**PHD PROJECT**

**GOOGLE EARTH ENGINE CODES (original dummy)**

/\*

Using Spectral Indices

Computing Landsat8 spectral indices (NDVI, EVI, NDWI, MNDWI, MI) for Kenya from 2020-2021

NDVI = normalized difference vegetation index (monitor vegetation health and biomass density; sensitive to soil and atmosphere)

EVI = enhanced vegetation index (tracks vegetation greenness and reduces effects of soil and atmosphere; useful in high-biomass areas)

NDWI = normalized difference water index (differentiate water bodies from land)

MNDWI = modified normalized difference vegetation index (water body detection in complex scenarios)

MI = moisture index (soil and vegetation moisture content)

\*/

//Kenya = roi

//var roi = ee.FeatureCollection('USDOS/LSIB\_SIMPLE/2017').filter(ee.Filter.eq('country\_na','Kenya'));

//Cloud masking

//Apply cloud masking and scale factors

function cloudmask(image)

{

/\*

Bit 0 - Fill

Bit 1 - Dilated Cloud

Bit 2 - Cirrus

Bit 3 - Cloud

Bit 4 - Cloud Shadow

\*/

var qaMask = image.select('QA\_PIXEL').bitwiseAnd(parseInt('11111', 2)).eq(0);

var saturationMask = image.select('QA\_RADSAT').eq(0);

return image.updateMask(qaMask).updateMask(saturationMask);

}

//Apply scaling factors

function applyScaleFactors(image)

{

var opticalBands = image.select('SR\_B.').multiply(0.0000275).add(-0.2);

var thermalBands = image.select('ST\_B.\*').multiply(0.00341802).add(149);

return image.addBands(opticalBands, null, true).addBands(thermalBands, null, true);

}

//Filter to 2021 Landsat 8 image over

var dataset = ee.ImageCollection('LANDSAT/LC08/C02/T1\_L2')

.filter(ee.Filter.date('2020-01-01', '2022-01-01'))

.filter(ee.Filter.bounds(Kenya))

.map(applyScaleFactors)

.map(cloudmask);

//Create a median composite

var image = dataset.median();

print(image);

var rgbVis = {min:0, max:0.3, bands: ['SR\_B4', 'SR\_B3', 'SR\_B2']};

Map.addLayer(image.clip(Kenya), rgbVis, 'Landsat 8 Image');

Map.centerObject(Kenya, 5.5);

/\*

Create NDVI image to monitor vegetation health and biomass density

NDVI = ((NIR – Red) / (NIR + Red))

NIR = B5 in Landsat8 and B8 in Sentinel-2

Red = B4 in Landsat8 and B4 in Sentinel-2

\*/

var ndvi = image.normalizedDifference(['SR\_B5', 'SR\_B4']).rename(['ndvi']);

/\*

Create NDWI differentiate water bodies from land

NDWI = ((Green – NIR) / (Green + NIR))

Green = B3 in Landsat8 and B3 in Sentinel-2

NIR = B5 in Landsat8 and B8 in Sentinel-2

\*/

var ndwi = image.normalizedDifference(['SR\_B3', 'SR\_B5']).rename(['ndwi']);

/\*

Create MNDWI image for water body detection in complex scenarios

MNDWI = ((Green – SWIR) / (Green + SWIR))

Green = B3 in Landsat8 and B3 in Sentinel-2

SWIR = B6 in Landsat8 and B11 in Sentinel-2

\*/

var mndwi = image.normalizedDifference(['SR\_B3', 'SR\_B6']).rename(['mndwi']);

/\*

Create moisture index image for soil and vegetation moisture content

MI = ((NIR – SWIR) / (NIR + SWIR))

NIR = B5 in Landsat8 and B8 in Sentinel-2

SWIR = B6 in Landsat8 and B11 in Sentinel-2

\*/

var mi = image.normalizedDifference(['SR\_B5', 'SR\_B6']).rename(['mi']);

//Create EVI image for tracking vegetation greenness and reduces effects of soil and atmosphere

// EVI = 2.5 \* ((Band 5 - Band 4) / (Band 5 + 6 \* Band 4 - 7.5 \* Band 2 + 1))

var evi = image.expression( '2.5 \* ((NIR - RED) / (NIR + 6 \* RED - 7.5 \* BLUE + 1))',

{

'BLUE': image.select ('SR\_B2'),

'RED': image.select ('SR\_B4'),

'NIR': image.select ('SR\_B5'),

}).rename('evi');

//Visuaization parameters

var ndviVis = {min: 0, max: 0.8, palette: ['white', 'green']};

var mndwiVis = {min: 0, max: 0.8, palette: ['white', 'blue']};

//Add EVI, MNDWI and NDVI images to Map

Map.addLayer(ndvi.clip(Kenya), ndviVis, 'NDVI');

Map.addLayer(ndwi.clip(Kenya), mndwiVis, 'NDWI');

Map.addLayer(mndwi.clip(Kenya), mndwiVis, 'MNDWI');

Map.addLayer(mi.clip(Kenya), mndwiVis, 'MI');

Map.addLayer(evi.clip(Kenya), ndviVis, 'EVI');

**UPDATED CODE (10 April 2025)**

/\*

Using Spectral Indices

Computing Landsat8 spectral indices (NDVI, EVI, NDWI, MNDWI, MI) for Kenya from 2020-2021

NDVI = Normalized difference vegetation index (monitor vegetation health and biomass density; sensitive to soil and atmosphere)

EVI = Enhanced vegetation index (tracks vegetation greenness and reduces effects of soil and atmosphere; useful in high-biomass areas)

LAI = Leaf area index

SAVI = Soil adjusted vegetation index (monitor vegetation health and biomass density with adjustments that account for soil brightness in areas with sparse vegetation)

MSAVI = Modified soil adjustment vegetation index

FVC = Fractional Vegetation Cover

NDWI = Normalized difference water index (differentiate water bodies from land)

MNDWI = Modified normalized difference vegetation index (water body detection in complex scenarios)

MI = Moisture index (soil and vegetation moisture content)

\*/

//Kenya = roi

//var roi = ee.FeatureCollection('USDOS/LSIB\_SIMPLE/2017').filter(ee.Filter.eq('country\_na','Kenya'));

//Cloud masking

//Apply cloud masking and scale factors

function cloudmask(image) {

/\*

Bit 0 - Fill

Bit 1 - Dilated Cloud

Bit 2 - Cirrus

Bit 3 - Cloud

Bit 4 - Cloud Shadow

\*/

var qaMask = image.select('QA\_PIXEL').bitwiseAnd(parseInt('11111', 2)).eq(0);

var saturationMask = image.select('QA\_RADSAT').eq(0);

return image.updateMask(qaMask).updateMask(saturationMask);

}

//Apply scaling factors

function applyScaleFactors(image) {

var opticalBands = image.select('SR\_B.').multiply(0.0000275).add(-0.2);

var thermalBands = image.select('ST\_B.\*').multiply(0.00341802).add(149);

return image.addBands(opticalBands, null, true).addBands(thermalBands, null, true);

}

//Filter to 2021 Landsat 8 image over

var dataset = ee.ImageCollection('LANDSAT/LC08/C02/T1\_L2')

.filter(ee.Filter.date('2020-01-01', '2022-01-01'))

.filter(ee.Filter.bounds(Kenya))

.map(applyScaleFactors)

.map(cloudmask);

//Create a median composite and clip to roi/Kenya

var image = (dataset.median().clip(Kenya));

print(image);

var rgbVis = {min:0, max:0.3, bands: ['SR\_B4', 'SR\_B3', 'SR\_B2']};

Map.addLayer(image, rgbVis, 'Landsat 8 Image');

Map.centerObject(Kenya, 5.5);

/\*

Create NDVI image to monitor vegetation health and biomass density

NDVI = ((NIR – Red) / (NIR + Red))

NIR = B5 in Landsat8 and B8 in Sentinel-2

Red = B4 in Landsat8 and B4 in Sentinel-2

\*/

var ndvi = image.normalizedDifference(['SR\_B5', 'SR\_B4']).rename(['ndvi']);

/\*

Create NDWI differentiate water bodies from land

NDWI = ((Green – NIR) / (Green + NIR))

Green = B3 in Landsat8 and B3 in Sentinel-2

NIR = B5 in Landsat8 and B8 in Sentinel-2

\*/

var ndwi = image.normalizedDifference(['SR\_B3', 'SR\_B5']).rename(['ndwi']);

/\*

Create MNDWI image for water body detection in complex scenarios

MNDWI = ((Green – SWIR) / (Green + SWIR))

Green = B3 in Landsat8 and B3 in Sentinel-2

SWIR = B6 in Landsat8 and B11 in Sentinel-2

\*/

var mndwi = image.normalizedDifference(['SR\_B3', 'SR\_B6']).rename(['mndwi']);

/\*

Create moisture index image for soil and vegetation moisture content

MI = ((NIR – SWIR) / (NIR + SWIR))

NIR = B5 in Landsat8 and B8 in Sentinel-2

SWIR = B6 in Landsat8 and B11 in Sentinel-2

\*/

var mi = image.normalizedDifference(['SR\_B5', 'SR\_B6']).rename(['mi']);

/\*

EVI = 2.5 \* ((Band 5 - Band 4) / (Band 5 + 6 \* Band 4 - 7.5 \* Band 2 + 1))

var evi = image.expression(

'2.5 \* ((NIR - RED) / (NIR + 6 \* RED - 7.5 \* BLUE + 1))', {

'BLUE': image.select ('SR\_B2'),

'RED': image.select ('SR\_B4'),

'NIR': image.select ('SR\_B5'),

}).rename('evi');

\*/

/\*

LAI = (3.618\*EVI — 0.118)

var lai = image.expression(

'((3.618) \* (2.5 \* ((NIR - RED) / (NIR + 6 \* RED - 7.5 \* BLUE + 1))) - 0.118)', {

'BLUE': image.select('SR\_B2'),

'RED': image.select('SR\_B4'),

'NIR': image.select('SR\_B5'),

}).rename('evi');

\*/

/\*

SAVI = ((NIR-Red)/(NIR+Red+0.5))\*1.5

var savi = image.expression(

'((NIR-RED) / (NIR + RED + 0.5)) \* 1.5', {

'RED': image.select('SR\_B4'),

'NIR': image.select('SR\_B5'),

}).rename('savi');

\*/

/\*

MSAVI = (2NIR + 1−sqrt((2NIR + 1)2−8(NIR−RED)))/2

var msavi = image.expression(

'(2 \* NIR + 1 - sqrt(pow((2 \* NIR + 1), 2) - 8 \* (NIR - RED)) ) / 2', {

'RED': image.select('SR\_B4'),

'NIR': image.select('SR\_B5'),

}).rename('msavi');

\*/

/\*

FVC = (NDVI - NDVIs) / (NDVIv - NDVIs)

NDVIs: The NDVI value representing bare soil conditions (FVC = 0).

NDVIv: The NDVI value representing pure green vegetation conditions (FVC = 1)

var fvc = image.expression('(NDVI - NDVIs) / (NDVIv - NDVIs)',

{'NDVI': image.select('NDVI'), 'NDVIs': 0.1, 'NDVIv': 0.8 })\*/

Create FVC image

var fvc = ndvi

.expression(

'ndvi\_sq / (1 + ndvi\_sq)',

{

ndvi\_sq: ndvi.select('ndvi').pow(2)

}

)

.rename('fvc');

\*/

//Visuaization parameters

var ndviVis = {min: 0, max: 0.8, palette: ['white', 'green']};

var mndwiVis = {min: 0, max: 0.8, palette: ['white', 'blue']};

//Vegetation colour palette

//{min:-0.05, max:0.5, palette: ndvipalette}

//var ndvipalette = 'FFFFFF,CE7E45,DF923D,F1B555,FCD163,99B718,74A901,66A000,529400,3E8601,207401,056201,004C00,023B01,012E01,011D01,011301';

//Add EVI, MNDWI and NDVI images to Map

Map.addLayer(ndvi, ndviVis, 'NDVI');

Map.addLayer(ndwi, mndwiVis, 'NDWI');

Map.addLayer(mndwi, mndwiVis, 'MNDWI');

Map.addLayer(mi, mndwiVis, 'MI');

Map.addLayer(evi, ndviVis, 'EVI');

Map.addLayer(lai, ndviVis, 'LAI');

Map.addLayer(savi, ndviVis, 'SAVI');

Map.addLayer(msavi, ndviVis, 'MSAVI');

Map.addLayer(fvc, ndviVis, 'FVC');

for (var year = 2001; year <= 2020; year++)

for (var month = 1; month <= 12; month++)

fire(year, month)

function fire(year, month) {

var startDate = ee.Date.fromYMD(year, month, 1)

var endDate = startDate.advance(1, 'month')

...

}