**Applied Remote Sensing Training (ARSET) Program**

Species Distribution Modeling with Google Earth Engine

# Optional Hands-On Activities

These activities are **optional** for your own practice and are **not** required to earn a certificate for the training.

## PART 1

The following exercises have been developed with the idea that you can practice your javaScript coding skills. Open the GEE code editor. If you can't find the code editor, you can navigate to it under Google Cloud > Earth Engine > Overview > Connect > Code Editor. Note that the files in this repository will open in a read-only mode. To save your changes, click "save" and select "save a copy."

1. In the code editor panel of the Google Earth Engine interface, use JavaScript code to create 3 different objects. You can give any name to these objects, but each object needs to be a different data type. Print your objects in the console by clicking the run button and make sure your code runs without errors.
2. Here is a list of JavaScript code commands. Each line of code contains an error (You can run the line of code to see the error). Find the error and fix it, so the code runs successfully.

```
print(this is my first code);

print(4 + "4");

var mylist = [0,1,2,a,d,c];
print(mylist)
```

```
var number = 45;
print(Number)

var mysecondlist [1,2,4];
print(mysecondlist)

var mydictionary = [
   first: 1,
   second: 'a string',
   third: [1,2,3,5,8,12]
];
print (mydictionary);
```

3. Using the search tab on the Docs panel, find the function add for ee.Number objects. Using this function, add 5 to 5 and store the result as a new variable. Print this variable to the console.
4. It is important that you understand the difference between JavaScript objects and Earth Engine objects. As an exercise, transform the following JavaScript objects into their equivalent Earth Engine object (the ee.Thing).

```
var animal = 'Tiger';
var number = 78;
var list = [1,2,3];
```

5. The following function does not work. Find the error and fix it:

```
var multiplyby2 = function(number) {
  var result = number.multiply(2);
  return result;
};
print('Result:', multiplyby2(4));
```

Access the solutions code.

Grant covered the basics of image, image collection, feature and feature collection operations in Google Earth Engine. The set of tools you have learned are the building blocks for more

complex analyses like SDMs. Thus, it is important to make sure you are familiar with these operations.

Here are some activities that will help you practice the skills we covered. This assignment consists of four activities with increasing levels of difficulty.

1 [Easy]

- Select any image dataset from the GEE catalog
- Export this image as a GeoTIFF to your Google Drive for a study area of your choosing (tip: don't make your study area too big or the export can take a long time)

2 [Moderate]

- Select any area of interest and create a time series chart to compare mean NDVI and EVI from the MODIS satellite (Collection Name: MOD13Q1.006 Terra Vegetation Indices 16-Day Global 250m of MODIS)

3 [Advanced]

- Write a function to calculate a Modified Soil Adjusted Vegetation Index band (MSAVI) - - https://www.usgs.gov/landsat-missions/landsat-modified-soil-adjusted-vegetation-index
- Map this function across a Landsat 8 collection to add an MSAVI band to each image
- Create a median pixel composite and add this layer to your map window

Access the solutions code.

## PART 2

The idea of this assignment is that you can practice modifying the general workflow we presented in the workshop for implementing species distribution models in GEE.

The assignment consists of four activities with increasing levels of difficulty. Completing all of them is optional.

1 [Easy]

Modify the general workflow presented to fit an SDM using presence only data for the Mountain tapir (*Tapirus pinchaque*). The data is available at the following asset: users/ramirocrego84/Tapirus_pinchaque

The model needs to include the following:

Grain size: 1000 m

Predictor variables: elevation, slope, temperature seasonality, max temperature of warmest month, min temperature of coldest month, mean annual precipitation, mean NDVI from MODIS for the last 10 years (MODIS/061/MOD13Q1) and human modification index (CSP/HM/GlobalHumanModification)

Define area to create pseudo-absences using option 3 (clusters).

Define your own block size

Classifier: random forest

Iterations: 5

Access the solution.

2 [Moderate]

Modify the previous model to:

Use the country of Ecuador as your area of interest (Make sure you extract the geometry)

Keep only the presence data that are within continental Ecuador (excluding the Galapagos Islands)

Create pseudo-absences randomly within the country boundary

Define a 250 m grain size

Use the gradient tree boost classifier

Access the solution.

3 [Advanced]

Modify the model from Activity 1 to extract the variable importance from each iteration of the Random Forest.

To extract the variable importance from a classifier, you will need to modify the SDM function. First, you will need to create a new object using the following code: ee.Feature(null, ee.Dictionary(ClassifierPr.explain()).get('importance')). Here, ClassifierPr is the name of the classifier you trained.

Then you will need to modify the functions to extract results from the output list (now you should have 5 resulting objects per iteration instead of 4) and create a new function to extract the variable importance

Finally, calculate the mean variable importance for each predictor variable across all iterations. To get the average value across properties of features in a feature collection you use the function 'reduceColumns'. You also need to add the function 'repeat' to the reducer to apply it across the multiple properties.

Access the solution.

Activity 4 [Advanced]

Access the script which has a subset of the Wood Thrush data used in the case study 3 of the manuscript 'Implementing SDMs in GEE'

Create a cloud free mosaic using Sentinel2 (COPERNICUS/S2). Filter the image collection to years 2018-2019-2020 and for the months of  May and June. Use the following bands 'B2','B3','B4','B5','B6','B7','B8','B11', and B12' as predictor variables.

Create a habitat suitability model and export the resulting image as an asset.

Export the accuracy metrics to your Google Drive

Note: Exporting the outputs of this code can take between 30 and 50 minutes.

Access the solution.