Python_example_lab3

August 8, 2018

```
In [1]: %matplotlib inline
        import matplotlib.pyplot as plt
        import numpy as np
        from netCDF4 import Dataset
```

1 Import data

```
In [2]: datadir = r'C:\Users\STUDENT\SATELLITE_DATA/'
        data = Dataset(datadir+'20180802000000-OSISAF-L3C_GHRSST-SSTsubskin-SEVIRI_SST-ssteqc_me
```

```
In [3]: #print(data.variables) <-- use this print statement to print out detailed info about the

        # I have commented this cell out due to the large amount of output it prints
```

```
OrderedDict([('time', <class 'netCDF4._netCDF4.Variable'>
int32 time(time)
    long_name: reference time of sst file
    standard_name: time
    axis: T
    units: seconds since 1981-01-01 00:00:00
unlimited dimensions: time
current shape = (1,)
filling on, default _FillValue of -2147483647 used
), ('lat', <class 'netCDF4._netCDF4.Variable'>
float32 lat(lat)
    long_name: latitude
    standard_name: latitude
    axis: Y
    units: degrees_north
    valid_min: -90.0
    valid_max: 90.0
    comment: geographical coordinates, WGS84 projection
unlimited dimensions:
current shape = (2400,)
filling on, default _FillValue of 9.969209968386869e+36 used
), ('lon', <class 'netCDF4._netCDF4.Variable'>
float32 lon(lon)
```

```
    long_name: longitude
    standard_name: longitude
    axis: X
    units: degrees_east
    valid_min: -180.0
    valid_max: 180.0
    comment: geographical coordinates, WGS84 projection
unlimited dimensions:
current shape = (2400,)
filling on, default _FillValue of 9.969209968386869e+36 used
), ('sea_surface_temperature', <class 'netCDF4._netCDF4.Variable'>
int16 sea_surface_temperature(time, lat, lon)
    _FillValue: -32768
    long_name: sea surface subskin temperature
    standard_name: sea_surface_subskin_temperature
    units: kelvin
    add_offset: 273.15
    scale_factor: 0.01
    valid_min: -300
    valid_max: 4500
    depth: 1 millimeter
    source: SEVIRI
    comment: Temperature of the subskin of the ocean
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('sst_dtime', <class 'netCDF4._netCDF4.Variable'>
int32 sst_dtime(time, lat, lon)
    _FillValue: -2147483648
    long_name: time difference from reference time
    units: seconds
    add_offset: 0.0
    scale_factor: 1.0
    valid_min: -2147483647
    valid_max: 2147483647
    comment: time plus sst_dtime gives seconds after 00:00:00 UTC January 1, 1981
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('sses_bias', <class 'netCDF4._netCDF4.Variable'>
int8 sses_bias(time, lat, lon)
    _FillValue: -128
    long_name: SSES bias estimate
    units: kelvin
    add_offset: 0.0
    scale_factor: 0.01
    valid_min: -127
    valid_max: 127
    comment: Bias estimate derived using the techniques described at http://www.ghrsst.org/SSES-
unlimited dimensions: time
```

```
current shape = (1, 2400, 2400)
filling on), ('sses_standard_deviation', <class 'netCDF4._netCDF4.Variable'>
int8 sses_standard_deviation(time, lat, lon)
    _FillValue: -128
    long_name: SSES standard deviation
    units: kelvin
    add_offset: 1.0
    scale_factor: 0.01
    valid_min: -127
    valid_max: 127
    comment: Standard deviation estimate derived using the techniques described at http://www.gh
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('dt_analysis', <class 'netCDF4._netCDF4.Variable'>
int8 dt_analysis(time, lat, lon)
    _FillValue: -128
    long_name: deviation from SST analysis or reference climatology
    units: kelvin
    add_offset: 0.0
    scale_factor: 0.1
    valid_min: -127
    valid_max: 127
    comment: The difference between this SST and the previous day's SST analysis
    reference: OSTIA
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('wind_speed', <class 'netCDF4._netCDF4.Variable'>
int8 wind_speed(time, lat, lon)
    _FillValue: -128
    long_name: 10m wind speed
    standard_name: wind_speed
    units: m s-1
    height: 10 m
    add_offset: 0.0
    scale_factor: 1.0
    valid_min: 0
    valid_max: 127
    comment: These wind speeds were created by the ECMWF and represent winds at 10 metres above
    source: WSP-ECMWF-Forecast
    time_offset: 0.0
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('sea_ice_fraction', <class 'netCDF4._netCDF4.Variable'>
int8 sea_ice_fraction(time, lat, lon)
    _FillValue: -128
    long_name: sea ice fraction
    standard_name: sea_ice_area_fraction
    units:
```

```
    add_offset: 0.0
    scale_factor: 0.01
    valid_min: 0
    valid_max: 100
    comment: Fractional sea ice cover from OSISAF ice product
    source: ICE-OSISAF
    time_offset: -36.0
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('aerosol_dynamic_indicator', <class 'netCDF4._netCDF4.Variable'>
int8 aerosol_dynamic_indicator(time, lat, lon)
    _FillValue: -128
    long_name: aerosol dynamic indicator
    units:
    add_offset: 0.0
    scale_factor: 0.1
    valid_min: 0
    valid_max: 127
    comment:
    source: sources_of_adi
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('adi_dtime_from_sst', <class 'netCDF4._netCDF4.Variable'>
int8 adi_dtime_from_sst(time, lat, lon)
    _FillValue: -128
    long_name: time difference of ADI data from sst measurement
    units: hour
    add_offset: 0.0
    scale_factor: 0.1
    valid_min: -127
    valid_max: 127
    comment: Difference in hours between the ADI and SST data
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('sources_of_adi', <class 'netCDF4._netCDF4.Variable'>
int8 sources_of_adi(time, lat, lon)
    _FillValue: -128
    long_name: sources of aerosol dynamic indicator
    valid_min: 0
    valid_max: 2
    flag_meanings: no_data AOD-NAAPS-ADI SDI-OSISAF-ADI
    flag_values: [0 1 2]
    comment: This variable provides a pixel by pixel description of where aerosol optical depth
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('l2p_flags', <class 'netCDF4._netCDF4.Variable'>
int32 l2p_flags(time, lat, lon)
    long_name: L2P flags
    valid_min: 0
    valid_max: 15
    flag_meanings: microwave land ice lake
    flag_masks: [1 2 4 8]
    comment: These flags are important to properly use the data.
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on, default _FillValue of -2147483647 used
), ('quality_level', <class 'netCDF4._netCDF4.Variable'>
int8 quality_level(time, lat, lon)
    _FillValue: -128
    long_name: quality level of SST pixel
    valid_min: 0
    valid_max: 5
    flag_meanings: no_data bad_data worst_quality low_quality acceptable_quality best_quality
    flag_values: [0 1 2 3 4 5]
    comment: These are the overall quality indicators and are used for all GHRSST SSTs
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('satellite_zenith_angle', <class 'netCDF4._netCDF4.Variable'>
int8 satellite_zenith_angle(time, lat, lon)
    _FillValue: -128
    long_name: satellite zenith angle
    units: angular_degree
    add_offset: 0.0
    scale_factor: 1.0
    valid_min: -90
    valid_max: 90
    comment: The satellite zenith angle at the time of the SST observations.
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('solar_zenith_angle', <class 'netCDF4._netCDF4.Variable'>
int8 solar_zenith_angle(time, lat, lon)
    _FillValue: -128
    long_name: solar zenith angle
    units: angular_degree
    add_offset: 90.0
    scale_factor: 1.0
    valid_min: -90
    valid_max: 90
    comment: The solar zenith angle at the time of the SST observations.
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('or_latitude', <class 'netCDF4._netCDF4.Variable'>
int16 or_latitude(time, lat, lon)
    _FillValue: -32768
    long_name: original latitude of the SST value
    standard_name: latitude
    units: degrees_north
    add_offset: 0.0
    scale_factor: 0.01
    valid_min: -9000
    valid_max: 9000
    comment: Original latitude of the SST value
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on), ('or_longitude', <class 'netCDF4._netCDF4.Variable'>
int16 or_longitude(time, lat, lon)
    _FillValue: -32768
    long_name: original longitude of the SST value
    standard_name: longitude
    units: degrees_east
    add_offset: 0.0
    scale_factor: 0.01
    valid_min: -18000
    valid_max: 18000
    comment: Original longitude of the SST value
unlimited dimensions: time
current shape = (1, 2400, 2400)
filling on)])
```

```
In [4]: # Print the names of all of the variables in "data"
        print(data.variables.keys())
```

```
odict_keys(['time', 'lat', 'lon', 'sea_surface_temperature', 'sst_dtime', 'sses_bias', 'sses_sta
```

2 Define variables

```
In [3]: sst = data.variables['sea_surface_temperature']
        lon = data.variables['lon']
        lat = data.variables['lat']
        print('Shape of sst is ',sst.shape)
        print('Shape of lon is ',lon.shape)
        print('Shape of lat is ',lat.shape)
```

```
Shape of sst is  (1, 2400, 2400)
Shape of lon is  (2400,)
Shape of lat is  (2400,)
```

3 Convert from Kelvin to Celsius

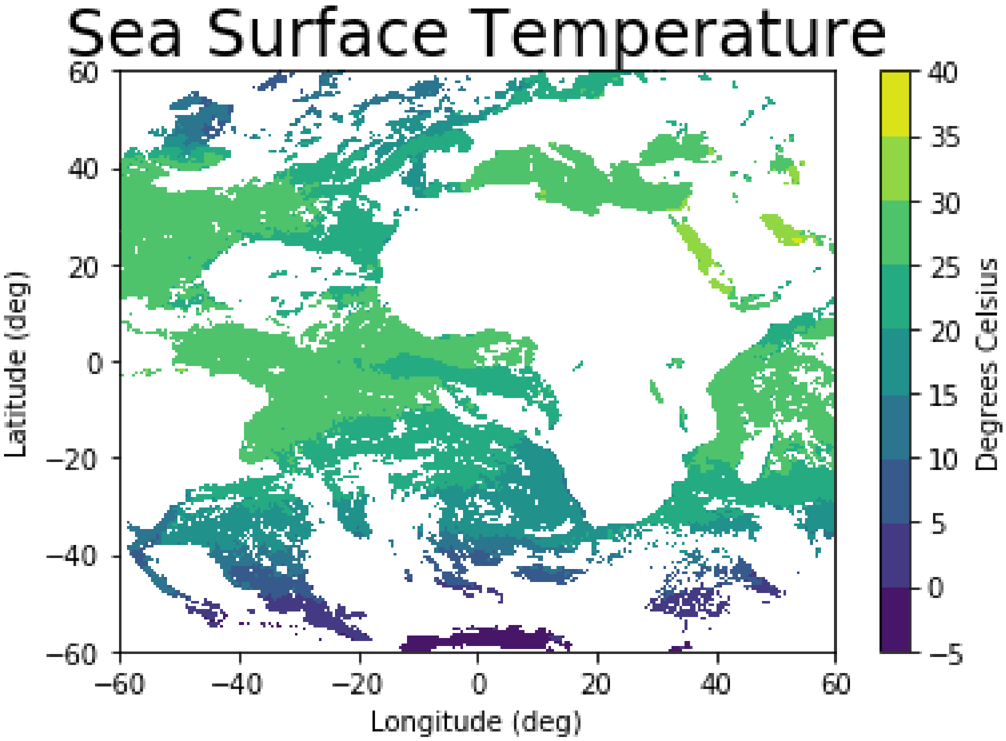
```
In [4]: sst_C = sst[:,:,:] - 273
```

4 Plot the data!

```
In [9]: LON,LAT = np.meshgrid(lon,lat)

        plt.contourf(LON,LAT,sst_C[0,:,:])
        plt.colorbar(label='Degrees Celsius')
        plt.title('Sea Surface Temperature',fontsize=24)
        plt.xlabel('Longitude (deg)')
        plt.ylabel('Latitude (deg)')
        plt.savefig('Figures/SST_satellite_COESSING.jpg')
```

```
Out[9]: <matplotlib.text.Text at 0x120c43400>
```



```
In [ ]:
```