Advanced Remote Sensing Lab 2: Introduction to Google Earth Engine in R

Victor Gutierrez ([victorhugo@temple.edu](mailto:victorhugo@temple.edu))

### Lab due

Febraury 02 2021

### Goals

1. To become familiar with the layout and functionality of the GEE platform.
2. To become familiar with basic JavaScript coding in general and applied to GEE
3. To introduce available data, basic object types and manipulation in GEE

### Total score

The lab counts for up to 3 points towards the final grade of the course.

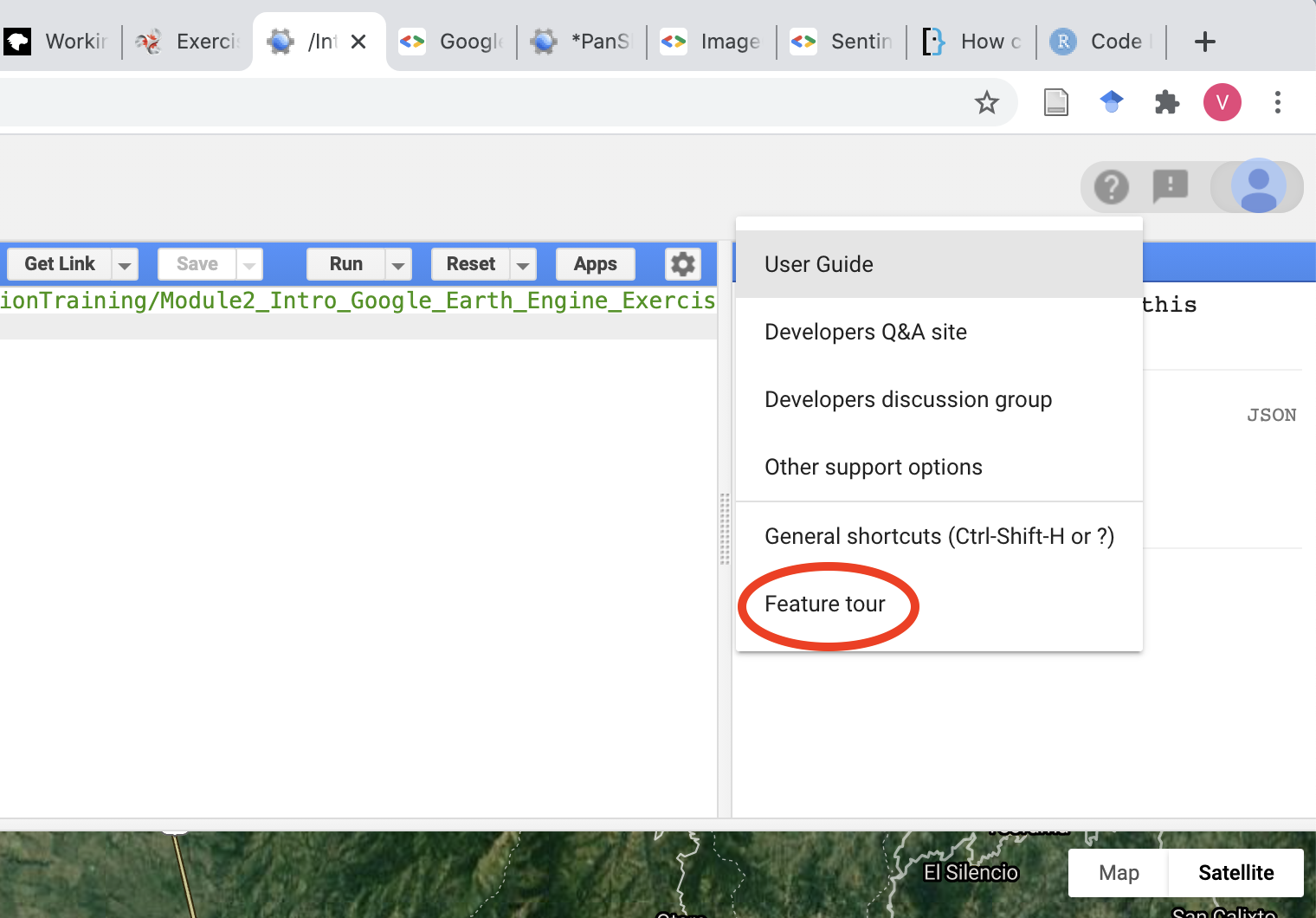
Have fun!

### Lab instructions

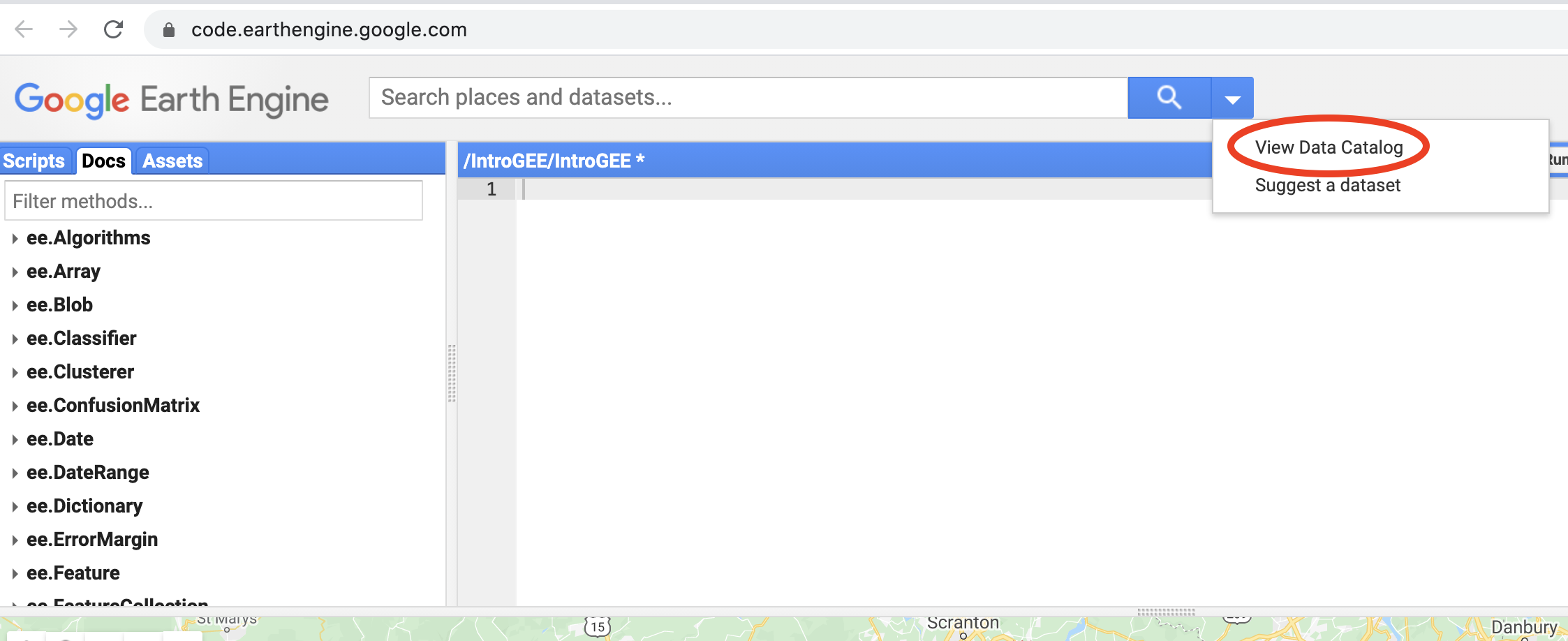
#### 1. Exploring GEE

Launch Google Earth engine in your browser (preferably chrome): <https://code.earthengine.google.com/>

Go to the help menu (? symbol in the top-left of the interface) and perform a “Feature Tour”



Click on the “Data Sets” tab at the top of the search panel. Then on the drop down menu select Landsat/Landsat Collection 1 Level-1/Landsat 8 OLI/TIRS C1 Level-1. Browse the datasets available



#### 2. Image visualization in GEE

Copy and paste the code below

// Get the image.  
var lc8 = ee.Image('LANDSAT/LC8\_L1T\_TOA/LC80140322016081LGN00')  
  
// Add the image to the map.  
Map.addLayer(lc8);  
  
// Center the map display on the image.  
Map.centerObject(lc8, 8);

You can see that the image looks very dark. Can you explain why?

Check the parameterization of the addLayer() function by searching on the Docs tab. Notice the name of the second parameter (visparams). Check the visualization parameters here: <https://developers.google.com/earth-engine/guides/image_visualization>

Use either of the options below to visualize the image. Modify visparams to improve the visualization as shown below. You can read about gamma correction here: <https://www.cambridgeincolour.com/tutorials/gamma-correction.htm>. Add also a name to the layer.

//OPTION 1: show layers individually  
Map.addLayer(lc8,  
{bands: 'B6, B5, B4', min: 0.05, max: 0.8, gamma: 1.6}, 'Landsat8TOA');  
  
//OPTION 2: show layers as a string  
Map.addLayer(lc8,  
{bands: ['B6', 'B5', 'B4'], min: 0.05, max: 0.8, gamma: 1.6}, 'Landsat8TOA');

#### 3. Explore some of the settings and the work environment.

Change the second argument of the centerObject() function from 8 to 10 and run. What happens? What happens when you change it to 6?

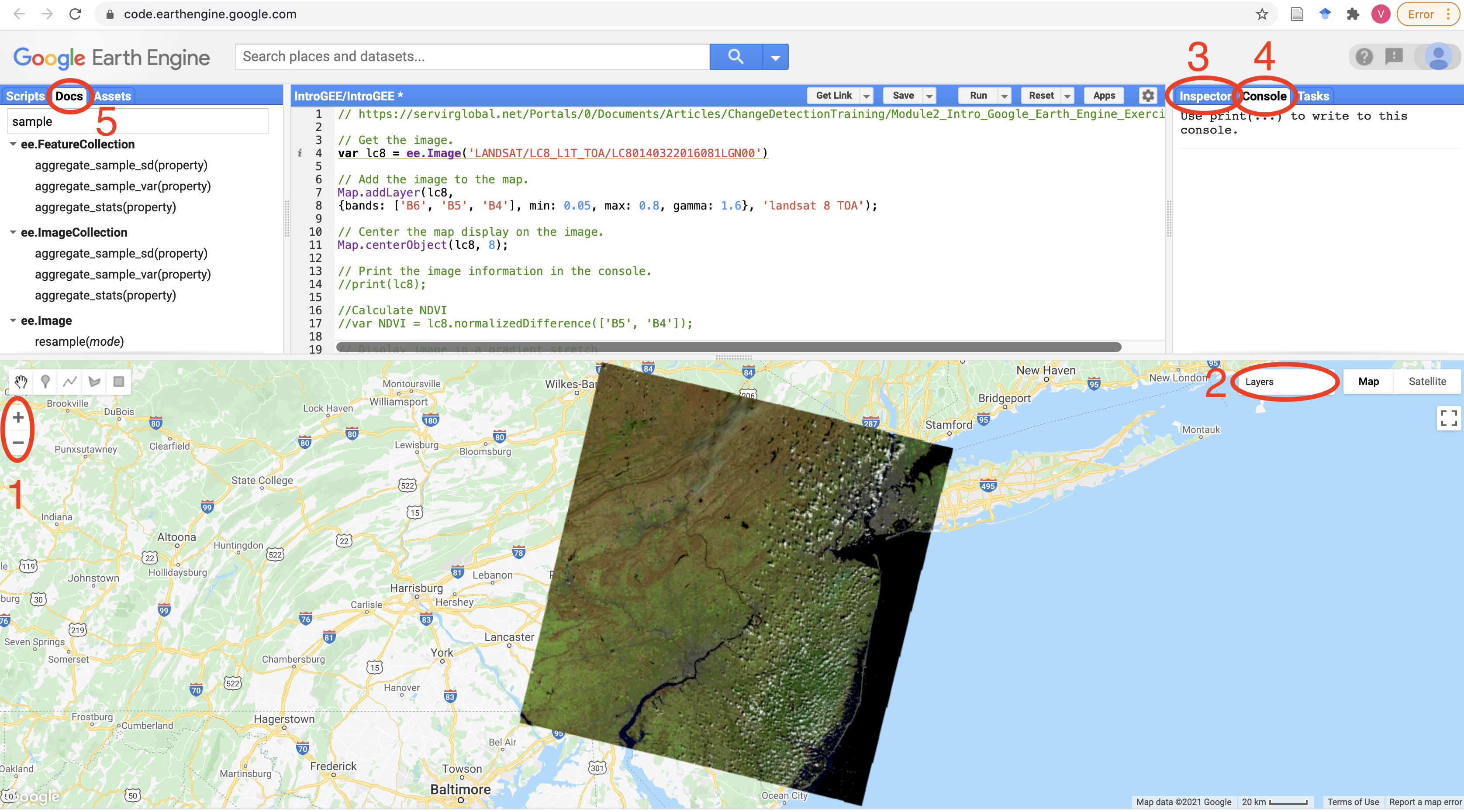
Explore the map window tool by zooming in and out (red circle #1).

Use the layers tool to turn the open layer on and off. Notice that the name is the same as the one that you specified in the addLayer() function (red circle #2).

Click on the inspector, then click anywhere in the image and explore the information that appears in the inspector. Make sure you expand each attribute (red circle #3).

Write the code below and explore the image properties in the console (red circle #4).

print(lc8);



#### 4. Calculate the NDVI for your image and display it.

Copy and paste the code below. The "palette argument uses a CSS style. You can check the color codes here: <https://www.quackit.com/css/css_color_codes.cfm>

//Calculate NDVI  
var NDVI = lc8.normalizedDifference(['B5', 'B4']);  
  
// Display image in a gradient stretch  
Map.addLayer (NDVI, {min: -0.2, max: 0.5, palette: ['FFFFFF', '339900']}, "NDVI");

#### 5. Mask clouds.

Comment the lines used to calculate the NDVI and display it so that those lines are not run. The code should look like this

// Get the image.  
var lc8 = ee.Image('LANDSAT/LC8\_L1T\_TOA/LC80140322016081LGN00')  
// Add the image to the map.  
Map.addLayer(lc8,  
{bands: ['B6', 'B5', 'B4'], min: 0.05, max: 0.8, gamma: 1.6}, 'landsat 8 TOA');  
  
// Center the map display on the image.  
Map.centerObject(lc8, 8);  
  
//Calculate NDVI  
//var NDVI = lc8.normalizedDifference(['B5', 'B4']);  
  
// Display image in a gradient stretch  
//Map.addLayer (NDVI, {min: -0.2, max: 0.5, palette: ['FFFFFF', '339900']}, "NDVI");

Select the “Docs” tab (red circle #5) and search for “cloud”. Inspect the function “ee.Algorithms.Landsat.simpleCloudScore(image)”. Then write the code below:

// Add the cloud likelihood band to the image.  
var cloudscore = ee.Algorithms.Landsat.simpleCloudScore(lc8);  
  
// Add the cloud image to the map. This will display the first three bands as R, G, B by default.  
Map.addLayer(cloudscore, {}, 'Cloud Likelihood, all bands');

Click on the “inspector tab and click on different locations in the image. You can see that the new cloudscore image added a band named”clouds" to the original dataset. Let’s select that band and display it.

// Since you are interested in only the cloud layer, specify just this band to be displayed in the parameters of the Map.addLayer statement.   
Map.addLayer(cloudscore, {bands: 'cloud'}, 'Cloud Likelihood');

Select the inspector tab and then in different locations of the image to assess how the cloud score values differ in cloudy and non-cloudy areas. Based on your assessment, select the threshold that you think is best to discriminate between cloudy and non-cloudy areas. Let’s use that threshold to mask clouds. In the code below, I selected 40

// Isolate the cloud likelihood band.  
var cloudLikelihood = cloudscore.select('cloud');  
  
// Compute a mask in which pixels below the threshold are 1 (you can check the lt function in the Docs tab.  
var cloudPixels = cloudLikelihood.lt(40);  
  
// Add the image to the map.  
Map.addLayer(cloudPixels, {}, 'Cloud Mask');

In the “layers” tab (red circle #2) uncheck all the layers except the cloud mask and the Landsat toa layers. Select and de-select the cloud mask to compare with the RGB and assess whether you are satisfied with the result obtained. If not, change the threshold and run again.

Use the created mask to remove clouds from the Landsat image and check the results:

// Mask out the Landsat image.  
var lc8\_NoClouds = lc8.updateMask (cloudPixels);  
  
// Review the result.  
Map.addLayer(lc8\_NoClouds, {bands: ['B6', 'B5', 'B4'], min: 0.1, max: 0.5}, 'Landsat8scene\_cloudmasked');

Zoom into the Philadelphia area. How did the cloud masking perform in urban areas? Check the hazy cloud in the north-west. How well was the hazy area masked?

#### 6. Perform an unsupervised classification:

// select the bounding box of a Landsat-8 image to sample pixels  
var region = lc8\_NoClouds.geometry()  
print(region)  
  
// Select random pixels for training the unsupervised classification  
var training = lc8\_NoClouds.sample({  
 region: region,  
 scale: 30,  
 numPixels: 5000  
});  
  
// train cluster based on sampled pixels  
var clusterer = ee.Clusterer.wekaKMeans(15).train(training);  
  
// classify the entire image based on teh trained cluster  
var result = lc8\_NoClouds.cluster(clusterer);  
  
// Display the classified image with random colors.  
Map.addLayer(result.randomVisualizer(), {}, 'clusters');

Explore the results and modify the number of clustes in the variable “clusterer” as needed until the number of classes descriminate reasonably well the land covers in the landscape.

### Homework

Apply the steps described above to visualize and mask the following image: LC80440332016115LGN00. Then use it to produce the normalized burnt ratio (NBR) and perform a supervised classification.

Change the appropriate arguments in the script to produce the following outputs. Produce a screenshot of each one of them and submit them as part of your report here: <https://forms.office.com/Pages/ResponsePage.aspx?id=74FucSK1c0SOMRC9Asz25dmnkCS0Q29AsedCc0cCybpUOVlNOFFKNUQxNFJRSENNSFZLVVhWUEZRTy4u>

1. An optimal RGB visualization of the input image using bands 5, 4, and 3. Please modifiy the stretch and gamma parameters to optimize visualization. Include the function from GEE in your report
2. An NBR image with a color palette from red, to yellow, to blue You can find the NDBI formula here: <https://www.usgs.gov/core-science-systems/nli/landsat/landsat-normalized-burn-ratio>. Include the function from GEE in your report.
3. A cloud mask produced based on an optimal threshold applied to the cloudscore image.
4. An unsupervised classification applied to the masked image with the number of clusters that best represent the land cover types in the original image. Enter in the report the number of classes that you used.