Geographic Information Systems

# Computing zonal statistics with Rasterstats in Python

Asked 10 months ago Modified 10 months ago Viewed 3k times

1

I have a satellite imagery .tif file, and a corresponding .shp file with polygons over buildings. For each band (3 bands total) in the image, I'd like to calculate zonal statistics within each polygon. So within polygon 1, band 1 had a mean of x, min of y, and max of z, and so on for the other two bands and the rest of the polygons.

I am trying to do this with rasterstats zonal_stats function. I have tried a few different ways with the same result and I'm not sure what's going on.

The shapefile has a number of attributes and then of course the geometry information

The simplest I've tried:

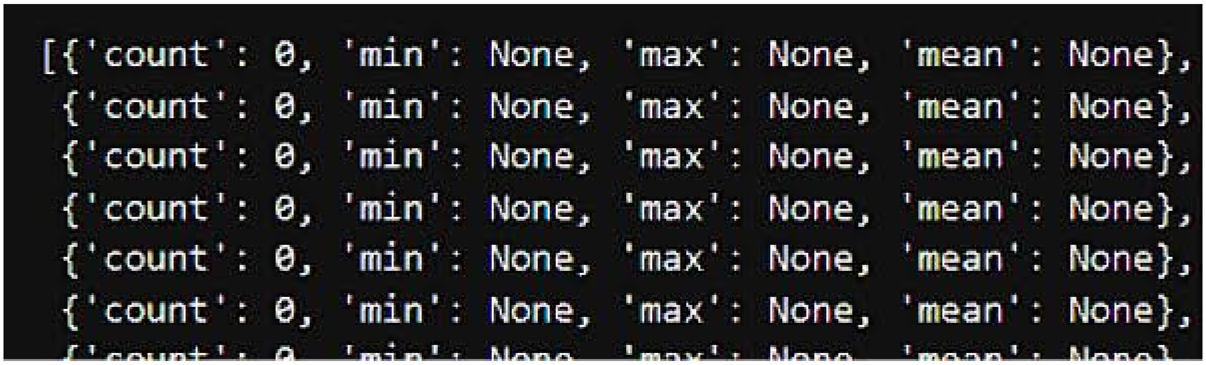
```
from rasterstats import zonal_stats
raster_path  = 'data/image.tif'
shp_path = 'data/building.shp'
stats = zonal_stats(shp_path, raster_path)
```

Another way I've tried:

```
dataset = rasterio.open(raster_path)
zone = geopandas.read_file(shp_path)
arr = dataset.read(1)
affine=dataset.transform
stats = zonal_stats(zone, arr, affine=affine)
```

I have also seen in some examples where more complex things are done with the shapefile, like some transformations it seems, but I'm not sure why.

In any case, all the variations I've tried have produced the following result. The zonal statistics are 0 or None. Can anyone with more knowledge about this tell me what this means, or why it might be happening? And how I might be able to get this to work? I can provide more info about the image and shapefile if need be.



```
[{'count': 0, 'min': None, 'max': None, 'mean': None},
 {'count': 0, 'min': None, 'max': None, 'mean': None},
 {'count': 0, 'min': None, 'max': None, 'mean': None},
 {'count': 0, 'min': None, 'max': None, 'mean': None},
 {'count': 0, 'min': None, 'max': None, 'mean': None},
 {'count': 0, 'min': None, 'max': None, 'mean': None},
 {'count': 0, 'min': None, 'max': None, 'mean': None}
```

EDIT: I found some examples about getting the shapefile to the same crs as the raster. However, my shapefile does not have a crs. When I try to print the crs of the shapefile, I get None returned. And so when I try to transform it to the raster crs, I get the error message because there is no crs to transform. I find this very strange. Does anyone have an idea of why this might be so? Is there a problem with the shapefile?

python raster geopandas zonal-statistics rasterstats

Share Improve this question Follow edited Sep 23, 2022 at 15:02 asked Sep 22, 2022 at 20:17

Ryan
**13** 1 4

2 Answers

Sorted by:
Highest score (default)

2

The most common issue with this is your shapefile and raster do not have the same CRS. You can try fixing this with geopandas, converting the shapefile CRS to the raster CRS.

```
default_crs = 'EPSG:4326'

dataset = rasterio.open(raster_path)
arr = dataset.read(1)
affine=dataset.transform

zone = geopandas.read_file(shp_path)
# Set CRS to the default if needed. This does not
# transform the shapefile to a difference CRS.
if zone.crs is None:
    zone = zone.set_crs(default_crs)

# Transform to a difference CRS.
zone = zone.to_crs(dataset.crs)
stats = zonal_stats(zone, arr, affine=affine)
```

Share Improve this answer Follow edited Sep 23, 2022 at 17:28 answered Sep 22, 2022 at 21:19

Shawn
**1,606** 9 21

0

I was able to get this to work, thanks Shawn for pointing me in the right direction. It turned out my shapefile initially had no crs, so I had to write `zone.set_crs(epsg"4326", inplace=True)` . I just chose 4326 randomly to start. Then I set the shapefile crs to the raster crs with `zone=zone.to_crs(dataset.crs)` like Shawn pointed out. And now it's working.

Share Improve this answer Follow

answered Sep 23, 2022 at 15:14
Ryan
**13** 1 4