MODIS & VIIRS Data: Learning with the ORNL DAAC

McNelis, J.J., Shrestha, R., Boyer, A. G., Klaassen, S., Thornton, M.M., Wilson, B.E.
Oak Ridge National Laboratory Distributed Active Archive Center

The ORNL DAAC wants you to be a MODIS and VIIRS data power user!

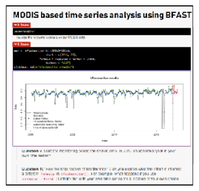
The ORNL DAAC has built a collection of resources for students, instructors, and researchers that use our MODIS and VIIRS services as tools for learning. The resources include classroom exercises, materials contributed by our active user community, and other e-Learning content developed by our staff. The ORNL DAAC has the resources that you need to be a more informed user of MODIS and VIIRS data, regardless of experience.

The Oak Ridge National Laboratory Distributed Active Archive Center (ORNL DAAC) for Biogeochemical Dynamics is one of the NASA Earth Observing System Data and Information System (EOSDIS) data centers. ORNL DAAC is operated by the ORNL Environmental Sciences Division and is responsible for data archival product development and distribution, and user support for biogeochemical and ecological data and models.

https://daac.ornl.gov Contact us: **uso@daac.ornl.gov**

Classroom Exercises

MODIS based time series analysis using BFAST



Jan Verbesselt
Associate Professor
Laboratory of GIS and Remote Sensing
Wageningen University
The Netherlands
Change Detection and Monitoring

This exercise is an introduction to MODIS NDVI time series analysis in R and implementation of the BFAST library for the decomposition of the time series into trend, season, and remainder components. This type of analysis is useful for detecting and characterizing change within the time series.

You will learn how to access MODIS NDVI fixed site data in CSV format, visualize the time series, perform quality filtering, and compute BFAST metrics within the R environment. This exercise is based on data for the Gelderland Loobos Site in The Netherlands.

MODIS based time series analysis using BFAST

The latest addition to the **Classroom Exercises** page

https://modis.ornl.gov/resources.html

Watch and Learn

What? NASA Earthdata Webinar August 2018

Why? Learn features and capabilities of ORNL DAAC's MODIS and VIIRS services:

- Visualization capabilities
- Placing Global Tool order
- Accessing Fixed Sites data
- Command-line access via the API
- Jupyter notebook demonstration

How? Watch on YouTube!
https://daac.ornl.gov/workshops/MODIS-VIIRS_Webinar_20180816.html

Community Tools

MODISTools R Package

Dr. Koen Hufkens
Computational & Applied Vegetation Ecology Lab
Ghent University
Belgium
Koen Hufkens

MODISTools package for Web Service access in R

Learn the MODIS & VIIRS API

The ORNL DAAC GitHub page is the place to learn how to use the MODIS & VIIRS API — your command-line interface to land products subsets from the MODIS & VIIRS instruments. Call web service functions using your favorite scripting language.

Bash
```
$ curl -X GET --header 'Accept: application/json'
```

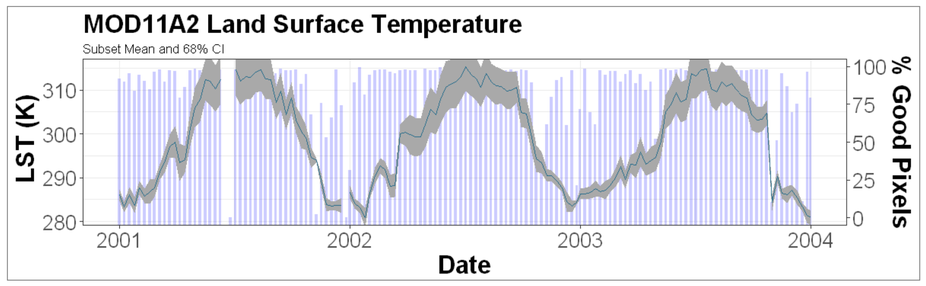
Python
```
> requests.get('<url>', headers=header).text
```

R
```
> GET('<url>')
```

AND MORE!

Example subset request:

Url	https://modis.ornl.gov/rst/api/v1/
Product	MOD11A2/
Function	subset?
Latitude	latitude=35
Longitude	&longitude=-85
Band	&band=LST_Day_1km
Start date	&startDate=A2001001
End date	&endDate=A2004001
Y window	&kmAboveBelow=25
X window	&kmLeftRight=25



Land Surface Temperature time series from our subset request

```
{
  "xllcorner": "-7766047.74",
  "yllcorner": "3868661.22",
  "cellsize": 926.6254330558338,
  "nrows": 51,
  "ncols": 51,
  "band": "LST_Day_1km",
  "units": "Kelvin",
  "scale": "0.02",
  "latitude": 35.0,
  "longitude": -85.0,
  "header": "<URL>",
  "subset": [
     {
      "modis_date": "A2001049",
      "calendar_date":"2001-02-18",
      "band": "LST_Day_1km",
      "tile": "h11v05",
      "proc_date": "2015113205847",
      "data": [14259, 14167, 14168, ... ]
     }, ...
  ]
}
```

JSON output from the example request

```
> lststats = pd.DataFrame(        # Get statistics
    np.column_stack([
     lstdata_filt_scale.mean(axis=1),
     lstdata.std(axis=1),
     lstdata.apply(
     lambda row:(float(row.count())/(ncol*nrow))*100)
   ]), columns=['mean', 'sd', 'quality'], index=dates
  )
> matplotlib.plot(lststats)      #  Plot !
```

Tutorial: https://github.com/ornldaac/modis_restservice_qc_filter_Python/
Apply quality filtering and plot a MODIS or VIIRS time series R and Python

Tutorial: https://github.com/ornldaac/webinar-modis-viirs-august2018
The API supplies all of the metadata required for gridding! Learn how!

```
> raster <- raster(
     vals = subset$data$data,
     nrows = subset$nrow,
     ncols = subset$ncol,
     xmn = subset$xllcorner,
     xmx = subset$ncol+ (subset
     ymn = subset$yllcorner,
     ymx = yllcorner + (subset$
     crs = '+proj=sinu +lon_0=0
     resolution = subset$cellsi
  )

> plot(raster)    #  Plot !
```