## ECE471 - Homework #1

Course: Fundamentals of Remote Sensing and Earth Observation

Instructor: Krishna Karra

Email: [krishna.karra@gmail.com](mailto:krishna.karra@gmail.com)

All homework materials can be found in this [google drive folder](https://drive.google.com/drive/folders/1wYwfxP882qSENMdlYFA1EyubzWvWmvvT?usp=sharing).

*Assigned:* Monday, Feb 1, 2021

*Due:* Friday, Feb 19, 2021

Please submit your assignment by sharing a Google Colaboratory notebook shared with the instructor by the due date. All code and accompanying written portions should occur within a single notebook.

You are expected to provide written explanations and accompanying images, graphs and visualizations when appropriate. Your code is expected to contain comments and be straightforward to follow for someone knowledgeable in Python.

Some students may want to perform initial development and experimentation on their local machines rather than directly in Google Colaboratory. This is totally fine, but when finished, you should copy over your final result into Google Colaboratory.

### Problem 1

In this first first problem, you will work with GeoTIFF and GeoJSON files, and use GDAL to manipulate geospatial data. You will also use the Python scientific stack to implement simple image processing algorithms, composite (e.g. temporal) operations and remote sensing indices from band data.

For this problem, you will be analyzing and processing imagery from Sentinel 2 (L1C, Top of Atmosphere) taken over the greater Santa Fe metro area from 2019 to 2020. Each GeoTIFF file contains seven bands *[red, green, blue, nir, swir1, swir2, alpha]*. Upon examination of this dataset, you will notice that the resolution and coordinate reference system of each file does not match.

There is a zip file that contains the contents of this dataset, called *s2\_santafe.zip.*

#### Task 1 - Align the dataset

For the first part of this problem, you are asked to create a spatially aligned dataset from the provided dataset. Specifically, every file in your output dataset should be at the same resolution, coordinate reference system and spatial extent. I would recommend projecting all your images to UTM. Your code should take the provided input dataset and write out the output dataset. If using GDAL on the command line rather than with the Python bindings, your code can be a bash script or the accompanying equivalent in Python.

There is a GeoJSON file that contains the spatial extent that each image in your output dataset should match, called *santafe\_crop.geojson*.

#### Task 2 - Analyze the dataset

Once you have created an aligned dataset, you will perform some analysis on this dataset.

First, you are asked to compute a histogram of values across the entire temporal stack each of the six bands (excluding the alpha band).

Next, across the temporal stack, you are asked to:

* Find the greenest scene (e.g. most vegetated scene -> max(NDVI))
* Find the snowiest scene (NDSI)
* Find the cloudiest scene
* Find the brightest scene

Note that your outputs should be from the result of your technique / code / algorithm running on the stack of imagery. You are NOT allowed to produce your answers simply through visual inspection of the data, although you will certainly want to inspect the data closely to figure out what approach to take. For each answer, provide the scene ID and the corresponding image to answer each question.

Finally, you are asked to create composite images (e.g. reduce the stack of imagery to a single image) of varying kinds:

* mean
* min
* max
* median
* greenest pixel (e.g. argmax NDVI)
* 85% greenest pixel

For each temporal operation, your output should be a GeoTIFF file that contains the georeferenced composite image.

For the purposes of measuring cloud cover across the scene, you will want to implement a simple cloud masking algorithm. Don’t go for something perfect, rather get something that works reasonably well. Note that this same cloud masking algorithm can be used for creation of composite imagery, by masking each image by your derived cloud mask. Finally, note that you do not need to necessarily use every single image / pixel for your composite operation.

### Grading

You can earn 10 points (out of 100 for the class) by doing the following:

Problem 1

* Create an aligned GeoTIFF dataset (3 points)
* Implement a reasonable cloud mask algorithm (2 point)
* Produce correct outputs for single scenes (3 points)
* Create high quality composite images in GeoTIFF format (2 points)

The instructor will award a maximum of 3 bonus points for this assignment. You can achieve bonus points by:

* Implementing a particularly excellent cloud masking algorithm for Problem 1 (1-2 points)
* Providing compelling visualizations that go along with your analysis (1-2 points)

### Plagiarism

Internet research and collaboration with other students is highly encouraged. However, copying code from another student directly is strictly forbidden. Any student may be called upon to do a detailed code walkthrough with the instructor after submission of the assignment. Failure or an inability to explain your code can result in a serious violation of ethics guidelines and possible disciplinary action.