Geoprocessing in python (SS 2019)

Geography Department, Humboldt-University Berlin
Matthias Baumann, Dirk Pflugmacher

Assignment III – Finding overlap areas of raster-files and calculate basic stats

Due date	Sunday, 12.5.2018 10pm
Submission form	1. Questions in relation to the scripting task within a moodle-quiz. 2. py-Script
Evaluation criteria	• Correctness of the asked results (50%) → moodle quiz • Functionality of the py-script (50%).

Goal of this assignment

The overarching goal of this assignment is to get familiar with the basic structure of *gdal*-datasets, and how to make use of them in an efficient way. Important components here are the *GeoTransform, Raster(XY)Size* and corresponding elements. In addition, you will need to load the raster-files into arrays in order to do some basic manipulations. A good use of this assignment (not mandatory, though) is to start developing some smaller tools (i.e., *function*), which you may then be able to use in assignments of the following weeks again without modifying them too much. A potential example for that may be a function to calculate the coordinates of the extent of an image.

Exercise – finding overlap area across raster files and calculate basic image statistics.

You are being asked to calculate some basic image statistics of several raster files. These files represent median EVI composites of a study region in northwest Argentina, globally one of the areas with the highest deforestation rates. You have overall data of five years (2000, 2005, 2010, 2015, 2018), and all of them are stored as individual *tif*-files. It turns out that all the files have different spatial extents, which means that image statistics across images (i.e., years) are only meaningful for the overlap area of all raster files.

Your tasks in this assignment are as follows: **first**, you will have to identify what the overlap area between the different raster files is (i.e., the maximum common extent). Once you have found this area, please store the coordinates of all four corners and fill them later in the respective quiz question in *moodle* (rounded to three decimals). **Second**, apply image slicing using *numpy*. Here, I want you to only load the part of the raster files into individual arrays that is contained in the overlap area. Please write down the size of the overlap (i.e., the array dimensions) and insert the numbers in the respective quiz question in *moodle*. **Third**, for each year please calculate the *mean, median, min, max, range, standard deviation* for each year, store the numbers, and insert them into the respective quiz question in moodle. In addition, calculate the same statistics for the image differences between 2000-2010, and 2010-2018.

Format of the assignment – Expert puzzle

This week you will work for the first time in a setting that is called "expert puzzle". The larger group is subdivided into smaller groups, and each group will work on a smaller working package related to the overall task described above. The individual groups are described below, and each group has *60-90min* to get all the required information/skills/concepts ready – these are the *expert-groups*. Once the time is over, we will randomly shuffle the *expert groups* in a way, so that in each new group there will be at least one person from each expert group – these are the *working groups*. Within each working group, the expert knowledge will be shared, and all components put together in order to develop a concept to write the script (incl. smaller code fragments). You have a maximum of *30min* to develop such a concept. After that, the scripting can start. The working groups then decide together on how to proceed – whether they want to continue together writing the code in class and/or meet during the week, or whether they want to coordinate their efforts through moodle or via email. In the end, each group member has to do the submission individually.

Important in this format is that you are strongly involved in your expert-group. You need to know and understand what you found out, and you should take notes as you will later-on have to provide your expertise in the working groups!

Group I – the "Overlap"-group

The task of this group is to provide the basis for the script to run with the geospatial data. You need to find out the required commands/functions/etc. by working through the provided materials. Import points to consider (among others that I want you to think about) for this tasks are:

- How to open a raster-file?
- What are the individual parts of the GeoTransform, the projection etc. and how can I make best use of it?
- How is the overlap area defined conceptually?
- …

You find some reading materials on how to do that in moodle, but also classic google-searches will help you. A good start could be to write some pseudo-code for the operation; but you can also start writing some python code already in case you want to run tests.

Group II – the "numpy" group

The task of this group is to explore what the basic numpy-elements are that you need for this assignment. Important points for this group include (but are not limited to):

- How do I create slices of arrays?
- How do I perform array operations?
- How do I perform operations with multiple arrays?
- …

Most of the required information for this are in the readings provided in moodle. If you find some spare time in your group, you can make use of it by already writing down the basic calculations with some "test -arrays" (e.g., comparable to those in the slides).

Group III – the gdal-numpy-transfer group

This group has the most challenging task. You will have to look through the literature in order to find out, how to translate geo-coordinates (which you will receive information about from group I) into array-coordinates (which will be needed by group II). The task is challenging, as we have not covered the individual functions during today's lecture part. Yet, there are some specific text passages in the moodle-materials (i.e., Chapter 9 of the book *Geoprocessing in python* with emphasize on section 9.3.1 ("Using real world coordinates" p. 193-196)), which will help you. Pay attention to the following commands/functions:

- GetGeoTransform()
- gdal.InvGeoTransform(), gdal.ApplyGeoTransform()
- map()
- ReadAsArray()

My recommendation would be to use one of the raster-files for the assignment to "play around" with the functions to see what they are actually doing.