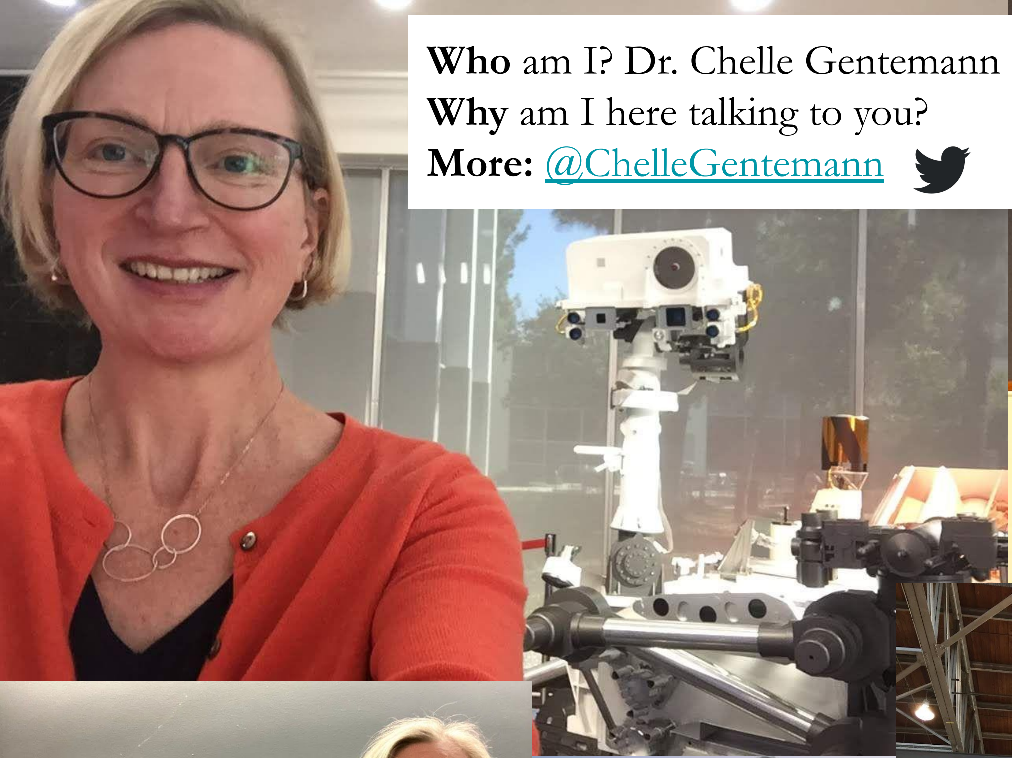
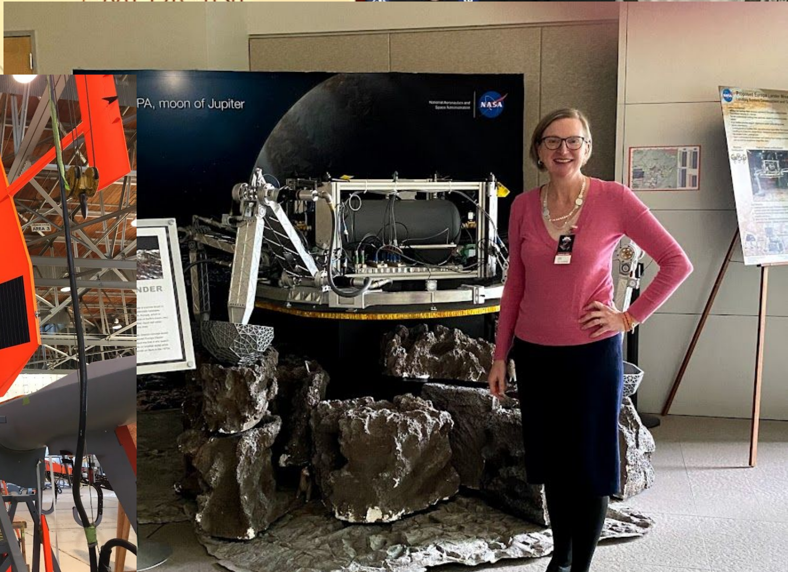

WE
PERSEVERE

Who am I? Dr. Chelle Gentemann
Why am I here talking to you?
More: [@ChelleGentemann]





The foundation of climate change: the Greenhouse Effect

who discovered the greenhouse effect

All News Images Videos Maps More Tools

About 20,400,000 results (0.71 seconds)

John Tyndall

Hear this out loud

John Tyndall set the foundation for our modern understanding of the greenhouse effect, climate change, meteorology, and weather. But did he 'discover' it? 160 years ago, on 18 May 1859, the Irish physicist John Tyndall wrote in his journal 'the subject is completely in my hands'. May 17, 2019

ART. XXXI.—*Circumstances affecting the Heat of the Sun's Rays;* by EUNICE FOOTE.

(Read before the American Association, August 23d, 1856.)

MY investigations have had for their object to determine the different circumstances that affect the thermal action of the rays of light that proceed from the sun.

Several results have been obtained.

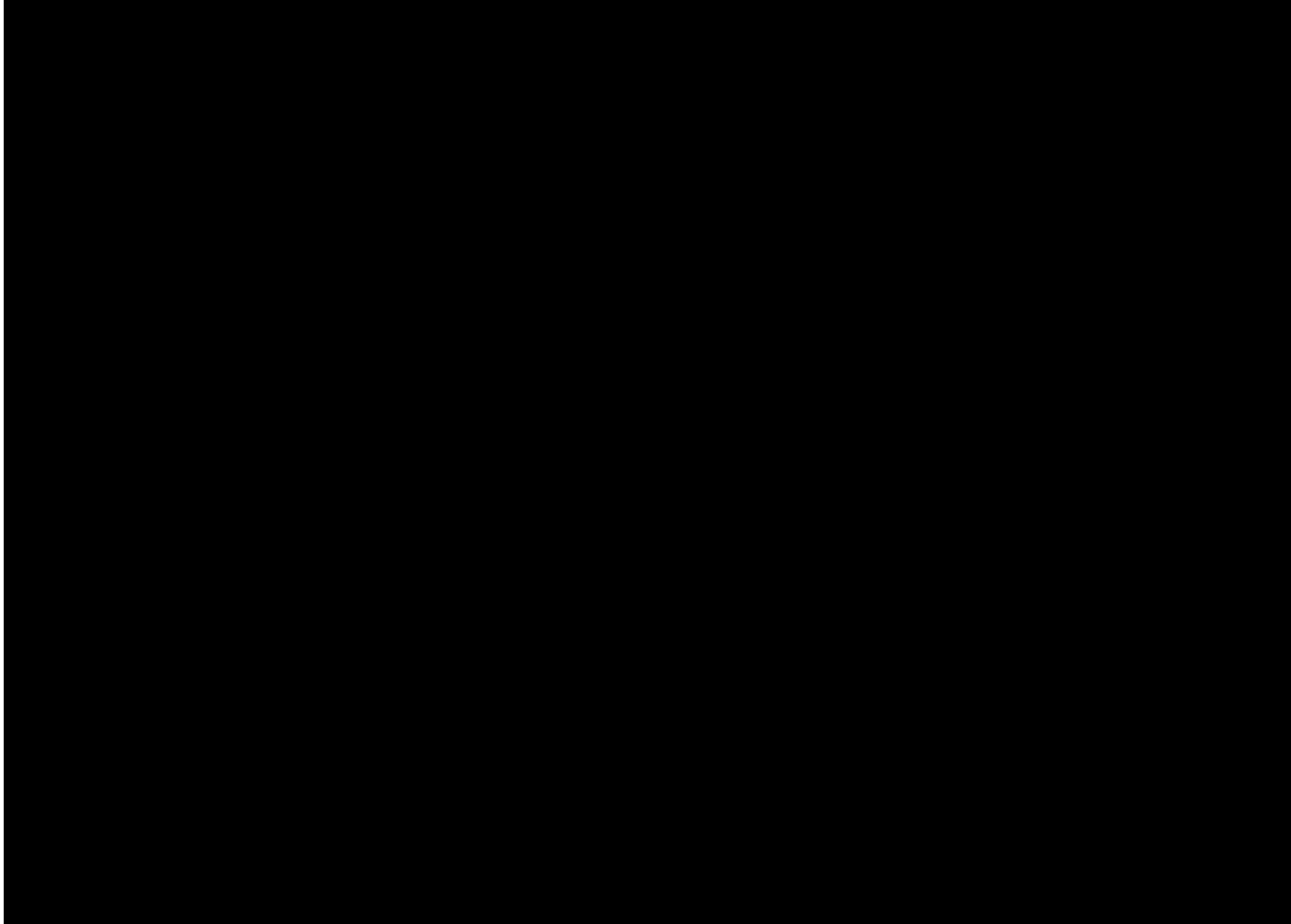
First. The action increases with the density of the air, and is diminished as it becomes more rarified.

The experiments were made with an air-pump and two cylindrical receivers of the same size, about four inches in diameter and thirty in length. In each were placed two thermometers, and the air was exhausted from one and condensed in the other. After both had acquired the same temperature they were placed in the sun, side by side, and while the action of the sun's rays rose to 110° in the condensed tube, it attained only 88° in the other. I had no means at hand of measuring the degree of condensation or rarefaction.

The observations taken once in two or three minutes, were as follows:

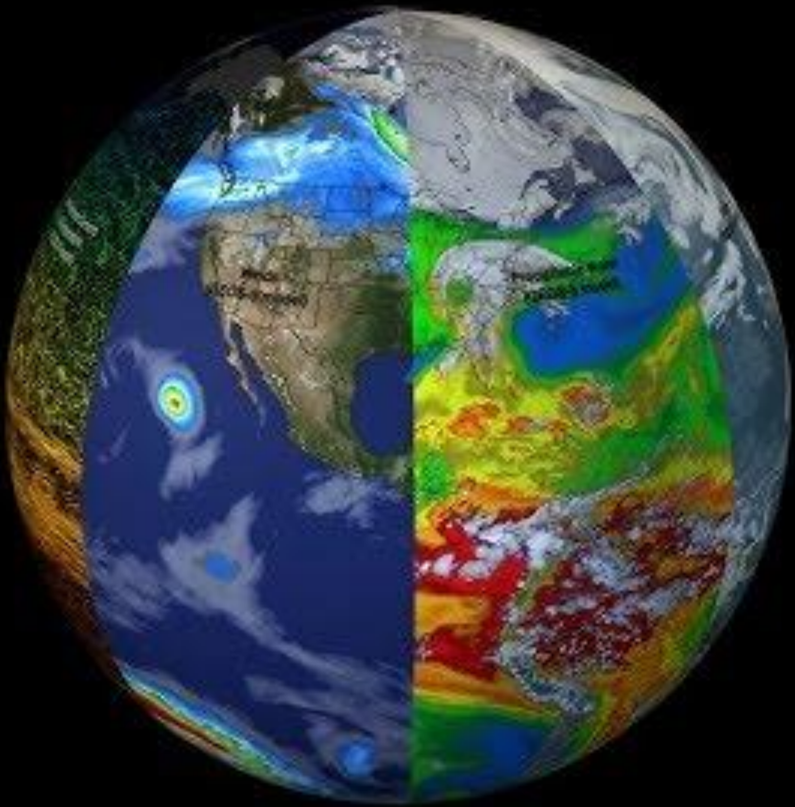
Exhausted Tube		Condensed Tube.	
In shade.	In sun.	In shade.	In sun.
75	80	75	80
76	82	78	95

Climate Change



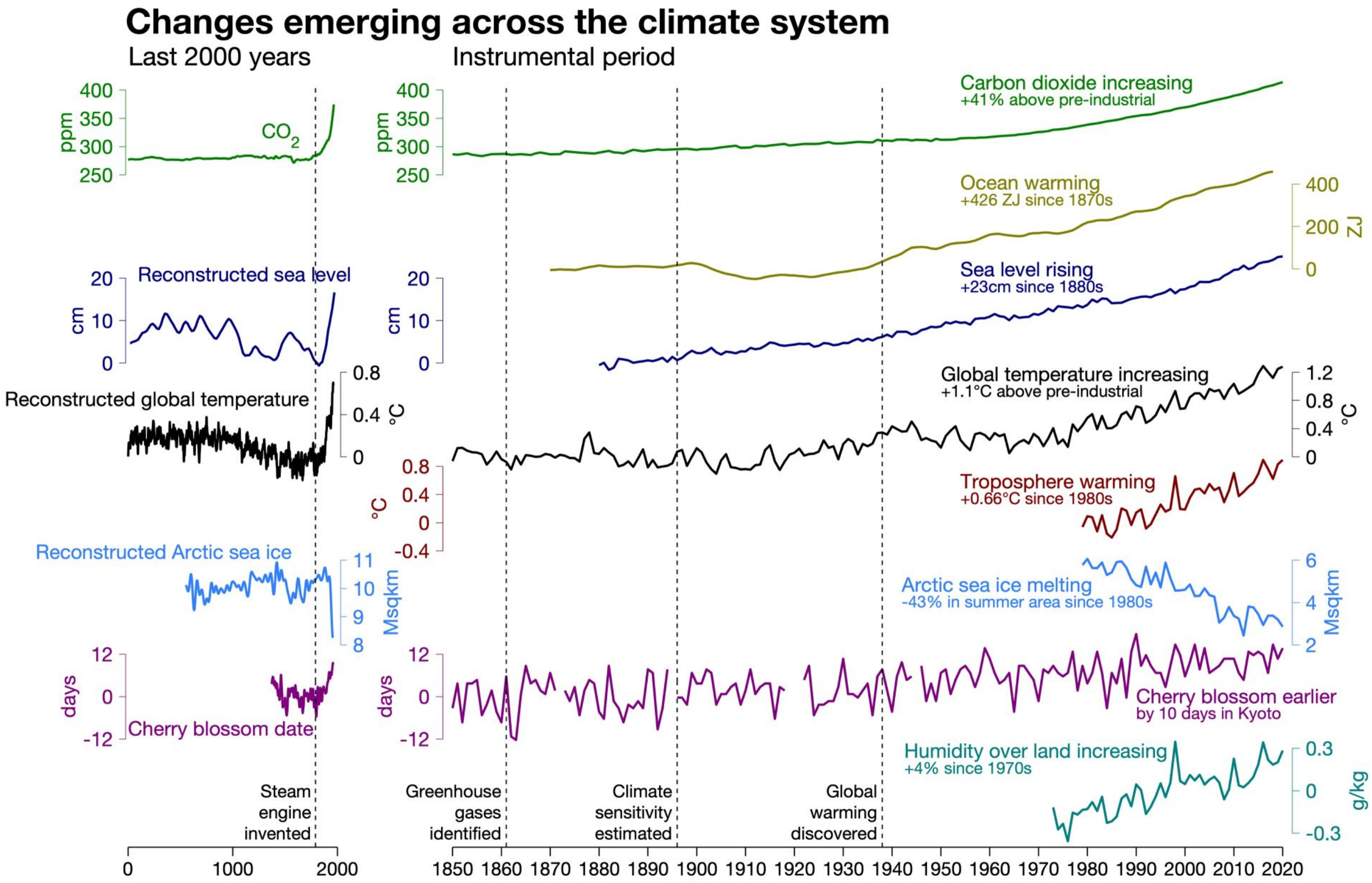
A simplified animation of the greenhouse effect. Credit: NASA/JPL-Caltech

Video credit:
NASA SVS



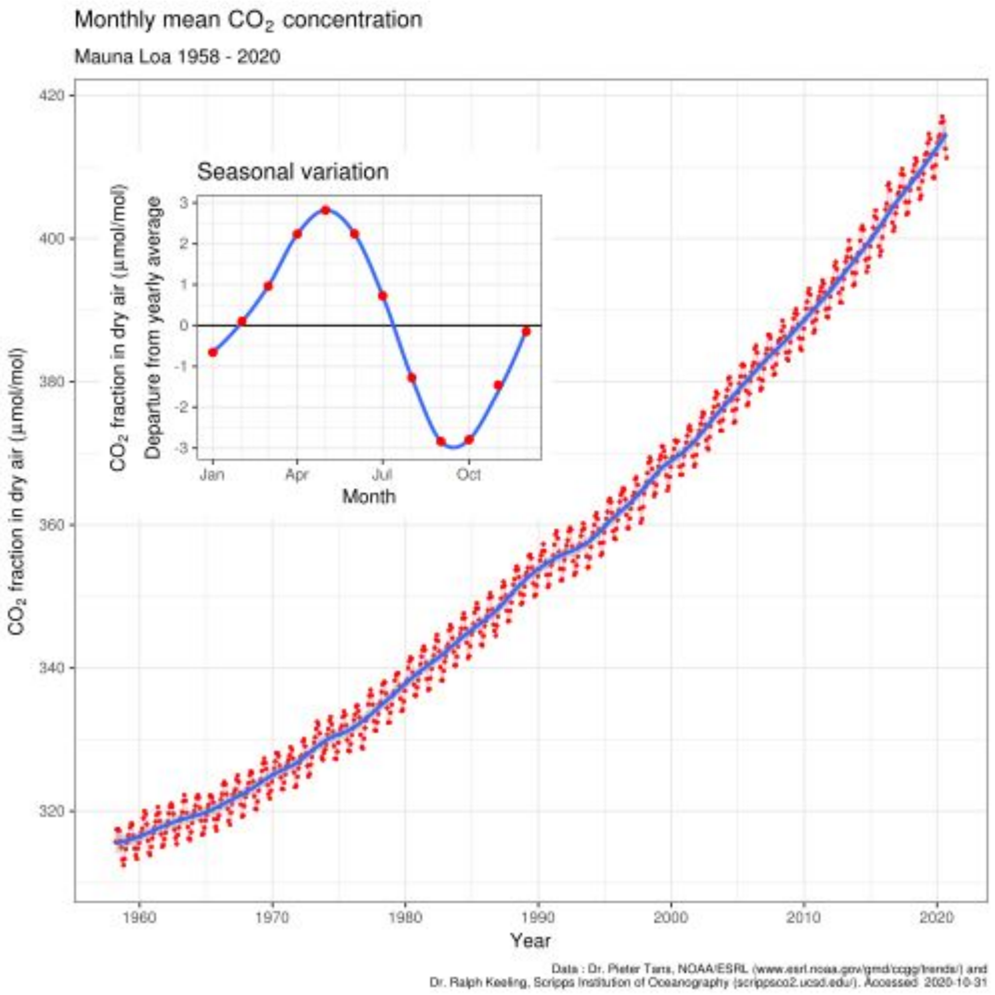
Video credit:
NASA SVS

What does the data say about our climate?

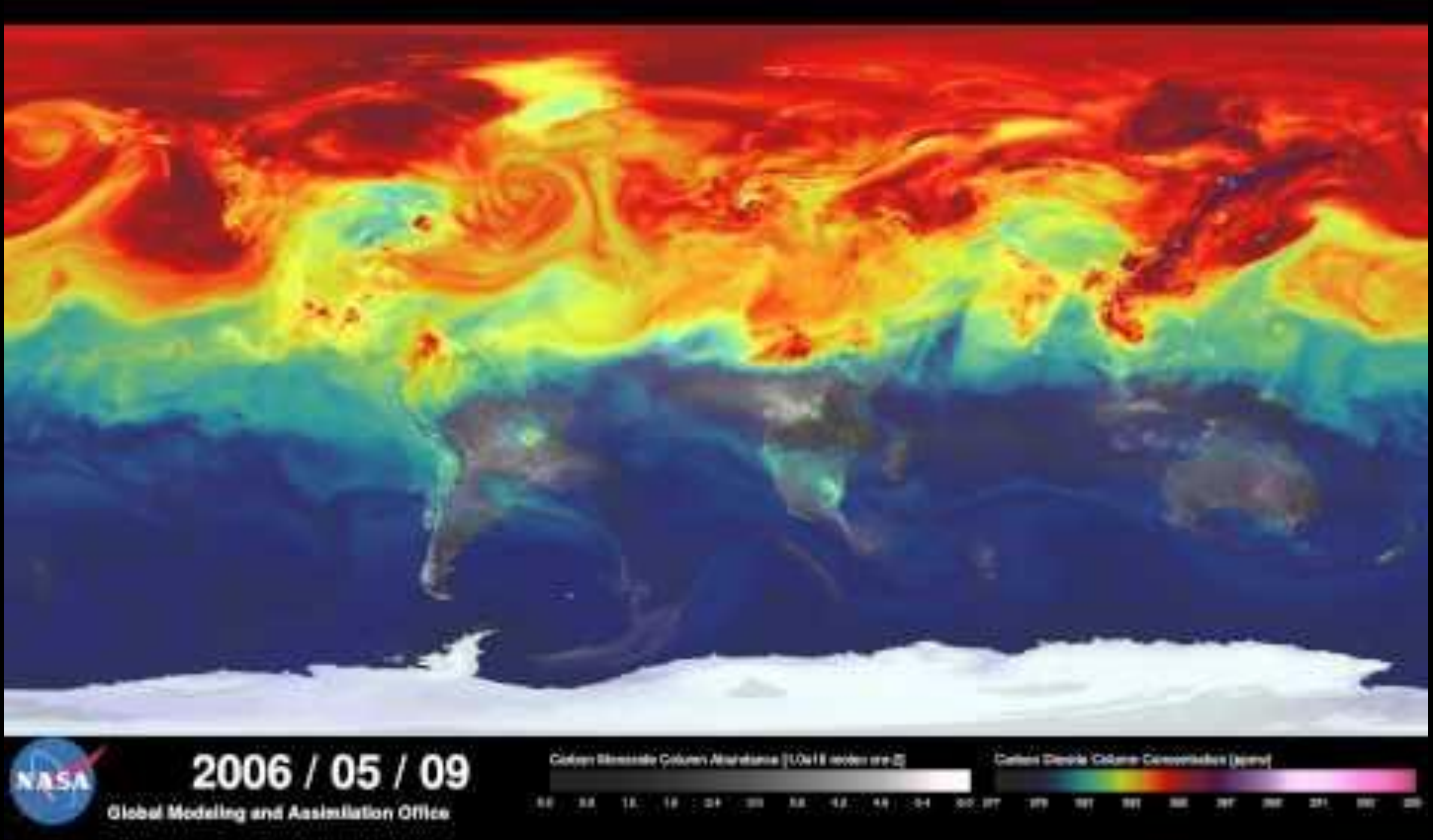


Keeling Curve

- In 1958 Keeling got a grant to begin monitoring CO2 in Hawaii.
- Roger Revelle (a famous scientist) argued that they just needed a snapshot - CO2 was too variable - and another snapshot 20 years later to show that CO2 was increasing
- Keeling advocated for precise measurements over time. By the mid-1960s we had both a measurement of the Earth’s breathing and the global increase in CO2.
- How would you remake this figure?



https://pubs.acs.org/doi/10.1021/ac1001492



Video credit:
NASA SVS

Calculating the Greenhouse Effect

Goal 1: Calculate how the temperature is changing with increasing CO2.

The Planetary energy balance:

Energy (from the Sun) absorbed = Energy emitted

$$E_{in} = E_{out}$$

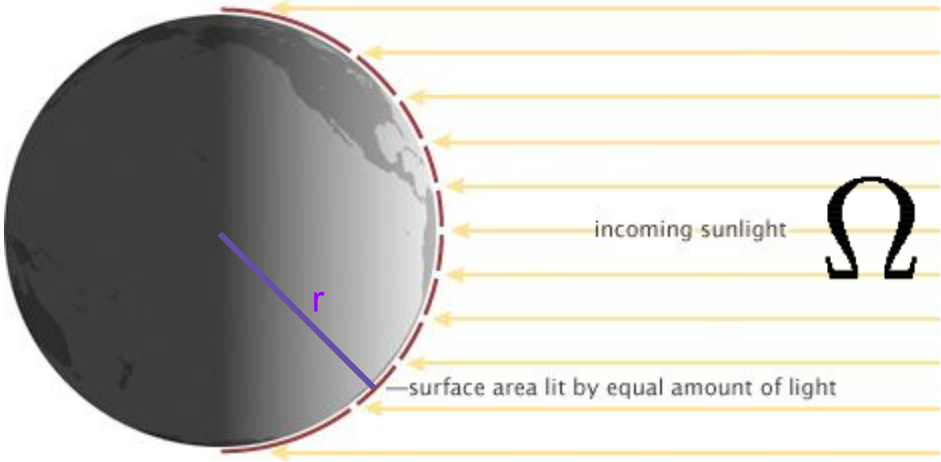
How does CO2 affects this?

A) The Sun emits radiation that is absorbed by the Earth (~30% is reflected by clouds, ice/snow, desert, this is the albedo)
B) The Earth emits radiation according to Stephan-Boltzman's Law: the rate that a body emits radiation (per unit area) is directly proportional to the body's absolute temperature to the fourth power (blackbody radiation)
C) The emitted radiation doesn't all go back into space……..

$$\boxed{E_{out} = \sigma T^4}$$

Calculate energy in

A) The Sun emits radiation Ω absorbed by the Earth



E_{in} Energy in equals the incoming sunlight (W/m^2) multiplied by the area (m^2) to get W

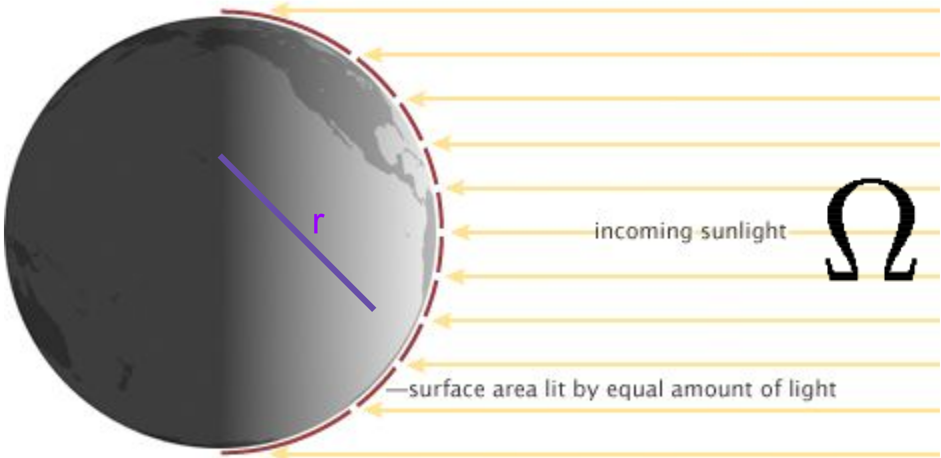
E_{in} = Incoming sunlight x area

$$E_{in} = \Omega \pi r^2$$

NASA illustrations by Robert Simmon

Some of the energy in is reflected by the atmosphere

A) The Sun emits radiation absorbed by the Earth (some is reflected by the atmosphere)



Albedo = the fraction of radiation reflected back to space by the atmosphere

The amount that gets through:

$$(1-A)$$

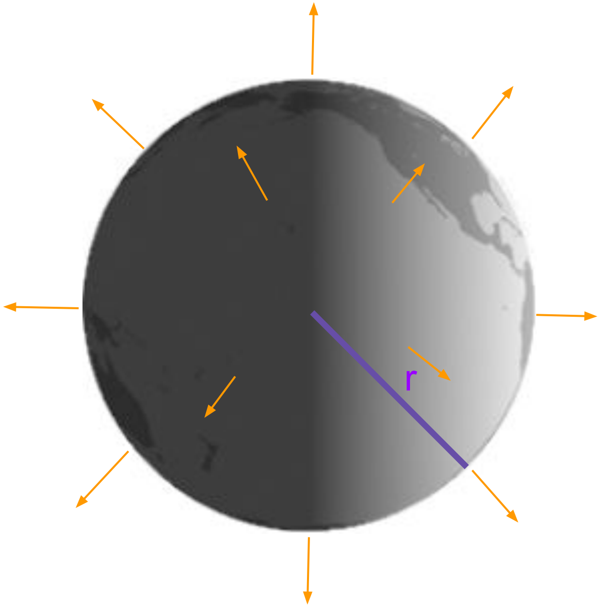
$$E_{in} = \Omega(1-A)\pi r^2$$

Reflective surface

NASA illustrations by Robert Simmon

Calculate Energy out

B) The Earth emits blackbody radiation



Energy out equals the emitted radiation (W/m^2) multiplied by the area (m^2) to get W

$$E_{out}$$

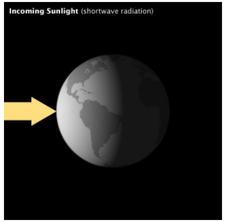
E_{out} = Outgoing radiation x area

$$E_{out} = \sigma T^4 4\pi r^2$$

NASA illustrations by Robert Simmon

Energy in = Energy out

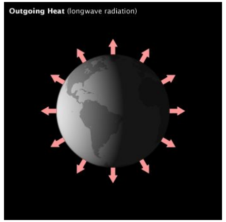
Goal 1: Calculate how the temperature is changing with increasing CO2



$$E_{in} = \Omega(1 - A)\pi r^2$$

Reflective surface

$$E_{out} = \sigma T^4 4\pi r^2$$



$$E_{in} = E_{out}$$

Planetary energy balance

$$\Omega(1 - A)\pi r^2 = \sigma T^4 4\pi r^2$$

$$T = \sqrt[4]{\frac{\Omega(1 - A)}{4\sigma}}$$

NASA illustrations by Robert Simmon

Calculating temperature

Goal 1: Calculate how the temperature is changing with increasing CO2

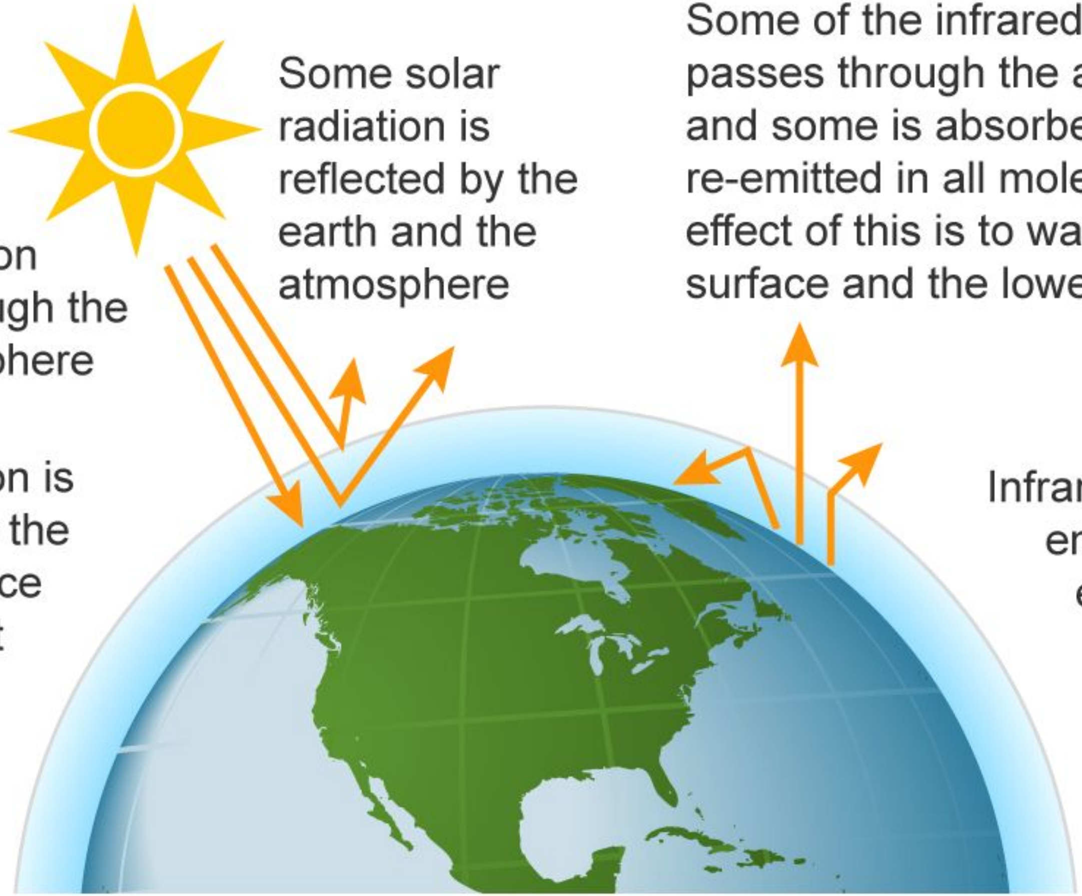
What is the Earth's temperature?

$$\Omega = 1372 \text{ Wm}^{-2}$$
$$\text{A} = 0.3$$
$$\sigma = 5.67\text{e}^{-8} \text{ Wm}^{-2}\text{K}^{-4}$$

$$T = \sqrt[4]{\frac{\Omega(1-A)}{4\sigma}}$$

~255 K ~-16 C ~1 F

The greenhouse effect



Our atmosphere is like a blanket

$$E_{in} = E_{out}$$ Planetary energy balance

$$\Omega(1-A)\pi r^2 = \sigma T^4 4\pi r^2$$ without an atmosphere

$$\Omega(1-A)\pi r^2 + \boxed{\Delta E 4\pi r^2} = \sigma T^4 4\pi r^2$$ **with** an atmosphere

$\Omega = 1372\ \mathrm{Wm^{-2}}$

$A = 0.3$

$\sigma = 5.67e^{-8}\ \mathrm{Wm^{-2}K^{-4}}$

$T = 288\ \mathrm{K}$

- Solve for the greenhouse effect!
- What happens to the temperature if we increase the greenhouse effect?
- What happens to the temperature if we decrease/increase the albedo?

Calculating the Temperature dependence on CO2

Goal 1: Calculate how the temperature is changing with increasing CO2

CO2 is ~380ppm (parts per million)

$$\Omega(1-A)\pi r^2 + \Delta E 4\pi r^2 = \sigma T^4 4\pi r^2$$

$$\Delta E = 133.26 + 0.044 \times CO2$$

As CO2 increases what happens to the temperature?

Plot the results

Use the equation to calculate the increase in temperature with time due to the increase in CO2

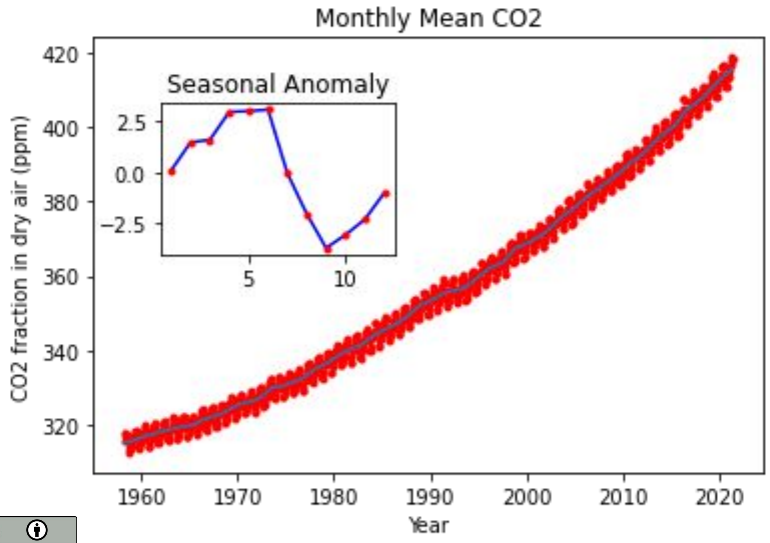
$$\Omega(1-A)\pi r^2 + \Delta E 4\pi r^2 = \sigma T^4 4\pi r^2$$

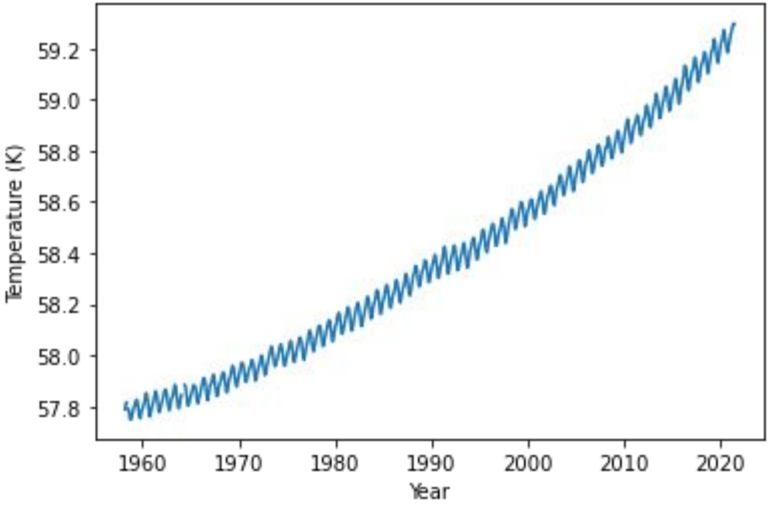
$\Omega = 1372\ Wm^{-2}$

$A = 0.3$

$\sigma = 5.67e^{-8}\ Wm^{-2}K^{-4}$

$$\Delta E = 133.26 + 0.044 \times CO2$$





Global warming is a climate crisis

The equation provides solutions to global warming:

$$\Omega(1-A)\pi r^2 + \Delta E 4\pi r^2 = \sigma T^4 4\pi r^2$$

Change the planetary albedo to reflect more radiation to space (increase aerosols, clouds, make surface more reflective)

Reduce greenhouse gases (CO2, Methane)

RESEARCH ARTICLE | ATMOSPHERIC SCIENCE

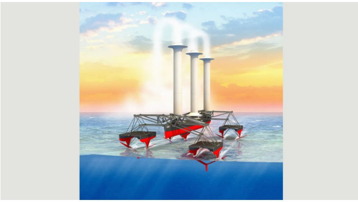
Toward practical stratospheric aerosol albedo modification: Solar-powered lofting

RU-SHAN GAO, KAREN H. ROSENLOF, BERND KÄRCHER, SIMONE TILMES, OWEN B. TOON, CHRISTOPHER MALONEY, AND PENGFEI YU Authors Info & Affiliations

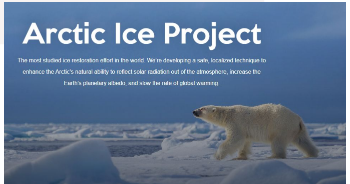
SCIENCE ADVANCES • 14 May 2021 • Vol 7, Issue 20 • DOI: 10.1126/sciadv.abe3416

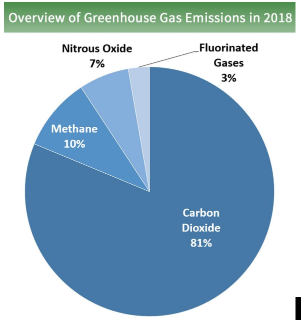
Real Ice

Real Ice pledges to help Indigenous people obtain re-icing machines that can increase ice thickness and re-strict ice melt in Arctic regions. We aim to achieve this by replenishing Arctic ice, using concept tested, wind powered, re-icing machines.



Stephen Salter believes that a fleet of 300 of his autonomous ships could reduce global temperatures by 1.5C (Credit: James MacNeill)



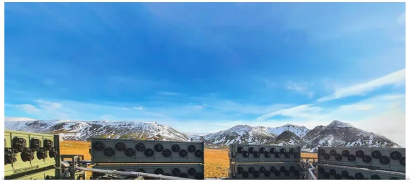


U.S. Environmental Protection Agency (2020). Inventory of U.S. Greenhouse Gas Emissions and Sinks: 1990-2018

Climeworks Turns On the World's Largest Carbon Capture and Storage Plant

The Iceland operation can remove 4,400 tons of CO2 from the air each year.

By Lloyd Alter | Fact checked by Haley Mast on September 14, 2021 12:34PM EDT



Published: 25 May 2017

Iron-dumping ocean experiment sparks controversy

Jeff Tollefson

Nature 545, 393–394 (2017) | Cite this article

510 Accesses | 4 Citations | 343 Altmetric | Metrics

AR6 Climate Change 2021: The Physical Science Basis

Changing by Alisa Singer
"As we witness our planet transforming around us we watch, listen, measure … respond."
www.environmentalgraphiti.org – 2021 Alisa Singer.

[Credit: NASA]

“Recent changes in the climate are widespread, rapid, and intensifying, and unprecedented in thousands of years.

[Credit: Peter John Maridable | Unsplash]

"Unless there are immediate, rapid, and large-scale reductions in greenhouse gas emissions, limiting warming to 1.5°C will be beyond reach.

[Credit: Yoda Adaman | Unsplash]

"It is indisputable that human activities are causing climate change, making extreme climate events, including heat waves, heavy rainfall, and droughts, more frequent and severe.

INTERGOVERNMENTAL PANEL ON climate change

[Credit: Hong Nguyen | Unsplash]

“Climate change is already affecting every region on Earth, in multiple ways.

The changes we experience will increase with further warming.

ipcc
INTERGOVERNMENTAL PANEL ON climate change

WMO UNEP

[Credit: Jenn Caselle | UCSB]

“There’s no going back from some changes in the climate system…

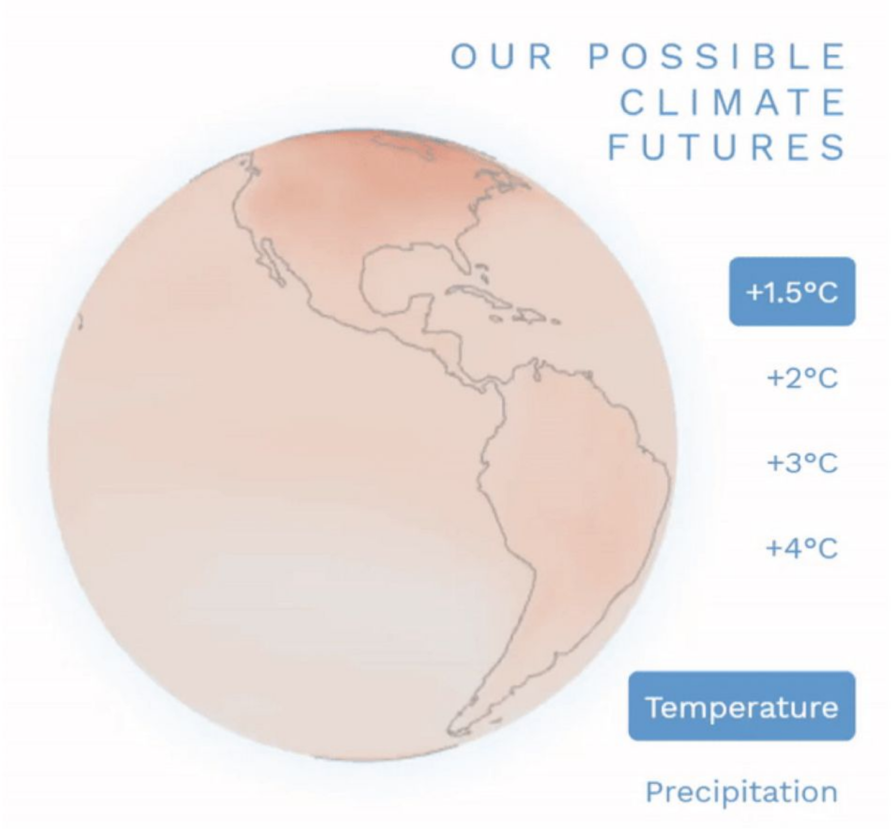
ipcc
INTERGOVERNMENTAL PANEL ON climate change
WMO UNEP

[Credit: Andy Mahoney | NSIDC]

“…However, some changes could be slowed and others could be stopped by limiting warming.

INTERGOVERNMENTAL PANEL ON climate change

Interactive atlas



https://interactive-atlas.ipcc.ch/

#IPCCData #IPCCAtlas

BY THE NUMBERS

Author Team

234 authors from **65** countries

28% women, **72%** men

30% new to the **IPCC**

Review Process

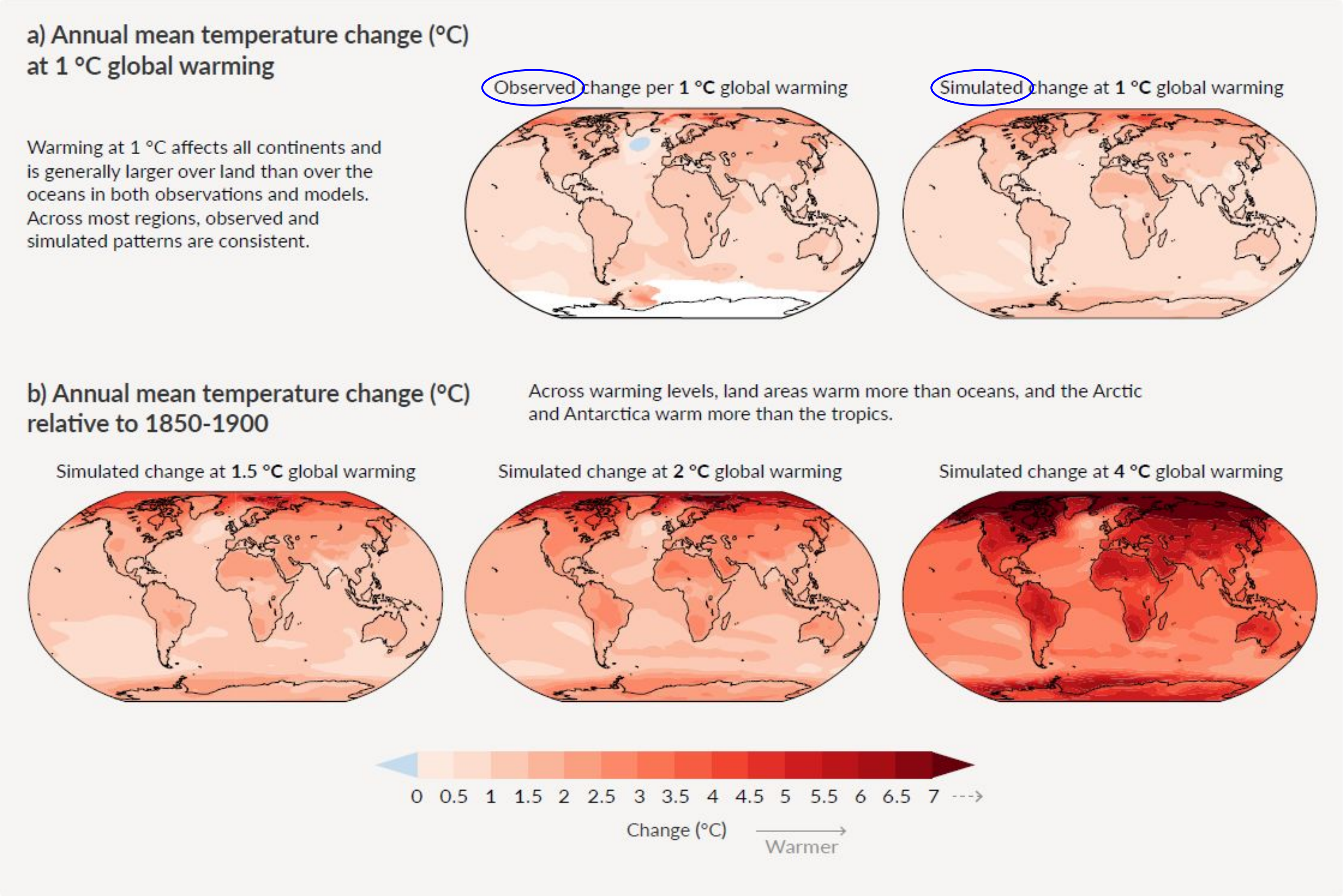
14,000 scientific publications assessed

78,000+ review comments

46 countries commented on Final Government Distribution

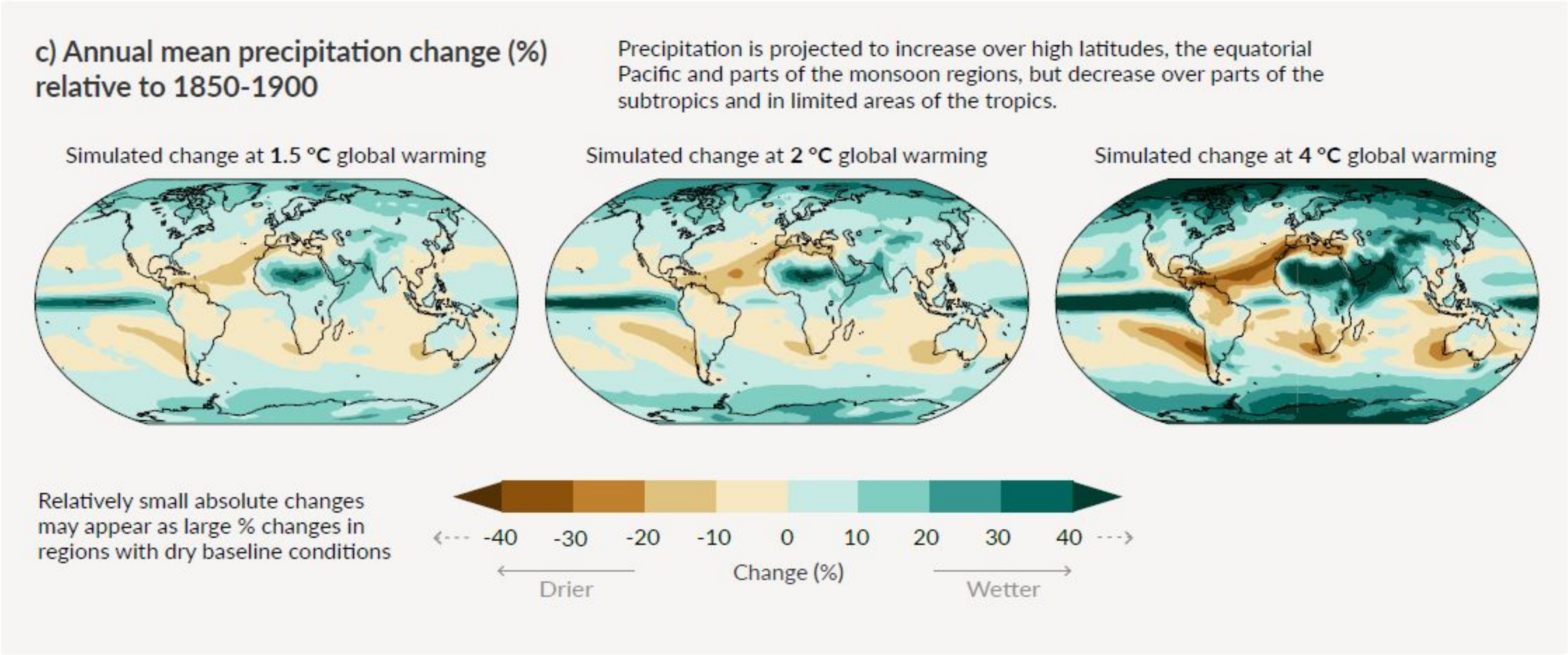

Image credit: NOAA

With every increment of global warming, changes get larger



https://www.ipcc.ch/report/ar6/wg1/#FullReport

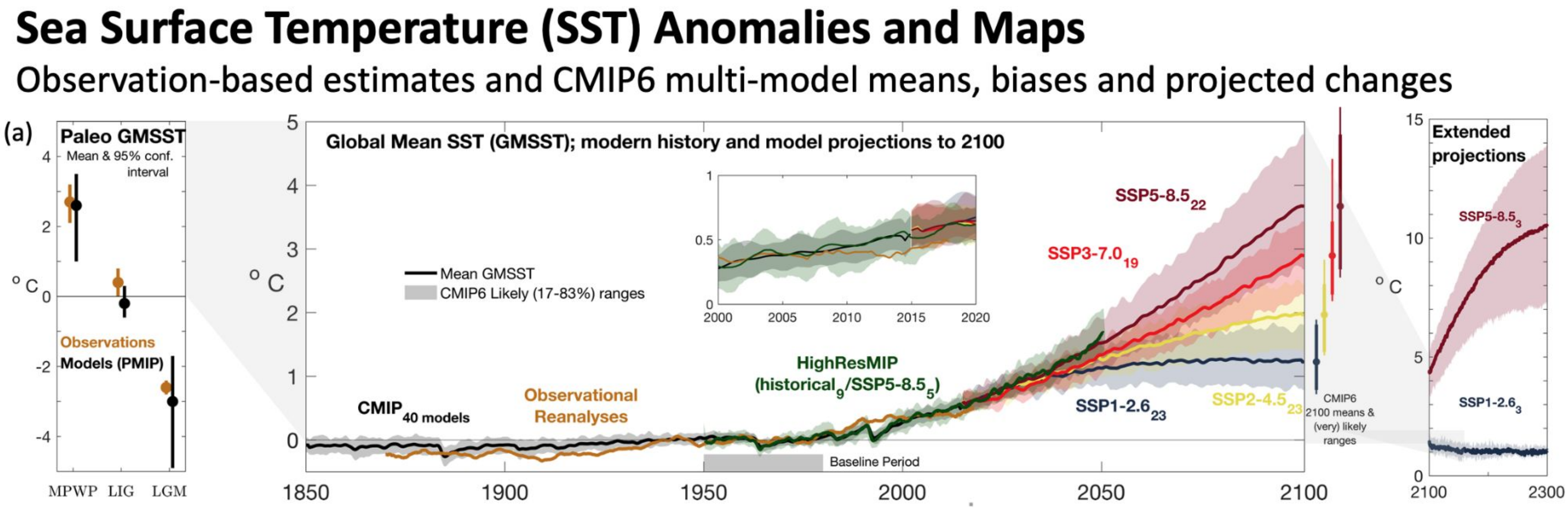
… in precipitation



CC-relationship water vapor - temperature - pressure
https://www.jbarisk.com/news-blogs/the-physics-of-precipitation-in-a-warming-climate/

https://www.ipcc.ch/report/ar6/wg1/#FullReport

Our trajectory from data and models



https://raw.githubusercontent.com/BrodiePearson/IPCC_AR6_Chapter9_Figures/main/Plotting_code_and_data/Fig9_03_SST/Fig9_03_SST.png

Extremes are the new normal

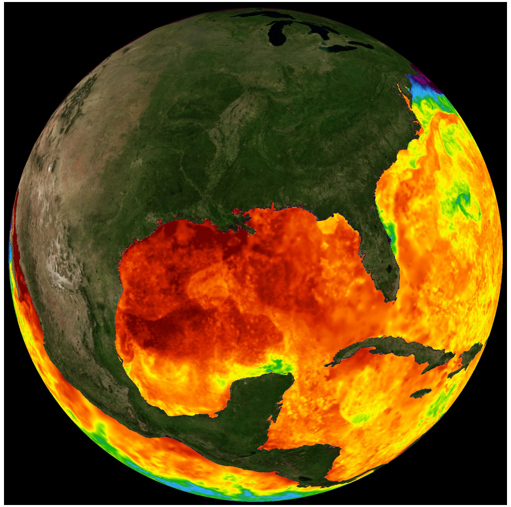
ENVIRONMENT | NEWS

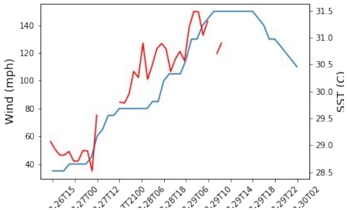
How climate change is fueling hurricanes like Ida

Hurricanes feed off heat, a growing source of fuel in a warming world.

BY SARAH GIBBENS

PUBLISHED AUGUST 31, 2021 • 8 MIN READ





The Creek Fire, in the Sierra National Forest in California, has burned hundreds of thousands of acres. Its spread was fueled by the preesnce of many dead, super dry trees; climate change contributed to both their death and their dryness.
PHOTOGRAPH BY STUART PALLEY, NATIONAL GEOGRAPHIC

SCIENCE | NEWS

The science connecting wildfires to climate change

A heating-up planet has driven huge increases in wildfire area burned over the past few decades.

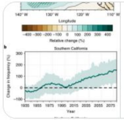
Climate crisis likely creating extreme winter weather events, says report

Arctic change increased chances of tightly spinning winds above North Pole, authors say, boosting chances of extreme weather

▲'How could ma
arguing,' said Juc
Xinhua/Rex/Shut

Daniel Swain
@Weather_West

It is worth noting that this exact situation--an extremely strong atmospheric river bringing brief period of record rainfall in midst of severe and temperature-amplified drought--is what we expect to see in California with #ClimateChange. #CAwx #CAwater



nature.com
Increasing precipitation volatility in...
Nature Climate Change - California recently experienced a rapid shift from multi-year drought ...

6:08 PM · Oct 24, 2021 · Twitter Web App

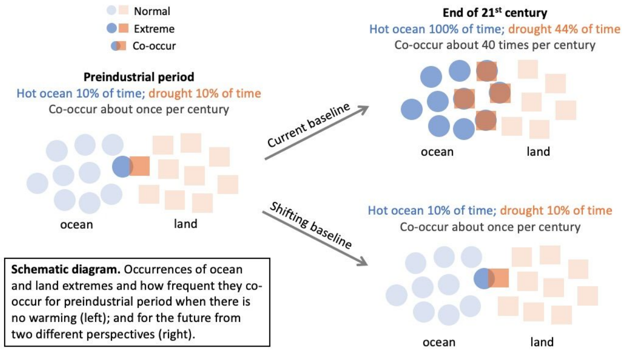
Extremes are the new normal

A decade ago - scientists would argue - we can't attribute any single weather event to climate.

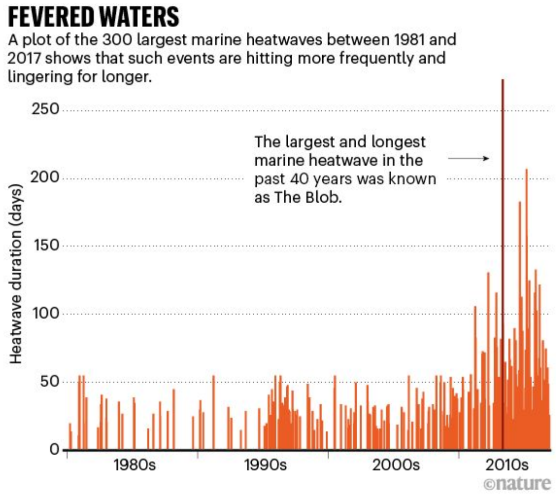
In the last decade, we have all experienced major shifts in our climate through changes in our local weather and scientists have figured out 'climate event attribution'

They look at the probability of the occurrence of an event (eg. a temperature extreme) in models run without human-influence and then compare it to the probability in models run with human-influence.

How?



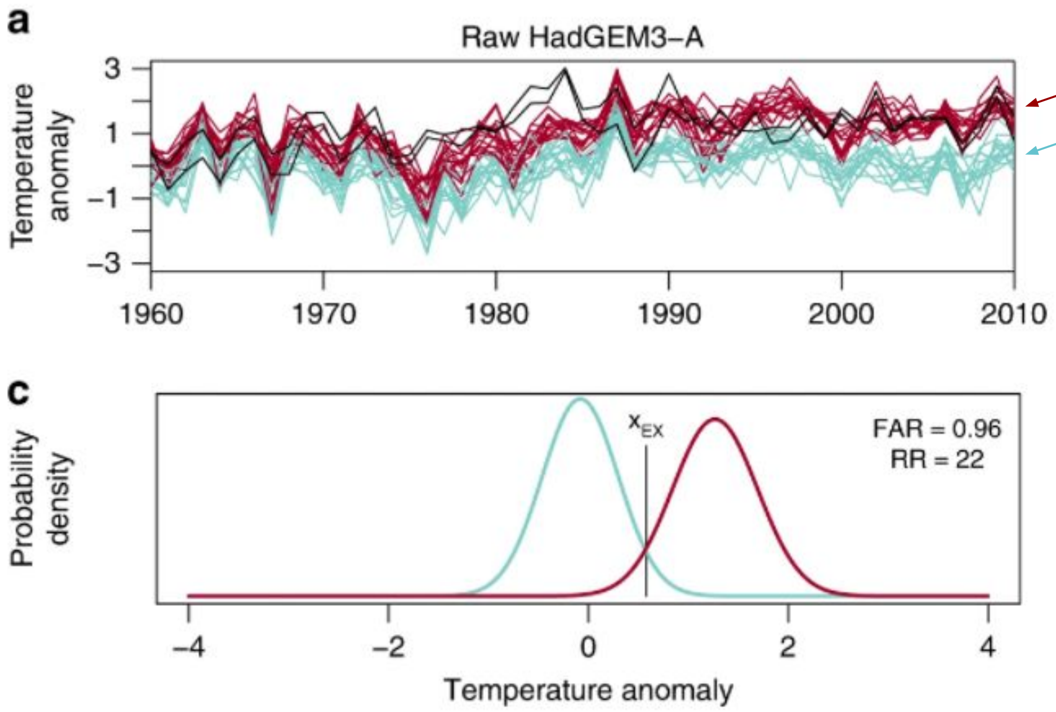
https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2021GL092765



https://www.nature.com/articles/d41586-021-01142-4

Extremes are the new normal

The probability of an event changes



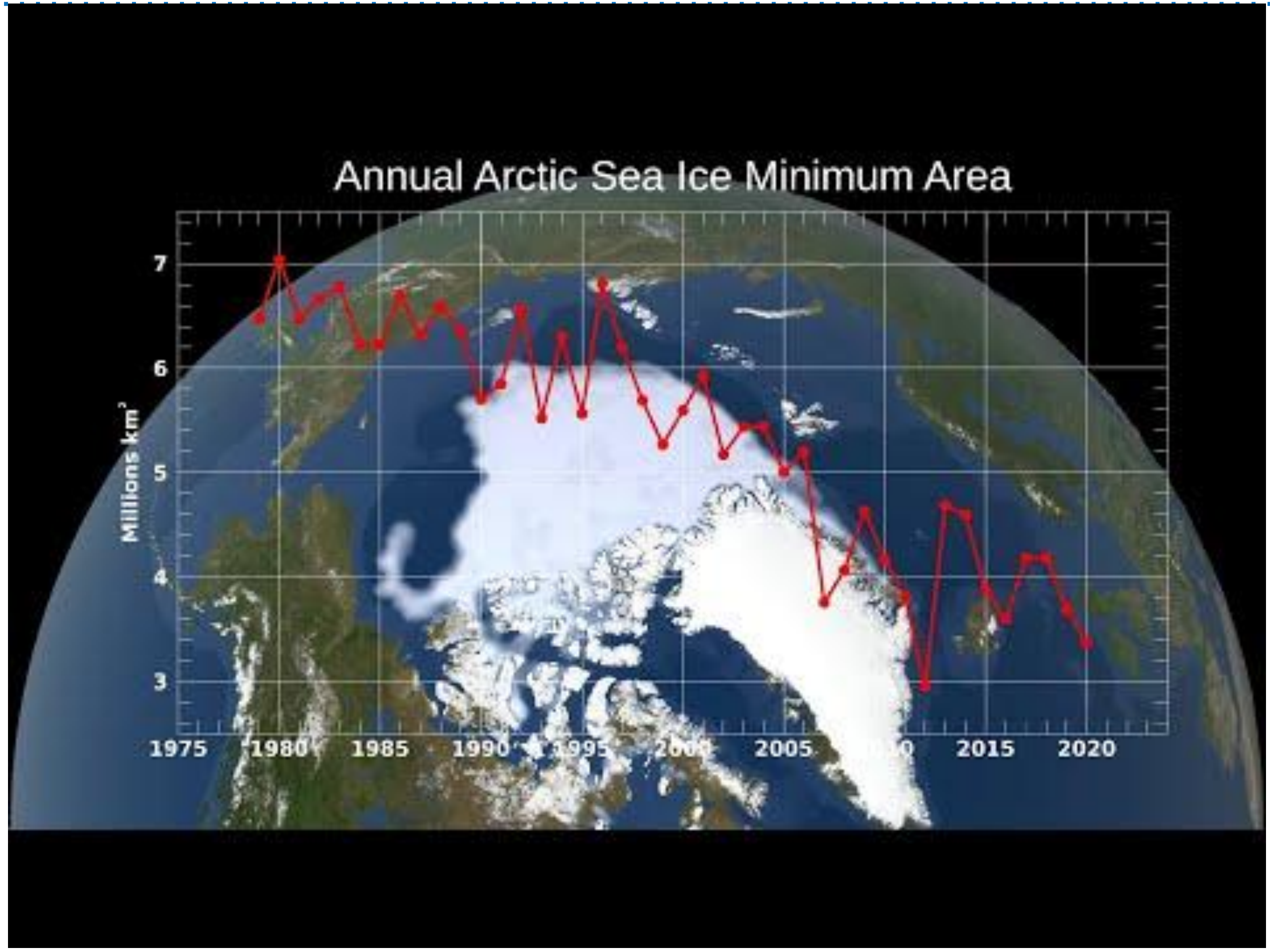
Model runs with all forcings

Model runs with only natural

How likely is it that there will be a 2 degree anomaly?

https://www.nature.com/articles/s41467-019-09729-2/figures/1

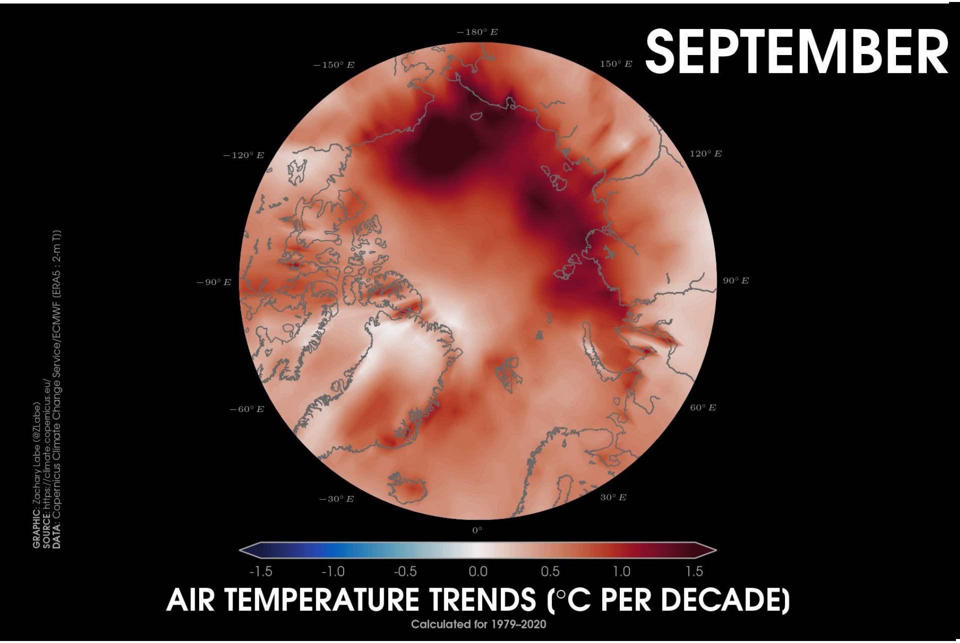
Positive Feedbacks in the Climate System

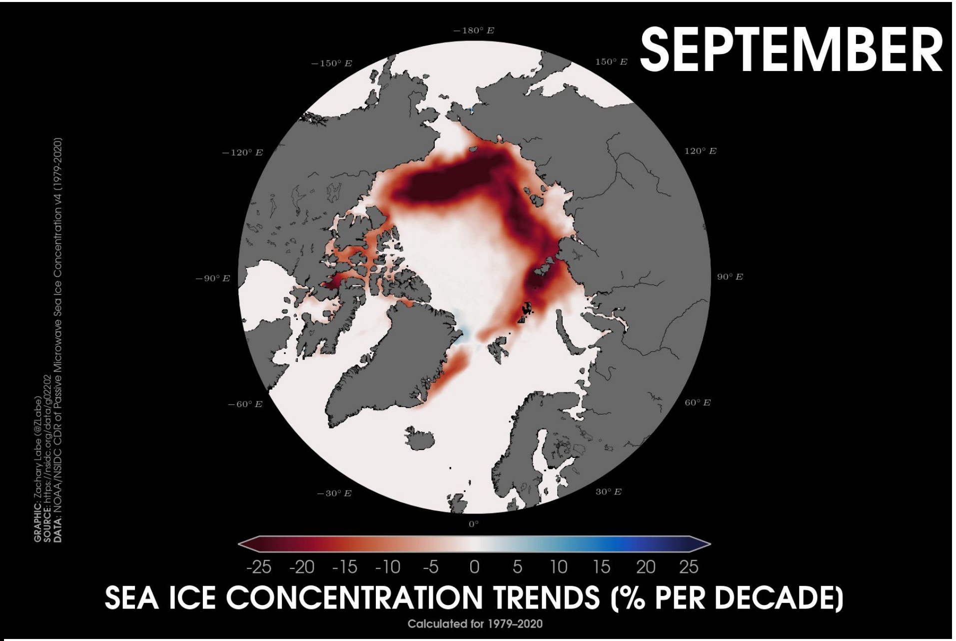


Goal: Positive feedbacks in the climate system

The Earth is a system - a complicated game of dominos

As air temperature increases, sea ice concentration decreases…..





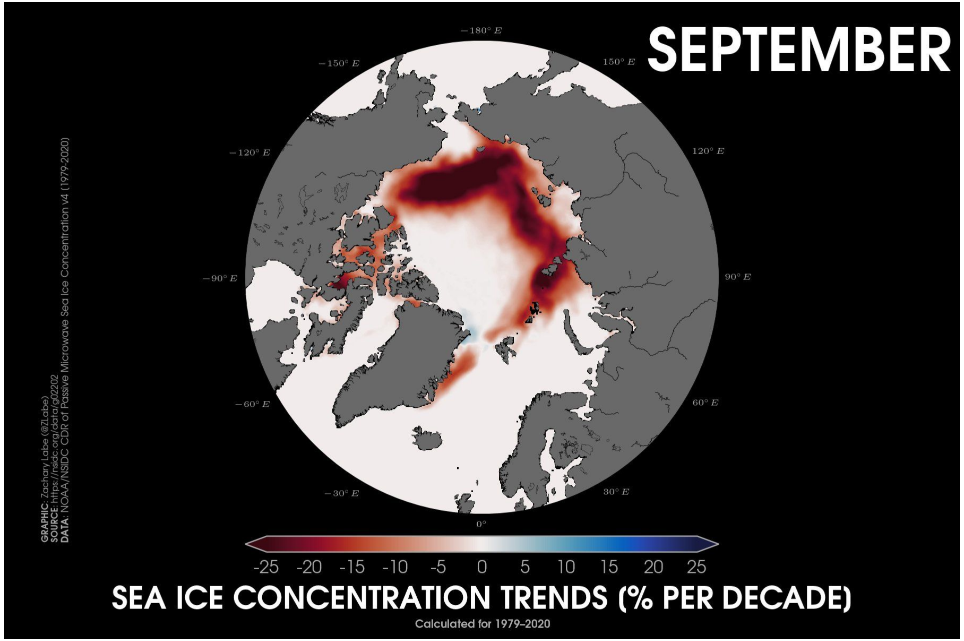
Goal: Arctic Amplification: a positive feedback

A positive feedback amplifies the initial perturbation

reflectivity!

Albedo feedback:

1. The albedo of sea ice is ~0.5-0.7. Most sunlight is reflected back to space.
2. The albedo of the ocean is 0.04. Most sunlight is absorbed and warms the seawater.
3. A warmer ocean melts more sea ice.

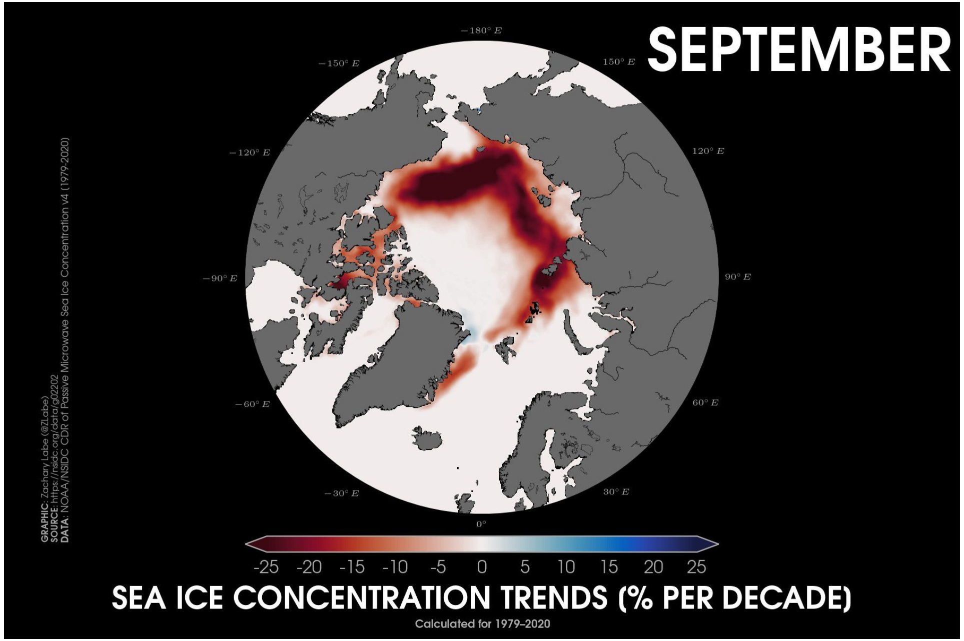


Goal: Arctic Amplification: a positive feedback

A positive feedback amplifies the initial perturbation

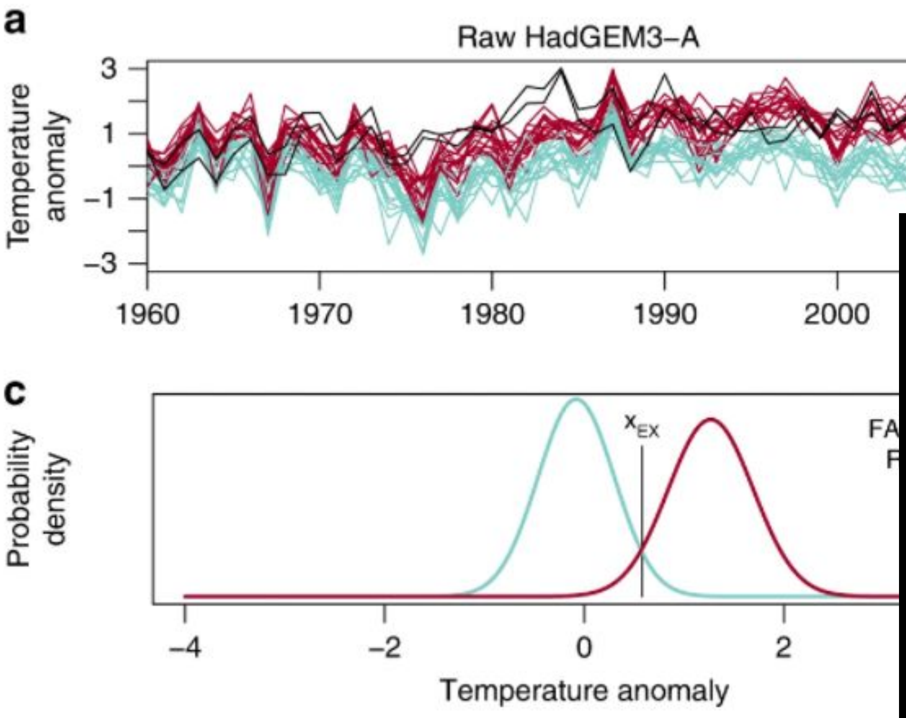
Cloud feedback:

1. As seaice melts, more ocean is exposed and more moisture is absorbed by the atmosphere
2. More moisture == more clouds
3. Clouds trap longwave radiation, warming the Arctic, melting more seaice.



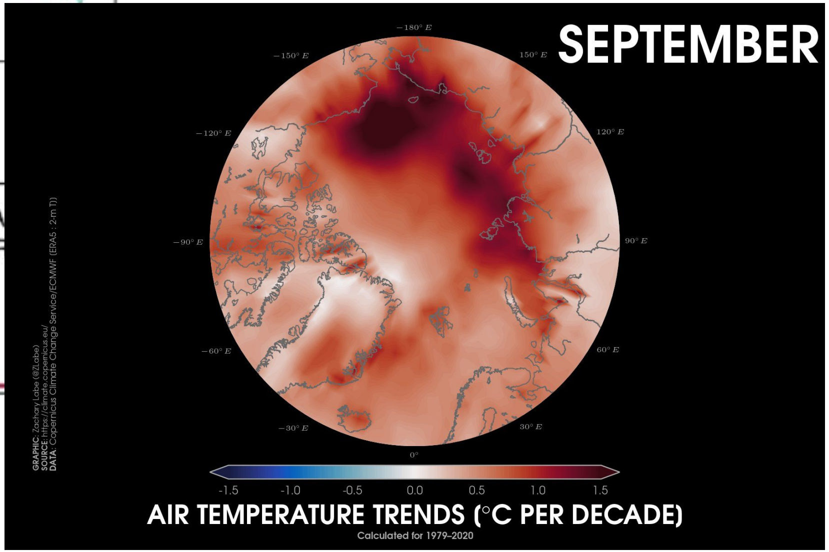
Can we look at this ourselves? --- this is global data --- 3D data

….. Latitude, longitude, and time…. How do we handle that?



Model runs with all forcings

Model runs with only natural



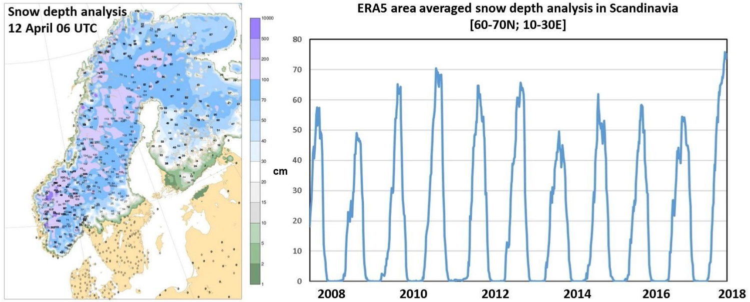
https://www.nature.com/articles/s41467-019-09729-2/figures/1

Learning objective: climate data analysis

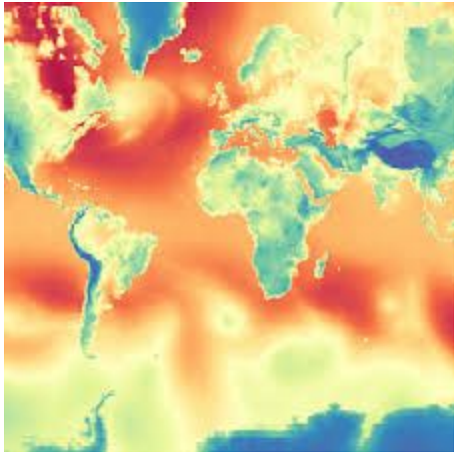
- 3D data (time,latitude,longitude)
- Introduction to Xarray python library
- Xarray Probability density functions
- Xarray linear regression: calculating mean trends
- Xarray global analysis of trends

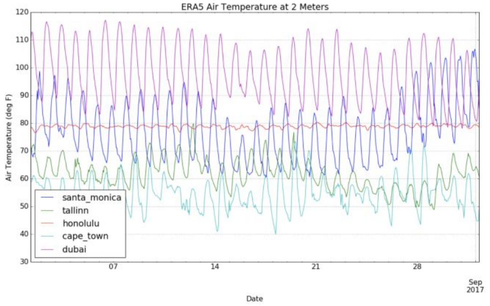
ERA5 - 5th gen ECMWF atmospheric global climate ReAnalyses

- ERA5 combines vast amounts of historical observations into global estimates using advanced modelling and data assimilation systems.
- From 1979 - 2019, hourly estimates of atmospheric, land and oceanic climate variables.
- 30 km global grid, with 137 levels from the surface up to a height of 80 km.



An ECMWF snow depth analysis for Scandinavia using ERA5 data shows the highest levels in a decade.





https://medium.com/planet-os/era5-reanalysis-data-on-s3-cee2f22422ae

https://www.ecmwf.int/en/forecasts/datasets/reanalysis-datasets/era5

Vast amount of data

- The number of observations assimilated in ERA5 has increased from approximately 0.75 million per day on average in 1979 to around 24 million per day by the end of 2018
- ~14,000 GB (14 TB)
- A key dataset used for understanding our weather and climate, but inaccessible to all but a few privileged institutions

PANGEO

A community platform for Big Data geoscience

contributors 71 | discourse 783 users | chat on gitter | follow @pangeo_data 3.9k

This website contains general information about the Pangeo project. For news and updates about Pangeo, check out our Medium blog and our Twitter feed. To engage with the Pangeo community, head over to our Discourse forum or browse our GitHub repos.

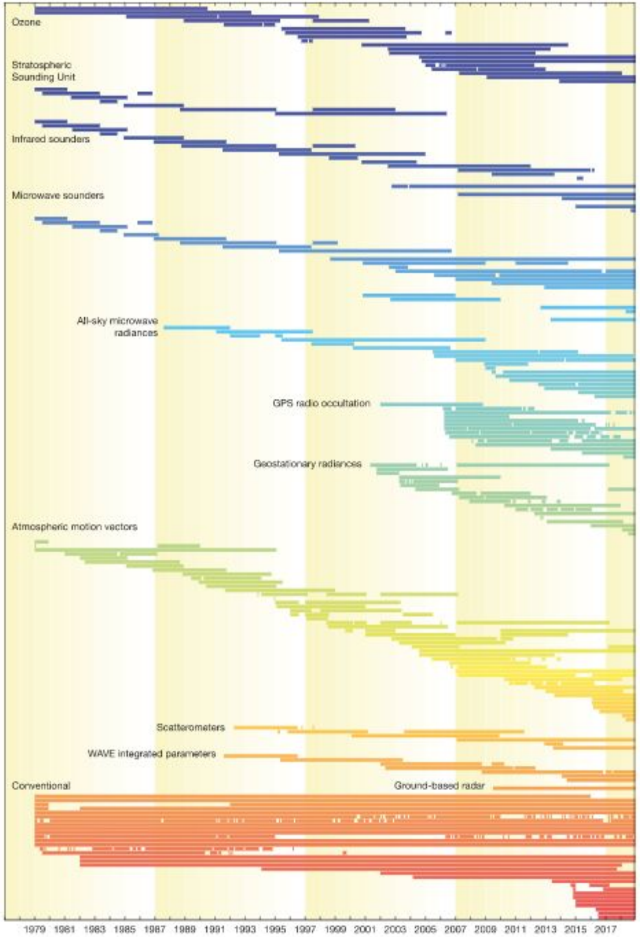


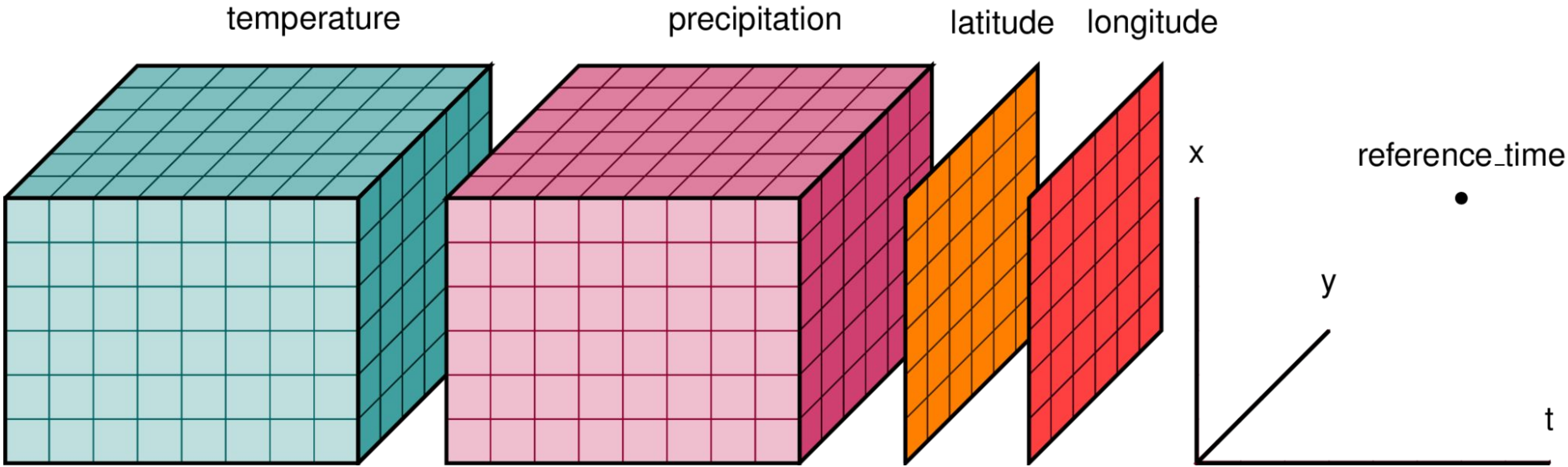
Figure 1 Data usage in ERA5 for the segment from 1979. Each horizontal bar represents the use of a particular satellite instrument or ground-based radar or a particular source of conventional data, such as weather stations, aircraft, ships, buoys and radiosondes. (Image courtesy of Paul Poli)

Data Frames are not enough: not all data is tabular

pandas

xarray

Xarray



‘data variables’ : temperature and precipitation “data variables”
‘coordinate variables’: they label the points along the dimensions.

Xarray introduces labels in the form of dimensions, coordinates and attributes on top of raw NumPy-like multidimensional arrays, which allows for a more intuitive, more concise, and less error-prone developer experience.

Xarray: Dataset

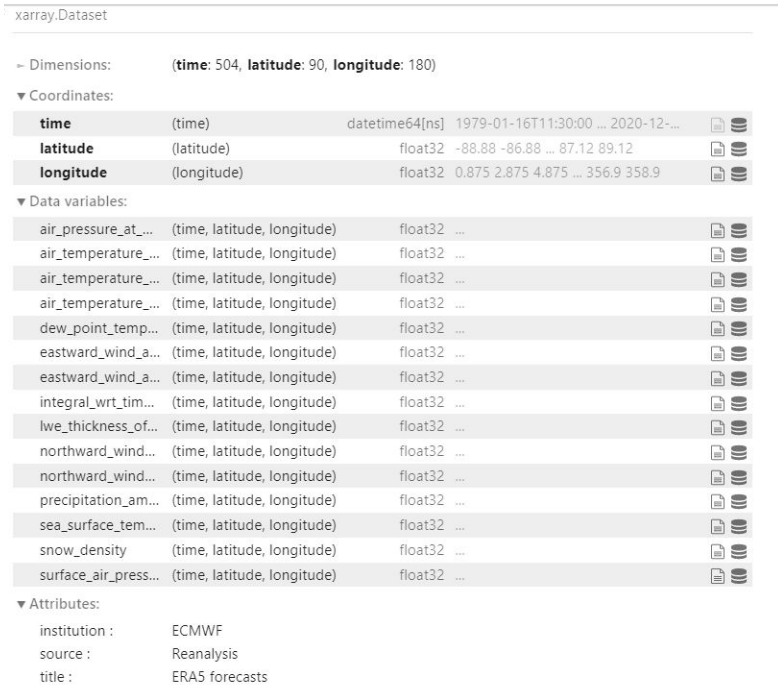
Xarray Datasets are essentially groups of DataArrays.

This is **really** valuable when you are looking at datasets that have multidimensional groups of data, for example, temperature, precipitation, cloud cover.

Some Xarray methods can be applied to all that Dataset contains.

For example, you can subset a Dataset and the subset, interpolate, calculate a mean, and it will do this across all the DataArrays the Dataset contains.

ERA5



Xarray read in a Dataset

```python
import xarray as xr
ds = xr.open_dataset('./../data/era5_monthly_2deg_aws_v20210920.nc' )
ds
```

xarray.Dataset

► Dimensions: (**time**: 504, **latitude**: 90, **longitude**: 180)

▼ Coordinates:

time	(time)	datetime64[ns]	1979-01-16T11:30:00 ... 2020-12-...
latitude	(latitude)	float32	-88.88 -86.88 ... 87.12 89.12
longitude	(longitude)	float32	0.875 2.875 4.875 ... 356.9 358.9

▼ Data variables:

air_pressure_at_...	(time, latitude, longitude)	float32	...
air_temperature_...	(time, latitude, longitude)	float32	...
air_temperature_...	(time, latitude, longitude)	float32	...
air_temperature_...	(time, latitude, longitude)	float32	...
dew_point_temp...	(time, latitude, longitude)	float32	...
eastward_wind_a...	(time, latitude, longitude)	float32	...
eastward_wind_a...	(time, latitude, longitude)	float32	...
integral_wrt_tim...	(time, latitude, longitude)	float32	...
lwe_thickness_of...	(time, latitude, longitude)	float32	...
northward_wind...	(time, latitude, longitude)	float32	...
northward_wind...	(time, latitude, longitude)	float32	...
precipitation_am...	(time, latitude, longitude)	float32	...
sea_surface_tem...	(time, latitude, longitude)	float32	...
snow_density	(time, latitude, longitude)	float32	...
surface_air_press...	(time, latitude, longitude)	float32	...

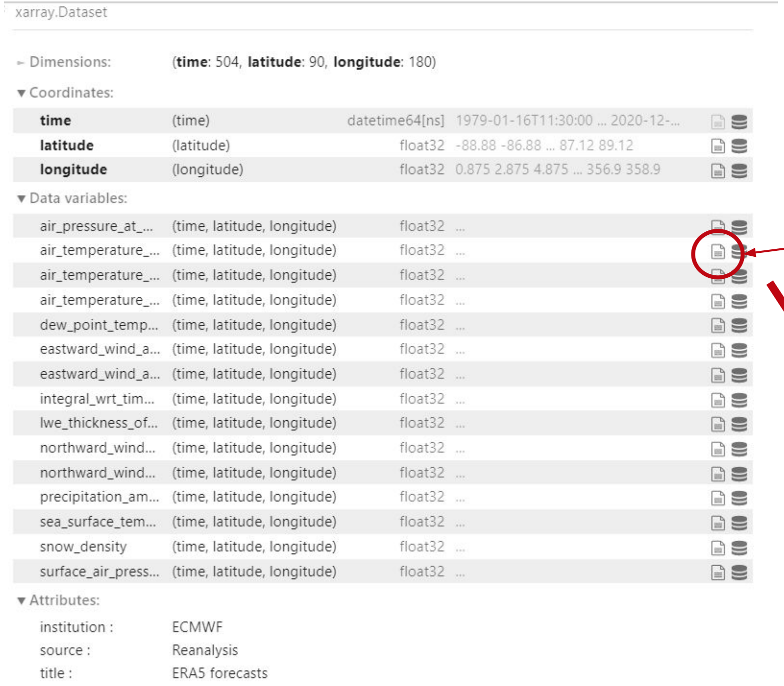
▼ Attributes:

institution :	ECMWF
source :	Reanalysis
title :	ERA5 forecasts

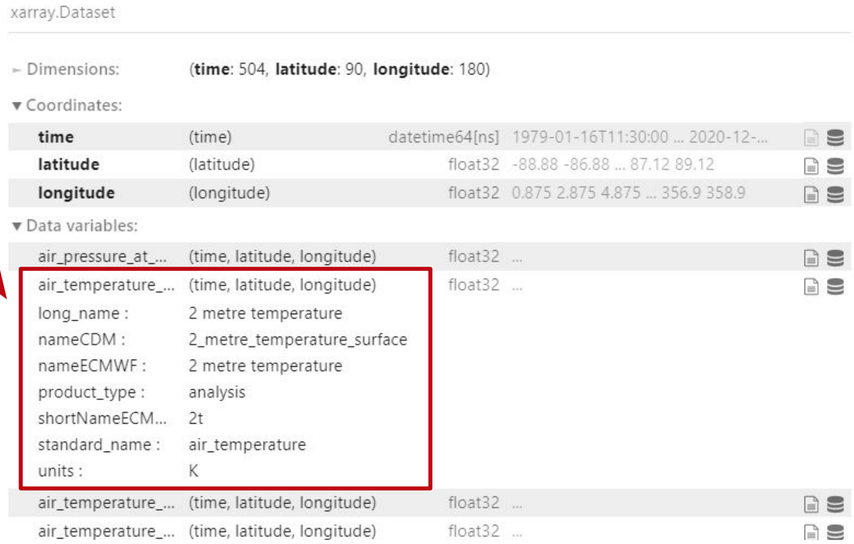
- 3D data
- Dimensions: 504x90x180
- Coordinates: time,latitude,longitude
- 15 data variables (in DataArrays)
- Attributes

Explore the data

Xarray allows you easily explore the data



Look at data attributes



Coordinates versus dimensions

- DataArray objects inside a Dataset may have any number of dimensions but are presumed to share a common coordinate system.
- Coordinates can also have any number of dimensions but denote constant/independent quantities, unlike the varying/dependent quantities that belong in data
- A dimension is just a name of an axis, like ‘time’

```
ds.dims
```

```
Frozen({'time': 504, 'latitude': 90, 'longitude': 180})
```

```
ds.coords
```

```
Coordinates:
  * time         (time) datetime64[ns] 1979-01-16T11:30:00 ... 2020-12-16T11:30:00
  * latitude     (latitude) float32 -88.88 -86.88 -84.88 ... 85.12 87.12 89.12
  * longitude    (longitude) float32 0.875 2.875 4.875 6.875 ... 354.9 356.9 358.9
```

Text from Xarray docs http://xarray.pydata.org/en/stable/user-guide/data-structures.html

DataArray are data variables in a Dataset

- A DataArray holds a multi-dimensional information
- DataArray objects inside a Dataset may have any number of dimensions but are presumed to share a common coordinate system.
- You can explore the data easily using either syntax

```
ds["air_temperature_at_2_metres"]
```

```
ds.air_temperature_at_2_metres
```

xarray.DataArray 'air_temperature_at_2_metres' (**time**: 504, **latitude**: 90, **longitude**: 180)

```
[8164800 values with dtype=float32]
```

▼Coordinates:

time	(time)	datetime64[ns]	1979-01-16T11:30:00 ... 2020-12-...
latitude	(latitude)	float32	-88.88 -86.88 ... 87.12 89.12
longitude	(longitude)	float32	0.875 2.875 4.875 ... 356.9 358.9

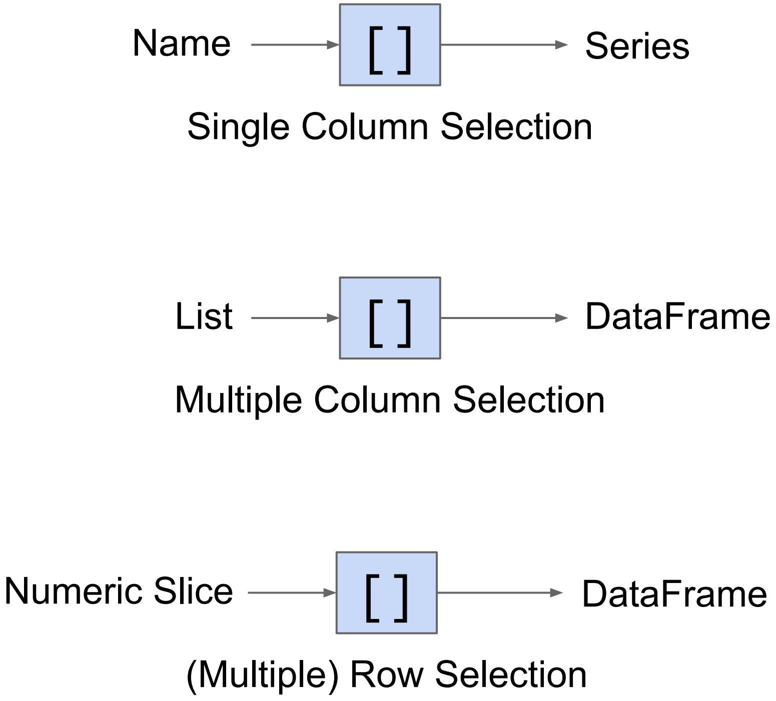
▼Attributes:

long_name :	2 metre temperature
nameCDM :	2_metre_temperature_surface
nameECMWF :	2 metre temperature
product_type :	analysis
shortNameECM...	2t
standard_name :	air_temperature
units :	K

Text from Xarray docs http://xarray.pydata.org/en/stable/user-guide/data-structures.html

Review: DataFrame access: [], loc, iloc

`[]: flexible, confusing?`



`iloc`: integer/positional

- Always 0-based, for rows and columns.
- Slices as usual, end-point exclusive.
- Use carefully (error prone).

`loc`: Labels

- Strings, integers - row/column labels
- Lists - similar, but always return dataframes
- Slices of labels: **end-point inclusive**!
- Boolean arrays: “mask” selection.

New: DataArray access: [], sel, isel

[]: flexible, confusing?

- Only for DataArrays

```python
point =  ds.air_temperature_at_2_metres[0,26,119]
```

isel: integer/positional

- Always 0-based
- Slices as usual, **end-point exclusive**.
- Use carefully (error prone).

```python
point = ds.isel(time=0,
                latitude=26,
                longitude=119)
```

sel: coordinates

- Strings, integers - coordinates*
- Slices: **end-point inclusive**!

```python
point = ds.sel(time="1979-01",
               latitude=37.125,
               longitude=238.875)
```

▾ Coordinates:

time	(time)	datetime64[ns]	1979-01-16T11:30:00 ... 2020-12-...
latitude	(latitude)	float32	-88.88 -86.88 ... 87.12 89.12
longitude	(longitude)	float32	0.875 2.875 4.875 ... 356.9 358.9

Find data at a point or in a region:

```
point = ds.sel(time="1979-01",
               latitude=37.125,
               longitude=238.875)
```

```
region = ds.sel(time="1979-01",
                latitude=slice(30,40),
                longitude=slice(230,250))
```

`sel`: coordinates

- Strings, integers - coordinates*
- Slices of coordinates: **end-point inclusive**!

Xarray helps you understand your code.

Xarray has all sort of high-level cool tricks built in

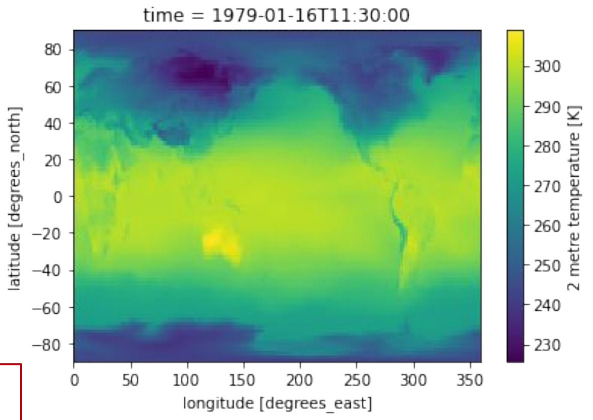
Select data and plot data in one line (using matplotlib)

```
ds.air_temperature_at_2_metres.sel(time="1979-01").plot()
```

Select a variable

Select coordinate

Apply a method()

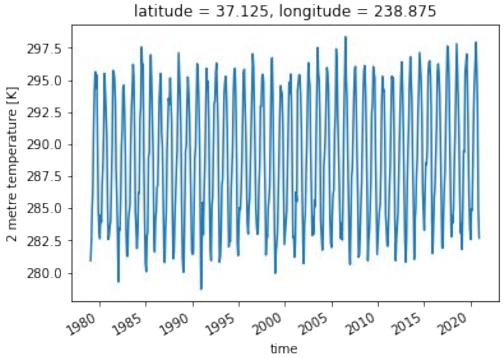



```
ds.air_temperature_at_2_metres.sel(
                                   latitude=37.125,
                                   longitude=238.875).plot()
```

Select a variable

Select coordinate

Apply a method()



Methods can be called across a DataArray or a Dataset -- LAZY

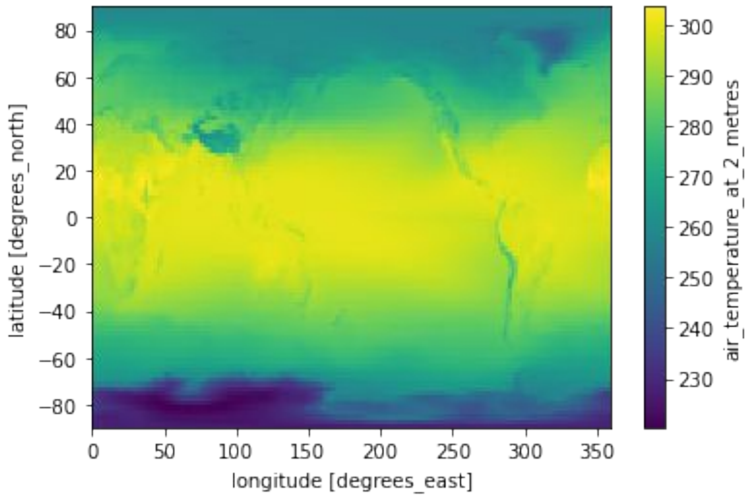
Select data and plot data in one line (using matplotlib)

```
ds.air_temperature_at_2_metres.mean("time").plot()
```

Select a variable

Apply a method() across a coordinate

Apply a method()



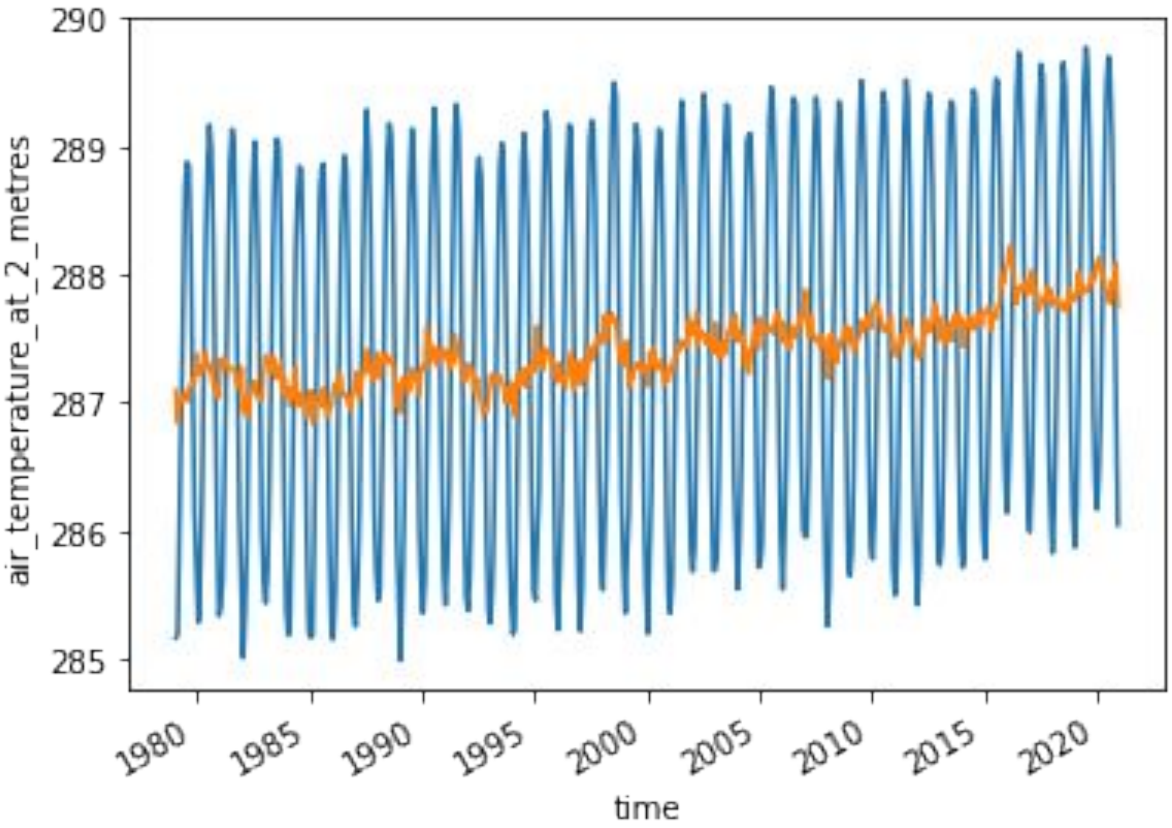
Same thing, but across all variables

Lazy

```
mean_map = ds.mean("time")
mean_map.air_temperature_at_2_metres.plot()
```

Plot the global trend in a variable

- Calculate a time series
- Take out the annual cycle
- Plot the trend

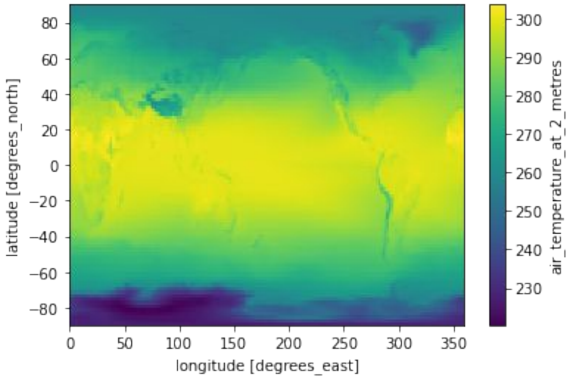


Goal: Calculate time series

Xarray has high level methods like .mean(), .std(), etc.

```
ave = ds.mean("time")
ave.air_temperature_at_2_metres.plot()
```

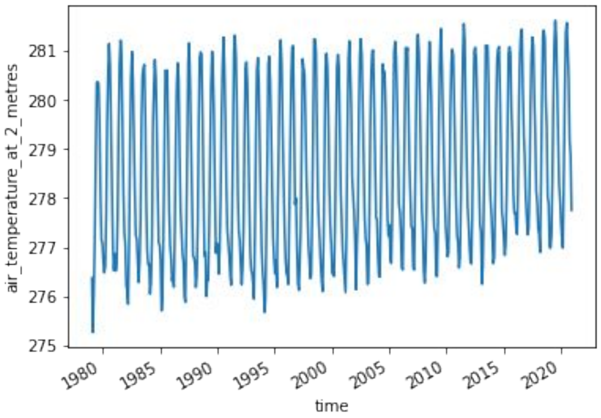
Take the mean across all time



Take the mean across all locations

```
ave = ds.mean(("latitude", "longitude"))
ave.air_temperature_at_2_metres.plot()
```

Does that look right?



Goal: Understand what .mean() does

With great power comes great responsibility

```
ave = ds.mean()
ave
```

Take the mean across all coordinates

xarray.Dataset

- Dimensions:
- Coordinates: (0)
- Data variables:

air_pressure_at_...	()	float32	1.01e+05
air_temperature_...	()	float32	278.5
air_temperature_...	()	float32	278.6
air_temperature_...	()	float32	278.4
dew_point_temp...	()	float32	274.0
eastward_wind_a...	()	float32	0.014
eastward_wind_a...	()	float32	-0.05225
integral_wrt_tim...	()	float32	5.908e+05
lwe_thickness_of...	()	float32	1.143
northward_wind...	()	float32	0.1978
northward_wind...	()	float32	0.1884
precipitation_am...	()	float32	9.783e-05
sea_surface_tem...	()	float32	286.6
snow_density	()	float32	128.7
surface_air_press...	()	float32	9.669e+04

- Attributes:

institution :	ECMWF
source :	Reanalysis
title :	ERA5 forecasts

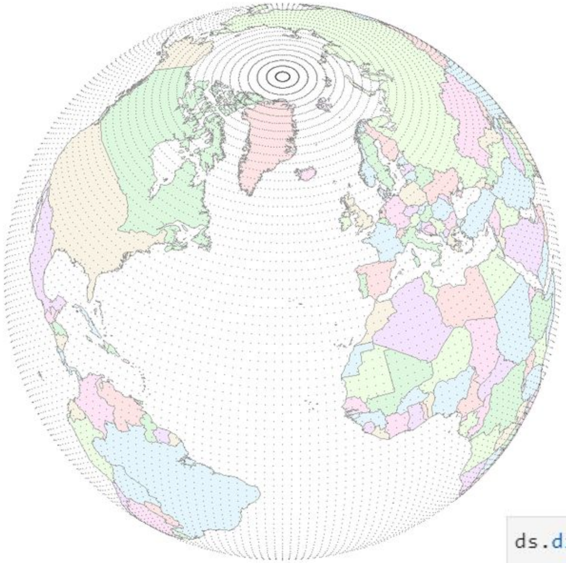
Does that look right?

ds.mean()

ds.weighted(weights).mean()

The map is flat - but the Earth is not - Gaussian grid

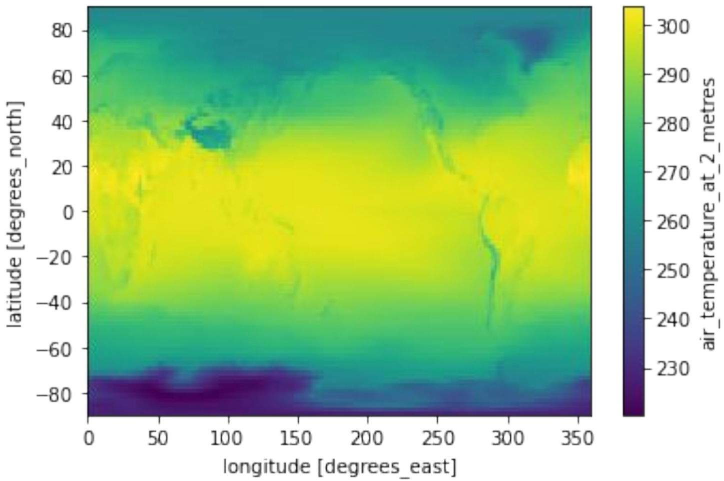
Programs aren't smart - you are - so what went wrong?



Gridded data is nice to work with but what does it represent?

How many grid points are at 90N (the North Pole)?

How many grid points are at the Equator?




```
ds.dims
```

```
Frozen({'time': 504, 'latitude': 90, 'longitude': 180})
```

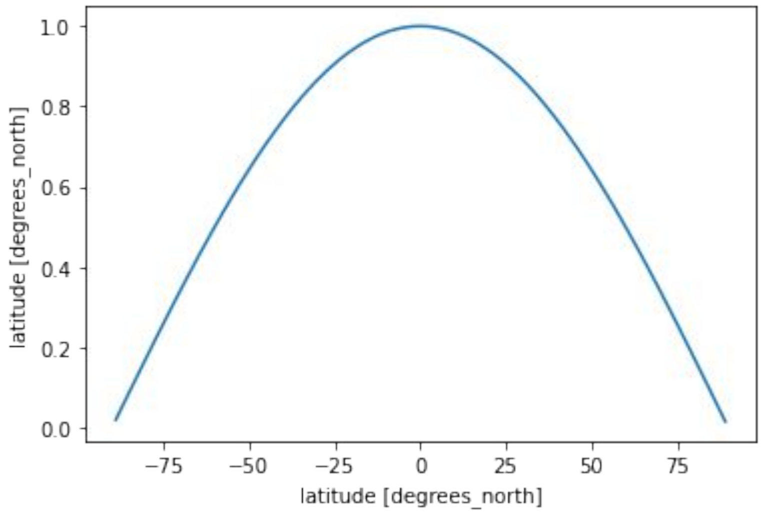
```
ds.coords
```

```
Coordinates:
  * time       (time) datetime64[ns] 1979-01-16T11:30:00 ... 2020-12-16T11:30:00
  * latitude   (latitude) float32 -88.88 -86.88 -84.88 ... 85.12 87.12 89.12
  * longitude  (longitude) float32 0.875 2.875 4.875 6.875 ... 354.9 356.9 358.9
```

Weight your data

Xarray provides the ability to weight your data

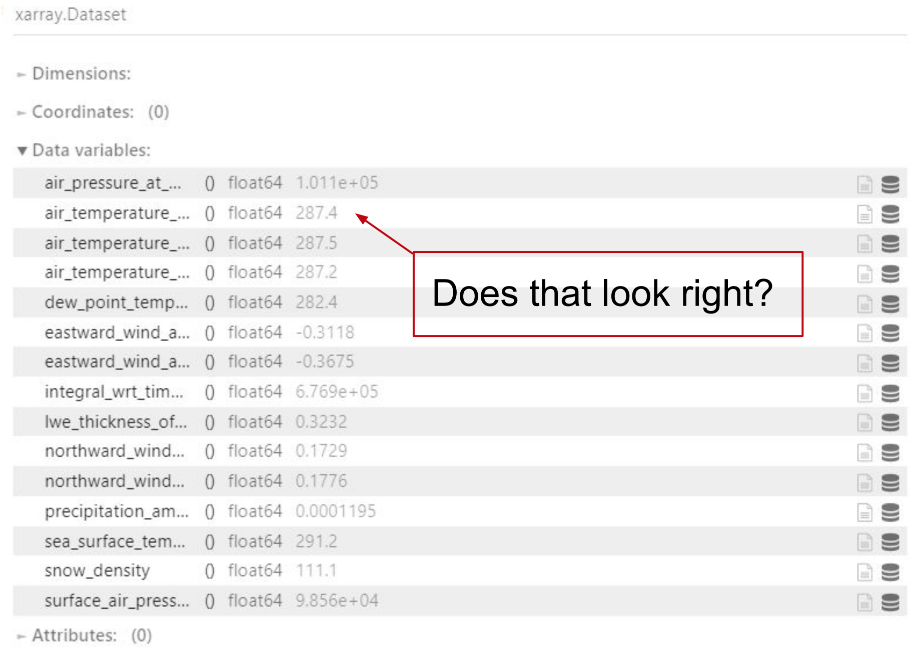
```
weights = np.cos(np.deg2rad(ds.latitude))
weights.name = "weights"
weights.plot()
```



Goal: Examine average values - weighted version

Xarray methods like .weighted() can be combined with .mean()

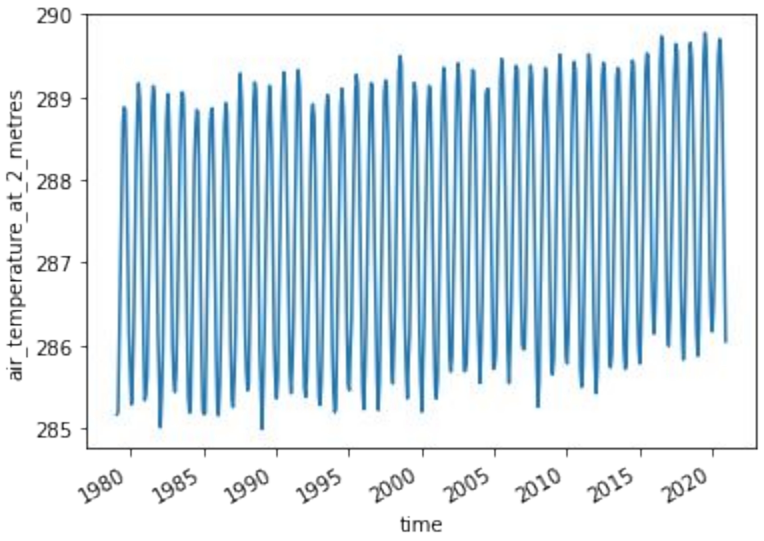
```
ds_weighted = ds.weighted(weights)
weighted_mean = ds_weighted.mean()
weighted_mean
```



Goal: Weighted global time series data

You can create means across coordinates: eg. latitude and longitude

```
ds_weighted = ds.weighted(weights)
weighted_mean = ds_weighted.mean(("latitude", "longitude"))
weighted_mean.air_temperature_at_2_metres.plot()
```



Take out the annual cycle using .groupby()

Use .groupby on a coordinate

pandas.DataFrame.groupby

`DataFrame.groupby(by=None, axis=0, level=None, as_index=True, sort=True, group_keys=True, squeeze=NoDefault.no_default, observed=False, dropna=True)` [source]

Group DataFrame using a mapper or by a Series of columns.

A groupby operation involves some combination of splitting the object, applying a function, and combining the results. This can be used to group large amounts of data and compute operations on these groups.

Parameters: **by** : ***mapping, function, label, or list of labels***

Used to determine the groups for the groupby. If `by` is a function, it's called on each value of the object's index. If a dict or Series is passed, the Series or dict VALUES will be used to determine the groups (the Series' values are first aligned; see `.align()` method). If an ndarray is passed, the values are used as-is to determine the groups. A label or list of labels may be passed to group by the columns in `self`. Notice that a tuple is interpreted as a (single) key.

axis : ***{0 or 'index', 1 or 'columns'}, default 0***

Split along rows (0) or columns (1).

level : ***int, level name, or sequence of such, default None***

If the axis is a MultiIndex (hierarchical), group by a particular level or levels.

as_index : ***bool, default True***

For aggregated output, return object with group labels as the index. Only relevant for DataFrame input. as_index=False is effectively "SQL-style" grouped output.

sort : ***bool, default True***

Sort group keys. Get better performance by turning this off. Note this does not influence the order of observations within each group. Groupby preserves the order of rows within each group.

GroupBy: split-apply-combine

xarray supports "group by" operations with the same API as pandas to implement the split-apply-combine strategy:

- Split your data into multiple independent groups.
- Apply some function to each group.
- Combine your groups back into a single data object.

Group by operations work on both `Dataset` and `DataArray` objects. Most of the examples focus on grouping by a single one-dimensional variable, although support for grouping over a multi-dimensional variable has recently been implemented. Note that for one-dimensional data, it is usually faster to rely on pandas' implementation of the same pipeline.

Resampling and grouped operations

Datetime components couple particularly well with grouped operations (see GroupBy: split-apply-combine) for analyzing features that repeat over time. Here's how to calculate the mean by time of day:

```
In [23]: ds.groupby("time.hour").mean()
Out[23]:
<xarray.Dataset>
Dimensions:  (hour: 4)
Coordinates:
  * hour     (hour) int64 0 6 12 18
Data variables:
    foo      (hour) float64 728.0 729.0 730.0 731.0
```

Goal: Calculate annual cycle

Can use .groupby & .mean

```
annual_cycle = weighted_mean.groupby("time.month").mean()
annual_cycle.air_temperature_at_2_metres.plot()
```

xarray.Dataset

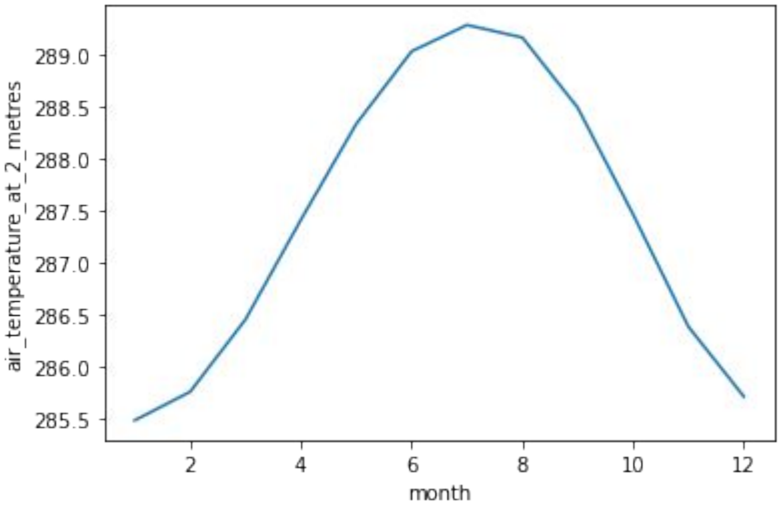
► Dimensions: (**month**: 12, **latitude**: 90, **longitude**: 180)

▼ Coordinates:

latitude	(latitude)	float32	-88.88 -86.88 ... 87.12 89.12
longitude	(longitude)	float32	0.875 2.875 4.875 ... 356.9 358.9
month	(month)	int64	1 2 3 4 5 6 7 8 9 10 11 12

► Data variables: (15)

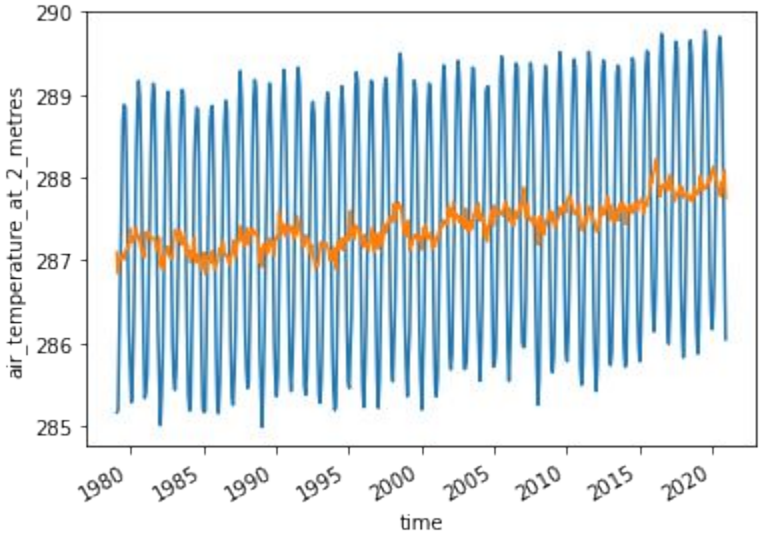
► Attributes: (0)



Put it all together and plot the trend

Can use .groupby & .mean

```
weighted_mean = ds_weighted.mean(("latitude", "longitude"))  #weighted mean time series
annual_cycle = weighted_mean.groupby("time.month").mean()    #calculate annual cycle
annual_cycle += annual_cycle.mean()                          #add back in the mean value

weighted_trend = weighted_mean.groupby("time.month") - annual_cycle
weighted_mean.air_temperature_at_2_metres.plot()
weighted_trend.air_temperature_at_2_metres.plot()
```



Goal: Are extremes more likely? PDF analysis.

xarray.plot.hist

`xarray.plot.hist(darray, figsize=None, size=None, aspect=None, ax=None, xincrease=None, yincrease=None, xscale=None, yscale=None, xticks=None, yticks=None, xlim=None, ylim=None, **kwargs)` [source]

Histogram of DataArray.

Wraps `matplotlib.pyplot.hist()`.

Plots *N*-dimensional arrays by first flattening the array.

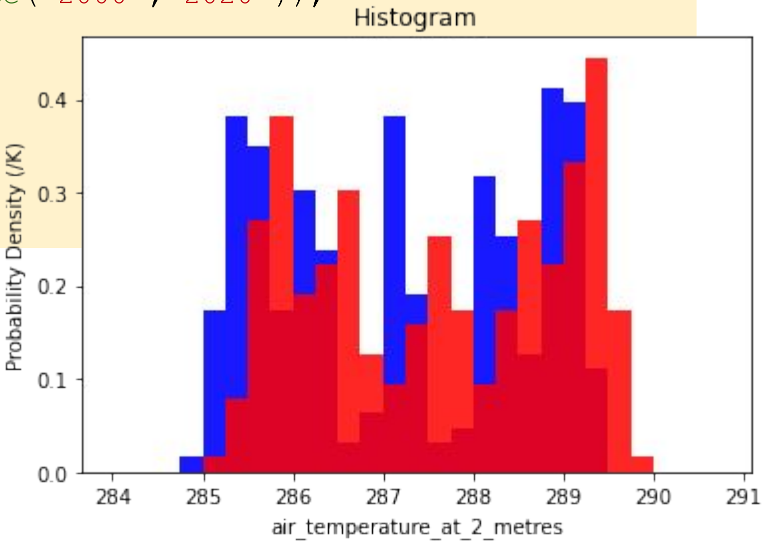
Parameters

- **darray** (`DataArray`) – Can have any number of dimensions.
- **figsize** (`tuple`, *optional*) – A tuple (width, height) of the figure in inches. Mutually exclusive with `size` and `ax`.
- **aspect** (scalar, *optional*) – Aspect ratio of plot, so that `aspect * size` gives the *width* in inches. Only used if a `size` is provided.
- **size** (scalar, *optional*) – If provided, create a new figure for the plot with the given size: *height* (in inches) of each plot. See also: `aspect`.
- **ax** (`matplotlib axes object`, *optional*) – Axes on which to plot. By default, use the current axes. Mutually exclusive with `size` and `figsize`.
- ****kwargs** (**optional**) – Additional keyword arguments to `matplotlib.pyplot.hist()`.

```
xr.plot.hist(darray,
    bins=bin_array,
    density=True,
    alpha=.9,
    color="b",
)
```

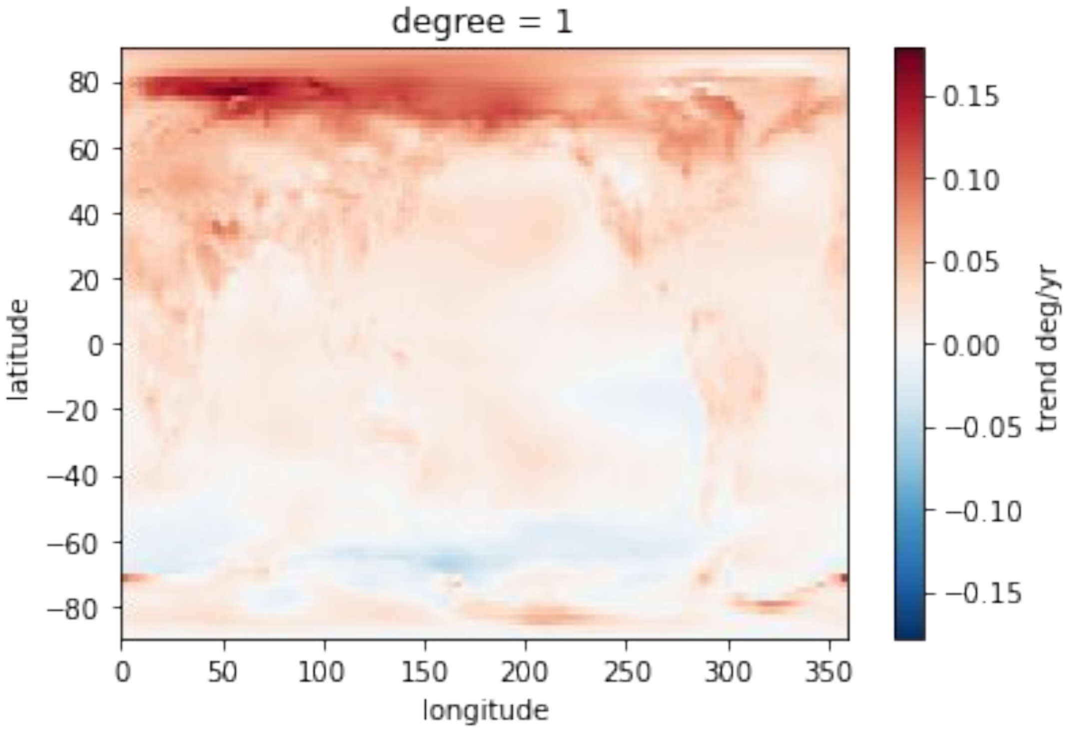
ERA5 temperature PDFs

```
bins = np.arange(284, 291)
xr.plot.hist(
    weighted_mean.air_temperature_at_2_metres .sel(time=slice("1980","2000")),
    bins=bins,
    density=True,
    alpha=.9,
    color="b",
)
xr.plot.hist(
    weighted_mean.air_temperature_at_2_metres .sel(time=slice("2000","2020")),
    bins=bins,
    density=True,
    alpha=.85,
    color="r",
)
plt.ylabel("Probability Density (/K)" )
```



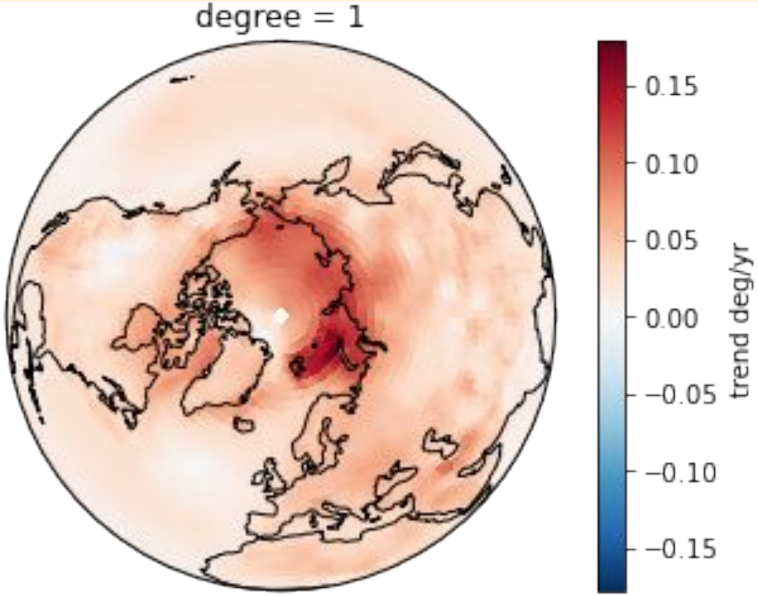
Goal: Can we plot the trend with our ERA5 data?

```
pfit = ds.air_temperature_at_2_metres .polyfit ("time", 1)
pfit.polyfit_coefficients[ 0] *= 3.154000000101e+16
pfit.polyfit_coefficients [0].plot (cbar_kwargs={ "label":"trend deg/yr" })
```



Goal: How about all fancy on a globe?

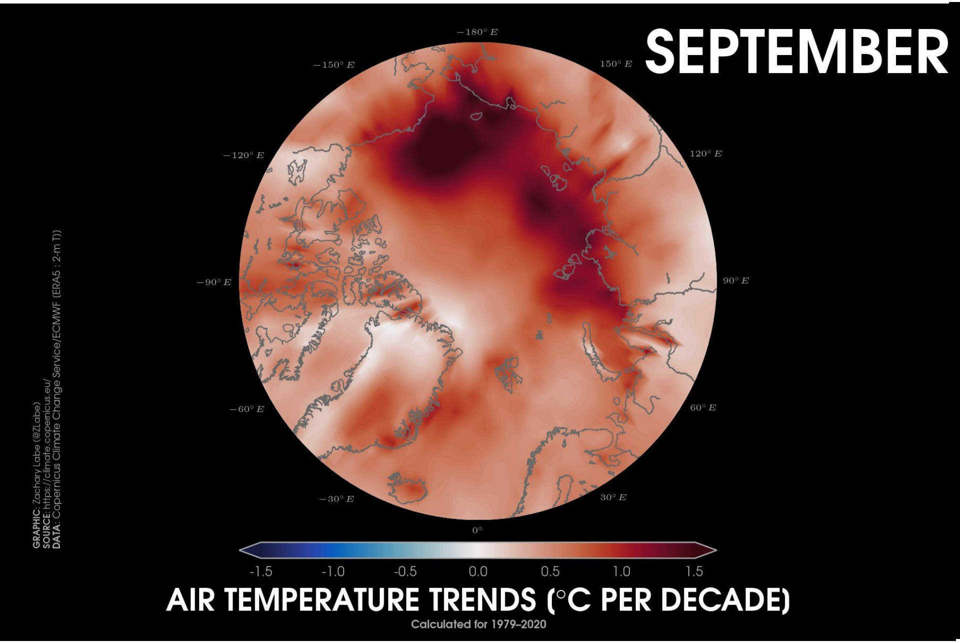
```
import cartopy.crs as ccrs

p = pfit.polyfit_coefficients[ 0].plot(
    subplot_kws =dict(projection=ccrs.Orthographic( 0, 55), facecolor="gray"),
    transform =ccrs.PlateCarree(central_longitude=0),
    cbar_kwargs ={"label": "trend deg/yr" },
)
p.axes.coastlines()
```

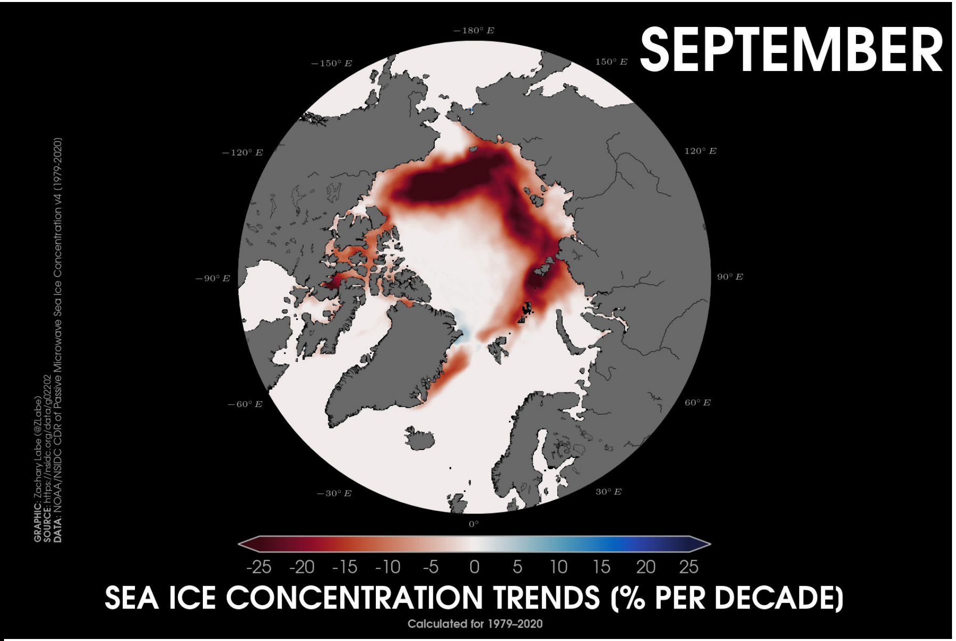


Goal: Positive feedbacks in the climate system

The Earth is a system - a complicated game of dominos

As air temperature increases, sea ice concentration decreases…..





Models and data tell us about our weather and climate

Xarray is a powerful tool to analyze climate data.

I've just given some simple examples here, but really, we need more eyes on the data, more eyes on the models.

The models use 'parameterizations' which are approximations for all sorts of different physical processes. Each parameterization uses coefficients derived fro data science - but the experiments may be limited or imperfect. Often parameters are adjusted to compensate for an error, but then end up causing other issues, and we are all still working on this.

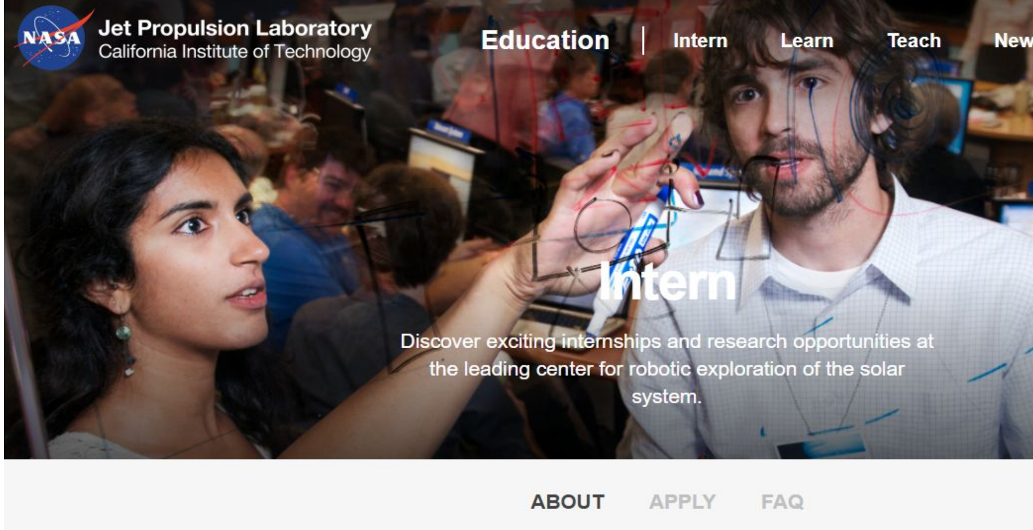
We need more of you, more data scientists working with climate and weather scientists to look at this data, helping to find new discoveries, amplify messages about changes to our climate and their impacts, and build machine learning models to replace old parameterizations.

I’m just an oceanographer

I started off like you, I’m just an oceanographer trying to find my way through the data science world. You are all data scientists maybe trying to find something interesting to work on.

Well, we all need your help. We need minds like yours to help solve the problems our and previous generations have caused. We need your voices, backed by solid data science, to mitigate what is going on, to change our trajectory.

I’ve shown you how to take pandas and leap to another level library, Xarray, you can do tutorials, leap to Scipy, or other libraries. I hope you enjoyed this small tour of Xarray and climate science.



Internships and Other Student Work Opportunities

Internships and Fellowships

NASA internships, fellowships and scholarships leverage NASA's unique missions and programs to enhance and increase the capability, diversity and size of the nation's future STEM (science, technology, engineering and math) workforce.

NASA's Goddard Space Flight Center offers hundreds of internship opportunities each year across four campuses located at:

- Greenbelt, Maryland
- Wallops Flight Facility, Wallops Island, Virginia
- Goddard Institute For Space Studies, New York City
- Independent Verification and Validation Facility, Fairmont, West Virginia

2013 Summer Interns in the Code 130 Office of Communications at NASA's Goddard Space Flight Center in Greenbelt, Maryland. [L-R] Talya Lerner, Sawyer Rosenstein, Paul Gabreilsen, Kevin McLaughlin, Crystal Garner, Kasha Patel.
Credits: NASA's Goddard Space Flight Center

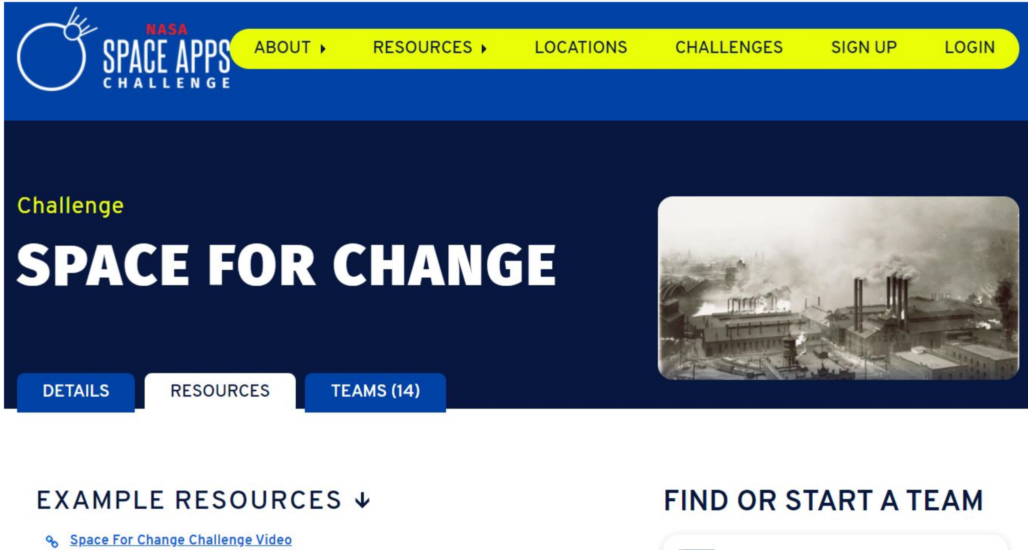
Internships are available at all levels of education from high school to graduate. Internships provide students with the opportunity to participate in either research or other experiential learning, under the guidance of a mentor at a NASA installation.

Eligibility Requirements

- U.S. citizenship
- GPA: 3.0 on a 4.0 scale
- High school students
  - At least 16 years of age and a current sophomore, junior or senior
- Undergraduate or graduate students
  - At the time the opportunity begins, must be accepted/enrolled full-time in an accredited U.S. college or university

To apply for NASA internships, fellowships and scholarships, visit NASA's OSSI site: https://intern.nasa.gov

For additional information: (GSFC-Education@mail.nasa.gov)



NASA Transform to OPen Science (TOPS)

- TOPS 5-year initiative will act as a catalyst to ***jump-start*** a suite of coordinated activities designed to rapidly transform science
- Designate **2023 as the Year of Open Science (YOOS)**

Activities

- TOPS open science platform
- Summer schools, internships, hackathons, etc.

https://science.nasa.gov/open-science/

https://github.com/nasa/Transform-to-Open-Science

Extra slides

Learning objectives

1. Show the power of the stack they know about for thinking about climate change, and grow their knowledge about that stack.
2. Expose students to data represented with structures other than tabular dataframes. Learn a little about xarray for querying/analysis.
    2.1. Moving from 2D to 3D data
        2.1.1. Concept of coordinates and dimensions
        2.1.2. Concept of metadata and understanding data
        2.1.3. Concept of histograms and data exploration
    2.2. Introduce Xarray
    2.3. How to read in data
    2.4. Selecting data .sel versus .isel compare to .iloc .loc
    2.5. Methods .mean .std .plot
    2.6. histograms
3. Examples of data with spatial and/or temporal structure.
    3.1. Satellite data structures
    3.2. Gridded data structures
4.
5. Illustrate the relationship between models grounded in physics and the kinds of data we have access to from simulations, field sensors and remote sensing.
6.
7.

Google questions

Slide 15 - hide solution have students calculate earth temperature in K, C, and deg F.

Slide 58 - ask students why they think the mean temperature is 10 deg below what we expect - why is it too cold?

- Quick discussion.
- Basic idea: evidence for climate change illustrated from one dataset, analyzed/visualized with xarray.
- Learning objectives:
  - Introduce them to basic physical mechanisms of climate change and the evidence we see in the data.
  - Show the power of the stack they know about for thinking about climate change, and grow their knowledge about that stack.
    - Expose students to data represented with structures other than tabular dataframes. Learn a little about xarray for querying/analysis.
    - Discuss data with spatial and/or temporal structure.
  - Illustrate the relationship between models grounded in physics and the kinds of data we have access to from simulations, field sensors and remote sensing. How do we reason about a physical model -- other ideas to validate our models - conservation of energy -

Adding in the Greenhouse Effect

We need to add the emitted radiation from the atmosphere to the E_{in} side

$$E_{in} = E_{out}$$

$$\Omega(1-A)\pi r 2 = \sigma T^4 4\pi r 2$$

$$\Omega(1-A)\pi r 2 + \boxed{\Delta E 4\pi r 2} = \sigma T^4 4\pi r 2$$

The Albedo (A) is ~.3 (on average)

Incoming solar radiation (   ) is ~1400 W/m^2

σ = 5.67x10-8 W/(m^2K^4)

T = 288 K

Solve for: ΔE

Water Vapor is a greenhouse gas

"the water-holding capacity of the atmosphere increases by about 7% for every 1°C rise in temperature"

specific latent heat of evaporation of water

saturation vapor pressure

$$\frac{\mathrm{d}e_s}{\mathrm{d}T} = \frac{L_v(T)e_s}{R_v T^2}$$

gas constant of water vapor

temperature

Read in a CSV file

SUGGEST this slide &* next 2 move to end ? do in lab?

Goal 1: Calculate how the temperature is changing with increasing CO2 by using actual CO2 data collected at Mauna Loa. Original (uncleaned) data is here.

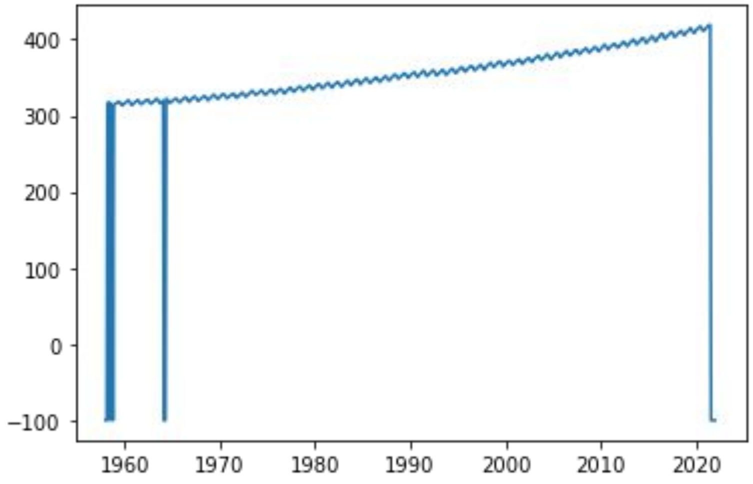
```
file = "./data_d100/monthly_in_situ_co2_mlo_cleaned.csv"
data = pd.read_csv(file)
data.head()
```

	year	month	date_index	fraction_date	c02	data_adjusted_season	data_fit	data_adjusted_seasonally_fit	data_filled	data_adjuste
0	1958	1	21200	1958.0411	-99.99	-99.99	-99.99	-99.99	-99.99	
1	1958	2	21231	1958.1260	-99.99	-99.99	-99.99	-99.99	-99.99	
2	1958	3	21259	1958.2027	315.70	314.43	316.19	314.90	315.70	
3	1958	4	21290	1958.2877	317.45	315.16	317.30	314.98	317.45	
4	1958	5	21320	1958.3699	317.51	314.71	317.86	315.06	317.51	

Plot the CO2 timeseries

What is going on? Why are their drops in the data?

```
import matplotlib.pyplot as plt
plt.plot(data["fraction_date"], data["c02"])
```

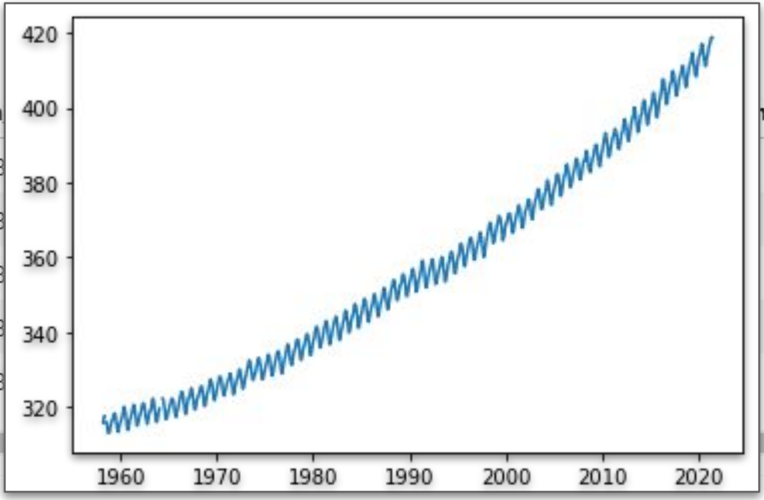


	year	month	date_index	fraction_date	c02	data_a
0	1958	1	21200	1958.0411	-99.99	
1	1958	2	21231	1958.1260	-99.99	
2	1958	3	21259	1958.2027	315.70	
3	1958	4	21290	1958.2877	317.45	
4	1958	5	21320	1958.3699	317.51	

Read in a CSV file

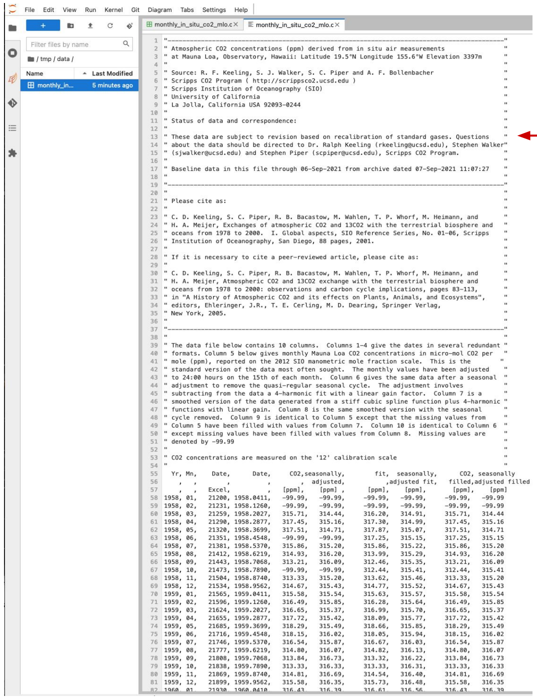
```python
file = "./data/monthly_in_situ_co2_mlo_cleaned.csv"
data = pd.read_csv(file,na_values=-99.99)
plt.plot(data["fraction_date"], data["c02"])
```

	year	month	date_index	fraction	...ally_fit	data_filled	data_adjusted_seasonal
0	1958	1	21200	1958	NaN	NaN	
1	1958	2	21231	1958	NaN	NaN	
2	1958	3	21259	1958	314.90	315.70	
3	1958	4	21290	1958	314.98	317.45	
4	1958	5	21320	1958	315.06	317.51	

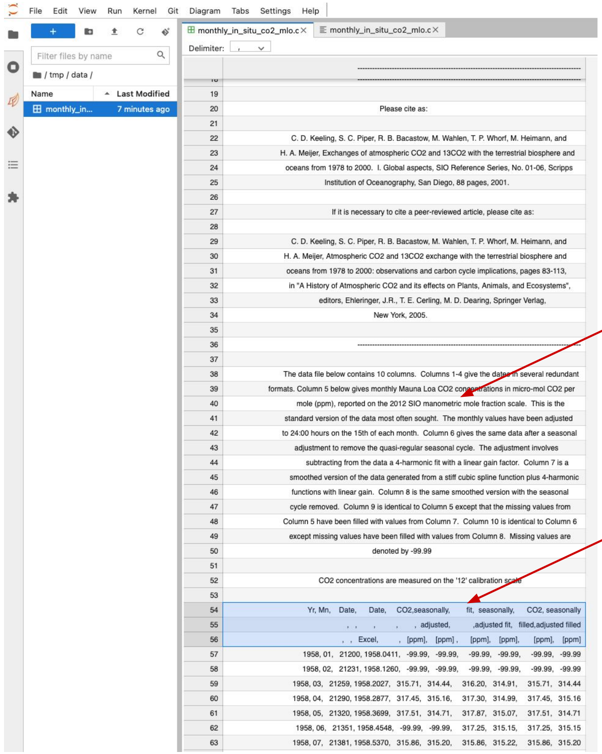


Goal: Understand real data is often a hot mess

```
file = "./data/monthly_in_situ_co2_mlo.csv"
```



A lot of text describing how to cite the data at the top of the csv file

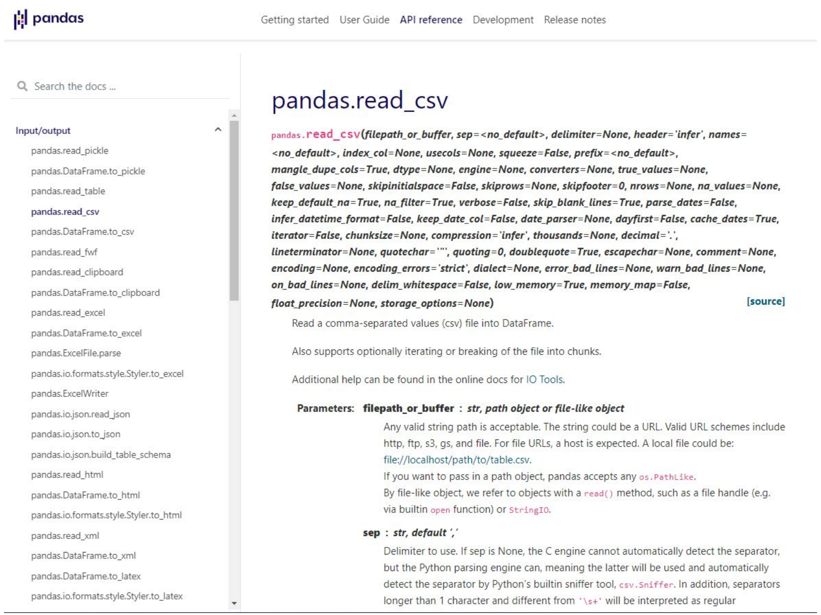


Even more text

Oh wait! Here is some data, but the column labels are split across multiple rows????

Goal: Recognize your real friends who are always there for you

Goal: Try to use the original data - you will want that citation info when you decide to publish results



pandas

Getting started User Guide API reference Development Release notes

Search the docs ...

Input/output

- pandas.read_pickle
- pandas.DataFrame.to_pickle
- pandas.read_table
- pandas.read_csv
- pandas.DataFrame.to_csv
- pandas.read_fwf
- pandas.read_clipboard
- pandas.DataFrame.to_clipboard
- pandas.read_excel
- pandas.DataFrame.to_excel
- pandas.ExcelFile.parse
- pandas.io.formats.style.Styler.to_excel
- pandas.ExcelWriter
- pandas.io.json.read_json
- pandas.io.json.to_json
- pandas.io.json.build_table_schema
- pandas.read_html
- pandas.DataFrame.to_html
- pandas.io.formats.style.Styler.to_html
- pandas.read_xml
- pandas.DataFrame.to_xml
- pandas.DataFrame.to_latex
- pandas.io.formats.style.Styler.to_latex

pandas.read_csv

pandas.**read_csv**(*filepath_or_buffer, sep=<no_default>, delimiter=None, header='infer', names=<no_default>, index_col=None, usecols=None, squeeze=False, prefix=<no_default>, mangle_dupe_cols=True, dtype=None, engine=None, converters=None, true_values=None, false_values=None, skipinitialspace=False, skiprows=None, skipfooter=0, nrows=None, na_values=None, keep_default_na=True, na_filter=True, verbose=False, skip_blank_lines=True, parse_dates=False, infer_datetime_format=False, keep_date_col=False, date_parser=None, dayfirst=False, cache_dates=True, iterator=False, chunksize=None, compression='infer', thousands=None, decimal='.', lineterminator=None, quotechar='"', quoting=0, doublequote=True, escapechar=None, comment=None, encoding=None, encoding_errors='strict', dialect=None, error_bad_lines=None, warn_bad_lines=None, on_bad_lines=None, delim_whitespace=False, low_memory=True, memory_map=False, float_precision=None, storage_options=None*) [source]

Read a comma-separated values (csv) file into DataFrame.

Also supports optionally iterating or breaking of the file into chunks.

Additional help can be found in the online docs for IO Tools.

Parameters: **filepath_or_buffer** : *str, path object or file-like object*

Any valid string path is acceptable. The string could be a URL. Valid URL schemes include http, ftp, s3, gs, and file. For file URLs, a host is expected. A local file could be: file://localhost/path/to/table.csv.
If you want to pass in a path object, pandas accepts any `os.PathLike`.
By file-like object, we refer to objects with a `read()` method, such as a file handle (e.g. via builtin `open` function) or `StringIO`.

sep : *str, default ','*

Delimiter to use. If sep is None, the C engine cannot automatically detect the separator, but the Python parsing engine can, meaning the latter will be used and automatically detect the separator by Python's builtin sniffer tool, `csv.Sniffer`. In addition, separators longer than 1 character and different from `'\s+'` will be interpreted as regular

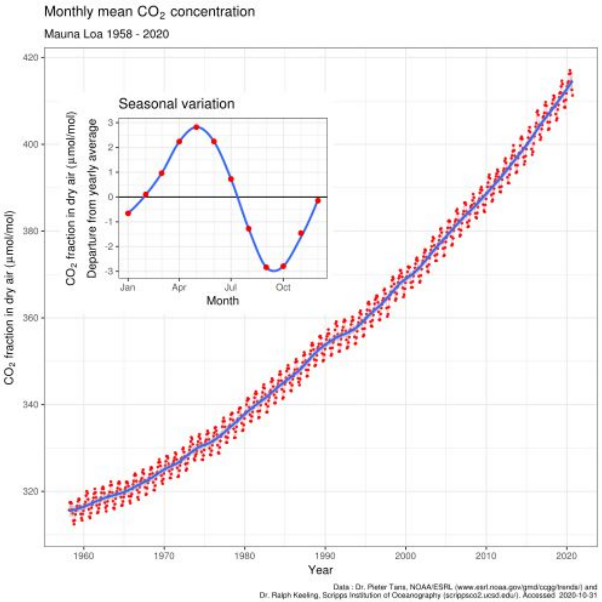
Arguments

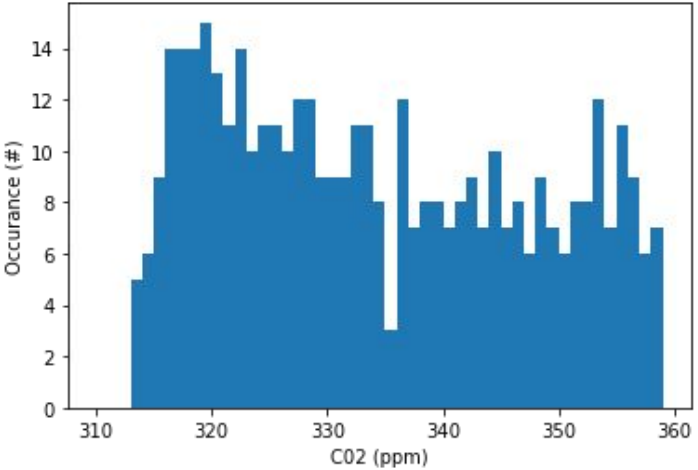
Filepath
header='infer'
names=<no_default>
skiprows=None
na_values=None

How do you calculate probability of event occurrence?

A histogram tells you how many times a particular value occurred in your dataset.

```
import numpy as np
plt.hist(data["c02"], bins=np.arange(310, 360, 1))
plt.ylabel("Occurrence (#)"), plt.xlabel("C02 (ppm)")
```

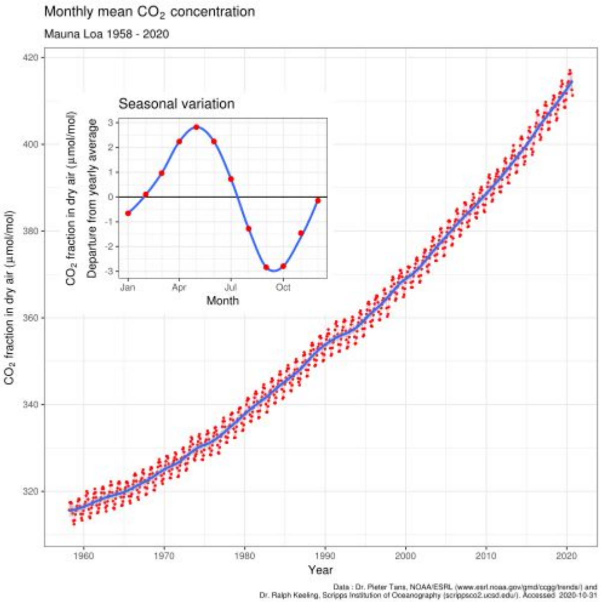


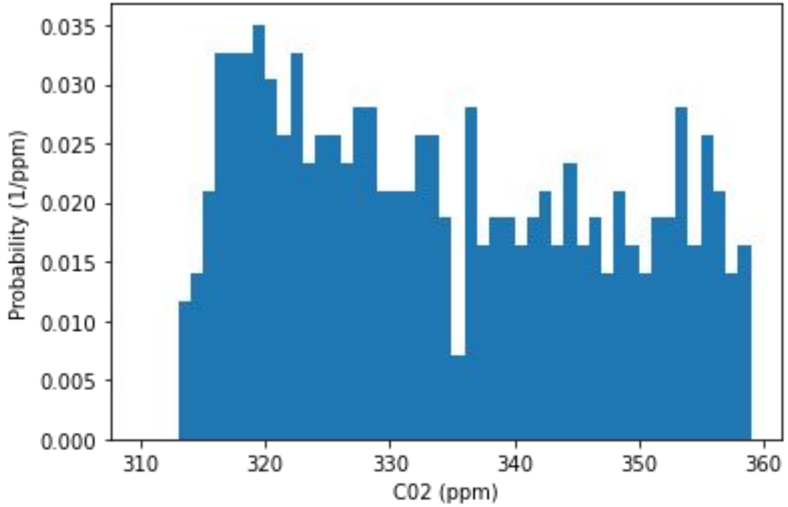


How do you calculate probability of event occurrence?

A probability density function (PDF) tells you the probability of a particular value occurred in your dataset.

```
plt.hist(data["c02"], bins=np.arange(310, 360, 1), density=True)
plt.ylabel("Probability (1/ppm)"), plt.xlabel("C02 (ppm)")
```





Extremes are the new normal

A decade ago - scientists would argue - we can't attribute any single weather event to climate.

In the last decade, we have all experienced major shifts in our climate through changes in our local weather and scientists have figured out 'climate event attribution'