

Tutorial: Dynamic Curve Number

NASA DEVELOP Riley County Water Resources – Summer 2020

Table of Contents

Procedure Summary	1
Part 1: Google Earth Engine Processing.....	2
Step 1.1 Import Riley County Study Area Shapefile into Google Earth Engine	2
Step 1.2 Prepare the Riley County Study Area Asset for Use**	3
Step 1.3 Import USGS Landsat Satellite Surface Reflectance Data Into Google Earth Engine.....	3
Step 1.4 Create a NDVI Function	4
Step 1.5 Apply the Function	5
Step 1.6 Recalibrate NDVI.....	5
Step 1.7 Export the NDVI Image with the Appropriate Scale and Region	5
Part 2: ArcGIS Pro Processing	6
Step 2.1 Import NDVI GeoTiffs into Arc Pro Document	6
Step 2.2 Calculate Dynamic Curve Number by using Raster Calculator on GeoTiffs	6

Procedure Summary



This material is based upon work supported by NASA through contract NNL16AA05C. Any mention of a commercial product, service, or activity in this material does not constitute NASA endorsement. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Aeronautics and Space Administration and partner organizations.

Part 1: Google Earth Engine Processing

Creating and Exporting NDVI Tiffs in Google Earth Engine

The DEVELOP Riley County Water Resources team is not pursuing a software release on the Google Earth Engine code we wrote to produce the NDVI output maps. As a result, we are unable to share any of the code we created from scratch. The following steps document the process for users to follow and write their own. All screenshots and code links are from publicly available code provided by Google Earth Engine Tutorials and other open source sites and are cited accordingly.

There is more than one way to correctly calculate NDVI through Google Earth Engine, or GEE. The general flow we followed is outlined here, and the specific steps we used are outlined below:

- Import Study Boundary
- Import Landsat image collections
- Create an NDVI Calculation Function that includes:
 - Filtered Boundary
 - Filtered Dates
 - NDVI function
 - Mapping the NDVI function over filtered image collection by mean or maximum NDVI
- Apply the NDVI Calculation Function to the year of interest
- Recalibrate NDVI imagery to prepare for export
- Export the NDVI GeoTiff

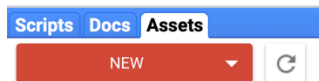
Step 1.1 Import Riley County Study Area Shapefile into Google Earth Engine

The Riley County Study Area Shapefile will be used to set the boundaries of the NDVI processing we wish to conduct. This helps limit the amount of time the processing takes and allows for the exported product to be constrained to the study area of interest.

Here is a link to [sign up for Google Earth Engine](#). You will need a gmail account.

Here is a [Google Earth Engine Tutorial](#) page that explains managing Assets

1. Upload the Shapefile using the “Assets” and “New” in the left top ribbon of your Google Earth Engine code page.



2. Hover your mouse over the new asset and click the blue arrow. This will import the Boundary Shapefile as a usable variable in your current script



3. In the upper section of your script where variables are located, rename the Shapefile variable with a desired variable name, such as RC_Boundary

This material is based upon work supported by NASA through contract NNL16AA05C. Any mention of a commercial product, service, or activity in this material does not constitute NASA endorsement. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Aeronautics and Space Administration and partner organizations.

Step 1.2 Prepare the Riley County Study Area Asset for Use**

The asset needs a little more processing before it is ready to be used in the bulk of the script.

1. Add an arbitrary property to the Riley County Study Area Asset.
2. Create a mask for Imagery and apply this to the variable for the study area asset variable that now has an arbitrary property.

**Skip this step if your asset boundary is showing up; this step is helpful for additional asset processing to ensure that the desired boundary is applied like a mask to the NDVI imagery.

Step 1.3 Import USGS Landsat Satellite Surface Reflectance Data Into Google Earth Engine

The U.S. Geological Survey (USGS) provides Landsat surface reflectance data for public access, available for direct import in Google Earth Engine. These data have been atmospherically corrected using [LaSRC](#) and include a cloud, shadow, water and snow mask produced using [CFMASK](#), as well as a per-pixel saturation mask. See Table 1 for data products used in this study.

Table 1
NASA Earth Observations for Riley County Water Resources Study

Data Product	Description	Collection Snippet	Years Covered in This Study
USGS Landsat 5 Surface Reflectance Tier 1	This dataset is the atmospherically corrected surface reflectance from the Landsat 5 ETM sensor.	ee.ImageCollection("LANDSAT/LT05/C01/T1_SR")	2006-2012
USGS Landsat 7 Surface Reflectance Tier 1	This dataset is the atmospherically corrected surface reflectance from the Landsat 7 ETM+ sensor.	ee.ImageCollection("LANDSAT/LE07/C01/T1_SR")	2013
USGS Landsat 8 Surface Reflectance Tier 1	This dataset is the atmospherically corrected surface reflectance from the Landsat 8 OLI/TIRS sensors.	ee.ImageCollection("LANDSAT/LC08/C01/T1_SR")	2013-2019

1. In the search tab above the middle of the script, search the image collection name based on the Table above.
 - a. Import Landsat 5, then Landsat 7, then Landsat 8.

USGS Landsat 5 Surface Reflectance Tier 1



Dataset Availability

- true

Dataset Provider

USGS

Collection Snippet

```
ee.ImageCollection("LANDSAT/LT05/C01/T1_SR")
```

Tags

global sr reflectance
ledaps cloud fmask
cfmask lt05 landsat

DESCRIPTION BANDS IMAGE PROPERTIES

This dataset is the atmospherically corrected surface reflectance from the Landsat 5 ETM sensor. These images contain 4 visible and near-infrared (VNIR) bands and 2 short-wave infrared (SWIR) bands processed to orthorectified surface reflectance, and one thermal infrared (TIR) band processed to orthorectified brightness temperature. The VNIR and SWIR bands have a resolution of 30m / pixel. The TIR band, while originally collected with a resolution of 120m / pixel (60m / pixel for Landsat 7) has been resampled using cubic convolution to 30m.

These data have been atmospherically corrected using [LEDAPS](#), and include a cloud, shadow, water and snow mask produced using [CFMASK](#), as well as a per-pixel saturation mask.

Strips of collected data are packaged into overlapping "scenes" covering approximately 170km x 183km using a [standardized reference grid](#).

See also [the USGS page on SR QA bands](#).

SR can only be produced for Landsat assets processed to the [L1TP level](#)

Data provider notes:

CLOSE

IMPORT

2. Rename the 3 different image collections with appropriate variables.

Step 1.4 Create a NDVI Function

The function you create in this step will be used to calculate NDVI when applied to different years of interest. It is important to note that the inputs in this function will change depending on the study year and study period, so the instructions below outline how to create the entire function, but the parameters of the function will have to be adjusted accordingly.

Here is a [Google Earth Engine tutorial](#) on filtering image collections and a [Google Earth Engine tutorial](#) on creating NDVI functions

1. Create a Landsat import function with a cloud mask
 - a. Create a name for your function, such as 'NDVI_Calculation' and set that equal to a function with a variable for year. Variable names such as "Year" or "year" are appropriate.
 - b. Import your desired Landsat image collection
 - i. Filter by date, with your "year" variable + ' - MONTH - DAY'
 1. For example, if you want to calculate the NDVI in May, put in (-year+ '05-01', year+ '05-31')
 - ii. Filter by bounds, with your Riley County Boundary Variable
2. Add NDVI to image collection to reduce by NDVI
3. Map the NDVI over the image collection you filtered above in step 1, and reduce by quality of NDVI band OR reduce by mean NDVI
 - a. To reduce by quality of NDVI Band, which will return the greenest pixel composite, or maximum NDVI, include .qualityMosaic('NDVI')
 - b. To reduce by the mean NDVI, replace .qualityMosaic('NDVI') with .mean()
 - c. Update your mask with the variable you created in
4. Return the final output that was created in step 3

This material is based upon work supported by NASA through contract NNL16AA05C. Any mention of a commercial product, service, or activity in this material does not constitute NASA endorsement. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Aeronautics and Space Administration and partner organizations.

Step 1.5 Apply the Function

Apply the NDVI function to the year of interest. This will include naming the new image variable you create, and setting that equal to the name of your function with your year variable of interest. Here is an example from the [Google Earth Engine Tutorial: Mapping over an ImageCollection](#):

```
var *your variable name here* = *your NDVI function*('2007');
```

Step 1.6 Recalibrate NDVI

The function written in Step 4 will create a new NDVI band to add to the image collection. This NDVI band will store values as floating points, which is different than an integer value that the rest of the bands are stored as. GEE will not export a singular image layer with multiple data types. That is why it is important to convert the other bands to match as floats

1. Convert all data in image to a floating point using the .toFloat() function ([Google Earth Engine, Array Transformations](#))
2. Rename the converted NDVI band to “NDVI”

Step 1.7 Export the NDVI Image with the Appropriate Scale and Region

Export your NDVI Tiff to the google drive associated with your google earth engine account. See the code below, taken from the [Export Images Google Earth Engine Tutorials](#) page, for how to specify the details of your export:

```
Export.image.toDrive({  
  image: *your image variable name here*,  
  description: *desired title for the exported Tiff*,  
  scale: 30, //Landsat imagery is taken at a 30m pixel resolution  
  region: *the Riley County Boundary Shapefile name*  
});
```

Proceed to Part 2: ArcGIS Pro Processing

Part 2: ArcGIS Pro Processing

Importing and Processing NDVI GeoTiffs in ArcGIS Pro to Create Curve Number Maps

Step 2.1 Import NDVI GeoTiffs into Arc Pro Document

1. Ensure that the GeoTiffs that were saved in your Google Drive from GEE are now accessible in a folder on the computer desktop

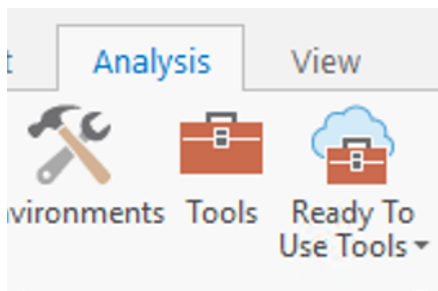


2. Using the “Add Data” button, import the NDVI layer or layers of interest

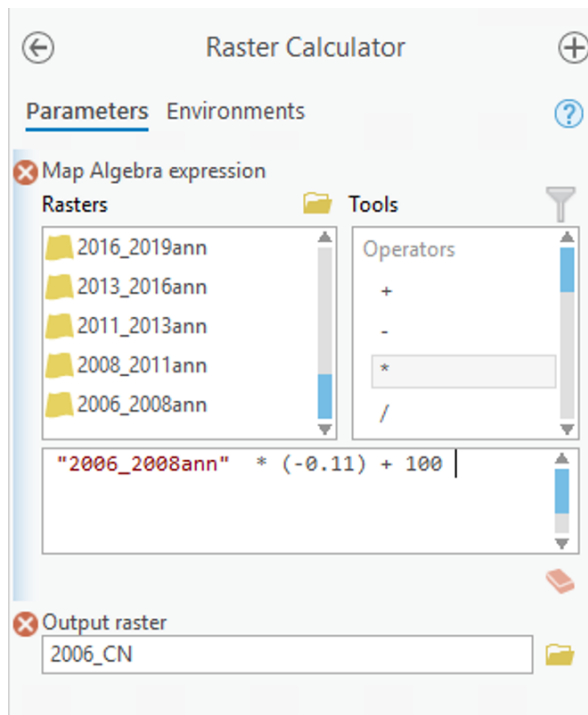
Step 2.2 Calculate Dynamic Curve Number by using Raster Calculator on GeoTiffs

The final curve number maps come from processing the NDVI layers with the phenology-based curve number equation outlined in Muche et. Al.

1. Open the Raster Calculator tool from Analysis Tools



2. Select the NDVI layer of interest
3. Apply the following equation: “NDVI GeoTiff Name” * (-0.11) + 100
4. Name the resulting CN output raster appropriately



5. Click “Run”

Congratulations! You should now have a Dynamic NDVI-incorporated CN map! Adjust the symbology according to style preference.