) simple issue to get started. Or you can join one of our Collaboration Cafes. See https://earthaccess.readthedocs.io/en/stable/calendar/ for the next Collaboration Cafe.

is there a limit to how much data you are allowed to stream at once? for example, could i stream conus-wide imagery
If you are using cloud compute, you can scale your compute to meet your needs.

can this replace google earth engine totally in streaming data
This is not a replacement for Google EE, but an alternative. GEE is a platform with flexible compute of data that was processed by Google to be more analysis friendly. Streaming data with earthaccess means streaming the data as it exist in the NASA archives and further processing (reprojecting, regridding, scaling etc) may still be required.

Could I have a quick review on the registration process to use the data?
Refer to the earthdata login getting started page:
https://www.earthdata.nasa.gov/data/earthdata-login

related to reproducibility, can streaming credentials be saved in a database, so that the same url could be used later?
Great question! Yes, you can save the urls for the data you used. In the demonstration we saved the search results to a variable, `results`. You can get the urls for each file from results using the `data_links()` method that is part of the granules in the `results` list. In the demonstration I used a Python *list comprehension* to get the size of each granule and sum those sizes to get the total

download size. You can use the same approach to get the data links:
`[g.data_links() for g in results]`
You could then write these links to a file for later use.

We have plans to add this functionality to `earthaccess` but it this work is still in progress.

Is stream same as access? What are the download limits per day / per IP?
Streaming is an access pattern, yes. There are limits to concurrent requests and IPs can get throttled but there is no limit to the amount of data a user can stream or download.

if we use colab or jupyter so then we won't run of memory?
Yes, we can use Colab or any other compute platform and stream data from NASA with earthaccess, the benefit of cloud compute is that we can scale it to meet our needs.

To use the earthdata portal, do you have select the data accessibility (Giovanni, Harmony API, etc. to retrieve data via streaming)?
No you do not need to select data accessibility. `earthaccess` is just another way to access NASA data. You can use the approach we demonstrated in the webinar to access any dataset. You might want to use Giovanni or Worldview to explore datasets first. Once you know which dataset you want and you know the `short_name`, then you can use `earthaccess` to search for granules and access that data.

How do we know what datasets are available for streaming and what their names are?
You can find the `short_name` of a dataset from the dataset landing page or you can use `earthaccess.search_datasets`. The landing page will also tell you the file format used for the data. See the answer to the question below for a list of file formats that can be streamed.

Does the streaming work for all NASA data?
It depends on the data format more than the science behind the data. If it's netcdf4, hdf5, COG, zarr, it is streamable.

Do the docs have references and names to which datasets are available? eg humidity, CO2...?
You can explore and learn about the NASA Earthdata archive at earthdata.nasa.gov