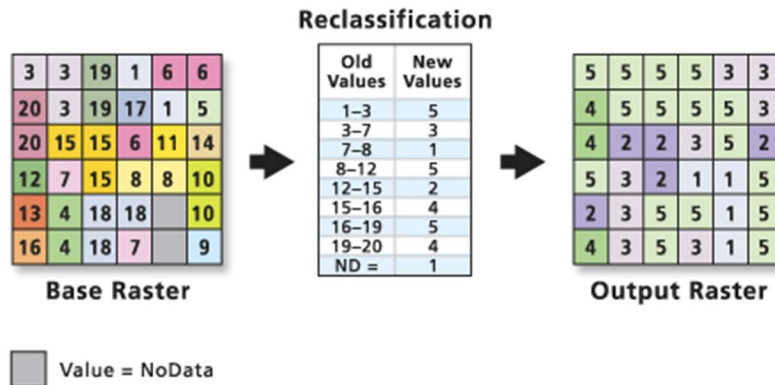


Lab 3: Working with raster data

Reclassifying Raster Data

In this lab, you will reclassify a raster dataset in Python. When you reclassify a raster, you create a new file in which each pixel is mapped to a new value based on some approach. This approach can vary depending upon your science question. For example, you can see that all cells that contain the values 1-3 in the figure below are assigned the new value of 5.



Dataset:

We will work with two raster datasets: a DTM (digital terrain model) and a DSM (digital surface model) of the same area. By performing a subtraction operation, we will get the difference between them, which indicate the heights of the geographic objects (mostly trees). We will then classify the trees based on their heights.

- “pre_DTM.tif”: DTM data file.
- “pre_DTM_hill.tif”: hillshade visualization of the DTM file; This is for visualization purpose only in order to give you an intuitive idea of the data. We will NOT use this file.
- “pre_DSM.tif”: DSM data file.
- “pre_DSM_hill.tif”: hillshade visualization of the DSM file; This is for visualization purpose only in order to give you an intuitive idea of the data. We will NOT use this file.

Tasks:

- What are the numbers of band(s) of these two raster datasets? (10 pts)
- What are the heights and widths of these two raster datasets? (10 pts)
- Plot out these two datasets using their default visualization. (10 pts)

Next, read the values (elevations) from the first band of these two datasets. Because both datasets have null values, we will use *masked=True*. So your code should look like below:

```
dtm_data = dtm.read(1,masked=True)
```

Subtracting the two gives us the heights of the geographic features:

```
heights = dsm_data - dtm_data
```

- What are the maximum and minimum heights? (20 pts)

- Create a histogram of the heights, and set the bins as *range(0,31,2)*. (10 pts)

In the next step, we will reclassify the raster cells based the heights. Particularly, we will have:

- No trees: (0m - 2m tall)
- Short trees: (2m - 7m tall)
- Medium trees: (7m - 12m tall)
- Tall trees: (> 12m tall)

We will reclassify the raster using the rule below. Notice in the matrix below we use *Inf* to represent the largest or max value found in the raster. So our assignment is as follows:

- 0 - 2 meters -> 1
- 2 - 7 meters -> 2 (short trees)
- 7 - 12 meters -> 3 (medium trees)
- > 12 meters -> 4 (tall trees)

np.digitize function

Numpy has a function called *digitize* that is useful for classifying the values in an array. This function will replace each data point with an integer corresponding to the index of the value range it belongs to. You can learn more about this function here:

<https://numpy.org/doc/stable/reference/generated/numpy.digitize.html> You can also Google more examples about how to use np.digitize.

- Use the np.digitize function to perform the reclassification of heights, and visualize the reclassified raster. (30 pts)

- Save your reclassification result into a new raster (10 pts)

Please submit your work as a Jupyter Notebook:

- Lab3_FirstName_LastName.ipynb