
spacegrids Documentation

Release

Willem Sijp

May 21, 2015

1	Contents:	1
1.1	Spacegrids	1
1.2	spacegrids Package	3
2	Indices and tables	5

Contents:

1.1 Spacegrids

Spacegrids is an open source library providing a Numpy array with grids, labelled axes and associated grid-related mathematical methods such as regridding and integration. Spacegrids provides an object data model of Netcdf data that ensures consistency between a Numpy data array and its grid under common operations (and so avoiding common pitfalls related to axis interpretation), and much more. It is a write less do more library for everyday use.

These [interactive plots from Netcdf data](#) are based on Spacegrids.

The Field, Gr (grid) and Coord objects make everyday use easy:

```
>>> import spacegrids as sg
>>> D = sg.info(nonick = True)
>>> P = sgPproject(D['my_project'] , nonick = True)
>>> P.load(['temperature','u'])
>>> # obtain the axes objects under their names T,X,Y,Z:
>>> for c in P['some_experiment'].axes:
>>>     exec c.name + ' = c' # now we can refer to X,Y
>>> TEMP = P['some_experiment']['temperature']
>>> U = P['some_experiment']['u'] # zonal velocity
>>> TEMP_sliced = TEMP[Y,:50] # slice. Note Y axis object
>>> m_TEMP = TEMP_sliced/(X*Y) # take hor. mean
>>> TEMP_regridded = TEMP.regrid(U.gr) # U grid differs
```

1.1.1 Features

- A numpy array with grid allowing automatic alignment and dimension broadcasting
- Easy to use and intuitive regridding functionality
- A data object model corresponding closely to Netcdf
- Easier IO via abstraction of IO with multiple Netcdf files
- Makes working with output of many experiments easy via aggregation methods
- The Field class eliminates errors arising from picking the wrong array index
- Quicker plotting due to automatic labels, axes etc.
- Distance-related methods such as spatial differentiation and integration on sphere
- Extensive unit tests and documentation

There is lots of documentation, both in the source code and elsewhere. Other documentation can be found at:

- [a practical tutorial](#)
- [a more advanced tutorial](#)
- [an overview of all classes, methods and functions](#)
- [interactive Netcdf data plots](#) based on Spacegrids

1.1.2 Installation

Install spacegrids simply by running (on command line):

```
pip install spacegrids
```

Dependencies: numpy, scipy, matplotlib (NetCDF4 optional). On Ubuntu/ Debian, install dependencies via package manager if pip install fails:

```
apt-get install python-{tk,numpy,matplotlib,scipy}
```

On Mac, pip can be installed via “sudo easy_install pip”.

1.1.3 Contribute

- Issue Tracker: github.com/willo12/spacegrids/issues
- Source Code: github.com/willo12/spacegrids

1.1.4 Support

If you are having issues, please let us know.

1.1.5 License

The project is licensed under the BSD license.

1.1.6 Other Packages

A related promising package is dimarray. It considers dimension names as a fundamental property of an array, and as such supports netCDF I/O format. It makes use of it in binary operations (broadcasting), transforms and indexing. It includes some of the nice features of pandas (e.g. axis alignment, optional nan skipping) but extends them to N dimensions, with a behaviour closer to a numpy array. Some geo features are planned (weighted mean for latitude, indexing modulo 360 for longitude, basic regridding) but dimarray should remain broad in scope. Other packages are [larry](#), [pandas](#) and [iris](#).

1.2 spacegrids Package

1.2.1 spacegrids Package

1.2.2 abstract Module

1.2.3 expercls Module

1.2.4 fieldcls Module

1.2.5 ops Module

1.2.6 plotting Module

1.2.7 projectcls Module

1.2.8 utilsg Module

Indices and tables

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