Department of Geography

CCST9083 Earth as Seen by Satellite

**Laboratory 4: Mapping Flooded areas in Google Earth Engine**

In this lab exercise, you will use the web-based code editor to generate a flood map in China's Henan province using Google Earth Engine. To access the Code Editor, simply browse for https://code.earthengine.google.com/.

Radar images from Sentinel-1 *(Sentinel-1 SAR GRD: C-band Synthetic Aperture Radar Ground Range Detected, log scaling)* will be used for this study, and more details can be explored by searching for the data in the search box. A window with a description of the data will open.
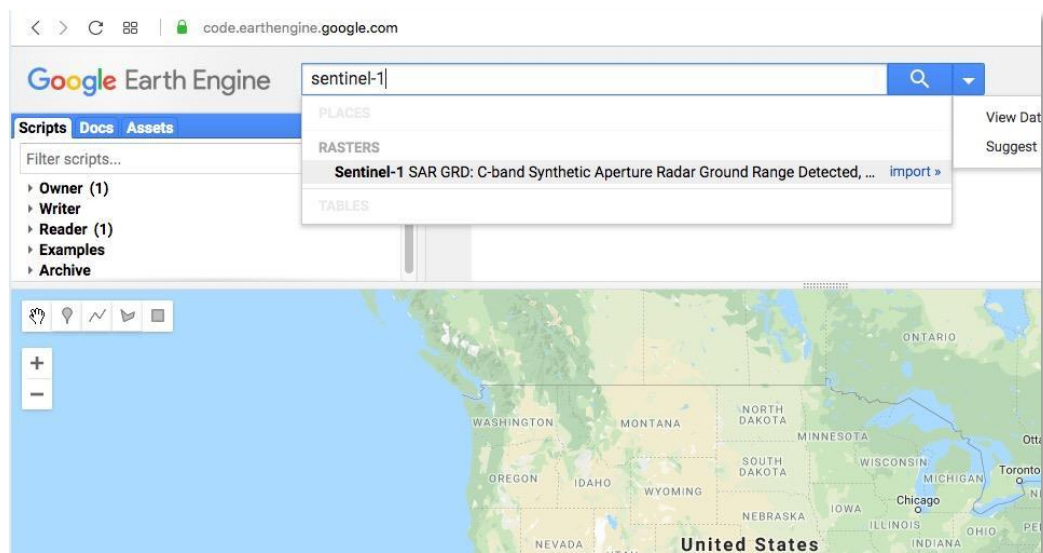
1. Area of interest

Our study is focused on floods that occurred in July of 2021 in Henan province, China. Draw a rectangle or polygon covering **Zhengzhou and Xinxiang**. The geometries drawn on the map will automatically be reflected in the Code Editor.
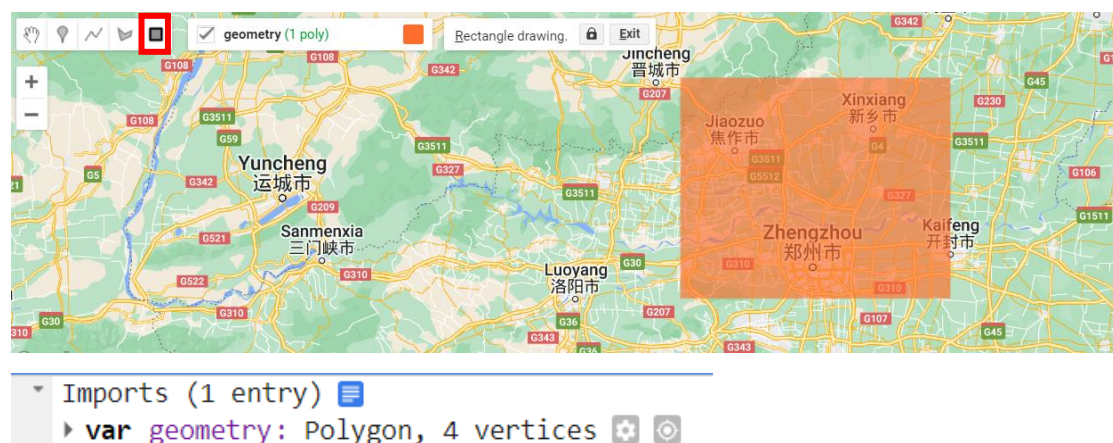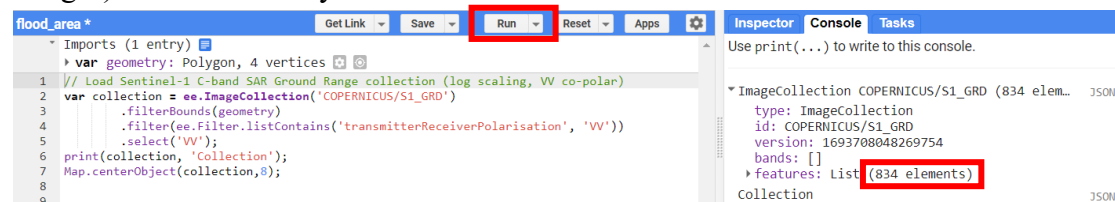
2. Filter the Sentinel-1 Data by Area

Load Sentinel-1 C-band SAR Ground Range collection through the code below:

```
// Load Sentinel-1 C-band SAR Ground Range collection (log scaling, VV co-polar)
var collection = ee.ImageCollection('COPERNICUS/S1_GRD')
        .filterBounds(geometry)
        .filter(ee.Filter.listContains('transmitterReceiverPolarisation', 'VV'))
        .select('VV');
print(collection, 'Collection');

Map.centerObject(collection,9);
```

Click **Run** button in the top menu. The right window will show the results for VV (834 images) in selected study area.


3. Filter the Sentinel-1 Data by Date

Filter the Sentinel-1 Data during the flood period (July, 2021):

```
// Filter by date
var before = collection.filterDate('2021-06-25', '2021-07-10').mosaic();
var after = collection.filterDate('2021-07-16', '2021-07-30').mosaic();

print(before, 'Before');
print(after, 'After');
```

Click **Run** button in the top menu. The right window will show the results in selected study area before and after flooding.
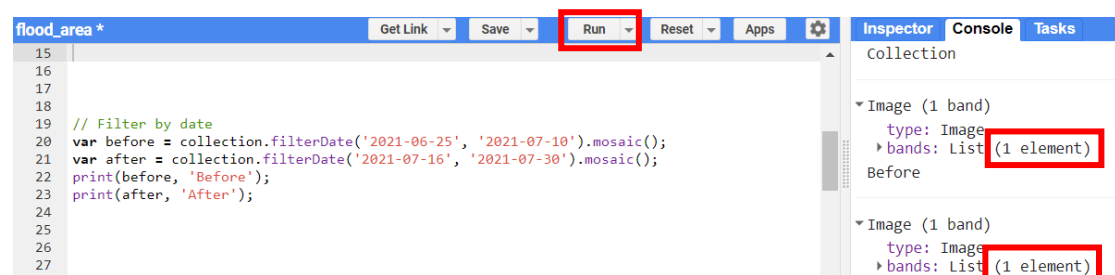
4. Display the Sentinel-1 Data

```
Map.addLayer(before, {min:-30,max:0}, 'Before flood');
Map.addLayer(after, {min:-30,max:0}, 'After flood');
```

Click the **Run** button in the top menu. The map window will display the Sentinel-1 images. Layer manager enables you to check different layers.
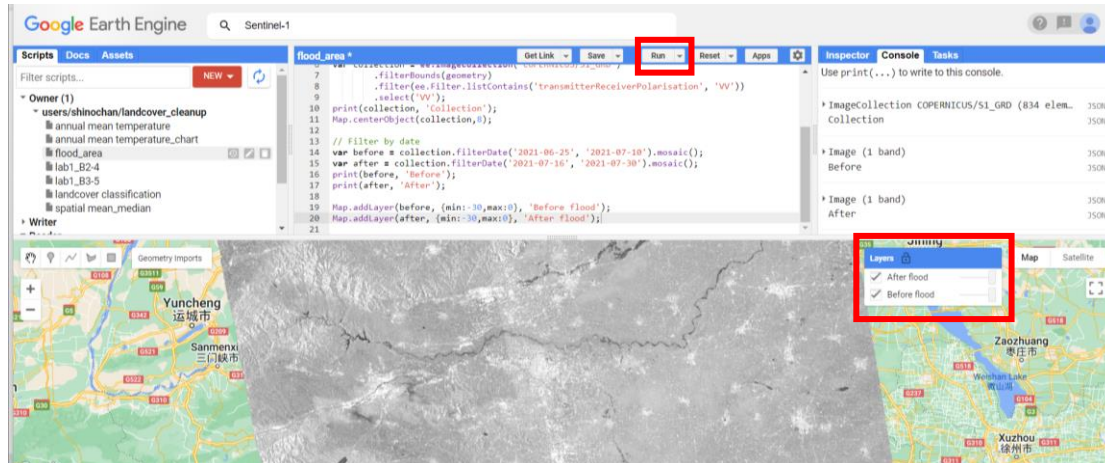
5. Apply smoothing filter

Smoothing filter is used to calculate the difference between 'Before' and 'After':

```
//Apply filter to reduce speckle
var diff = after.subtract(before);
Map.addLayer(diff, {min:-10,max:10}, 'After - before', 1);

var SMOOTHING_RADIUS = 100;
var diff_smoothed = after.focal_median(SMOOTHING_RADIUS, 'circle', 'meters')
.subtract(before.focal_median(SMOOTHING_RADIUS, 'circle', 'meters'));
Map.addLayer(diff_smoothed, {min:-10,max:10}, 'diff smoothed', 1);
```
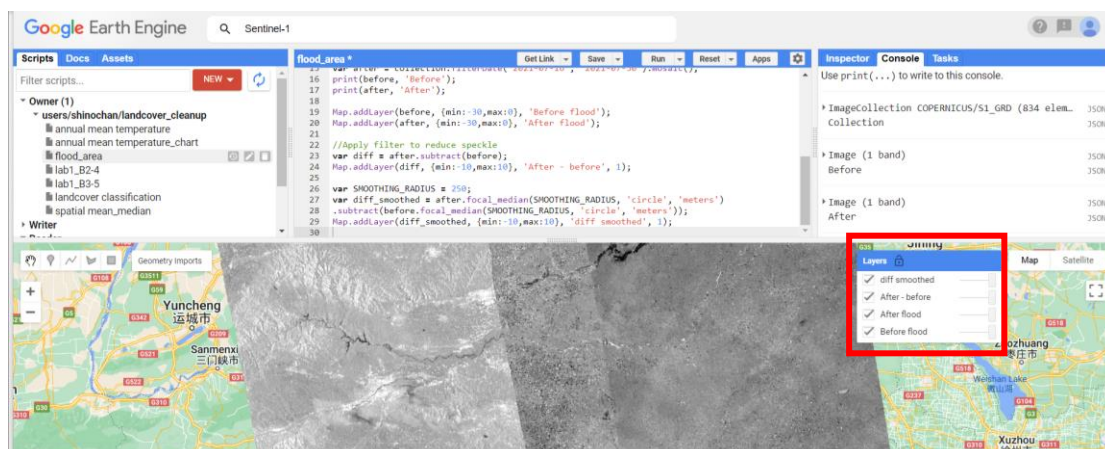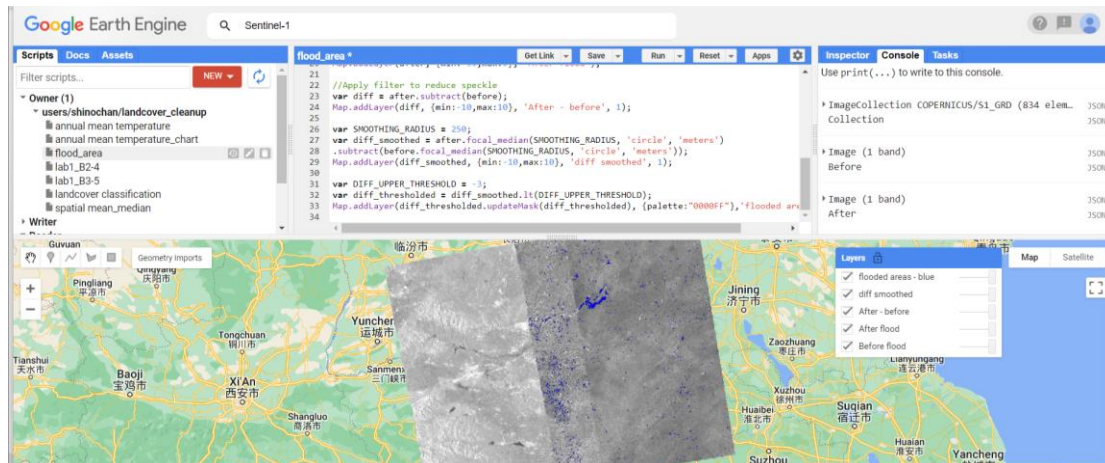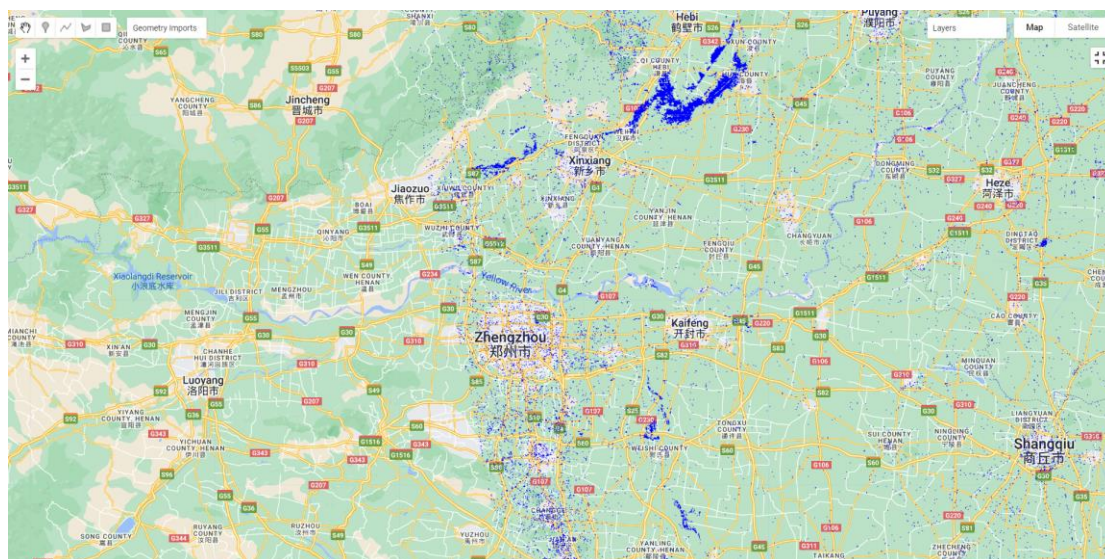
6. Apply a Threshold

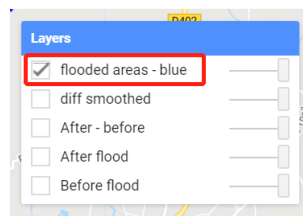Apply a threshold to identify flood areas:

```
var DIFF_UPPER_THRESHOLD = -3;
var diff_thresholded = diff_smoothed.lt(DIFF_UPPER_THRESHOLD);
Map.addLayer(diff_thresholded.updateMask(diff_thresholded),
{palette:"0000FF"},'flooded areas - blue',1);
```
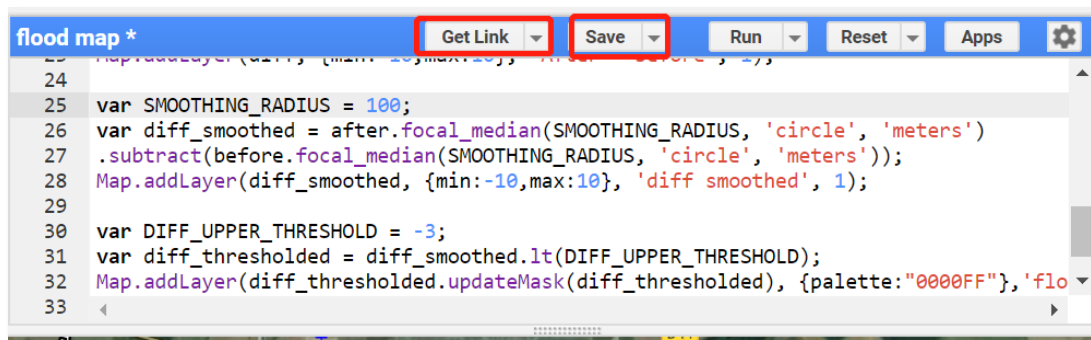

7. Display flood map

Go to the Layers manager, untick all the layers except the **'flooded areas – blue'**. Then toggle full-screen view.

8.   Save and <u>get the link</u> (optional).


```
flood map *                    Get Link  ▾    Save  ▾      Run  ▾    Reset  ▾    Apps    ⚙

24
25   var SMOOTHING_RADIUS = 100;
26   var diff_smoothed = after.focal_median(SMOOTHING_RADIUS, 'circle', 'meters')
27   .subtract(before.focal_median(SMOOTHING_RADIUS, 'circle', 'meters'));
28   Map.addLayer(diff_smoothed, {min:-10,max:10}, 'diff smoothed', 1);
29
30   var DIFF_UPPER_THRESHOLD = -3;
31   var diff_thresholded = diff_smoothed.lt(DIFF_UPPER_THRESHOLD);
32   Map.addLayer(diff_thresholded.updateMask(diff_thresholded), {palette:"0000FF"},'flo
33
```

< End >