

GCC Satellite Indicators Pipeline for CO₂ Thesis

This notebook builds a **reproducible pipeline** to extract annual satellite-based indicators for the GCC countries, to be used later in national and Saudi-only CO₂ forecasting models.

Scope:

- Countries: Saudi Arabia, United Arab Emirates, Qatar, Kuwait, Oman, Bahrain
- Spatial unit: national boundaries (GAUL level 0, FAO GAUL 2015)
- Temporal unit: yearly aggregates
- Main satellite indicators:
 - Long-term NDVI (PKU GIMMS NDVI, AVHRR+MODIS consolidated, 1990–2022)
 - MODIS NDVI (2000–2023)
 - MODIS daytime land surface temperature (LST, 2000–2023)
 - Sentinel-5P NO₂ column (2018–2023)
 - Sentinel-5P CH₄ column (2018–2023)

Core output:

- `gulf_satellite_indicators_by_country.csv`
One row per (country, year) with annual means of all available indicators.

This notebook focuses **only** on satellite data acquisition and annual aggregation.

The merging with national CO₂ and socioeconomic data is done in a separate modeling notebook.

```
In [1]: import ee
import pandas as pd
import numpy as np

from datetime import datetime

print("Imports loaded.")
```

Imports loaded.

```
In [2]: # Configure your Earth Engine project ID here
EE_PROJECT = "co2-project-472909"

def initialize_ee():
    """Initialize the Google Earth Engine client."""
    try:
        ee.Initialize(project=EE_PROJECT)
        print("Google Earth Engine initialized with project:", EE_PROJECT)
    except Exception as e:
        print("Initialization without authentication failed:", e)
        print("Attempting interactive authentication...")
        ee.Authenticate()
        ee.Initialize(project=EE_PROJECT)
        print("Google Earth Engine initialized after authentication.")
```

```
initialize_ee()
```

Google Earth Engine initialized with project: co2-project-472909

```
In [3]: # GCC countries of interest
GCC_COUNTRIES = [
    "Saudi Arabia",
    "United Arab Emirates",
    "Qatar",
    "Kuwait",
    "Oman",
    "Bahrain",
]

# National boundaries (FAO GAUL Level 0)
WORLD_LAYER_ID = "FAO/GAUL/2015/level0"
world_fc = ee.FeatureCollection(WORLD_LAYER_ID)

print("Countries:", GCC_COUNTRIES)
print("World layer:", WORLD_LAYER_ID)
```

Countries: ['Saudi Arabia', 'United Arab Emirates', 'Qatar', 'Kuwait', 'Oman', 'Bahrain']
World layer: FAO/GAUL/2015/level0

```
In [4]: # Temporal coverage per sensor (year when reasonably available)
GLOBAL_START_YEAR      = 1990      # overall start for the loop
MODIS_START_YEAR       = 2000
S5P_START_YEAR         = 2018

# Long-term NDVI from GIMMS NDVI3g (NASA/GIMMS/3GV0)
# Dataset availability: 1981-07-01 to 2013-12-16
# https://developers.google.com/earth-engine/datasets/catalog/NASA_GIMMS_3GV0
LONG_NDVI_START_YEAR   = 1990      # we use 1990 as start
LONG_NDVI_END_YEAR     = 2013      # dataset ends in 2013

END_YEAR               = 2023      # overall analysis end year

# Long-term NDVI (GIMMS NDVI3g)
GIMMS_NDVI_IC = ee.ImageCollection("NASA/GIMMS/3GV0")
LONG_NDVI_IC = GIMMS_NDVI_IC
LONG_NDVI_BAND = "ndvi"           # band name from GIMMS dataset
LONG_NDVI_SCALE = 9277            # meters (pixel size in GEE docs)

# MODIS: NDVI
MODIS_NDVI_IC = ee.ImageCollection("MODIS/061/MOD13A2") # 16-day, 1km
MODIS_NDVI_BAND = "NDVI"
MODIS_NDVI_SCALE = 500            # meters
MODIS_NDVI_FACTOR = 0.0001       # scale factor for MOD13A2 NDVI

# MODIS: Land Surface Temperature (Daytime)
MODIS_LST_IC = ee.ImageCollection("MODIS/061/MOD11A2") # 8-day, 1km
MODIS_LST_BAND = "LST_Day_1km"
MODIS_LST_SCALE = 1000            # meters
MODIS_LST_FACTOR = 0.02          # to convert to Kelvin

# Sentinel-5P: NO2
S5P_NO2_IC = ee.ImageCollection("COPERNICUS/S5P/OFFL/L3_NO2")
S5P_NO2_BAND = "NO2_column_number_density"
S5P_NO2_SCALE = 10000            # meters

# Sentinel-5P: CH4
```

```

SSP_CH4_IC = ee.ImageCollection("COPERNICUS/S5P/OFFL/L3_CH4")
SSP_CH4_BAND = "CH4_column_volume_mixing_ratio_dry_air"
SSP_CH4_SCALE = 10000 # meters

print("Temporal coverage (intended):")
print(f" Long-term NDVI (GIMMS NDVI3g): {LONG_NDVI_START_YEAR}-{LONG_NDVI_END_YEAR}")
print(f" MODIS (NDVI, LST): {MODIS_START_YEAR}-{END_YEAR}")
print(f" Sentinel-5P (NO2, CH4): {S5P_START_YEAR}-{END_YEAR}")

```

Temporal coverage (intended):

- Long-term NDVI (GIMMS NDVI3g): 1990-2013
- MODIS (NDVI, LST): 2000-2023
- Sentinel-5P (NO2, CH4): 2018-2023

```

In [5]: def get_country_geometry(country_name: str) -> ee.Geometry:
        """Return the national boundary geometry for a given country."""
        fc = world_fc.filter(ee.Filter.eq("ADM0_NAME", country_name))
        count = fc.size().getInfo()
        if count == 0:
            raise ValueError(f"No GAUL feature found for country: {country_name}")
        return fc.geometry()

```

```

In [6]: def annual_mean_over_geometry(
        image_collection: ee.ImageCollection,
        band: str,
        geom: ee.Geometry,
        year: int,
        scale: int,
        factor: float = 1.0,
    ) -> float:
    """
    Compute the annual mean of a given band over a geometry for a specific year.
    Returns a Python float or None if the result is missing.
    """

    start = ee.Date.fromYMD(year, 1, 1)
    end = start.advance(1, "year")

    img = (
        image_collection
        .filterDate(start, end)
        .select(band)
        .mean()
    )

    if factor != 1.0:
        img = img.multiply(factor)

    reduced = img.reduceRegion(
        reducer=ee.Reducer.mean(),
        geometry=geom,
        scale=scale,
        maxPixels=1e13,
    )

    value = reduced.get(band)
    try:
        value = value.getInfo()
    except Exception:
        value = None

    return value

```

In [7]: rows = []

```
print("Starting satellite indicators extraction...")
print(f"Time range: {GLOBAL_START_YEAR}-{END_YEAR}")
start_time = datetime.now()

for cname in GCC_COUNTRIES:
    print(f"\n=== Country: {cname} ===")
    try:
        geom = get_country_geometry(cname)
    except Exception as e:
        print(f" ERROR: cannot get geometry: {e}")
        continue

    for year in range(GLOBAL_START_YEAR, END_YEAR + 1):
        # Initialize values as None
        ndvi_long_mean = None          # PKU GIMMS NDVI (1990-2022)
        ndvi_modis_mean = None         # MODIS NDVI (2000+)
        lst_mean_k = None
        no2_mean = None
        ch4_mean = None

        try:
            # Long-term NDVI (PKU GIMMS) for 1990-2022
            if LONG_NDVI_START_YEAR <= year <= LONG_NDVI_END_YEAR:
                ndvi_long_mean = annual_mean_over_geometry(
                    LONG_NDVI_IC,
                    LONG_NDVI_BAND,
                    geom,
                    year,
                    int(LONG_NDVI_SCALE),
                    factor=1.0,
                )

            # MODIS NDVI: available from MODIS_START_YEAR onward
            if year >= MODIS_START_YEAR:
                ndvi_modis_mean = annual_mean_over_geometry(
                    MODIS_NDVI_IC,
                    MODIS_NDVI_BAND,
                    geom,
                    year,
                    MODIS_NDVI_SCALE,
                    factor=MODIS_NDVI_FACTOR,
                )

            # MODIS LST (Daytime) in Kelvin: available from MODIS_START_YEAR onward
            if year >= MODIS_START_YEAR:
                lst_mean_k = annual_mean_over_geometry(
                    MODIS_LST_IC,
                    MODIS_LST_BAND,
                    geom,
                    year,
                    MODIS_LST_SCALE,
                    factor=MODIS_LST_FACTOR,
                )

            # Sentinel-5P NO2: only from S5P_START_YEAR
            if year >= S5P_START_YEAR:
                no2_mean = annual_mean_over_geometry(
                    S5P_NO2_IC,
                    S5P_NO2_BAND,
```

```

        geom,
        year,
        S5P_NO2_SCALE,
        factor=1.0,
    )

# Sentinel-5P CH4: only from S5P_START_YEAR
if year >= S5P_START_YEAR:
    ch4_mean = annual_mean_over_geometry(
        S5P_CH4_IC,
        S5P_CH4_BAND,
        geom,
        year,
        S5P_CH4_SCALE,
        factor=1.0,
    )

row = {
    "country": cname,
    "year": year,
    "ndvi_long_mean": ndvi_long_mean,    # PKU GIMMS NDVI (1990-2022)
    "ndvi_modis_mean": ndvi_modis_mean,  # MODIS NDVI (2000+)
    "lst_day_k_mean": lst_mean_k,
    "no2_mean": no2_mean,
    "ch4_mean": ch4_mean,
}
rows.append(row)

# Verbose printing for readability (rounded values)
def fmt(v):
    return "NaN" if v is None else f"{v:.4e}"

print(
    f" Year {year}: "
    f"ndvi_long={fmt(ndvi_long_mean)}, "
    f"ndvi_modis={fmt(ndvi_modis_mean)}, "
    f"lst_K={fmt(lst_mean_k)}, "
    f"no2={fmt(no2_mean)}, "
    f"ch4={fmt(ch4_mean)}"
)

except Exception as e:
    print(f" ERROR during extraction for year {year}: {e}")
    rows.append(
        {
            "country": cname,
            "year": year,
            "ndvi_long_mean": None,
            "ndvi_modis_mean": None,
            "lst_day_k_mean": None,
            "no2_mean": None,
            "ch4_mean": None,
        }
    )

end_time = datetime.now()
print("\nExtraction finished.")
print("Elapsed time:", end_time - start_time)

```

Starting satellite indicators extraction...
Time range: 1990-2023

=== Country: Saudi Arabia ===

Year 1990: ndvi_long=1.0330e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1991: ndvi_long=9.8457e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1992: ndvi_long=9.8850e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1993: ndvi_long=1.0148e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1994: ndvi_long=9.8287e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1995: ndvi_long=1.0364e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1996: ndvi_long=1.0529e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1997: ndvi_long=1.0455e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1998: ndvi_long=1.1211e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1999: ndvi_long=1.0260e-01, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 2000: ndvi_long=9.8366e-02, ndvi_modis=9.7812e-02, lst_K=3.1628e+02, no2=NaN, ch4=NaN
Year 2001: ndvi_long=1.0560e-01, ndvi_modis=1.0399e-01, lst_K=3.1405e+02, no2=NaN, ch4=NaN
Year 2002: ndvi_long=1.0084e-01, ndvi_modis=1.0126e-01, lst_K=3.1363e+02, no2=NaN, ch4=NaN
Year 2003: ndvi_long=9.4512e-02, ndvi_modis=9.8661e-02, lst_K=3.1314e+02, no2=NaN, ch4=NaN
Year 2004: ndvi_long=9.2731e-02, ndvi_modis=1.0172e-01, lst_K=3.1320e+02, no2=NaN, ch4=NaN
Year 2005: ndvi_long=8.9654e-02, ndvi_modis=1.0073e-01, lst_K=3.1299e+02, no2=NaN, ch4=NaN
Year 2006: ndvi_long=8.9021e-02, ndvi_modis=9.9655e-02, lst_K=3.1275e+02, no2=NaN, ch4=NaN
Year 2007: ndvi_long=8.9148e-02, ndvi_modis=9.9332e-02, lst_K=3.1287e+02, no2=NaN, ch4=NaN
Year 2008: ndvi_long=8.9747e-02, ndvi_modis=9.6825e-02, lst_K=3.1254e+02, no2=NaN, ch4=NaN
Year 2009: ndvi_long=9.6290e-02, ndvi_modis=9.7693e-02, lst_K=3.1260e+02, no2=NaN, ch4=NaN
Year 2010: ndvi_long=9.7535e-02, ndvi_modis=9.9714e-02, lst_K=3.1355e+02, no2=NaN, ch4=NaN
Year 2011: ndvi_long=9.7878e-02, ndvi_modis=9.9363e-02, lst_K=3.1243e+02, no2=NaN, ch4=NaN
Year 2012: ndvi_long=1.0369e-01, ndvi_modis=9.7741e-02, lst_K=3.1263e+02, no2=NaN, ch4=NaN
Year 2013: ndvi_long=1.0407e-01, ndvi_modis=9.9655e-02, lst_K=3.1255e+02, no2=NaN, ch4=NaN
Year 2014: ndvi_long=NaN, ndvi_modis=1.0076e-01, lst_K=3.1346e+02, no2=NaN, ch4=NaN
Year 2015: ndvi_long=NaN, ndvi_modis=9.8456e-02, lst_K=3.1311e+02, no2=NaN, ch4=NaN
Year 2016: ndvi_long=NaN, ndvi_modis=1.0133e-01, lst_K=3.1375e+02, no2=NaN, ch4=NaN
Year 2017: ndvi_long=NaN, ndvi_modis=9.9416e-02, lst_K=3.1355e+02, no2=NaN, ch4=NaN
Year 2018: ndvi_long=NaN, ndvi_modis=9.8594e-02, lst_K=3.1308e+02, no2=5.9540e-05, ch4=1.8877e+03
Year 2019: ndvi_long=NaN, ndvi_modis=1.0541e-01, lst_K=3.1359e+02, no2=5.9507e-05, ch4=1.8940e+03
Year 2020: ndvi_long=NaN, ndvi_modis=1.0167e-01, lst_K=3.1387e+02, no2=5.7710e-05, ch4=1.9059e+03
Year 2021: ndvi_long=NaN, ndvi_modis=1.0223e-01, lst_K=3.1428e+02, no2=6.1491e-05, ch4=1.9184e+03
Year 2022: ndvi_long=NaN, ndvi_modis=9.9439e-02, lst_K=3.1263e+02, no2=5.9395e-05, ch4=1.9176e+03
Year 2023: ndvi_long=NaN, ndvi_modis=1.0902e-01, lst_K=3.1240e+02, no2=6.4158e-05, ch4=1.9199e+03

=== Country: United Arab Emirates ===

Year 1990: ndvi_long=9.5781e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1991: ndvi_long=8.8791e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1992: ndvi_long=9.1064e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1993: ndvi_long=9.5461e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1994: ndvi_long=9.3829e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1995: ndvi_long=9.5986e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1996: ndvi_long=9.7187e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1997: ndvi_long=9.6146e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1998: ndvi_long=9.7765e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1999: ndvi_long=9.4495e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 2000: ndvi_long=9.2479e-02, ndvi_modis=9.9950e-02, lst_K=3.1934e+02, no2=NaN, ch4=NaN
Year 2001: ndvi_long=9.8140e-02, ndvi_modis=1.0142e-01, lst_K=3.1660e+02, no2=NaN, ch4=NaN
Year 2002: ndvi_long=9.5636e-02, ndvi_modis=1.0296e-01, lst_K=3.1673e+02, no2=NaN, ch4=NaN
Year 2003: ndvi_long=8.8411e-02, ndvi_modis=1.0228e-01, lst_K=3.1531e+02, no2=NaN, ch4=NaN
Year 2004: ndvi_long=7.8825e-02, ndvi_modis=1.0277e-01, lst_K=3.1605e+02, no2=NaN, ch4=NaN
Year 2005: ndvi_long=7.6235e-02, ndvi_modis=1.0339e-01, lst_K=3.1573e+02, no2=NaN, ch4=NaN

Year 2006: ndvi_long=7.7627e-02, ndvi_modis=1.0315e-01, lst_K=3.1539e+02, no2=NaN, ch4=NaN
Year 2007: ndvi_long=7.9000e-02, ndvi_modis=1.0494e-01, lst_K=3.1560e+02, no2=NaN, ch4=NaN
Year 2008: ndvi_long=7.9654e-02, ndvi_modis=1.0316e-01, lst_K=3.1494e+02, no2=NaN, ch4=NaN
Year 2009: ndvi_long=8.7537e-02, ndvi_modis=1.0281e-01, lst_K=3.1497e+02, no2=NaN, ch4=NaN
Year 2010: ndvi_long=8.8362e-02, ndvi_modis=1.0589e-01, lst_K=3.1625e+02, no2=NaN, ch4=NaN
Year 2011: ndvi_long=8.7883e-02, ndvi_modis=1.0362e-01, lst_K=3.1538e+02, no2=NaN, ch4=NaN
Year 2012: ndvi_long=9.7139e-02, ndvi_modis=1.0206e-01, lst_K=3.1552e+02, no2=NaN, ch4=NaN
Year 2013: ndvi_long=9.5377e-02, ndvi_modis=1.0290e-01, lst_K=3.1546e+02, no2=NaN, ch4=NaN
Year 2014: ndvi_long=NaN, ndvi_modis=1.0310e-01, lst_K=3.1616e+02, no2=NaN, ch4=NaN
Year 2015: ndvi_long=NaN, ndvi_modis=1.0231e-01, lst_K=3.1638e+02, no2=NaN, ch4=NaN
Year 2016: ndvi_long=NaN, ndvi_modis=1.0255e-01, lst_K=3.1623e+02, no2=NaN, ch4=NaN
Year 2017: ndvi_long=NaN, ndvi_modis=1.0331e-01, lst_K=3.1690e+02, no2=NaN, ch4=NaN
Year 2018: ndvi_long=NaN, ndvi_modis=1.0097e-01, lst_K=3.1654e+02, no2=7.7884e-05, ch4=1.8977e+03
Year 2019: ndvi_long=NaN, ndvi_modis=1.0228e-01, lst_K=3.1601e+02, no2=7.8258e-05, ch4=1.9142e+03
Year 2020: ndvi_long=NaN, ndvi_modis=1.0531e-01, lst_K=3.1657e+02, no2=7.4916e-05, ch4=1.9232e+03
Year 2021: ndvi_long=NaN, ndvi_modis=1.0346e-01, lst_K=3.1720e+02, no2=8.1394e-05, ch4=1.9421e+03
Year 2022: ndvi_long=NaN, ndvi_modis=1.0345e-01, lst_K=3.1561e+02, no2=7.6671e-05, ch4=1.9306e+03
Year 2023: ndvi_long=NaN, ndvi_modis=1.0545e-01, lst_K=3.1523e+02, no2=8.1404e-05, ch4=1.9292e+03

=== Country: Qatar ===

Year 1990: ndvi_long=6.8131e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1991: ndvi_long=6.4200e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1992: ndvi_long=6.3493e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1993: ndvi_long=7.1483e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1994: ndvi_long=6.2511e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1995: ndvi_long=6.4567e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1996: ndvi_long=8.6406e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1997: ndvi_long=6.9536e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1998: ndvi_long=7.6425e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1999: ndvi_long=6.6459e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 2000: ndvi_long=6.1520e-02, ndvi_modis=7.4945e-02, lst_K=3.1443e+02, no2=NaN, ch4=NaN
Year 2001: ndvi_long=6.8911e-02, ndvi_modis=7.8273e-02, lst_K=3.1262e+02, no2=NaN, ch4=NaN
Year 2002: ndvi_long=6.3712e-02, ndvi_modis=7.5882e-02, lst_K=3.1261e+02, no2=NaN, ch4=NaN
Year 2003: ndvi_long=5.8854e-02, ndvi_modis=7.6057e-02, lst_K=3.1089e+02, no2=NaN, ch4=NaN
Year 2004: ndvi_long=3.6886e-02, ndvi_modis=7.7299e-02, lst_K=3.1138e+02, no2=NaN, ch4=NaN
Year 2005: ndvi_long=3.7505e-02, ndvi_modis=7.8126e-02, lst_K=3.1081e+02, no2=NaN, ch4=NaN
Year 2006: ndvi_long=3.9652e-02, ndvi_modis=7.7384e-02, lst_K=3.1098e+02, no2=NaN, ch4=NaN
Year 2007: ndvi_long=3.9660e-02, ndvi_modis=8.5816e-02, lst_K=3.1118e+02, no2=NaN, ch4=NaN
Year 2008: ndvi_long=4.1102e-02, ndvi_modis=7.6010e-02, lst_K=3.1075e+02, no2=NaN, ch4=NaN
Year 2009: ndvi_long=4.7767e-02, ndvi_modis=7.5072e-02, lst_K=3.1015e+02, no2=NaN, ch4=NaN
Year 2010: ndvi_long=4.6671e-02, ndvi_modis=7.9779e-02, lst_K=3.1181e+02, no2=NaN, ch4=NaN
Year 2011: ndvi_long=5.5629e-02, ndvi_modis=7.8239e-02, lst_K=3.1042e+02, no2=NaN, ch4=NaN
Year 2012: ndvi_long=6.3005e-02, ndvi_modis=7.7447e-02, lst_K=3.1109e+02, no2=NaN, ch4=NaN
Year 2013: ndvi_long=6.1512e-02, ndvi_modis=7.6845e-02, lst_K=3.1079e+02, no2=NaN, ch4=NaN
Year 2014: ndvi_long=NaN, ndvi_modis=8.0333e-02, lst_K=3.1182e+02, no2=NaN, ch4=NaN
Year 2015: ndvi_long=NaN, ndvi_modis=7.6260e-02, lst_K=3.1165e+02, no2=NaN, ch4=NaN
Year 2016: ndvi_long=NaN, ndvi_modis=8.0264e-02, lst_K=3.1181e+02, no2=NaN, ch4=NaN
Year 2017: ndvi_long=NaN, ndvi_modis=8.4110e-02, lst_K=3.1207e+02, no2=NaN, ch4=NaN
Year 2018: ndvi_long=NaN, ndvi_modis=7.7081e-02, lst_K=3.1216e+02, no2=9.9233e-05, ch4=1.9087e+03
Year 2019: ndvi_long=NaN, ndvi_modis=8.0815e-02, lst_K=3.1225e+02, no2=9.7117e-05, ch4=1.9250e+03
Year 2020: ndvi_long=NaN, ndvi_modis=8.1622e-02, lst_K=3.1237e+02, no2=9.6152e-05, ch4=1.9355e+03
Year 2021: ndvi_long=NaN, ndvi_modis=7.7446e-02, lst_K=3.1333e+02, no2=1.0067e-04, ch4=1.9544e+03

Year 2022: ndvi_long=NaN, ndvi_modis=7.7608e-02, lst_K=3.1126e+02, no2=9.9393e-05, ch4=1.9363e+03
Year 2023: ndvi_long=NaN, ndvi_modis=8.7153e-02, lst_K=3.1121e+02, no2=9.7870e-05, ch4=1.9308e+03

=== Country: Kuwait ===

Year 1990: ndvi_long=8.0747e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1991: ndvi_long=7.1514e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1992: ndvi_long=7.5447e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1993: ndvi_long=8.9106e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1994: ndvi_long=7.6980e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1995: ndvi_long=9.1072e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1996: ndvi_long=8.9383e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1997: ndvi_long=7.8597e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1998: ndvi_long=9.4654e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1999: ndvi_long=8.5770e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 2000: ndvi_long=8.2517e-02, ndvi_modis=8.4439e-02, lst_K=3.1463e+02, no2=NaN, ch4=NaN
Year 2001: ndvi_long=8.4369e-02, ndvi_modis=8.7605e-02, lst_K=3.1208e+02, no2=NaN, ch4=NaN
Year 2002: ndvi_long=8.1066e-02, ndvi_modis=8.9570e-02, lst_K=3.1209e+02, no2=NaN, ch4=NaN
Year 2003: ndvi_long=7.5838e-02, ndvi_modis=8.5518e-02, lst_K=3.1135e+02, no2=NaN, ch4=NaN
Year 2004: ndvi_long=5.9779e-02, ndvi_modis=8.7907e-02, lst_K=3.1172e+02, no2=NaN, ch4=NaN
Year 2005: ndvi_long=5.8421e-02, ndvi_modis=8.7046e-02, lst_K=3.1098e+02, no2=NaN, ch4=NaN
Year 2006: ndvi_long=5.9777e-02, ndvi_modis=9.1934e-02, lst_K=3.1161e+02, no2=NaN, ch4=NaN
Year 2007: ndvi_long=5.7155e-02, ndvi_modis=8.6422e-02, lst_K=3.1086e+02, no2=NaN, ch4=NaN
Year 2008: ndvi_long=5.5183e-02, ndvi_modis=7.9957e-02, lst_K=3.1015e+02, no2=NaN, ch4=NaN
Year 2009: ndvi_long=6.4866e-02, ndvi_modis=8.0275e-02, lst_K=3.1028e+02, no2=NaN, ch4=NaN
Year 2010: ndvi_long=7.0189e-02, ndvi_modis=8.5668e-02, lst_K=3.1254e+02, no2=NaN, ch4=NaN
Year 2011: ndvi_long=7.3504e-02, ndvi_modis=8.2894e-02, lst_K=3.1056e+02, no2=NaN, ch4=NaN
Year 2012: ndvi_long=8.1849e-02, ndvi_modis=8.1613e-02, lst_K=3.1089e+02, no2=NaN, ch4=NaN
Year 2013: ndvi_long=8.5679e-02, ndvi_modis=8.4757e-02, lst_K=3.1096e+02, no2=NaN, ch4=NaN
Year 2014: ndvi_long=NaN, ndvi_modis=8.9537e-02, lst_K=3.1224e+02, no2=NaN, ch4=NaN
Year 2015: ndvi_long=NaN, ndvi_modis=8.3948e-02, lst_K=3.1133e+02, no2=NaN, ch4=NaN
Year 2016: ndvi_long=NaN, ndvi_modis=9.0404e-02, lst_K=3.1250e+02, no2=NaN, ch4=NaN
Year 2017: ndvi_long=NaN, ndvi_modis=8.3906e-02, lst_K=3.1248e+02, no2=NaN, ch4=NaN
Year 2018: ndvi_long=NaN, ndvi_modis=8.0875e-02, lst_K=3.1175e+02, no2=1.1376e-04, ch4=1.9020e+03
Year 2019: ndvi_long=NaN, ndvi_modis=9.4517e-02, lst_K=3.1264e+02, no2=1.1675e-04, ch4=1.8997e+03
Year 2020: ndvi_long=NaN, ndvi_modis=8.8399e-02, lst_K=3.1322e+02, no2=1.1188e-04, ch4=1.9173e+03
Year 2021: ndvi_long=NaN, ndvi_modis=9.1581e-02, lst_K=3.1331e+02, no2=1.2114e-04, ch4=1.9281e+03
Year 2022: ndvi_long=NaN, ndvi_modis=8.7294e-02, lst_K=3.1109e+02, no2=1.2170e-04, ch4=1.9256e+03
Year 2023: ndvi_long=NaN, ndvi_modis=9.6466e-02, lst_K=3.1164e+02, no2=1.2288e-04, ch4=1.9266e+03

=== Country: Oman ===

Year 1990: ndvi_long=9.3142e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1991: ndvi_long=8.6809e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1992: ndvi_long=9.0676e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1993: ndvi_long=9.2442e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1994: ndvi_long=9.3599e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1995: ndvi_long=9.8351e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1996: ndvi_long=9.5980e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1997: ndvi_long=9.7359e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1998: ndvi_long=9.8122e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 1999: ndvi_long=9.5837e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
Year 2000: ndvi_long=9.2750e-02, ndvi_modis=9.8442e-02, lst_K=3.1825e+02, no2=NaN, ch4=NaN
Year 2001: ndvi_long=9.5289e-02, ndvi_modis=9.8737e-02, lst_K=3.1591e+02, no2=NaN, ch4=NaN
Year 2002: ndvi_long=9.2122e-02, ndvi_modis=9.9575e-02, lst_K=3.1568e+02, no2=NaN, ch4=NaN
Year 2003: ndvi_long=8.6344e-02, ndvi_modis=9.9752e-02, lst_K=3.1506e+02, no2=NaN, ch4=NaN

Year 2004: ndvi_long=7.7252e-02, ndvi_modis=1.0041e-01, lst_K=3.1533e+02, no2=NaN, ch4=NaN
 Year 2005: ndvi_long=7.5032e-02, ndvi_modis=1.0117e-01, lst_K=3.1518e+02, no2=NaN, ch4=NaN
 Year 2006: ndvi_long=7.6596e-02, ndvi_modis=1.0072e-01, lst_K=3.1492e+02, no2=NaN, ch4=NaN
 Year 2007: ndvi_long=7.5973e-02, ndvi_modis=1.0289e-01, lst_K=3.1499e+02, no2=NaN, ch4=NaN
 Year 2008: ndvi_long=7.7790e-02, ndvi_modis=9.8782e-02, lst_K=3.1444e+02, no2=NaN, ch4=NaN
 Year 2009: ndvi_long=8.4545e-02, ndvi_modis=9.9143e-02, lst_K=3.1510e+02, no2=NaN, ch4=NaN
 Year 2010: ndvi_long=8.2403e-02, ndvi_modis=1.0232e-01, lst_K=3.1574e+02, no2=NaN, ch4=NaN
 Year 2011: ndvi_long=8.4820e-02, ndvi_modis=1.0155e-01, lst_K=3.1460e+02, no2=NaN, ch4=NaN
 Year 2012: ndvi_long=9.6574e-02, ndvi_modis=1.0181e-01, lst_K=3.1478e+02, no2=NaN, ch4=NaN
 Year 2013: ndvi_long=9.4302e-02, ndvi_modis=1.0109e-01, lst_K=3.1499e+02, no2=NaN, ch4=NaN
 Year 2014: ndvi_long=NaN, ndvi_modis=1.0259e-01, lst_K=3.1524e+02, no2=NaN, ch4=NaN
 Year 2015: ndvi_long=NaN, ndvi_modis=1.0045e-01, lst_K=3.1568e+02, no2=NaN, ch4=NaN
 Year 2016: ndvi_long=NaN, ndvi_modis=1.0126e-01, lst_K=3.1562e+02, no2=NaN, ch4=NaN
 Year 2017: ndvi_long=NaN, ndvi_modis=1.0070e-01, lst_K=3.1589e+02, no2=NaN, ch4=NaN
 Year 2018: ndvi_long=NaN, ndvi_modis=9.9410e-02, lst_K=3.1546e+02, no2=5.3663e-05, ch4=1.8996e+03
 Year 2019: ndvi_long=NaN, ndvi_modis=1.0037e-01, lst_K=3.1530e+02, no2=5.4365e-05, ch4=1.9071e+03
 Year 2020: ndvi_long=NaN, ndvi_modis=1.0473e-01, lst_K=3.1568e+02, no2=5.1010e-05, ch4=1.9207e+03
 Year 2021: ndvi_long=NaN, ndvi_modis=1.0252e-01, lst_K=3.1589e+02, no2=5.5416e-05, ch4=1.9356e+03
 Year 2022: ndvi_long=NaN, ndvi_modis=1.0198e-01, lst_K=3.1478e+02, no2=5.2623e-05, ch4=1.9284e+03
 Year 2023: ndvi_long=NaN, ndvi_modis=1.0216e-01, lst_K=3.1392e+02, no2=5.7665e-05, ch4=1.9227e+03

=== Country: Bahrain ===

Year 1990: ndvi_long=8.2104e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1991: ndvi_long=7.7387e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1992: ndvi_long=7.8627e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1993: ndvi_long=8.4630e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1994: ndvi_long=8.0428e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1995: ndvi_long=8.5466e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1996: ndvi_long=8.6005e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1997: ndvi_long=8.1355e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1998: ndvi_long=8.8442e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 1999: ndvi_long=8.0288e-02, ndvi_modis=NaN, lst_K=NaN, no2=NaN, ch4=NaN
 Year 2000: ndvi_long=7.5861e-02, ndvi_modis=8.3367e-02, lst_K=3.1106e+02, no2=NaN, ch4=NaN
 Year 2001: ndvi_long=8.3353e-02, ndvi_modis=9.1234e-02, lst_K=3.0950e+02, no2=NaN, ch4=NaN
 Year 2002: ndvi_long=8.0824e-02, ndvi_modis=8.7549e-02, lst_K=3.0945e+02, no2=NaN, ch4=NaN
 Year 2003: ndvi_long=7.7157e-02, ndvi_modis=8.5056e-02, lst_K=3.0799e+02, no2=NaN, ch4=NaN
 Year 2004: ndvi_long=7.5371e-02, ndvi_modis=8.8198e-02, lst_K=3.0856e+02, no2=NaN, ch4=NaN
 Year 2005: ndvi_long=7.4270e-02, ndvi_modis=8.4099e-02, lst_K=3.0796e+02, no2=NaN, ch4=NaN
 Year 2006: ndvi_long=7.4867e-02, ndvi_modis=8.2937e-02, lst_K=3.0838e+02, no2=NaN, ch4=NaN
 Year 2007: ndvi_long=7.6478e-02, ndvi_modis=8.8652e-02, lst_K=3.0794e+02, no2=NaN, ch4=NaN
 Year 2008: ndvi_long=7.3313e-02, ndvi_modis=8.1607e-02, lst_K=3.0810e+02, no2=NaN, ch4=NaN
 Year 2009: ndvi_long=6.9038e-02, ndvi_modis=8.1587e-02, lst_K=3.0765e+02, no2=NaN, ch4=NaN
 Year 2010: ndvi_long=6.9637e-02, ndvi_modis=8.5441e-02, lst_K=3.0892e+02, no2=NaN, ch4=NaN
 Year 2011: ndvi_long=7.3893e-02, ndvi_modis=8.7675e-02, lst_K=3.0772e+02, no2=NaN, ch4=NaN
 Year 2012: ndvi_long=9.9456e-02, ndvi_modis=8.4315e-02, lst_K=3.0831e+02, no2=NaN, ch4=NaN
 Year 2013: ndvi_long=9.9577e-02, ndvi_modis=8.6845e-02, lst_K=3.0809e+02, no2=NaN, ch4=NaN
 Year 2014: ndvi_long=NaN, ndvi_modis=9.5261e-02, lst_K=3.0876e+02, no2=NaN, ch4=NaN
 Year 2015: ndvi_long=NaN, ndvi_modis=8.7366e-02, lst_K=3.0893e+02, no2=NaN, ch4=NaN
 Year 2016: ndvi_long=NaN, ndvi_modis=9.0922e-02, lst_K=3.0921e+02, no2=NaN, ch4=NaN
 Year 2017: ndvi_long=NaN, ndvi_modis=9.7272e-02, lst_K=3.0886e+02, no2=NaN, ch4=NaN
 Year 2018: ndvi_long=NaN, ndvi_modis=9.0080e-02, lst_K=3.0922e+02, no2=1.3198e-04, ch4=NaN
 Year 2019: ndvi_long=NaN, ndvi_modis=9.5867e-02, lst_K=3.0952e+02, no2=1.2941e-04, ch4=NaN
 Year 2020: ndvi_long=NaN, ndvi_modis=9.3564e-02, lst_K=3.0916e+02, no2=1.2826e-04, ch4=NaN
 Year 2021: ndvi_long=NaN, ndvi_modis=8.8233e-02, lst_K=3.1012e+02, no2=1.3273e-04, ch4=NaN
 Year 2022: ndvi_long=NaN, ndvi_modis=8.9287e-02, lst_K=3.0849e+02, no2=1.3223e-04, ch4=1.9084e+03

Year 2023: ndvi_long=NaN, ndvi_modis=9.6855e-02, lst_K=3.0889e+02, no2=1.3296e-04, ch4=1.9181e+03

Extraction finished.

Elapsed time: 0:15:05.077435

Interpretation of the satellite extraction log (example: Saudi Arabia)

- `ndvi_long` corresponds to the annual mean NDVI from the GIMMS NDVI3g product (NASA/GIMMS/3GV0), scaled to the physical range $[-1, 1]$. For Saudi Arabia, values are typically around 0.09–0.11, which is consistent with predominantly arid land cover.
- `ndvi_modis` corresponds to the annual mean MODIS NDVI (MOD13A2), after applying the 0.0001 scale factor. From 2000 onward, `ndvi_long` and `ndvi_modis` are broadly comparable in magnitude.
- `lst_K` is the annual mean daytime land surface temperature from MODIS (MOD11A2), expressed in Kelvin (e.g. 313 K $\approx$ 40 °C).
- `no2` and `ch4` are Sentinel-5P column averages and are only available from 2018 onwards; they are expected to be NaN before that year.

```
In [8]: sat_df = pd.DataFrame(rows)
sat_df = sat_df.sort_values(["country", "year"]).reset_index(drop=True)

print("\nPreview of satellite indicators (first 10 rows):")
print(sat_df.head(10))

print("\nNon-null counts by column:")
print(sat_df.notnull().sum())

print("\nYear coverage per country:")
print(sat_df.groupby("country")["year"].agg(["min", "max", "count"]))
```

Preview of satellite indicators (first 10 rows):

	country	year	ndvi_long_mean	ndvi_modis_mean	lst_day_k_mean	no2_mean	\
0	Bahrain	1990	0.082104	NaN	NaN	NaN	
1	Bahrain	1991	0.077387	NaN	NaN	NaN	
2	Bahrain	1992	0.078627	NaN	NaN	NaN	
3	Bahrain	1993	0.084630	NaN	NaN	NaN	
4	Bahrain	1994	0.080428	NaN	NaN	NaN	
5	Bahrain	1995	0.085466	NaN	NaN	NaN	
6	Bahrain	1996	0.086005	NaN	NaN	NaN	
7	Bahrain	1997	0.081355	NaN	NaN	NaN	
8	Bahrain	1998	0.088442	NaN	NaN	NaN	
9	Bahrain	1999	0.080288	NaN	NaN	NaN	

	ch4_mean
0	NaN
1	NaN
2	NaN
3	NaN
4	NaN
5	NaN
6	NaN
7	NaN
8	NaN
9	NaN

Non-null counts by column:

country	204
year	204
ndvi_long_mean	144
ndvi_modis_mean	144
lst_day_k_mean	144
no2_mean	36
ch4_mean	32

dtype: int64

Year coverage per country:

	min	max	count
country			
Bahrain	1990	2023	34
Kuwait	1990	2023	34
Oman	1990	2023	34
Qatar	1990	2023	34
Saudi Arabia	1990	2023	34
United Arab Emirates	1990	2023	34

```
In [9]: OUTPUT_CSV = "gulf_satellite_indicators_by_country.csv"
sat_df.to_csv(OUTPUT_CSV, index=False)

print(f"Saved satellite indicators to: {OUTPUT_CSV}")
```

Saved satellite indicators to: gulf_satellite_indicators_by_country.csv

```
In [10]: sa_sat = sat_df[sat_df["country"] == "Saudi Arabia"].copy()

print("Saudi Arabia - satellite indicators summary:")
print(sa_sat.describe(include="all"))

print("\nSaudi Arabia - yearly values (first 15 rows):")
print(sa_sat.head(15)[["year", "ndvi_long_mean", "ndvi_modis_mean", "lst_day_k_mean", "no2_mean", "ch4_mean"]])

print("\nSaudi Arabia - non-null flags per year:")
print(sa_sat.groupby("year").agg({"country": lambda x: x[0]}))
```

```
sa_sat.set_index("year")  
    ["ndvi_long_mean", "ndvi_modis_mean", "lst_day_k_mean", "no2_mean", "ch4_mean"]  
    ].notnull()  
)
```

Saudi Arabia - satellite indicators summary:

	country	year	ndvi_long_mean	ndvi_modis_mean	\
count	34	34.000000	24.000000	24.000000	
unique	1	NaN	NaN	NaN	
top	Saudi Arabia	NaN	NaN	NaN	
freq	34	NaN	NaN	NaN	
mean	NaN	2006.500000	0.099069	0.100436	
std	NaN	9.958246	0.006033	0.002722	
min	NaN	1990.000000	0.089021	0.096825	
25%	NaN	1998.250000	0.095845	0.098645	
50%	NaN	2006.500000	0.098654	0.099655	
75%	NaN	2014.750000	0.103657	0.101418	
max	NaN	2023.000000	0.112110	0.109025	

	lst_day_k_mean	no2_mean	ch4_mean
count	24.000000	6.000000	6.000000
unique	NaN	NaN	NaN
top	NaN	NaN	NaN
freq	NaN	NaN	NaN
mean	313.287814	0.000060	1907.229511
std	0.835583	0.000002	13.815490
min	312.399288	0.000058	1887.655050
25%	312.629725	0.000059	1896.929770
50%	313.126248	0.000060	1911.744107
75%	313.601949	0.000061	1918.188337
max	316.277366	0.000064	1919.905734

Saudi Arabia - yearly values (first 15 rows):

	year	ndvi_long_mean	ndvi_modis_mean	lst_day_k_mean	no2_mean	ch4_mean
136	1990	0.103298	NaN	NaN	NaN	NaN
137	1991	0.098457	NaN	NaN	NaN	NaN
138	1992	0.098850	NaN	NaN	NaN	NaN
139	1993	0.101477	NaN	NaN	NaN	NaN
140	1994	0.098287	NaN	NaN	NaN	NaN
141	1995	0.103644	NaN	NaN	NaN	NaN
142	1996	0.105289	NaN	NaN	NaN	NaN
143	1997	0.104549	NaN	NaN	NaN	NaN
144	1998	0.112110	NaN	NaN	NaN	NaN
145	1999	0.102601	NaN	NaN	NaN	NaN
146	2000	0.098366	0.097812	316.277366	NaN	NaN
147	2001	0.105601	0.103985	314.046051	NaN	NaN
148	2002	0.100836	0.101261	313.627827	NaN	NaN
149	2003	0.094512	0.098661	313.137876	NaN	NaN
150	2004	0.092731	0.101721	313.200964	NaN	NaN

Saudi Arabia - non-null flags per year:

	ndvi_long_mean	ndvi_modis_mean	lst_day_k_mean	no2_mean	ch4_mean
year					
1990	True	False	False	False	False
1991	True	False	False	False	False
1992	True	False	False	False	False
1993	True	False	False	False	False
1994	True	False	False	False	False
1995	True	False	False	False	False
1996	True	False	False	False	False
1997	True	False	False	False	False
1998	True	False	False	False	False
1999	True	False	False	False	False
2000	True	True	True	False	False
2001	True	True	True	False	False
2002	True	True	True	False	False
2003	True	True	True	False	False

2004	True	True	True	False	False
2005	True	True	True	False	False
2006	True	True	True	False	False
2007	True	True	True	False	False
2008	True	True	True	False	False
2009	True	True	True	False	False
2010	True	True	True	False	False
2011	True	True	True	False	False
2012	True	True	True	False	False
2013	True	True	True	False	False
2014	False	True	True	False	False
2015	False	True	True	False	False
2016	False	True	True	False	False
2017	False	True	True	False	False
2018	False	True	True	True	True
2019	False	True	True	True	True
2020	False	True	True	True	True
2021	False	True	True	True	True
2022	False	True	True	True	True
2023	False	True	True	True	True

satellite image preview (sample)

For illustration purposes, we can display an example satellite composite over Saudi Arabia for a given year. This is **not** part of the modeling pipeline, but it helps visually confirm what kind of signal the indicators are summarizing.

The cell below tries to:

- Build a yearly MODIS NDVI composite for a chosen year (e.g. 2020),
- Display it on an interactive map together with the Saudi Arabia boundary.

This visualization uses `geemap` (a convenient wrapper around Earth Engine + folium). If `geemap` is not installed, the cell will print a short message instead of raising an error.

```
In [17]: try:
import geemap, ee
print("geemap and earthengine-api already available.")
except ImportError:
import sys
print("geemap not found. Installing into current Jupyter kernel...")
!{sys.executable} -m pip install -q geemap
import geemap, ee
print("Installation complete.")
```

geemap not found. Installing into current Jupyter kernel...

WARNING: Ignoring invalid distribution ~ip (C:\Python312\Lib\site-packages)
WARNING: Ignoring invalid distribution ~ip (C:\Python312\Lib\site-packages)
ERROR: Could not install packages due to an OSError: [WinError 5] Access is denied: 'C:\\Python312\\share\\jupyter\\labextensions\\jupyter-leaflet'
Consider using the `--user` option or check the permissions.

[notice] A new release of pip is available: 25.2 -> 25.3

[notice] To update, run: C:\Python312\python.exe -m pip install --upgrade pip

```

-----
ModuleNotFoundError                                Traceback (most recent call last)
Cell In[17], line 2
      1 try:
----> 2     import geemap, ee
      3     print("geemap and earthengine-api already available.")

ModuleNotFoundError: No module named 'geemap'

During handling of the above exception, another exception occurred:

ModuleNotFoundError                                Traceback (most recent call last)
Cell In[17], line 8
      6 print("geemap not found. Installing into current Jupyter kernel...")
      7 get_ipython().system('{sys.executable} -m pip install -q geemap')
----> 8 import geemap, ee
      9 print("Installation complete.")

ModuleNotFoundError: No module named 'geemap'

```

```

In [19]: # interactive map preview for a single year (e.g. Saudi Arabia, 2020)
try:
    import geemap.foliummap as geemap
    from IPython.display import display

    preview_country = "Saudi Arabia"
    preview_year = 2020

    print(f"Building MODIS NDVI annual composite for {preview_country}, {preview_year} ...")

    geom = get_country_geometry(preview_country)
    start = ee.Date.fromYMD(preview_year, 1, 1)
    end = start.advance(1, "year")

    # MODIS NDVI composite for the selected year
    ndvi_img = (
        MODIS_NDVI_IC
        .filterDate(start, end)
        .select(MODIS_NDVI_BAND)
        .mean()
    )

    # Simple visualization parameters (values and palette may be adjusted as needed)
    ndvi_vis = {
        "min": 0.0,
        "max": 8000.0,
        "palette": ["brown", "yellow", "green"],
    }

    # Create map centered roughly on Saudi Arabia
    m = geemap.Map(center=[23.5, 44], zoom=4)
    m.addLayer(ndvi_img, ndvi_vis, f"MODIS NDVI {preview_year}")
    m.addLayer(geom, {"color": "red"}, f"{preview_country} boundary", False)

    m.addLayerControl()
    display(m)

except ImportError:
    print("geemap is not installed. To enable the interactive preview, install it via:")
    print("  pip install geemap")

```

```
except Exception as e:
    print("An error occurred while building the preview map:", e)
```

geemap is not installed. To enable the interactive preview, install it via:
`pip install geemap`

Next step: Integration with national CO₂ and socioeconomic data

This notebook produces a single, clean CSV:

- `gulf_satellite_indicators_by_country.csv`

with one row per (country, year) and the following columns:

- `country`
- `year`
- `ndvi_long_mean` (PKU GIMMS NDVI annual mean, 1990–2022)
- `ndvi_modis_mean` (MODIS NDVI annual mean, 2000–2023)
- `lst_day_k_mean` (MODIS daytime LST in Kelvin, 2000–2023)
- `no2_mean` (Sentinel-5P NO₂ column mean, 2018–2023)
- `ch4_mean` (Sentinel-5P CH₄ column mean, 2018–2023)

The intended usage in the **baseline / modeling notebook** is:

1. Load the national GCC (or Saudi-only) dataset used for Baseline 1 and Baseline 2.
2. Merge it with this satellite dataset on `["country", "year"]` using a left join.
3. Apply any interpolation or availability-aware imputation at the **merged** level, enforcing:
 - No satellite values before the first valid sensor year (no backward extrapolation),
 - Strict temporal alignment (no look-ahead across years).
4. Define a new “Baseline 3” for Saudi Arabia that augments the national feature set with selected satellite indicators and evaluate its performance against:
 - Baseline 1 (GCC panel),
 - Baseline 2 (Saudi-only national model),
 - Naive last-year baseline.

All interpolation logic and model training remain outside this notebook, to keep this file focused solely on **data acquisition and annual aggregation of satellite indicators**.