

ĐẠI HỌC QUỐC GIA HÀ NỘI
ĐẠI HỌC CÔNG NGHỆ



BÁO CÁO MÔN HỌC
MÔN: TRÍ TUỆ NHÂN TẠO
TUẦN 2

Thành viên:
Hoàng Quốc Anh
Nguyễn Bá Chung
Trần Hữu Quốc Đông
Phạm Quang Hùng
Ngô Thị Ngọc Quyên

MỤC LỤC

I. Đọc ảnh vệ tinh bằng Python	3
II. Thông kê từng band	3
1. Code	3
2. Bảng thống kê các band	3
III. Tính toán chỉ số NDWI, NDVI	4
1. Tính toán chỉ số NDWI	4
2. Tính toán chỉ số NDVI	4
3. Xuất bản đồ NDWI, NDVI dưới dạng tif	4
IV. Hiện thị bản đồ trên phần mềm Arcmap	5
1. Hiện thị bản đồ chỉ số NDWI	5
2. Hiện thị bản đồ chỉ số NDVI	6
V. Kết quả	6
1. Link code	6
2. Link ảnh tif	6

I. Đọc ảnh vệ tinh bằng Python

Để đọc được cũng như xử lý ảnh vệ tinh, Python có một thư viện hỗ trợ là GDAL. Để cài được thư viện, nhóm đề xuất sử dụng bộ công cụ quản lý thư viện Open Sources Conda (Anaconda).

Để nhập thư viện GDAL cũng như các thư viện cho xử lý ảnh, chương trình code sẽ như bảng dưới đây:

```
# Raster Processing Libraries
from osgeo import gdal
import os
import numpy as np
```

Để nhập ảnh vệ tinh vào ma trận ảnh, sử dụng thư viện GDAL với dòng lệnh như sau:

```
# Get Raster
data_path = 'Input/LANDSAT_7_Bands_30M.tif'
ds = gdal.Open(data_path, gdal.GA_ReadOnly)
data = ds.ReadAsArray()
transform = ds.GetGeoTransform()
```

Như vậy, ma trận ảnh vệ tinh sẽ là ma trận *data*.

II. Thông kê từng band

1. Code

Ma trận *data* bao gồm 7 band giống như ảnh đầu vào. Tuy nhiên, nếu ảnh đầu vào được xử lý để chuyển về hệ UTM qua Arcmap, các giá trị NoData sẽ bị chuyển về giá trị xấp xỉ vô cùng. Vậy nên, cần một bước xử lý để chuyển các giá trị nằm ngoài Histogram ảnh, dòng lệnh như sau:

```
# Update Band with Threshold
for i in range(data.shape[0]):
    for j in range(data.shape[1]):
        for k in range(data.shape[2]):
            if data[i][j][k] < -0.2:
                data[i][j][k] = None
```

Để thống kê cho các band, các giá trị NoData cần phải bỏ qua, thư viện numpy có các hàm nan có thể xử lý vấn đề này, đoạn lệnh in các giá trị thống kê như sau:

```
# Debug
for i in range(data.shape[0]):
    print("Band " + str(i + 1) + ": ", np.nanmax(data[i]),
          np.nanmin(data[i]), np.nanmedian(data[i]), np.nanmean(data[i]),
          np.nanstd(data[i]))
```

Các giá trị thống kê của ảnh được in ở trên console.

2. Bảng thống kê các band

	Max	Min	Median	Mean	StD
<i>Band 1</i>	0.4591750	-0.1827575	0.0405425	0.0443643	0.0184301
<i>Band 2</i>	0.5117275	-0.1735450	0.0493012	0.0526391	0.0198314
<i>Band 3</i>	0.5765175	-0.0496025	0.0875537	0.0886249	0.0220179
<i>Band 4</i>	0.6626200	-0.0498363	0.0763475	0.0794651	0.0302105
<i>Band 5</i>	0.7839500	-0.0024400	0.2783075	0.2730874	0.0835403
<i>Band 6</i>	1.1017813	-0.0000200	0.1782213	0.1718088	0.0639222
<i>Band 7</i>	1.2038200	0.0019600	0.0899325	0.1024409	0.0563200

III. Tính toán chỉ số NDWI, NDVI

1. Tính toán chỉ số NDWI

- Công thức tính chỉ số NDWI

$$NDWI = \frac{X_{green} - X_{nir}}{X_{green} + X_{nir}} = \frac{\text{band 3} - \text{band 5}}{\text{band 3} + \text{band 5}}$$

- Code:

```
# Get Bands Image
band3 = ds.GetRasterBand(3)
band3 = band3.ReadAsArray()
band5 = ds.GetRasterBand(5)
band5 = band5.ReadAsArray()

# Calculate NDWI
ndwi = (band3 - band5) / (band3 + band5)
```

Trong đó, *band3* và *band5* là 2 ma trận ảnh của band Green và NIR.

2. Tính toán chỉ số NDVI

- Công thức tính chỉ số NDVI

$$NDVI = \frac{X_{nir} - X_{red}}{X_{nir} + X_{red}} = \frac{\text{band 5} - \text{band 4}}{\text{band 5} + \text{band 4}}$$

- Code:

```
# Get Bands Image
band4 = ds.GetRasterBand(4)
band4 = band4.ReadAsArray()

# Calculate NDVI
ndvi = (band5 - band4) / (band5 + band4)
```

Trong đó, *band4* và *band5* là 2 ma trận ảnh của band Red và NIR.

3. Xuất bản đồ NDWI, NDVI dưới dạng tif

Để xuất các ma trận mới bao gồm *ndwi* và *ndvi* ra ảnh vệ tinh, chương trình cần một hàm xuất ảnh, có các đoạn lệnh như sau:

```
def array2raster(filename, transform, array):
    # Get Image Array Length
    rows, cols = array.shape

    # Create Raster
    driver = gdal.GetDriverByName('GTiff')
    out_raster = driver.Create(filename, cols, rows, 1,
gdal.GDT_Float64)

    # Set Transform
    out_raster.SetGeoTransform(transform)

    # Add Image to Raster
    outband = out_raster.GetRasterBand(1)
    outband.WriteArray(array)

    # Set Coordinate System and Project
    out_raster_project = gdal.osr.SpatialReference()
    out_raster_project.ImportFromEPSG(4326)
    out_raster.SetProjection(out_raster_project.ExportToWkt())
    outband.FlushCache()
```

Ma trận ảnh sẽ được truyền vào hàm *array2raster* để xử lý, chương trình như sau:

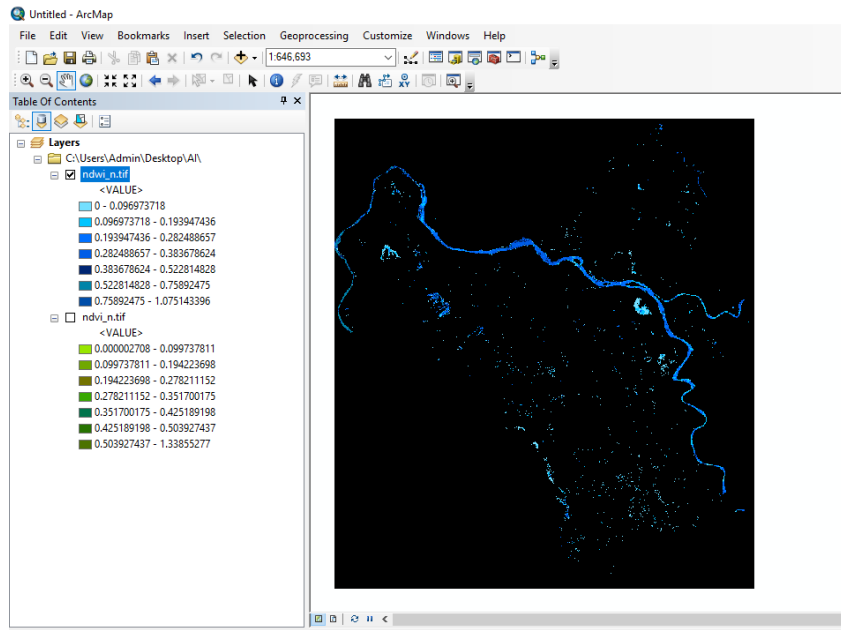
```
# Save NDWI and NDVI to New Raster
array2raster('Output/ndwi.tif', transform, ndwi)
array2raster('Output/ndvi.tif', transform, ndvi)
```

Cuối cùng, đầu ra sẽ là 2 ảnh vệ tinh *ndwi.tif* và *ndvi.tif* trong thư mục Output.

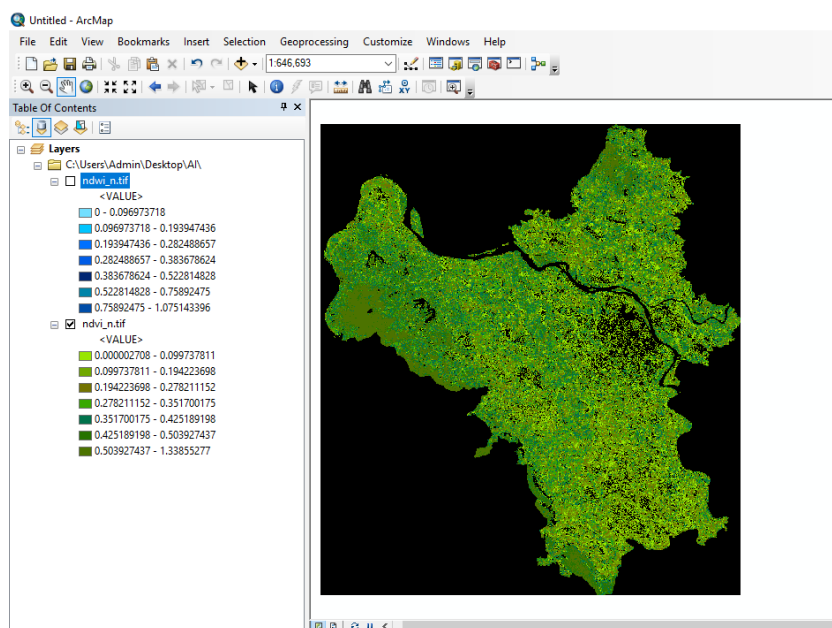
IV. Hiện thị bản đồ trên phần mềm Arcmap

- Add data *ndwi.tif* và *ndvi.tif* thu được ở phần III vào phần mềm Acrmap.
- Trên toolbars chọn *Geoprocessing* → *Search for tool* → *Calculate Statistics (data management)*
- Sử dụng tool '*Calculate Statistics*', chọn *input raster dataset* là ảnh tif cần xử lý
- Double click vào ảnh tif, chọn *Symbology* → *Classified* → số *classes* cho vùng nghiên cứu → chọn *symbol* cho từng *range* → *apply*

1. Hiện thị bản đồ chỉ số NDWI



2. Hiện thị bản đồ chỉ số NDVI



V. Kết quả

1. Link code

https://github.com/Windrist/BaiTap_AI_UET

2. Link ảnh tif

https://github.com/Windrist/BaiTap_AI_UET/tree/master/Output