Connecting Google Colab and Earth Engine for Satellite Imagery Analysis and Vegetation Monitoring

Dr Ana María Pacheco Pascagaza
University of Manchester

Towards the UK fire danger rating system (UKFDRS)

The project aims to establish and validate the scientific basis and essential components necessary for the development of a bespoke UK Fire Danger Rating System (FDRS).

This system will facilitate the evaluation of current and future fire regimes, enabling the assessment, management, and mitigation of wildfire impacts and the associated risk of uncontrollable wildfires throughout the UK.

MANCHESTER 1824
The University of Manchester

Agenda

- Introduction
- Google Earth Engine (GEE) – What it is? Benefits?
- Google Colab – What it is? Benefits?
- Connecting GEE API in Google Colab
- Importing and visualising datasets
- Data analysis applications
- Practical overview of the Colab notebook
- Limitations
- Questions

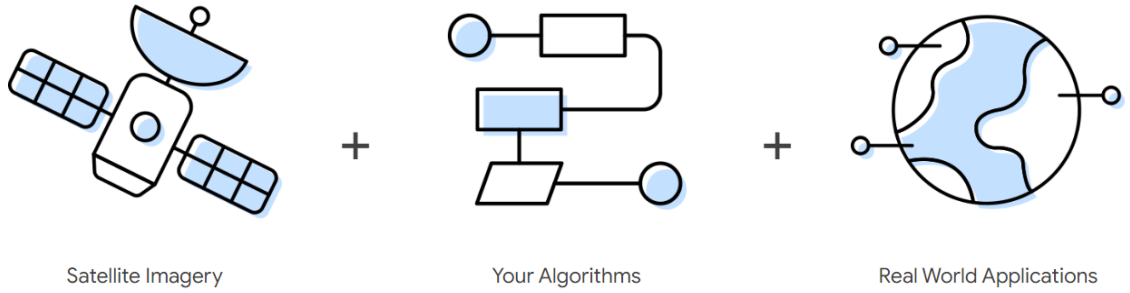
MANCHESTER 1824
The University of Manchester

Introduction

- The Earth is undergoing rapid changes, and environmental degradation is intensifying, significantly impacting vegetation patterns.
- Environmental monitoring, forest monitoring, land cover assessment, climate studies, fires and disaster management rely heavily on access to up-to-date satellite data.
- The ability to extract meaningful insights from these vast datasets is crucial for informed decision-making that help addressing environmental challenges.
- There is an urgent need for timely and efficient analysis of satellite imagery to monitor and understand these transformations.

Google Earth Engine

- A cloud-based geospatial platform developed by Google
- Allows their users to analyse and process large-scale Earth observation and geospatial data



Represents a valuable tool for researchers, scientists, environmentalists, and developers interested in connecting the power of geospatial data and analysis to address various Earth-related challenges and questions.

Source image: Google Earth Engine

MANCHESTER 1824
The University of Manchester

GEE Key Features and benefits

Data Archive

- Provides access to a massive archive of Earth observation and derived products .
- This data spans multiple decades and covers the entire globe in most of the cases.

Code editor

- You can write and run JavaScript or Python code within the Earth Engine Code Editor.

Geospatial Analysis

- Offers a wide range of geospatial analysis tools and functions on large datasets
- Examples: image processing, spatial analysis, time-series analysis, and machine learning.

Visualization

- You can visualise geospatial data and analyse results interactively on the map within the Code Editor, making it easier to explore and understand complex spatial patterns.

Collaboration

- Supports collaboration by enabling users to share their scripts, visualisations, and analysis results
- Collaborators can work on projects simultaneously and access shared data and code.

MANCHESTER 1824
The University of Manchester

Google Colab

- It is a free cloud-based platform provided by Google
- It allows you to write and execute Python code in a web-based interactive environment
- It is presented in a Jupyter notebook format

MANCHESTER 1824
The University of Manchester

Google Colab Key Features

Free access

- Provides access to access to Graphics Processing Units (GPUs) and Tensor Processing Units (TPUs) at no cost.

Google drive

- It is integrated with Google Drive, making easy to store and share notebooks

Libraries

- It comes with a wide range of pre-installed Python libraries
- Example: pandas, scikit-learn, TensorFlow, PyTorch etc.

Notebooks

- Notebooks are interactive documents that combine code execution cells with formatted text and visualisations.
- You can write and run Python code, add explanations, charts, and images, creating a narrative around their data analysis or machine learning projects.

Collaboration

- Supports collaboration by enabling users to share their scripts, visualisations, and analysis results with others.
- Collaborators can work on projects simultaneously and access shared data and code.
- You can integrate Colab with version control (e.g. GitHub) to track changes, and edit.

MANCHESTER 1824
The University of Manchester

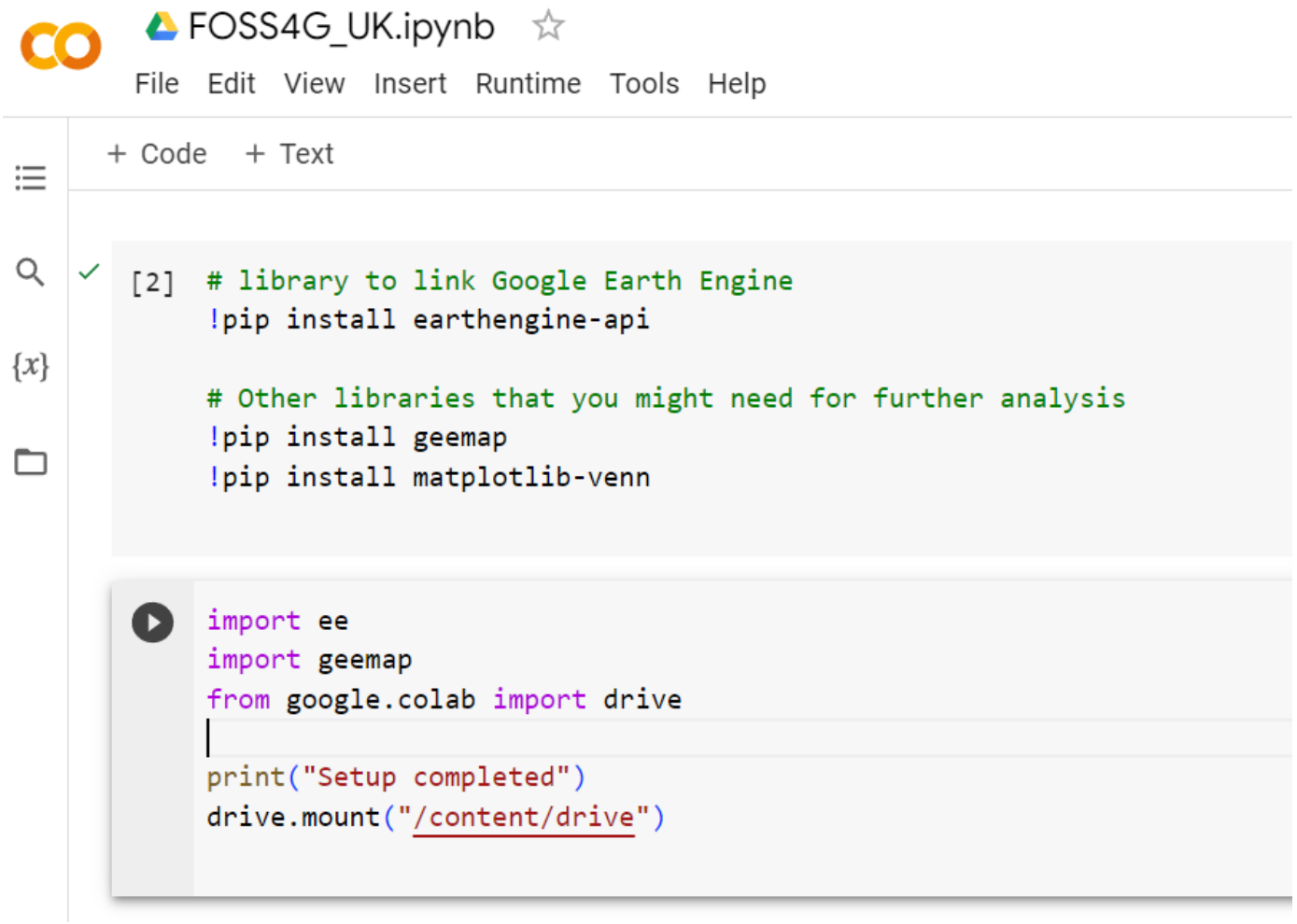
Let’s explore a Colab Jupyter notebook and how you can connect it with Google Earth Engine

MANCHESTER 1824
The University of Manchester

Setting up

STEP 1: Install your libraries

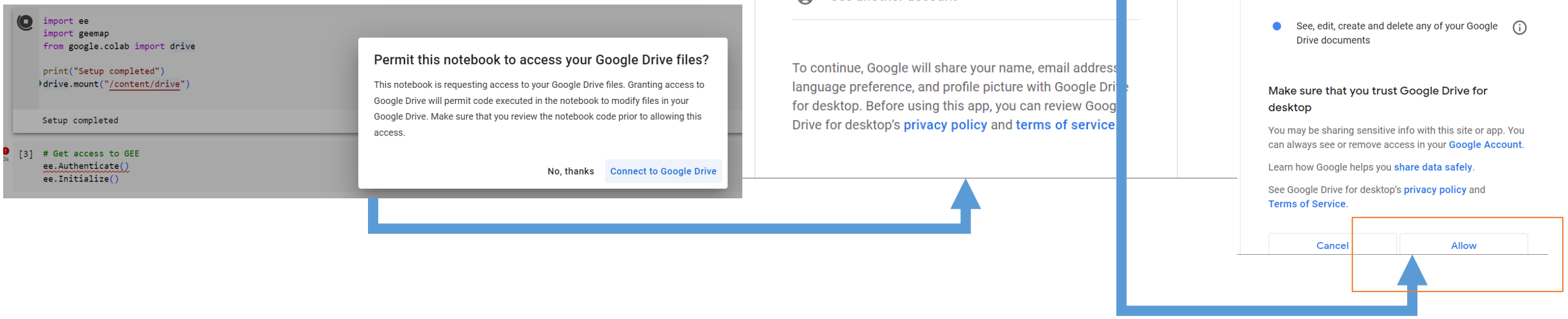
STEP 2: Import your libraries and link to your google drive



Link with Google drive

1. You need to access your google account

ee.Autenticate()
ee.Initialize()



MANCHESTER 1824
The University of Manchester

Autentication

```
# Get access to GEE
ee.Authenticate()
ee.Initialize()
```

1

```
To authorize access needed by Earth Engine, open the following URL in a web browser and follow the instructions. If the web browser does not start automatically, please manually browse the URL below.

    https://code.earthengine.google.com/client-auth?scopes=https%3A//www.googleapis.com/auth/earthengine%20https%3A//www.googleapis.com/auth/devstorage.full_control&request_id=QPqhgKGLauGHu6-MA8-v_bv1ULjAn

The authorization workflow will generate a code, which you should paste in the box below.
Enter verification code:
```

7

2

Google Earth Engine

Notebook Authenticator

Active account: **ukfdrs@gmail.com** SWITCH ACCOUNT

Cloud Project: **ee-ukfdrs** CHOOSE PROJECT

Data access: ☐ Use **read-only scopes**

WARNING: ONLY PROCEED IF YOU NEED TO ACCESS EARTH ENGINE FROM A NOTEBOOK

The token that you generate here will allow access to your Google account. Ensure that you understand the notebook that you are running.

Any code that you include in the notebook (and anyone with access to the notebook kernel) will be able to copy or change your data. Enable read-only scopes above to prevent data changes.

If you are not running a notebook, or you don't understand these warnings, then the link that sent you here may be trying to trick you. Do not proceed!

GENERATE TOKEN

3

Sign in with Google

Choose an account

to continue to

Earth Engine Notebook Client - ukfdrs@gmail.com

UK FDRS
ukfdrs@gmail.com

Use another account

4

Google

Google hasn't verified this app

You've been given access to an app that's currently being tested. You should only continue if you know the developer that invited you.

Continue

Back to safety

5

Earth Engine Notebook Client - ukfdrs@gmail.com wants access to your Google Account

ukfdrs@gmail.com

Select what

Earth Engine Notebook Client - ukfdrs@gmail.com

can access

Select all

View and manage your Google Earth Engine data. Learn more

Manage your data and permissions in Cloud Storage and see the email address for your Google Account. Learn more

Make sure that you trust Earth Engine Notebook Client - ukfdrs@gmail.com

You may be sharing sensitive info with this site or app. You can always see or remove access in your Google Account.

Learn how Google helps you share data safely.

See Earth Engine Notebook Client - ukfdrs@gmail.com's privacy policy and Terms of Service

Cancel

Continue

6

Make sure that you trust Earth Engine Notebook Client - ukfdrs@gmail.com

Earth Engine Notebook Client - ukfdrs@gmail.com is requesting access to your Google Account. To protect your account, don't continue unless you know and trust this app.

Sign in or provide access to Earth Engine Notebook Client - ukfdrs@gmail.com

To sign in or provide access:

1. Copy the authorisation code from the authorisation code section.
2. Navigate to Earth Engine Notebook Client - ukfdrs@gmail.com.
3. Paste the authorisation code on the Earth Engine Notebook Client - ukfdrs@gmail.com screen.

Authorisation code

Please copy this code, switch to your application and paste it there:

4/1Adeu5E PulDq2U3n 4ALNTW14-
PHslIJwoF HG5bIgBdY C8CYGx8aP6L

Don't sign in or provide access to Earth Engine Notebook Client - ukfdrs@gmail.com

If you don't want to continue, close this window.

Datasets available in GEE achive

Category	Datasets	Description
Optical Satellite Data	Landsat 8 Surface Reflectance, Sentinel-2 Surface Reflectance, MODIS Land Surface Temperature	Satellite imagery datasets providing reflectance, temperature, and other Earth observation data.
Radar Satellite Data	ALOS PALSAR, Sentinel-1	Radar satellite datasets offering various applications including terrain mapping and monitoring.
Topography	SRTM (Shuttle Radar Topography Mission)	Elevation and topography data derived from radar measurements.
Land Cover Data	Copernicus Global Land Cover Layers, MODIS Land Cover Type	Global land cover classification datasets at various resolutions and time periods.
Climate Datasets	ERA5 Climate Reanalysis, CHIRPS Precipitation Data	Climate datasets offering temperature, precipitation, and other meteorological variables.
Derived Maps	Global Forest Change, NASA Global Fire Maps, GEDI Biomass	Maps derived from satellite data, including forest change, fire occurrence, and biomass.

See more at: Earth Engine Data Catalog | Google for Developers

MANCHESTER 1824
The University of Manchester

Define parameters you want to consider

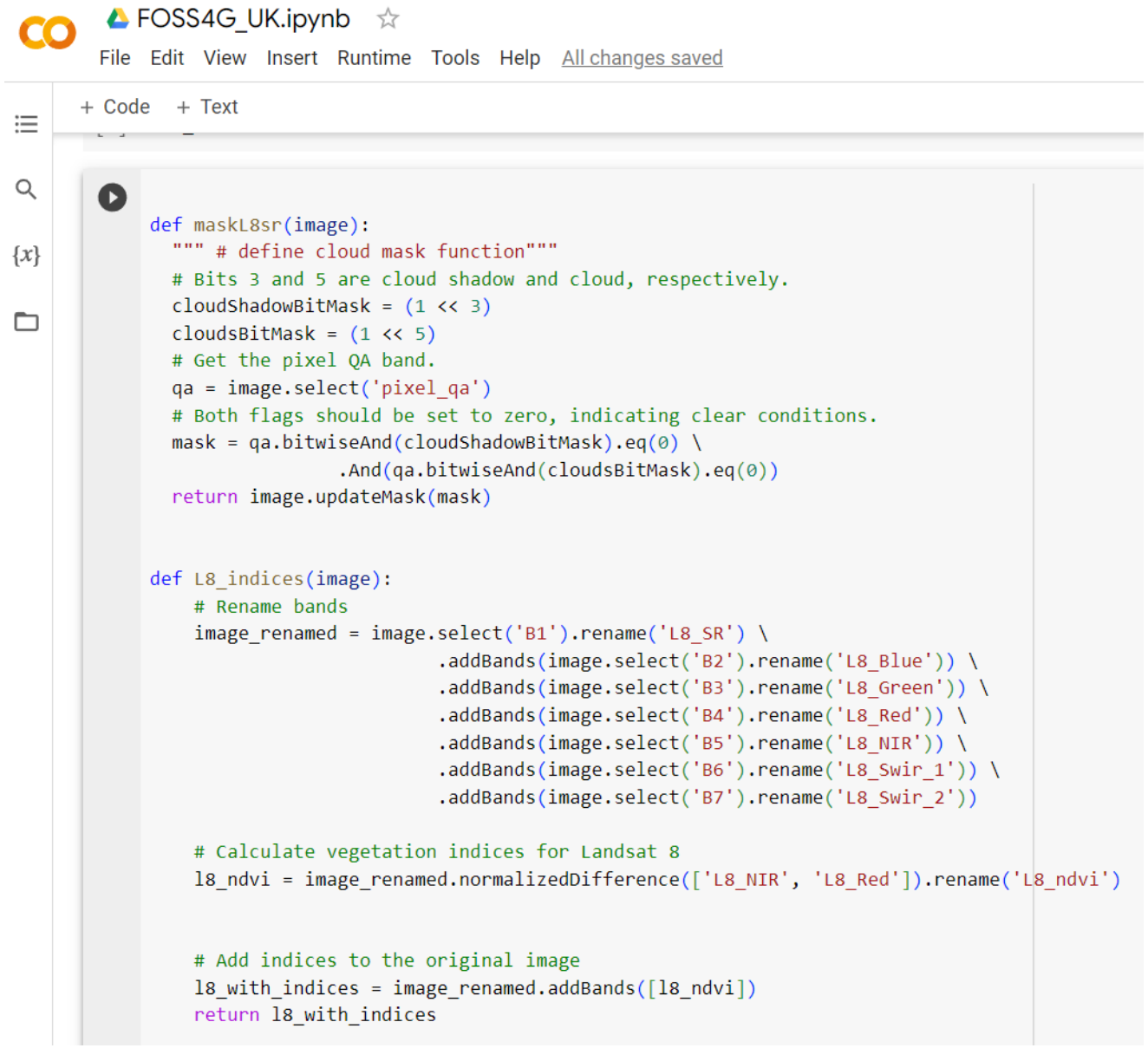
- Scale
- Area
- Time period

```
# Spatial resolution
scale = 30

# Define the coordinates of the polygon
coordinates = [[[-8.709412085888312, 60.8875076094291],
                [-8.709412085888312, 49.5],
                [1.8209834219241872, 49.5],
                [1.8209834219241872, 60.8875076094291]]]


# Study area
polygon = ee.Geometry.Polygon(coordinates)

# Time period
start_date = "2017-01-01"
end_date = "2022-12-31"
```

MANCHESTER 1824
The University of Manchester

Define functions that you want to apply to your imagery

- Cloud masking
- Vegetation indices

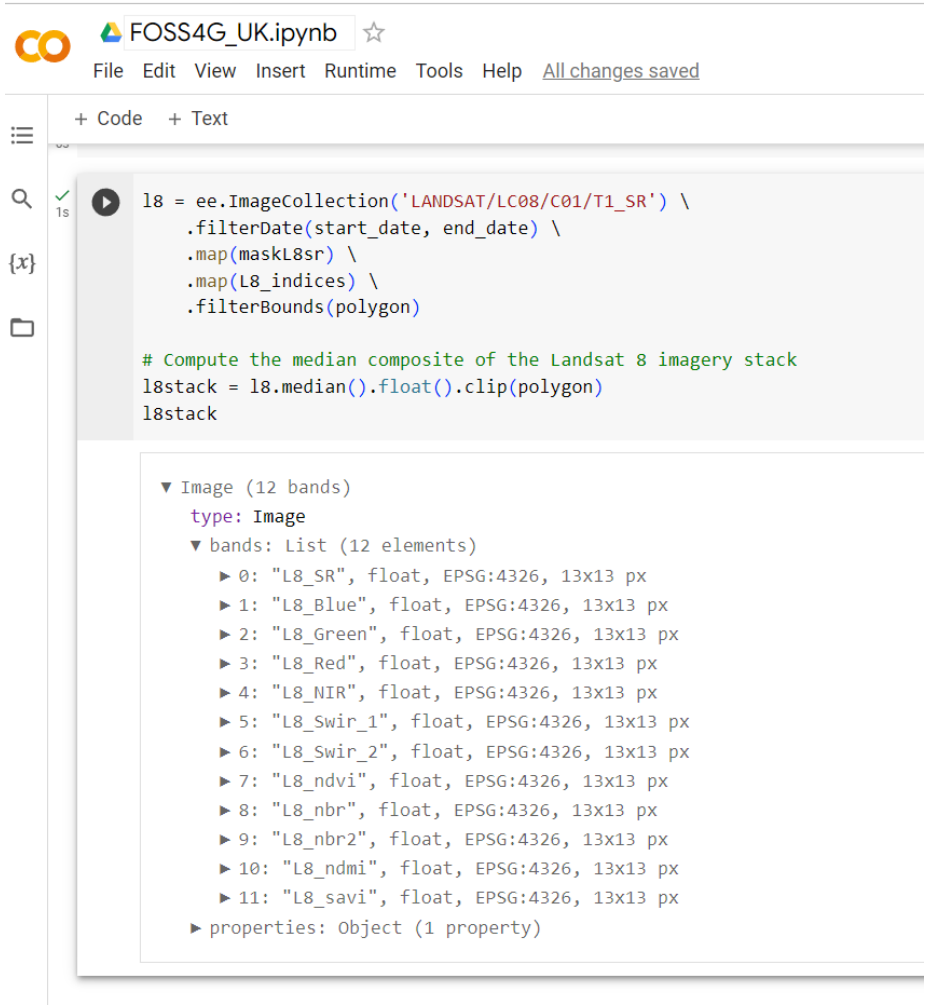



```
def maskL8sr(image):
  """ # define cloud mask function"""
  # Bits 3 and 5 are cloud shadow and cloud, respectively.
  cloudShadowBitMask = (1 << 3)
  cloudsBitMask = (1 << 5)
  # Get the pixel QA band.
  qa = image.select('pixel_qa')
  # Both flags should be set to zero, indicating clear conditions.
  mask = qa.bitwiseAnd(cloudShadowBitMask).eq(0) \
                 .And(qa.bitwiseAnd(cloudsBitMask).eq(0))
  return image.updateMask(mask)


def L8_indices(image):
    # Rename bands
    image_renamed = image.select('B1').rename('L8_SR') \
                         .addBands(image.select('B2').rename('L8_Blue')) \
                         .addBands(image.select('B3').rename('L8_Green')) \
                         .addBands(image.select('B4').rename('L8_Red')) \
                         .addBands(image.select('B5').rename('L8_NIR')) \
                         .addBands(image.select('B6').rename('L8_Swir_1')) \
                         .addBands(image.select('B7').rename('L8_Swir_2'))

    # Calculate vegetation indices for Landsat 8
    l8_ndvi = image_renamed.normalizedDifference(['L8_NIR', 'L8_Red']).rename('L8_ndvi')


    # Add indices to the original image
    l8_with_indices = image_renamed.addBands([l8_ndvi])
    return l8_with_indices
```

MANCHESTER 1824
The University of Manchester

Loading Satellite data

Define functions that you want to apply to your imagery



Visualisation of Satellite data

FOSS4G_UK.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

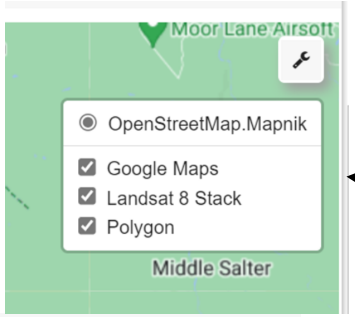
+ Code + Text

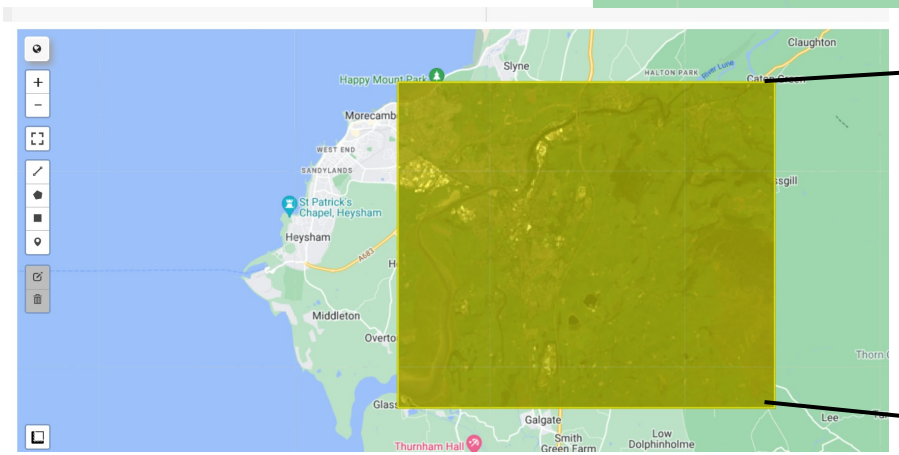
RAM Disk

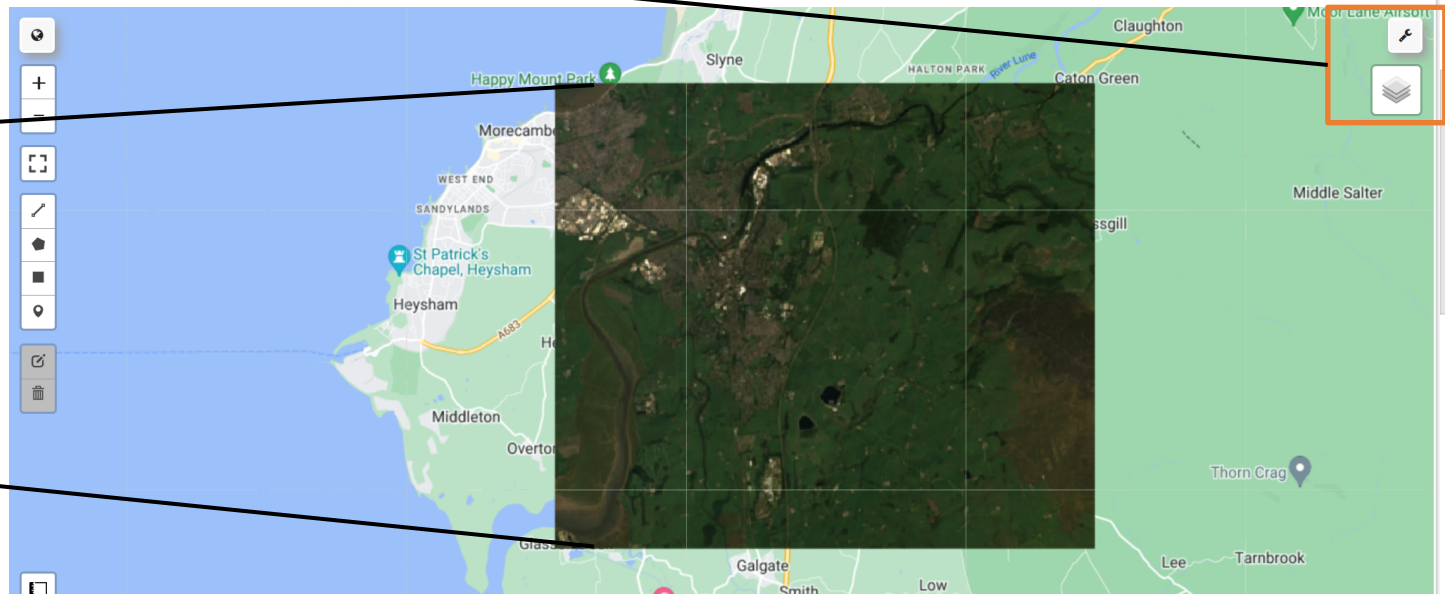
Image visualisation

```
[17] import folium
```

```
# Initialize a Map
Map = geemap.Map()
Map.centerObject(polygon, 12)  # You can adjust the zoom level
# Add the l8stack image to the map

opacity = 0
vis_params = {
    'min': 0,
    'max': 3000,
    'bands': ['L8_Red', 'L8_Green', 'L8_Blue'],  # Adjust bands based on your specific visualization needs
}
Map.addLayer(l8stack, vis_params, 'Landsat 8 Stack')

# Add the polygon layer
Map.addLayer(polygon, {'color': 'yellow'}, 'Polygon')

# Add a layer control widget
Map.addLayerControl()

# Display the map
Map
```

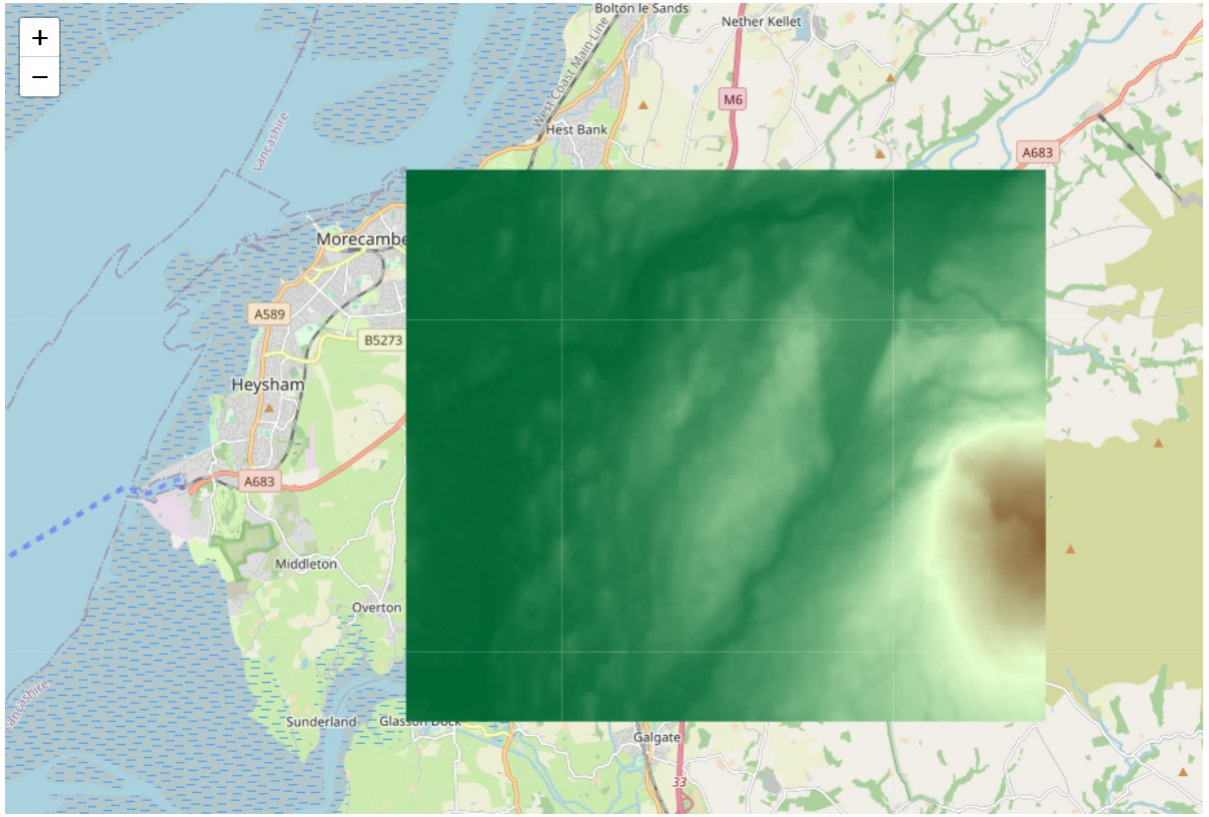






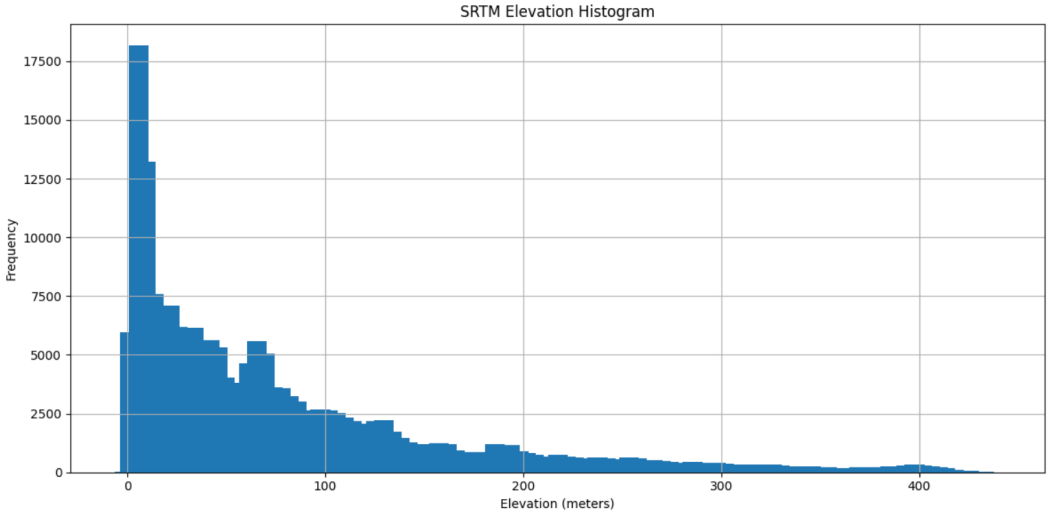
Visualisation of other available datasets and plotting histograms

```
display(m)
```



SRTM

```
[22] # Get the elevation band from the SRTM image
     elevation_band = srtm.select('elevation')

     # Create a histogram of elevation values
     histogram = elevation_band.reduceRegion(
         reducer=ee.Reducer.histogram(100),  # Number of bins for the histogram
         geometry=polygon,
         scale=30,  # Resolution in meters
         maxPixels=1e9
     )

     # Extract the histogram data as a list
     hist_data = histogram.get('elevation').getInfo()

     # Get the elevation values (X-axis)
     elevation_values = [float(x) for x in hist_data['bucketMeans']]

     # Get the frequency/count of elevation values (Y-axis)
     elevation_counts = [int(x) for x in hist_data['histogram']]

     # Plot the histogram
     plt.figure(figsize=(12, 6))
     plt.bar(elevation_values, elevation_counts, width=10)
     plt.title('SRTM Elevation Histogram')
     plt.xlabel('Elevation (meters)')
     plt.ylabel('Frequency')
     plt.grid(True)
     plt.tight_layout()
     plt.show()
```



MANCHESTER 1824
The University of Manchester

Applications

Application	Description	Google Earth Engine Functions
Temporal Analysis	Analysing changes over time in satellite data to detect trends and patterns.	ee.Reducer.mean(), ee.Reducer.median(), ee.Reducer.stdDev(), ee.ImageCollection.filterDate()
Land Cover Mapping	Classifying land use, tracking deforestation, monitoring urbanisation, and assessing vegetation changes.	ee.Image.clip(), ee.Image.normalizedDifference(), ee.Image.select(), ee.Image.expression()
Climate Studies	Studying climate change, weather patterns, and climate variables like temperature and precipitation.	ee.ImageCollection.filter() ee.ImageCollection.mean(), ee.ImageCollection.reduce()
Disaster Management	Responding to natural disasters by analysing pre- and post-disaster imagery to identify affected areas.	ee.ImageCollection.filterBounds(), ee.ImageCollection.mosaic(), ee.Image.reduceRegion()
Machine Learning	Leveraging machine learning algorithms like Random Forest, SVM, and others for classification and analysis.	ee.Classifier.smileRandomForest(), ee.Classifier.train(), ee.Classifier.predict(), ee.FeatureCollection()
Topography Analysis	Deriving elevation, slope, aspect, and other topographic information from elevation datasets.	ee.Image.gradient(), ee.Image.hillshade(), ee.Terrain.slope(), ee.Terrain.aspect()
Climate and Weather Data	Accessing climate data, weather data, and climate indices for research and analysis.	ee.ImageCollection("NASA/GLDAS/V021/NOAH/G025/T3H"), ee.ImageCollection("UCSB-CHG/CHIRPS/DAILY"), ee.ImageCollection("NASA/ORNL/DAYMET_V4")
Derived Maps	Generating thematic maps such as biomass, vegetation height, forest change, and fire risk.	Custom algorithms using ee.Image.expression(), ee.Image.updateMask(), and ee.Reducer functions.

MANCHESTER 1824
The University of Manchester

Limitations

	Google Earth Engine	Google Colab
Ease of Use	Requires proficiency in Earth Engine's JavaScript-based code.	Familiar Python interface for users.
Learning Curve	Steeper learning curve for newcomers.	More accessible for Python developers.
Access to Data	Extensive satellite and geospatial datasets available.	Limited to publicly available datasets.
Customisation	Highly customizable, but requires coding skills.	Customization through Python libraries.
Scalability	Excellent for handling large geospatial datasets.	Limited by Colab's CPU/GPU resources.
Interactive Analysis	Offers an interactive map-based interface.	Limited interactivity, primarily code.
Storage	Data is hosted on Google's servers, storage is not an issue.	Limited storage, temporary session data.
Cost	Free tier available, but costs can increase with extensive use.	Limited free GPU hours, additional costs for extended use.
Offline Use	Internet connection required to access datasets.	Internet connection required for usage.

MANCHESTER 1824
The University of Manchester

Let’s have a practical showcase of a Colab notebook to explore optical satellite data and derive a time series of the normalised vegetation index NDVI

FOSS4G_UK.ipynb - Colaboratory (google.com)

MANCHESTER 1824
The University of Manchester

Do you want to learn more?

- Explore the Geemap Github where you can find, videos, and tutorials that will helpful Tutorials – geemap
- Explore more about Google Colab at Welcome to Colaboratory - Colaboratory (google.com)

Any questions?

Thank you

Contact us:

Email: ana.pachecopascagaza@manchester.ac.uk
Website: https://ukfdrs.com/
Twitter: @ukfdrs
GitHub: https://github.com/UKFDRS