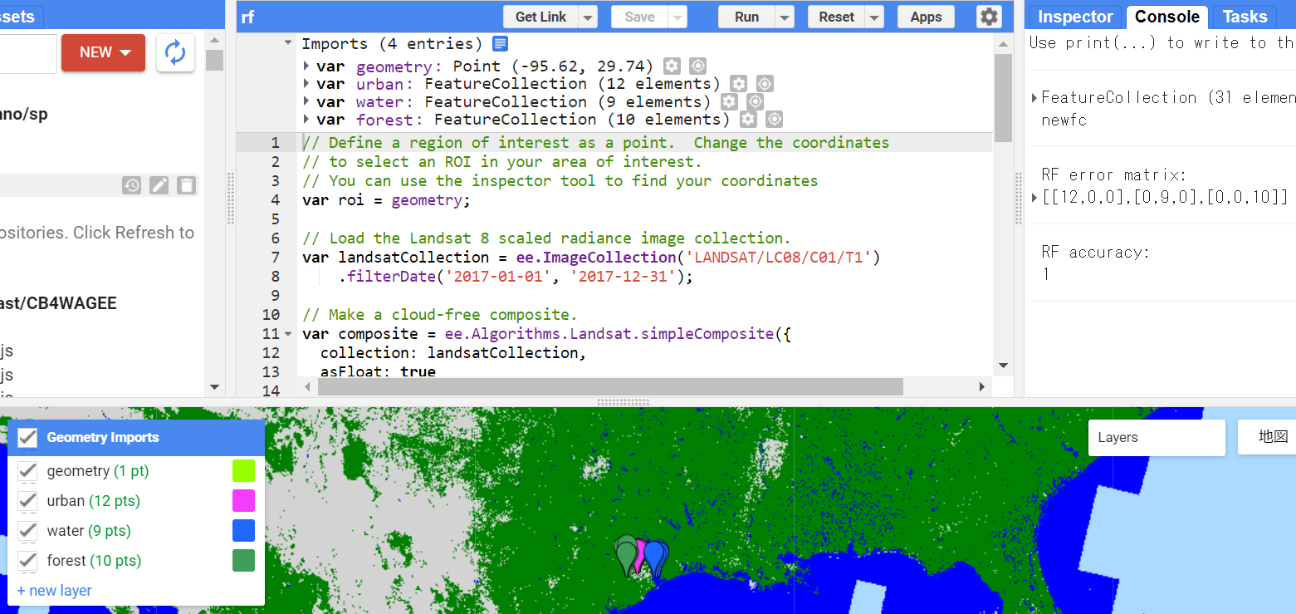
Load Landsat 8 images, composite, make a Random Forest classifier and train it.



// Define a region of interest as a point. Change the coordinates

// to select an ROI in your area of interest.

// You can use the inspector tool to find your coordinates

var roi = geometry;

// Load the Landsat 8 scaled radiance image collection.

var landsatCollection = ee.ImageCollection('LANDSAT/LC08/C01/T1')

.filterDate('2017-01-01', '2017-12-31');

// Make a cloud-free composite.

var composite = ee.Algorithms.Landsat.simpleComposite({

collection: landsatCollection,

asFloat: true

});

// Visualize the Composite

Map.addLayer(composite, {bands: ['B4', 'B3', 'B2'], max: 0.5, gamma: 2}, 'L8 Image', false);

// Merge points together

var newfc = water.merge(urban).merge(forest);

print(newfc, 'newfc')

// Select the bands for training

var bands = ['B2', 'B3', 'B4', 'B5', 'B6', 'B7'];

// Sample the input imagery to get a FeatureCollection of training data.

var training = composite.select(bands).sampleRegions({

collection: newfc,

properties: ['landcover'],

scale: 30

});

// Make a Random Forest classifier and train it.

var classifier = ee.Classifier.smileRandomForest(10).train({

features: training,

classProperty: 'landcover',

inputProperties: bands

});

// Classify the input imagery.

var classified = composite.select(bands).classify(classifier);

// Define a palette for the Land Use classification.

var palette = [

'D3D3D3', // urban (0) // grey

'0000FF', // water (1) // blue

'008000' // forest (2) // green

];

// Display the classification result and the input image.

Map.setCenter(-96.0171, 29.6803);

Map.addLayer(classified, {min: 0, max: 2, palette: palette}, 'Land Use Classification');

// Get a confusion matrix representing resubstitution accuracy.

print('RF error matrix: ', classifier.confusionMatrix());

print('RF accuracy: ', classifier.confusionMatrix().accuracy());