

sentinel-3-downloading-processing

October 24, 2023

1 Downloading Sentinel 3 data using python

```
[2]: #installing eumdac package for downloading sentinel 3 data
pip install eumdac
```

```
Requirement already satisfied: eumdac in d:\anaconda3\lib\site-packages (2.1.0)
Requirement already satisfied: requests>=2.5.0 in d:\anaconda3\lib\site-packages
(from eumdac) (2.28.1)
Requirement already satisfied: pyyaml in d:\anaconda3\lib\site-packages (from
eumdac) (6.0)
Requirement already satisfied: charset-normalizer<3,>=2 in
d:\anaconda3\lib\site-packages (from requests>=2.5.0->eumdac) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in d:\anaconda3\lib\site-packages
(from requests>=2.5.0->eumdac) (3.3)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in d:\anaconda3\lib\site-
packages (from requests>=2.5.0->eumdac) (1.26.11)
Requirement already satisfied: certifi>=2017.4.17 in d:\anaconda3\lib\site-
packages (from requests>=2.5.0->eumdac) (2022.9.14)
Note: you may need to restart the kernel to use updated packages.

WARNING: Ignoring invalid distribution -yproj (d:\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -yproj (d:\anaconda3\lib\site-packages)
```

```
[notice] A new release of pip is available: 23.2.1 -> 23.3
[notice] To update, run: python.exe -m pip install --upgrade pip
```

```
[103]: import eumdac
import datetime
import shutil
# Insert your personal key and secret into the single quotes
consumer_key = 'qa88SaI8oAEkvXPs2uruI3L9Ttga'
consumer_secret = 'H8WpHfekE2diMSRFwGSRY93zngYa'

credentials = (consumer_key, consumer_secret)

token = eumdac.AccessToken(credentials)

print(f"This token '{token}' expires {token.expiration}")
```

This token 'af9ecd7c-66fd-3d8b-9dd9-61d59cf9f4cc' expires 2023-10-24
14:00:15.272801

```
[28]: # Add vertices for polygon, wrapping back to the start point.
#polygon box created which includes the given lat and lon

datastore = eumdac.DataStore(token)
#print(datastore.collections)
selected_collection = datastore.get_collection('EO:EUM:DAT:0412')
ex=0.5
geometry = [[86.61137+ex, 20.24755+ex],[86.61137+ex, 20.96497+ex],[71.5649+ex,
↪20.96497+ex],[71.5649+ex, 20.24755+ex],[86.61137+ex, 20.24755+ex]]

# Set sensing start and end time
start = datetime.datetime(2023, 10, 18, 0, 0)
end = datetime.datetime(2023, 10, 18, 23, 59)

# Retrieve datasets that match our filter
products = selected_collection.search(
    geo='POLYGON(({})'.format(','.join(["{} {}".format(*coord) for coord in
↪geometry])),
    dtstart=start,
    dtend=end)

print(f'Found Datasets: {len(products)} datasets for the given time range')

for product in products:
    print(str(product))
```

```
Found Datasets: 13 datasets for the given time range
S3A_SL_2_WST____20231018T172738_20231018T173038_20231018T185023_0180_104_312_018
0_MAR_0_NR_003.SEN3
S3B_SL_2_WST____20231018T165143_20231018T165443_20231018T180124_0179_085_169_036
0_MAR_0_NR_003.SEN3
S3B_SL_2_WST____20231018T164843_20231018T165143_20231018T180122_0179_085_169_018
0_MAR_0_NR_003.SEN3
S3A_SL_2_WST____20231018T154938_20231018T155238_20231018T165302_0179_104_311_036
0_MAR_0_NR_003.SEN3
S3A_SL_2_WST____20231018T154638_20231018T154938_20231018T170323_0180_104_311_018
0_MAR_0_NR_003.SEN3
S3B_SL_2_WST____20231018T054049_20231018T054349_20231018T074717_0179_085_162_252
0_MAR_0_NR_003.SEN3
S3A_SL_2_WST____20231018T044144_20231018T044444_20231018T070206_0179_104_304_270
0_MAR_0_NR_003.SEN3
S3A_SL_2_WST____20231018T043844_20231018T044144_20231018T070204_0179_104_304_252
0_MAR_0_NR_003.SEN3
S3B_SL_2_WST____20231018T043331_20231018T061431_20231019T140951_6059_085_161_____
```

```

__MAR_O_NT_003.SEN3
S3B_SL_2_WST____20231018T040249_20231018T040549_20231018T060637_0179_085_161_270
O_MAR_O_NR_003.SEN3
S3B_SL_2_WST____20231018T035949_20231018T040249_20231018T060422_0179_085_161_252
O_MAR_O_NR_003.SEN3
S3A_SL_2_WST____20231018T033126_20231018T051226_20231019T142832_6059_104_303____
__MAR_O_NT_003.SEN3
S3B_SL_2_WST____20231018T025232_20231018T043331_20231019T140937_6059_085_160____
__MAR_O_NT_003.SEN3

```

[29]: *#downloading the available sentinel 3 WST data in the region*

```

import os

download_path = r"C:\Users\Legion\Desktop\climaview\2023_10_18" # Replace with
↳ your desired path

for product in products:
    with product.open() as fsrc, \
        open(os.path.join(download_path, fsrc.name), mode='wb') as fdst:
        shutil.copyfileobj(fsrc, fdst)
        print(f'Download of product {product} finished.')

print('All downloads are finished.')

```

```

Download of product S3A_SL_2_WST____20231018T172738_20231018T173038_20231018T185
023_0180_104_312_0180_MAR_O_NR_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T165143_20231018T165443_20231018T180
124_0179_085_169_0360_MAR_O_NR_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T164843_20231018T165143_20231018T180
122_0179_085_169_0180_MAR_O_NR_003.SEN3 finished.
Download of product S3A_SL_2_WST____20231018T154938_20231018T155238_20231018T165
302_0179_104_311_0360_MAR_O_NR_003.SEN3 finished.
Download of product S3A_SL_2_WST____20231018T154638_20231018T154938_20231018T170
323_0180_104_311_0180_MAR_O_NR_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T054049_20231018T054349_20231018T074
717_0179_085_162_2520_MAR_O_NR_003.SEN3 finished.
Download of product S3A_SL_2_WST____20231018T044144_20231018T044444_20231018T070
206_0179_104_304_2700_MAR_O_NR_003.SEN3 finished.
Download of product S3A_SL_2_WST____20231018T043844_20231018T044144_20231018T070
204_0179_104_304_2520_MAR_O_NR_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T043331_20231018T061431_20231019T140
951_6059_085_161_____MAR_O_NT_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T040249_20231018T040549_20231018T060
637_0179_085_161_2700_MAR_O_NR_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T035949_20231018T040249_20231018T060
422_0179_085_161_2520_MAR_O_NR_003.SEN3 finished.
Download of product S3A_SL_2_WST____20231018T033126_20231018T051226_20231019T142

```

832_6059_104_303_____MAR_0_NT_003.SEN3 finished.
Download of product S3B_SL_2_WST____20231018T025232_20231018T043331_20231019T140
937_6059_085_160_____MAR_0_NT_003.SEN3 finished.
All downloads are finished.

```
[1]: import xarray as xr
import rioxarray as rio
```

```
[96]: #Path where the data file is located
data = r"C:\Users\Legion\Desktop\COPERN~1\S3B_SL~1.SEN\S3B_SL~1.SEN\202310~1.NC"
```

```
[97]: #Read the data as dataset (DS)
DS = xr.open_dataset(data,engine='netcdf4')
```

```
[98]: #Show the metadata
DS.info
```

```
[98]: <bound method Dataset.info of <xarray.Dataset>
Dimensions:                               (time: 1, nj: 1200, ni: 1500, channel: 3)
Coordinates:
  lat                                       (nj, ni) float32 ...
  lon                                       (nj, ni) float32 ...
  * time                                   (time) datetime64[ns] 2023-10-19T18:03:32
Dimensions without coordinates: nj, ni, channel
Data variables: (12/22)
  adi_dtime_from_sst                      (time, nj, ni) float32 ...
  aerosol_dynamic_indicator                (time, nj, ni) float32 ...
  brightness_temperature                   (channel, time, nj, ni) float32 ...
  dt_analysis                             (time, nj, ni) float32 ...
  dual_nadir_sst_difference                (time, nj, ni) float32 ...
  l2p_flags                               (time, nj, ni) int16 ...
  ...                                     ...
  sses_standard_deviation                  (time, nj, ni) float32 ...
  sst_algorithm_type                       (time, nj, ni) int8 ...
  sst_dtime                                (time, nj, ni) timedelta64[ns] ...
  sst_theoretical_uncertainty              (time, nj, ni) float32 ...
  wind_speed                              (time, nj, ni) float32 ...
  wind_speed_dtime_from_sst                (time, nj, ni) float32 ...
Attributes: (12/48)
  Conventions:                            CF-1.6, Unidata Observation Dataset v1.0
  Metadata_Conventions:                   Unidata Dataset Discovery v1.0
  acknowledgment:                         European Commission Copernicus Programme
  cdm_data_type:                           swath
  comment:                                GHRSST SST L2P
  creator_email:                           ops@eumetsat.int
  ...                                     ...
  summary:                                Sentinel-3B SLSTR skin sea surface temperature
```

```

time_coverage_end:      20231019T180631Z
time_coverage_start:    20231019T180332Z
title:                  Sentinel-3B SLSTR L2P SST dataset
uuid:                   TBC
westernmost_longitude:  50.4677619934082>

```

[95]: DS

```

[95]: <xarray.Dataset>
Dimensions:                               (time: 1, nj: 1200, ni: 1500, channel: 3)
Coordinates:
  lat                                     (nj, ni) float32 ...
  lon                                     (nj, ni) float32 ...
  * time                                 (time) datetime64[ns] 2023-10-19T18:03:32
Dimensions without coordinates: nj, ni, channel
Data variables: (12/22)
  adi_dtime_from_sst                    (time, nj, ni) float32 ...
  aerosol_dynamic_indicator              (time, nj, ni) float32 ...
  brightness_temperature                 (channel, time, nj, ni) float32 ...
  dt_analysis                           (time, nj, ni) float32 ...
  dual_nadir_sst_difference              (time, nj, ni) float32 ...
  l2p_flags                             (time, nj, ni) int16 ...
  ...
  sses_standard_deviation               (time, nj, ni) float32 ...
  sst_algorithm_type                    (time, nj, ni) int8 ...
  sst_dtime                             (time, nj, ni) timedelta64[ns] ...
  sst_theoretical_uncertainty            (time, nj, ni) float32 ...
  wind_speed                            (time, nj, ni) float32 ...
  wind_speed_dtime_from_sst             (time, nj, ni) float32 ...
Attributes: (12/48)
  Conventions:                          CF-1.6, Unidata Observation Dataset v1.0
  Metadata_Conventions:                 Unidata Dataset Discovery v1.0
  acknowledgment:                       European Commission Copernicus Programme
  cdm_data_type:                         swath
  comment:                              GHRSSST SST L2P
  creator_email:                        ops@eumetsat.int
  ...
  summary:                              Sentinel-3B SLSTR skin sea surface temperature
  time_coverage_end:                    20231019T180631Z
  time_coverage_start:                  20231019T180332Z
  title:                                Sentinel-3B SLSTR L2P SST dataset
  uuid:                                 TBC
  westernmost_longitude:                50.4677619934082

```

[77]: DS.coords

```
[77]: Coordinates:
      lat      (nj, ni) float32 ...
      lon      (nj, ni) float32 ...
      * time    (time) datetime64[ns] 2023-10-19T18:03:32
```

```
[83]: DS.data_vars
```

```
[83]: Data variables:
      adi_dtime_from_sst      (time, nj, ni) float32 ...
      aerosol_dynamic_indicator (time, nj, ni) float32 ...
      brightness_temperature  (channel, time, nj, ni) float32 ...
      dt_analysis              (time, nj, ni) float32 ...
      dual_nadir_sst_difference (time, nj, ni) float32 ...
      l2p_flags                (time, nj, ni) int16 ...
      nadir_sst_theoretical_uncertainty (time, nj, ni) float32 ...
      nedt                     (channel, time, nj, ni) float32 ...
      probability_cloud_single_in (time, nj, ni) float32 ...
      probability_cloud_single_io (time, nj, ni) float32 ...
      quality_level            (time, nj, ni) int8 ...
      satellite_zenith_angle   (time, nj, ni) float32 ...
      sea_ice_fraction          (time, nj, ni) float32 ...
      sea_ice_fraction_dtime_from_sst (time, nj, ni) float32 ...
      sea_surface_temperature  (time, nj, ni) float32 ...
      sses_bias                 (time, nj, ni) float32 ...
      sses_standard_deviation  (time, nj, ni) float32 ...
      sst_algorithm_type        (time, nj, ni) int8 ...
      sst_dtime                 (time, nj, ni) timedelta64[ns] ...
      sst_theoretical_uncertainty (time, nj, ni) float32 ...
      wind_speed                (time, nj, ni) float32 ...
      wind_speed_dtime_from_sst (time, nj, ni) float32 ...
```

2 Extracting Sea Surface temperature data from geotiff(site2)

```
[126]: import rasterio
data = rasterio.open(r"C:\Users\Legion\Desktop\climaview\2023_10_18\SST_DATA_1.
↳tif")
```

```
[127]: #The exported geotiff file have two bands, band 0 is grey scale band and band 1
↳contains temp value
# so read data from the first band

z = data.read()[1]
# check the crs of the data
data.crs
```

```
[127]: CRS.from_wkt('GEOGCS["WGS 84",DATUM["World Geodetic System 1984",SPHEROID["WGS 84",6378137,298.257223563]],PRIMEM["Greenwich",0],UNIT["degree",0.0174532925199433,AUTHORITY["EPSG","9122"]],AXIS["Latitude",NORTH],AXIS["Longitude",EAST]]')
```

```
[128]: # check the bounding-box of the data
data.bounds
```

```
[128]: BoundingBox(left=59.349449157714844, bottom=9.60668590291341,
right=75.48120632425943, top=22.99787139892578)
```

```
[129]: # since the raster is in regular lon/lat grid (4326) we can use
# `dat.index()` to identify the index of a given lon/lat pair
# (e.g. it expects coordinates in the native crs of the data)

def getval(lon, lat):
    idx = data.index(lon, lat, precision=1E-6)
    print("Sea Surface Temperature of location",z[idx])
    return
getval(71.5649, 20.96497) #location given in the task
```

Sea Surface Temperature of location 28.179987

3 Extracting Sea Surface temperature data from geotiff(site1)

```
[135]: import rasterio
data = rasterio.open(r"C:\Users\Legion\Desktop\climaview\2023_10_18\SST_DATA_2.
    ↳tif")

#The exported geotiff file have two bands, band 0 is grey scale band and band 1_
    ↳contains temp value
# so read data from the first band[1]
z = data.read()[1]

# check the crs of the data
data.crs

# check the bounding-box of the data
data.bounds

# since the raster is in regular lon/lat grid (4326) we can use
# `dat.index()` to identify the index of a given lon/lat pair
# (e.g. it expects coordinates in the native crs of the data)
def getval(lon, lat):
    idx = data.index(lon, lat, precision=1E-6)
    print("Sea Surface Temperature of given location:",z[idx])
    return
```

```
getval(86.5899,20.2239) #the given loaction contains no SST value so i took the ↵  
↵near lat and lon value
```

Sea Surface Temperature of given location: 23.140009