Search projects

Help Donate Log in Register

georasters 0.5.12

✓ Latest version

`pip install georasters`

Last released: Jun 3, 2018

Tools for working with Geographical Information System Rasters

Navigation

Project description

Release history

Download files

Project links

Homepage

Statistics

GitHub statistics:

Stars: 75

Forks: 15

Open issues/PRs: 13

View statistics for this project via Libraries.io, or by using Google BigQuery

Meta

License: GNU General Public License v3 or later (GPLv3+) (GPLv3)

Author: Ömer Özak

gis, geospatial, geographic, raster, vector, zonal, statistics, spatial, analysis

Maintainers

omerozak

Classifiers

Project description

pypi v0.5.12 downloads no longer available build error coverage 30%

The `GeoRasters` package is a python module that provides a fast and flexible tool to work with GIS raster files. It provides the GeoRaster class, which makes working with rasters quite transparent and easy. In a way it tries to do for rasters what GeoPandas does for geometries.

It includes tools to

- Merge rasters
- Plot rasters
- Extract information from rasters
- Given a point (lat,lon) find its location in a raster
- Aggregate rasters to lower resolutions
- Align two rasters of different sizes to common area and size
- Get all the geographical information of raster
- Create GeoTiff files easily
- Load GeoTiff files as masked numpy rasters
- Clip raster using geometries
- Get zonal statistics using geometries
- Spatial analysis tools

Install

```
pip install git+git://github.com/ozak/georasters.git
pip install georasters
```

Example Usage: GeoRasters

```
import georasters as gr

# Load data
raster = './data/slope.tif'
data = gr.from_file(raster)

# Plot data
data.plot()

# Get some stats
data.mean()
data.sum()
```

Development Status

- 3 - Alpha

Intended Audience

- Developers
- Science/Research

License

- OSI Approved :: GNU General Public License v3 or later (GPLv3+)

Operating System

- OS Independent

Programming Language

- Python
- Python :: 2
- Python :: 2.7
- Python :: 3
- Python :: 3.4
- Python :: 3.5
- Python :: 3.6

Topic

- Scientific/Engineering :: GIS
- Utilities

```
data.std()

# Convert to Pandas DataFrame
df = data.to_pandas()

# Save transformed data to GeoTiff
data2 = data**2
data2.to_tiff('./data2')

# Algebra with rasters
data3 = np.sin(data.raster) / data2
data3.plot()

# Notice that by using the data.raster object,
# you can do any mathematical operation that handles
# Numpy Masked Arrays

# Find value at point (x,y) or at vectors (X,Y)
value = data.map_pixel(x,y)
Value = data.map_pixel(X,Y)
```

Example Merge GeoRasters:

```
import os
import georasters as gr
import matplotlib.pyplot as plt

DATA = "/path/to/tiff/files"

# Import raster
raster = os.path.join(DATA, 'pre1500.tif')
data = gr.from_file(raster)
(xmin, xsize, x, ymax, y, ysize) = data.geot

# Split raster in two
data1 = gr.GeoRaster(
    data.raster[:data.shape[0] / 2, :],
    data.geot,
    nodata_value=data.nodata_value,
    projection=data.projection,
    datatype=data.datatype,
)
data2 = gr.GeoRaster(
    data.raster[data.shape[0] / 2:, :],
    (xmin, xsize, x, ymax + ysize * data.shape[0] / 2, y, ysize),
    nodata_value=data.nodata_value,
    projection=data.projection,
    datatype=data.datatype,
)

# Plot both parts and save them
plt.figure(figsize=(12, 8))
data1.plot()
plt.savefig(os.path.join(DATA, 'data1.png'), bbox_inches='tight')
```

./tests/data/data1.png

```
plt.figure(figsize=(12,8))
data2.plot()
plt.savefig(os.path.join(DATA,'data2.png'), bbox_inches='tight')
```

./tests/data/data2.png

```
# Generate merged raster

data3 = data1.union(data2)
```

```
# Plot it and save the figure

plt.figure(figsize=(12,8))
data3.plot()
plt.savefig(os.path.join(DATA,'data3.png'), bbox_inches='tight')
```

./tests/data/data3.png

Another Merge:

Example Usage: Other functions

```
import georasters as gr

# Get info on raster
NDV, xsize, ysize, GeoT, Projection, DataType = gr.get_geo_info(raster)

# Load raster
data = load_tiff(raster)

# Find location of point (x,y) on raster, e.g. to extract info at that location
col, row = gr.map_pixel(x,y,GeoT[1],GeoT[-1], GeoT[0],GeoT[3])
value = data[row,col]

# Agregate raster by summing over cells in order to increase pixel size by e.g. 10
gr.aggregate(data,NDV,(10,10))

# Align two rasters
data2 = load_tiff(raster2)
(alignedraster_o, alignedraster_a, GeoT_a) = gr.align_rasters(raster, raster2, how=np.

# Create GeoRaster
A=gr.GeoRaster(data, GeoT, nodata_value=NDV)

# Load another raster
NDV, xsize, ysize, GeoT, Projection, DataType = gr.get_geo_info(raster2)
data = load_tiff(raster2)
B=gr.GeoRaster(data2, GeoT, nodata_value=NDV)

# Plot Raster
A.plot()

# Merge both rasters and plot
C=B.merge(A)
C.plot()
```

Issues

Find a bug? Report it via github issues by providing

- a link to download the smallest possible raster and vector dataset necessary to reproduce the error
- python code or command to reproduce the error
- information on your environment: versions of python, gdal and numpy and system memory