
pangaea Documentation

Release 0.0.4

Alan D. Snow

Jul 10, 2017

Contents:

1	xarray extension	3
2	Read in LSM files	7
3	Logging	9
4	Indices and tables	11

An xarray extension for gridded land surface & weather model output.

GitHub: <https://github.com/snowman2/pangaea>

class pangaia.LSMGridReader(xarray_obj)

This is an extension for xarray specifically designed for land surface models.

Read with pangaia example:

```
import pangaia as pa

with pa.open_mfdataset('/path/to/ncfiles/*.nc',
                      lat_var='lat',
                      lon_var='lon',
                      time_var='time',
                      lat_dim='lat',
                      lon_dim='lon',
                      time_dim='time') as xds:
    print(xds.lsm.projection)
```

Read with xarray example:

```
import xarray as xr

with pa.open_dataset('/path/to/file.nc') as xds:
    print(xds.lsm.projection)
```

affine

Affine() – The affine for the transformation.

center

Return the geographic center point of this dataset.

coords

Returns y, x coordinate arrays

Warning: The grids always be returned with [0,0] as Northeast and [-1,-1] as Southwest.

datetime

Get datetime object for time variable

dx

float – Pixel size in x direction.

dy

float – Pixel size in y direction.

epsg

str – EPSG code

geotransform

tuple – The geotransform for grid.

getvar (*variable*, *yslice*=*slice(None, None, None)*, *xslice*=*slice(None, None, None)*, *calc_4d_method*=*None*, *calc_4d_dim*=*None*)
 Get variable from model with subset options.

Warning: The grids will always be returned with [0,0] as Northeast and [-1,-1] as Southwest.

Parameters

- **variable** (*str*) – Name of variable in dataset.
- **yslice** (*slice*, optional) – Slice in y-direction of grid to extract data from.
- **xslice** (*slice*, optional) – Slice in x-direction of grid to extract data from.
- **calc_4d_method** (*str*) – Method to convert 4D variables to 3D variables (Ex. ‘mean’, ‘min’, or ‘max’).
- **calc_4d_dim** (*str*) – Dimension to reduce grid from 4D to 3D (Ex. ‘top_bottom’).

Returns

Return type `xarray.DataArray()`

latlon

Returns lat,lon arrays

Warning: The grids always be returned with [0,0] as Northeast and [-1,-1] as Southwest.

projection

`osgeo.osr.SpatialReference()` The projection for the dataset.

resample (*variable*, *match_grid*)

Resample data to grid.

Parameters

- **variable** (*str*) – Name of variable in dataset.
- **match_grid** (`gdal.Dataset()` or `sloot.grid.GDALGrid()`) – Grid you want the data resampled to match resolution. You can also pass the path to the grid.

to_datetime ()

Converts time to datetime.

to_projection (*variable*, *projection*)

Convert Grid to New Projection.

Parameters

- **variable** (*str*) – Name of variable in dataset.
- **projection** (*osr.SpatialReference()*) – Projection to convert data to.

Returns

Return type *xarray.Dataset()*

to_tif (*variable*, *time_index*, *out_path*)

Dump a variable at a time index to a geotiff.

Parameters

- **variable** (*str*) – Name of variable in dataset.
- **time_index** (*int*) – 0-based time index,
- **out_path** (*str*) – Path to output geotiff file,

to_utm (*variable*)

Convert Grid to UTM projection at center of grid.

Parameters **variable** (*str*) – Name of variable in dataset.

Returns

Return type *xarray.Dataset()*

x_size

int – Number of columns in the dataset.

y_inverted

Is the y-coord inverted

y_size

int – Number of rows in the dataset.

Read in LSM files

`pangaea.open_mfdataset` (*path_to_lsm_files*, *lat_var*, *lon_var*, *time_var*, *lat_dim*, *lon_dim*, *time_dim*,
lon_to_180=False, *coords_projected=False*, *loader=None*, *engine=None*,
autoclose=True)

Wrapper to open land surface model netcdf files using `xarray.open_mfdataset()`.

Warning: The time dimension and variable will both be renamed to 'time' to enable slicing.

Parameters

- **path_to_lsm_files** (*str*) – Path to land surface model files with wildcard. (Ex. `'/path/to/files/*.nc'`)
- **lat_var** (*str*) – Latitude variable (Ex. `lat`).
- **lon_var** (*str*) – Longitude variable (Ex. `lon`).
- **time_var** (*str*) – Time variable (Ex. `time`).
- **lat_dim** (*str*) – Latitude dimension (Ex. `lat`).
- **lon_dim** (*str*) – Longitude dimension (Ex. `lon`).
- **time_dim** (*str*) – Time dimension (ex. `time`).
- **lon_to_180** (*bool*, *optional*, *default=False*) – If `True`, will convert longitude from `[0 to 360]` to `[-180 to 180]`.
- **coords_projected** (*bool*, *optional*, *default=False*) – If `True`, it will assume the coordinates are already in the projected coordinate system.
- **loader** (*str*, *optional*, *default=None*) – If `'hrrr'`, it will load in the HRRR dataset.
- **engine** (*str*, *optional*) – See: `xarray.open_mfdataset()` documentation.
- **autoclose** (*str*, *optional*, *default=True*) – If `True`, will use `autoclose` option with `xarray.open_mfdataset()`.

Returns

Return type `xarray.Dataset()`

Read with pangaea example:

```
import pangaea as pa

with pa.open_mfdataset('/path/to/ncfiles/*.nc',
                      lat_var='lat',
                      lon_var='lon',
                      time_var='time',
                      lat_dim='lat',
                      lon_dim='lon',
                      time_dim='time') as xds:
    print(xds.lsm.projection)
```

`pangaea.log_to_console` (*status=True, level=None*)

Log events to the console.

Parameters

- **status** (*bool, Optional, Default=True*) – whether logging to console should be turned on(True) or off(False)
- **level** (*string, Optional, Default=None*) – level of logging; whichever level is chosen all higher levels will be logged. See: <https://docs.python.org/2/library/logging.html#levels>

`pangaea.log_to_file` (*status=True, filename='/home/docs/.cache/pangaea/log/pangaea.log', level=None*)

Log events to a file.

Parameters

- **status** (*bool, Optional, Default=True*) – whether logging to file should be turned on(True) or off(False)
- **filename** (*string, Optional, Default=None*) – path of file to log to
- **level** (*string, Optional, Default=None*) – level of logging; whichever level is chosen all higher levels will be logged. See: <https://docs.python.org/2/library/logging.html#levels>

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`

A

affine (pangaea.LSMGridReader attribute), 3

C

center (pangaea.LSMGridReader attribute), 3

coords (pangaea.LSMGridReader attribute), 3

D

datetime (pangaea.LSMGridReader attribute), 3

dx (pangaea.LSMGridReader attribute), 4

dy (pangaea.LSMGridReader attribute), 4

E

epsg (pangaea.LSMGridReader attribute), 4

G

geotransform (pangaea.LSMGridReader attribute), 4

getvar() (pangaea.LSMGridReader method), 4

L

latlon (pangaea.LSMGridReader attribute), 4

log_to_console() (in module pangaea), 9

log_to_file() (in module pangaea), 9

LSMGridReader (class in pangaea), 3

O

open_mfdataset() (in module pangaea), 7

P

projection (pangaea.LSMGridReader attribute), 4

R

resample() (pangaea.LSMGridReader method), 4

T

to_datetime() (pangaea.LSMGridReader method), 4

to_projection() (pangaea.LSMGridReader method), 4

to_tif() (pangaea.LSMGridReader method), 5

to_utm() (pangaea.LSMGridReader method), 5

X

x_size (pangaea.LSMGridReader attribute), 5

Y

y_inverted (pangaea.LSMGridReader attribute), 5

y_size (pangaea.LSMGridReader attribute), 5