TATSSI-documentation

October 8, 2020

Contents

General information

- **Version** v0.1-beta1
- **Title** Tools for Analyzing Time Series of Satellite Imagery
- **Author** Gerardo Lopez-Saldana [aut, cre], Inder Tecuapetla-Gómez [ctb]
- **Maintainer** Gerardo Lopez-Saldana [series_tiempo@conabio.gob.mx]
- **License** GNU Affero GPL (>=3)
- **Repository** GitHub
- **Date/Publication** 2020-09-25
- **DOI** 10.5281/zenodo.4050082

1 Tools for Analyzing Time Series of Satellite Imagery (TATSSI)

1.1 Description

TATSSI is an open source platform that provides routines for downloading, generating, gap-filling, smoothing, analyzing and exporting time series of the Moderate Resolution Imaging Spectroradiometer (MODIS) and Visible Infrared Imaging Radiometer Suite (VIIRS), see also (VIIRS), and some of their derivatives. Since the quality flags of these products are used in the generation of any time series, quality analysis (QA) is the cornerstone of any analysis made with TATSSI.

TATSSI has been written in Python and extensively tested in Linux (Ubuntu 18.04 and 20.04). TATSSI can also be run in a Windows machine through the WSL-Ubuntu interpreter. Instructions for TATSSI's installation can be found at this GitHub repository.

TATSSI is user-friendly as its routines can be accessed via a graphical user interface (GUI), a series of Jupyter Notebooks or command-line instructions.

TATSSI can process time series in mass either on a local system or on an external server (e.g. Google Cloud or Earth Engine) through the use of Python's library DASK.

TATSSI's routines are divided in three menus: Data, Time series and Quality. Figure 1 provides an overlook of TATSSI's menus structure and Figure 2 shows the flowchart of these routines.

Figure 1: TATSSI's menus structure.

NOTE: TATSSI's Jupyter Notebooks are already documented, so this manuscript is mainly concerned with parameters required in the GUI version of TATSSI.

1.2 Location of TATSSI's folders when using the WSL-Ubuntu interpreter

When TATSSI's repository at GitHub is cloned via the WSL-Ubuntu interpreter, a copy of the platform is saved in the following directory:

`C:\Users\userName\AppData\Local\Packages\CanonicalGroupLimited.X\LocalState\rootfs\home\`

where `userName` and `X` depend on the system and Ubuntu version used to run TATSSI. For instance, in the system used to generate this documentation, the WSL-Ubuntu interpreter uses the 18.04 version and $X =$ `Ubuntu18.04onWindows_79rhkp1fndgsc`.

2 Data menu

2.1 Translator

Allows converting to/from all GDAL supported formats. This is mainly an internal module that is used to translate input files into a Cloud Optimized GeoTIFF (COG) which is the default internal TATSSI format. It is possible to access to this module via the TATSSI's Jupyter Notebook; have to open the file `TATSSI_ImportExport.ipynb`.

Here is an example on how to use this module to convert the MOD13A2.006 EVI SubDataset product into a COG directly from a Python script:

```
# Import directly the TATSSI translate module
import os
from glob import glob
from TATSSI.input_output.translate import Translate
from TATSSI.input_output.utils import *

# Set data directory
DataDir = "/home/TATSSI/data/MOD13A2.006/"
# Get all HDF files into a list
fnames = os.path.join(DataDir, '*.hdf')
```

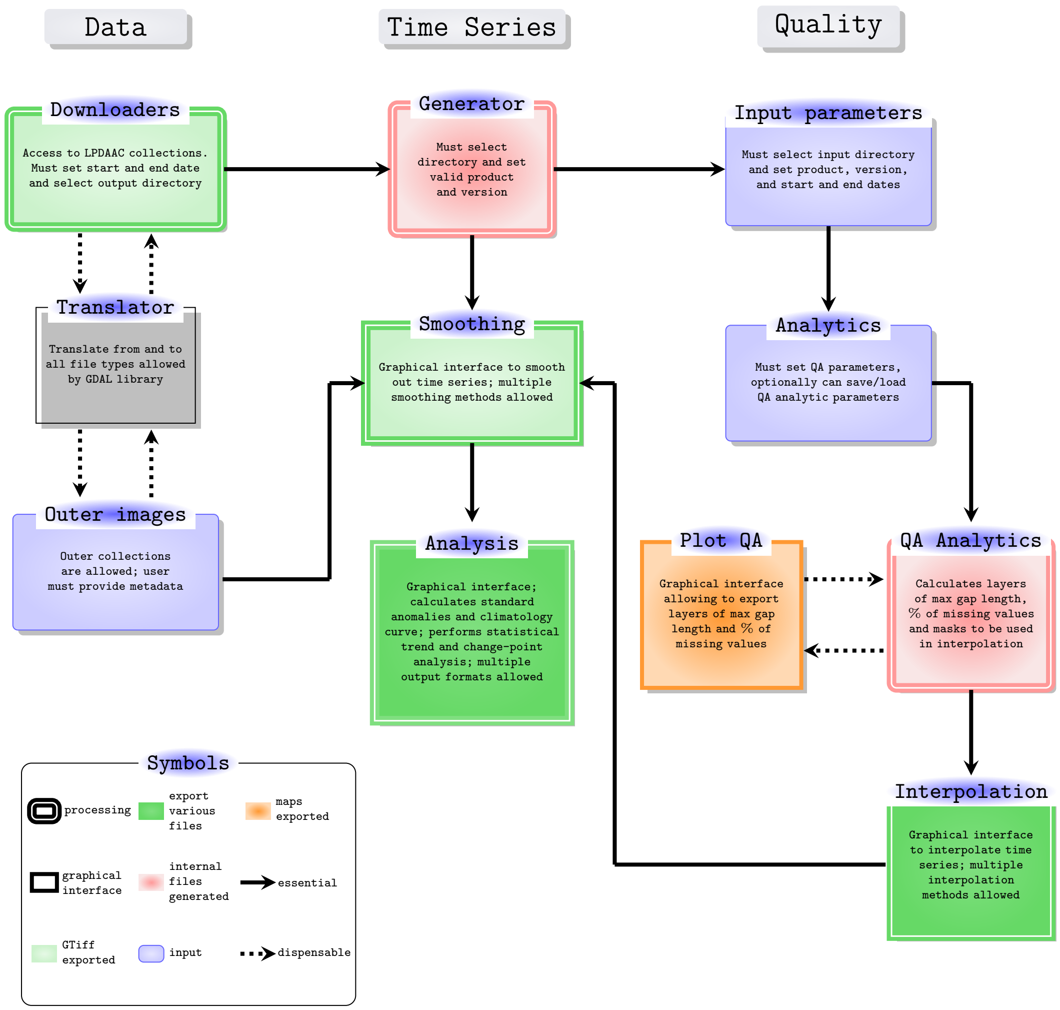


Figure 2: TATSSI's flowchart.

```
fnames = glob(fnames)
# Sort the list
fnames.sort()

for fname in fnames:
    # Get EVI QA SubDataset (SD) which is index 1
    sds = get_subdatasets(fname)
    vi_qa_sds = sds[1][0]

    # Set output file
    directory_name = os.path.dirname(os.path.abspath(fname))
    output_fname = os.path.join(directory_name,
                       os.path.basename(fname)[:-3] + 'EVI.tif')

    # Extract to a GeoTiff file
    Translate(vi_qa_sds, output_fname, 'GTiff')
```

2.2 Downloaders

Download most products available at the Land Processes Distributed Active Archive Center (LP DAAC). Figure 3 provides an overlook of this module. Beware that LP DAAC servers are usually offline on Wednesdays. Also, keep in mind that a successful download depends on a properly working internet connection.

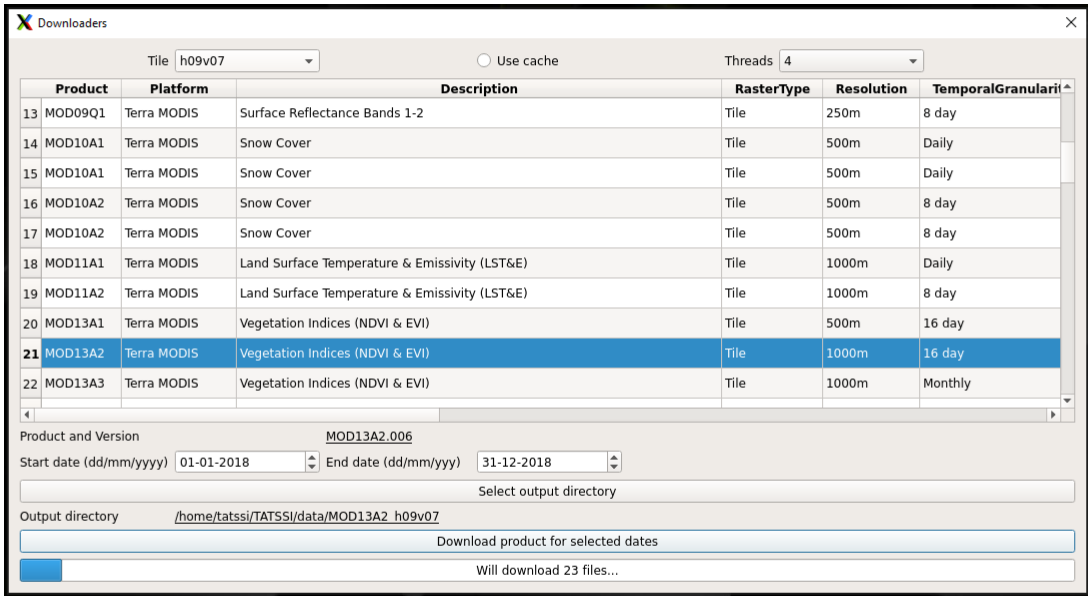


Figure 3: Downloaders module acquiring the 23 MOD13A2.006 files of the tile h09v07 from 2008.

2.2.1 Tile

Select an area of study, valid areas correspond to tiles following MODIS Land sinusoisal tile grid.

2.2.2 Use cache

Allows use of previously defined download setup stored at cache memory. The use of this parameter may come in handy when, for any reason, a download is interrupted.

2.2.3 Threads

To reduce downloading time, can select up to 6 threads.

2.2.4 Start date

Start date of products to download.

2.2.5 End date

End date of products to download.

2.2.6 Select output directory

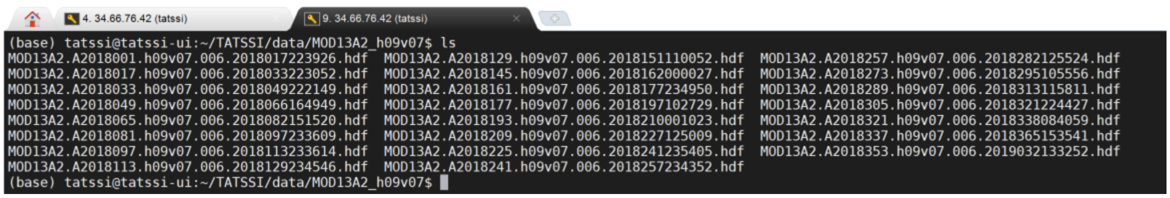
Window pops up allowing selection of a directory where files from chosen product will be saved.

2.2.7 Download product for selected dates

Once the above parameters have been entered and a **Product** selected (from products table), a click on this button will start downloading process.

2.2.8 Example

Figure 3 shows the setup for downloading the 23 files of the MOD13A2.006 product corresponding to tile h09v07 from 2018 using 4 threads. Figure 4 shows the output of this download; note that these files were saved at the directory: `/home/tatssi/TATSSI/data/MOD13A2_h09v07`.



```
(base) tatssi@tatssi-ui:~/TATSSI/data/MOD13A2_h09v07$ ls
MOD13A2.A2018001.h09v07.006.2018017223926.hdf  MOD13A2.A2018129.h09v07.006.2018151110052.hdf  MOD13A2.A2018257.h09v07.006.2018282125524.hdf
MOD13A2.A2018017.h09v07.006.2018033223052.hdf  MOD13A2.A2018145.h09v07.006.2018162000027.hdf  MOD13A2.A2018273.h09v07.006.2018295105556.hdf
MOD13A2.A2018033.h09v07.006.2018049222149.hdf  MOD13A2.A2018161.h09v07.006.2018177234950.hdf  MOD13A2.A2018289.h09v07.006.2018313115811.hdf
MOD13A2.A2018049.h09v07.006.2018066164949.hdf  MOD13A2.A2018177.h09v07.006.2018197102729.hdf  MOD13A2.A2018305.h09v07.006.2018321224427.hdf
MOD13A2.A2018065.h09v07.006.2018082151520.hdf  MOD13A2.A2018193.h09v07.006.2018210001023.hdf  MOD13A2.A2018321.h09v07.006.2018338084059.hdf
MOD13A2.A2018081.h09v07.006.2018097233609.hdf  MOD13A2.A2018209.h09v07.006.2018227125009.hdf  MOD13A2.A2018337.h09v07.006.2018365153541.hdf
MOD13A2.A2018097.h09v07.006.2018113233614.hdf  MOD13A2.A2018225.h09v07.006.2018241235405.hdf  MOD13A2.A2018353.h09v07.006.2019032133252.hdf
MOD13A2.A2018113.h09v07.006.2018129234546.hdf  MOD13A2.A2018241.h09v07.006.2018257234352.hdf
(base) tatssi@tatssi-ui:~/TATSSI/data/MOD13A2_h09v07$
```

Figure 4: Console look of output of Example above.

3 Time Series menu

All files distributed through the LP DAAC are available on a file per time step, either daily, 8-day, 16-day or monthly data. Since this is not the most efficient way to gather data for time series analysis when per-pixel operations for every time-step are required, TATSSI will generate time series in an internal format as a Cloud Optimized GeoTIFF (COG).

Generation, smoothing and analysis of time series of satellite images are allowed in this menu. Generation has to do with the creation of time series as a Cloud Optimised GeoTIFF (COG) masked by the user-defined QA Scientific Datasets (SDS) parameters selection. The smoothing module allows for the application of some smoothing methods to long time series of satellite imagery (datacubes). The analysis module provides routines allowing for calculation of standard anomalies, statistical trend analysis, STR decomposition of time series and estimation of inter annual climatology.

For TATSSI to process files provided from a source different to LP DAAC, the user must ensure that every file contains a date as a `metadata` based on filename. An example of how to add this `metadata` can be found in the Appendix.

3.1 Generator

Several of the MODIS and VIIRS products distributed on the LP DAAC are in HDF4 or HDF5 formats, given the hierarchical nature of these formats. This means that a single file can contain multiple subdatasets (referred hereafter as Scientific Datasets (SDS)).

TATSSI will create a time series for a product where the default behavior is:

1. For every file in the data directory that matches the product selected by the user:
   - Will import each band or SDS to the internal TATSSI format (Cloud Optimized GeoTiff)
2. For each QA layer associated with the product:
   - Will import it to the internal TATSSI format and perform the QA decoding
3. For each band or SDS and associated QA layers:
   - Will create GDAL VRTs layer stacks

User must set the following parameters:

3.1.1 Select data directory

Window pops up allowing selection of a directory where LP DAAC files have been previously stored. Once a proper data directory has been selected, Select spatial subset and Format | Driver | Extension buttons are

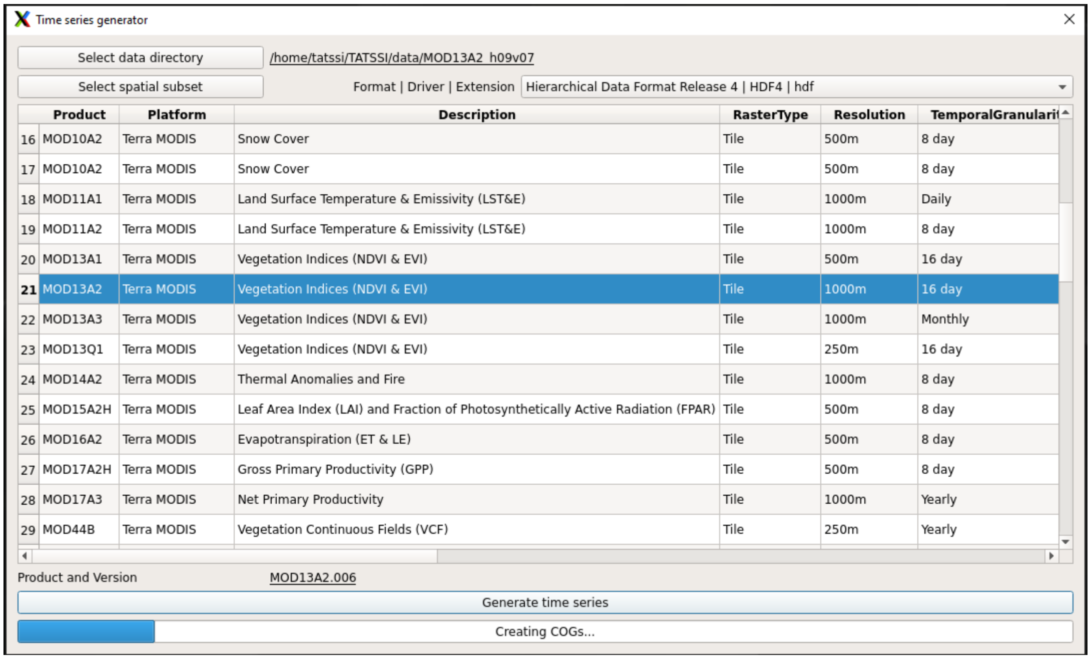


Figure 5: Time series generator decoding QA flags for MOD13A2.006 images from 2008.

habilitated.

3.1.2 Select spatial subset

Optional. Allows for setting a spatial subset used as a framework to generate a time series. When pressing this button a new window pops up, see Figure 6. Select the *Zoom to rectangle* button. On the shown image click and draw to define a rectangle; note that the image will adjust and only show the selected rectangle (spatial subset). To define a different rectangle on the selected product, press *Home* button and repeat previous step. Once the desired rectangle has been set, press the button *Get extent for subsetting* to indicate that the extent of the defined rectangle must be used to generate a time series. Close window.

3.1.3 Format | Driver | Extension

Optional. Allows selection of *Format* and GDAL Raster Drivers using the *Extension* of the first file in the selected data directory. Useful when files do not have a consistent extension or corresponding format.

3.1.4 Generate time series

Once a valid data directory, **Product** (from products table), and/or Select spatial subset, and/or Format | Driver | Extension parameters, have been set, a click on this button will start generating COG files. Note that this procedure generates as many time series as layers exist in the chosen product. For instance, pressing this button for the MOD13A2.006 product will result in 12 time series, see Example below for more details.

3.1.5 Example

This is a follow up to Example in Downloaders. In order to analyze the MOD13A2.006 files, TATSSI has to create an *xarray* to operate on these files as a large datacube. Figure 5 shows a setup to perform this process. Note that the default format/extension (HDF4) was utilized and the entire tile h09v07 was considered for processing. After decoding the QA flags, at the directory `/TATSSI/data/MOD13A2_h09v07`, in addition to the MOD13A2.006 files, there are now 12 new subdirectories; each subdirectory corresponds to a layer

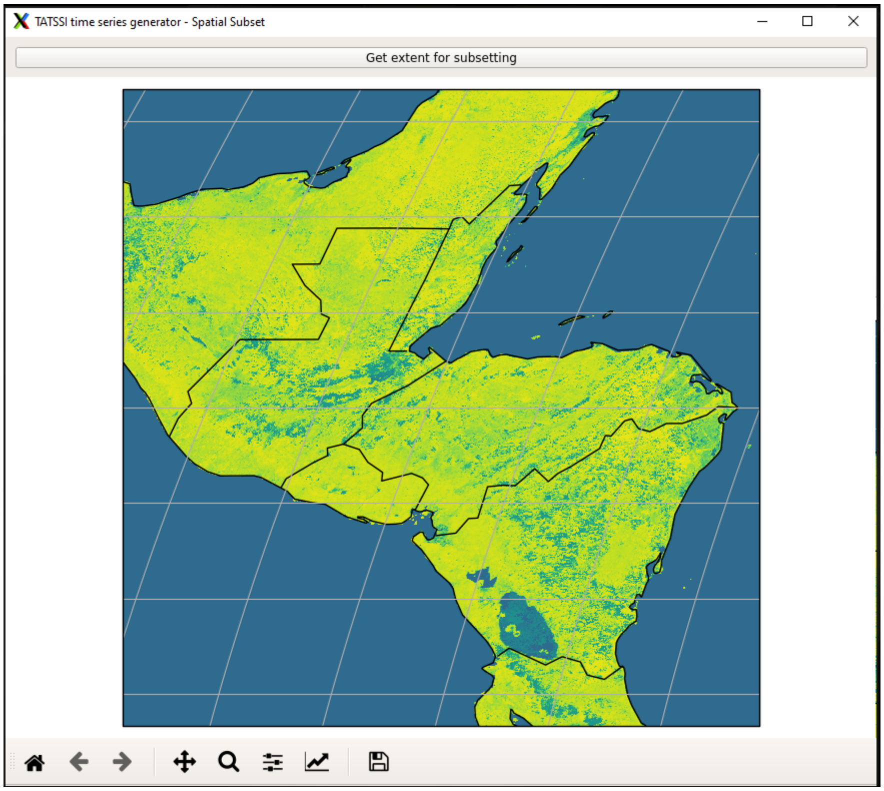


Figure 6: Selecting a spatial subset for MOD13A2.006 product of the tile h09v07.

of information from the MOD13A2.006 product. Within each subdirectory there are 23 Cloud Optimized GeoTIFFs (COGs); see Figure 7.

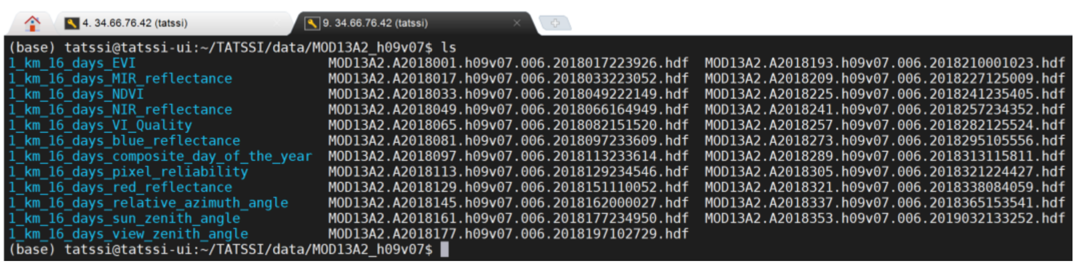


Figure 7: Console look of output of Example above; subdirectories names are shown in blue.

3.2 Smoothing

Window pops up allowing selection of a file containing an interpolated time series; this file is in a directory whose name is *interpolated*, see Figure 8 for an example; see Interpolation in QA Analytics for further details on how to intepolate a time series with TATSSI.

Once a valid interpolated time series has been chosen, a new window pops up and then the following parameters must be set:

3.2.1 Data variables

Scroll allowing selection of variable to smooth.

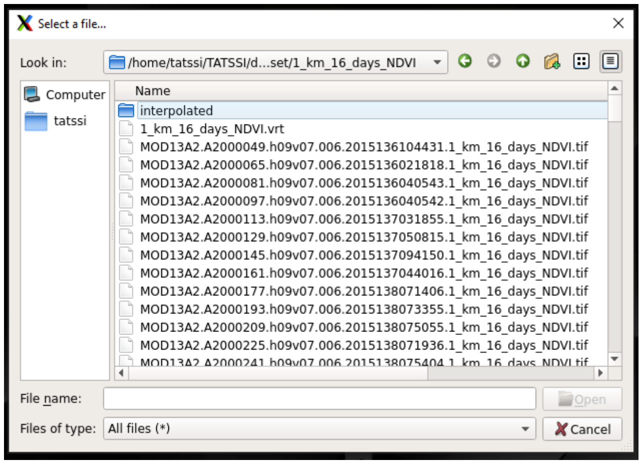


Figure 8: Accessing to interpolated time series for further smoothing.

3.2.2 Dates

Scroll allowing selection of files by date of acquisition. Files content (images) are displayed in the left visualization panel. Dates of acquisition are automatically uploaded and correspond to dates from the time span of the product chosen in Data variables.

3.2.3 Smoothing methods

These methods are based on functions from Python's statsmodel.tsa.api and numPy libraries.

smoothn provides a fast, automatized and discretized, robust spline smoothing for 1-dimensional data. Technical details about this method can be found at Garcia (2010).

3.2.4 Smoothing factor

Numeric value. Caution is advised when selecting this parameter as a small value yields a *rough* smoothing, which resembles linear interpolation, whereas a large value may create a *too wiggly* fit.

3.2.5 Smooth

Once valid smoothing parameters have been entered, a click on this button will start smoothing process, pixel-wise, and on the entire data cube.

3.2.6 Graphical controller buttons

Note that when hovering over left and right graphical panels, at the right bottom corner of the screen, coordinates and pixel values and date and time series values are displayed. By clicking on the following buttons, user can improve plots shown in graphical panels.

- Home. Reset original view, this has effect on either the left (resetting original map) or right (resetting previously chosen time series plot) panels.
- Left arrow. Back to previous view.
- Right arrow. Forward to next view.
- Crossroads. Left click on mouse button allows for panning on axes, right click on mouse button allows for zooming in/out; this applies to both left and right panels.
- Magnifying glass. Zoom to rectangle.

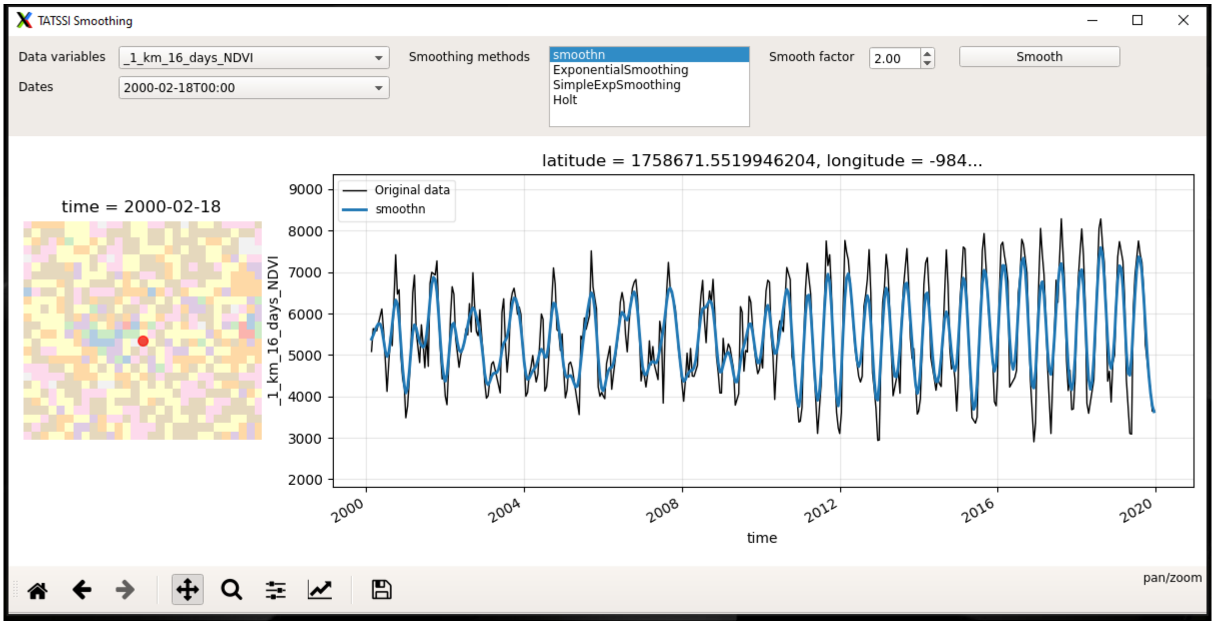


Figure 9: smoothn method, with factor 2, applied to NDVI layer of MOD13A2.006_subset images from 2000 to 2019. Pastel_r color palette and diamond marker style are shown in this plot.

- Configure subplots. Window pops up allowing to set margins of graphical layout: Border parameters (top, bottom, left, right) and Spacings (horizontal and vertical). Should the user find it useful, it is possible to *Export values* of these parameters. *Reset* allows to recover original configuration.
- Edit axis, curve and image parameters. Scroll menu pops up allowing to edit either left (map) or right (time series plot) graphical panels. For left and right panels can edit *Axes*, *Curves* and *Images*. In *Axes* can edit *Title*, *X-Axis* (farmost *Left* and *Right* values), *Y-Axis* (farmost *Bottom* and *Top* values), *Label* and *Scale*. In *Curves* can edit *Label*, *Line* (style, width and RGBA Color), *Marker* (style, size, face and edge RGBA colors). For left panel can edit *Images* parameters such as *Label*, *Colormap*, *Min* and *Max* values and *Interpolation* method (not to be confused with methods in Interpolation).
- Save the figure. Saves the current plot into a local image. Allowed types are: `EPS`, `JPEG/JPG`, `PGF`, `PDF`, `PNG`, `PS`, `RAW/RGBA`, `SVG/SVGZ`, `TIF/TIFF`.

3.2.7 Example

TATSSI comes with the folder `/TATSSI/data/MOD13A2.006_subset` which contains 457 MOD13A2.006 files corresponding to a spatial subset of the tile h09v07 from 2000 to 2019. In order to apply the routines of the smoothing module to this dataset an interpolation process must have been applied previously; see this Example for details on this process. Figure 8 shows how to access to the interpolated NDVI layer of the MOD13A2.006_subset. Figure 9 shows a further smoothing to a pixel of this NDVI layer. After pressing the *Smooth* button the resulting smoothed time series is saved at the same directory where the interpolate time series is located.

3.3 Analysis

This menu provides routines for performing basic analysis such as calculation of standard anomalies, statistical trend analysis, STR decomposition for time series at the pixel level, and inter annual climatology. Depending on the analysis its result can be exported as a series of COG files or as an image. The first step to analyze a time series consists of providing either an interpolated or a smoothed time series. Once a suitable time series has been chosen, a window pops up allowing for the selection of a series of parameters (described below) and showing two main visualization panels, see Figure 10. The left panel shows two images on the top and a climatology curve on the bottom; these images allow to visualize the behavior of the selected variable at two different dates within the year of interest. The right panel shows a seasonal, trend and residuals (STR) decomposition: (from top to bottom) the raw time series with total number of peaks and valleys as well as a summary of a trend analysis based on the Mann-Kendall test statistic; estimated trend via moving average with additional change point analysis based on the `R` package changepoint by Killick et al. (2016); estimated seasonal component; and residuals. The output of any of the analysis performed by this module is saved at the directory where the analyzed product was sourced from.

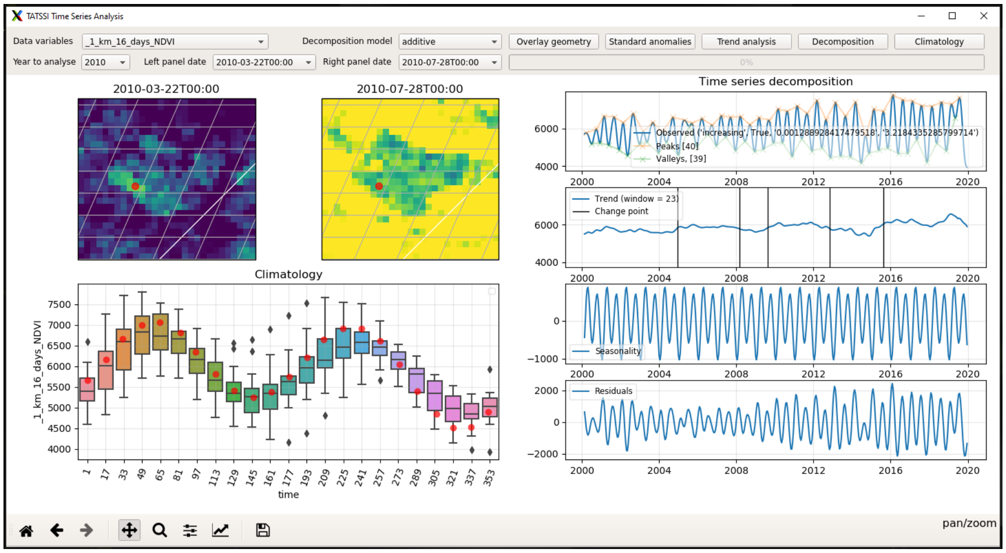


Figure 10: Analysis of a 16-days NDVI time series from 2000 to 2019. The annual climatology curve for the selected pixel (red dot in top left panels) shows a bimodal behavior. Also, from day 1 to 257, the chosen time series is slightly over the climatology's median. Although visually does not seem to be abnormalities in the seasonal component, a statistically significant positive trend is found at the raw data, as well as a series of significant change points in the trend at 2004, 2009, 2012 and 2017.

3.3.1 Data variables

Scroll allowing selection of variable to analyze.

3.3.2 Year to analyse

Scroll allowing selection of a year (within the time span of the series). The selection of a year has effect on the two images in the left top panel and in a subsequent Standard anomalies analysis.

3.3.3 Left panel date

Scroll allowing selection of a date/image within the time span of the series.

3.3.4 Right panel date

Scroll allowing selection of a date/image within the time span of the series.

3.3.5 Decomposition and BW

Scroll allowing selection of additive or multiplicative STR decomposition. BW stands for *bandwidth* used in trend estimation. See more details at Decomposition section below.

3.3.6 Overlay geometry

This button allows to upload a shapefile to be overlapped onto the image of the chosen data variable displayed in the left top pannels. This shapefile's projection and that of the data variable must be the same. This function improves visual inspection of the data variable, however, the final analysis will be performed on the entire original product.

3.3.7 Standard anomalies

Calculates anomalies as follows: For a time series $(x_t)_{t=1,\ldots,(P\times n)}$ with P periods (years) and n observations in each period, let μ_t and σ_t be, respectively, the overall mean and standard deviation across the P periods/years of the time series at a given time-point/date t:

$$\mu_t = \frac{1}{P}\sum_{k=0}^{P-1} x_{t+k\times n}, \quad \sigma_t = \sqrt{\frac{1}{P}\sum_{k=0}^{P-1}(x_{t+k\times n}-\mu_t)^2}, \quad t=1,\ldots,n.$$

For any time-point/date t in the period k, its corresponding anomaly is defined as:

$$A_{t,k} = (x_{t+k\times n}-\mu_t)/\sigma_t, \quad t=1,\ldots,n, \quad k=0,\ldots,P-1.$$

When this button is pressed the standard anomalies of the Year to analyse are calculated and saved.

3.3.8 Trend analysis

Performs a trend analysis on the selected time series utilizing the Mann-Kendall test. Briefly, and following Mann (1945) and Chapter 4 and 5 in Kendall (1962), for the time series $X_1,\ldots,X_n$ the statistic

$$S_n = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \operatorname{sign}(X_j - X_i),$$

where

$$\operatorname{sign}(X_j - X_i) = \begin{cases} +1 & \text{for } X_j - X_i > 0 \\ -1 & \text{for } X_j - X_i < 0 \end{cases}$$

can be used to determine whether there exist a trend in the time series; large values of $|S_n|$ provide evidence against the null hypothesis of no trend. In order to calculate the p-value of this null hypothesis it is convenient to utilize the normal approximation of the following statistic

$$T_n = \begin{cases} \frac{S_n-1}{\sqrt{\sigma_n}} & \text{for } S_n > 0 \\ \frac{S_n+1}{\sqrt{\sigma_n}} & \text{for } S_n < 0, \end{cases}$$

where $\sigma_n^2 = n(n-1)(2n+5)/18$.

In this module the null hypothesis $H_0 : |T_n| > |\widehat{T}_n|$ is used; the corresponding *p-value* is equal to $2\,P\{T_n > |\widehat{T}_n|\}$. When p-value $< \alpha$, for some significance level α, then the null hypothesis is rejected; $\alpha = 0.05$ is employed in this module. To ease interpretation, the variable `trend` is defined as follows:

$$\text{trend} = \begin{cases} -1 & \text{if } H_0 \text{ is rejected and } T_n < 0 \\ 0 & \text{if } H_0 \text{ is not rejected} \\ +1 & \text{if } H_0 \text{ is rejected and } T_n > 0 \end{cases}.$$

When this button is pressed, the statistic T_n, the corresponding p-value and `trend` are calculated and saved.

NOTE: Sometimes at the console the following `Warning:Metadata exceeding X bytes cannot be written into GeoTIFF. Transferred to PAM instead` may be displayed. This message means that the output content is large enough and that the metadata will be allocated in a companyon file.

3.3.9 Decomposition

When *additive* is selected in [Decomposition model], then the model

$$X_t = T_t + S_t + e_t$$

is employed, whereas

$$X_t = T_t \times S_t \times e_t$$

is used when *multiplicative* is chosen. The following explanation holds for a single time series but when this button is pressed this decomposition is performed on each time series in the data cube.

The trend component is estimated utilizing a moving average with a fixed bandwidth. The user can set the bandwidth value equal to a number in the set $\{3, 4, \ldots, T\}$ where T is the number of observations by period/year; by default, bandwidth is equal to the number of observations by period.

After removing the estimated trend from the original time series, the seasonal component is calculated by averaging, for each date, over all years. Finally, the residuals result from removing trend and seasonal components from the original time series. Note that the *trend*, *seasonal* and *residual* layers just described are saved.

3.3.10 Climatology

In the left bottom panel a climatology curve is shown. For each date, this curve shows (excluding outliers) the minimum, 25%, 50% and 75% quartiles as well as the maximum. The 50% quartile is also known as the median. These statistics correspond to the distribution of the values of the selected product at each date over all years. Some time series will behave off with respect to this climatology curve. This time series may exhibit outliers from below (values less than the minimum of climatology curve) or from above (values greater than the maximum of the climatology curve).

By pressing this button layers for the minimum, 25%, 50% and 75% quartiles as well as the maximum are calculated. Also, layers of outliers from below and above (for each date within the time span) are calculated. Along with these layers, TATSSI calculates mean, and standard deviation (sd). Lastly, when this button is pressed, Standard anomalies, for each year in the time series, are also calculated and saved.

3.3.11 Example

Figure 10 shows an overview of TATSSI's analysis module applied to the MOD13A2.006_subset time series imagery. When Standard anomalies, Trend analysis, Decomposition and Climatology buttons are pressed all the products shown in Figure 11 are saved.

4 Quality menu

This menu provides routines for screening clouds, cloud shadows and any artifacts using Quality Assessment (QA) flags. Through decoding of QA Scientific Datasets (SDS) associated with diverse MODIS & VIIRS datasets, time series are created, masked and saved as Cloud Optimised GeoTIFFs (COGs). Interpolation methods are provided to fill in the resulting gaps.

4.1 Analytics

Window pops up and then the following parameters must be set:

4.1.1 Select input data directory

Window pops up allowing the selection of a directory containing time series generated in TimeSeries/Generator, see Generator for more details. An error will be produced if a time series has not been previously generated.

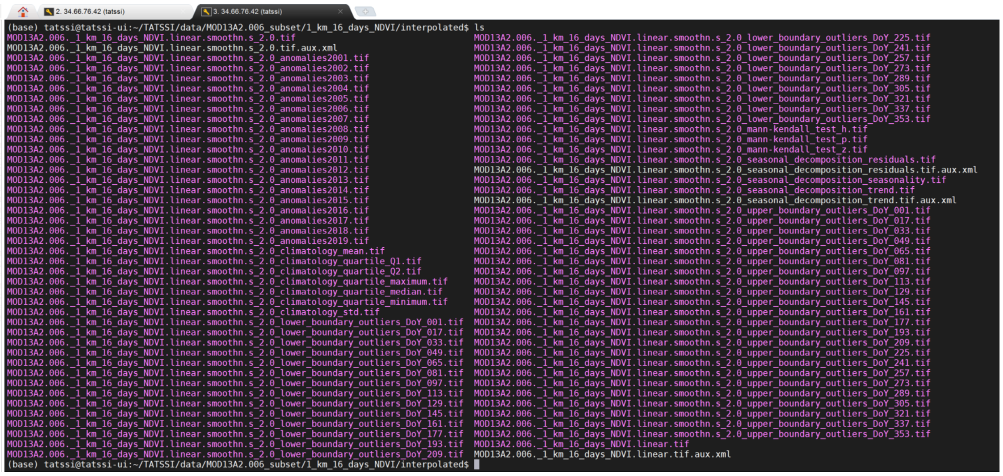


Figure 11: Console look of output of Example above; all the files resulted from the basic statistical analysis applied to the MOD13A2.006_subset are shown in purple.

4.1.2 Product

Name of product to analyze.

4.1.3 Version

Version of product to analyze.

4.1.4 Start date

Start date of time series to analyze.

4.1.5 End date

End date of time series to analyze.

4.1.6 Get time series QA definitions

Once valid Product, Version, Start date and End date have been chosen, a click on this button uploads product's QA definition.

4.1.7 QA definition

Allows selection of QA flags of chosen product. A single product can have more than one QA flag. For instance, the MOD13A2.006 at 1km spatial and 16-days time resolution comes with 2 QA flags (_1_km_16_days_VI_Quality and _1_km_16_days_pixel_reliability). These QA flags are summarized in a table displaying information such as **QualityLayer**, **Name**, **Value**, **Description** and **Acceptable**.

4.1.8 QA parameter name

This scroll displays the QA parameter name associated with the **QA definition** chosen above. Once a QA parameter has been chosen, at the right panel (Description) a list containing a complete classification of the chosen QA parameter is displayed. For instance, using the MOD13A2.006 aforementioned, if _1_km_16_days_pixel_reliability has been chosen in **QA definition** then **QA parameter name** displays MODLAND as the only option and Description shows 4 pixel reliability categories: Good data, Marginal data, Snow/Ice andx Cloudy data. In general, when selecting QA parameter categories (left mouse click on a

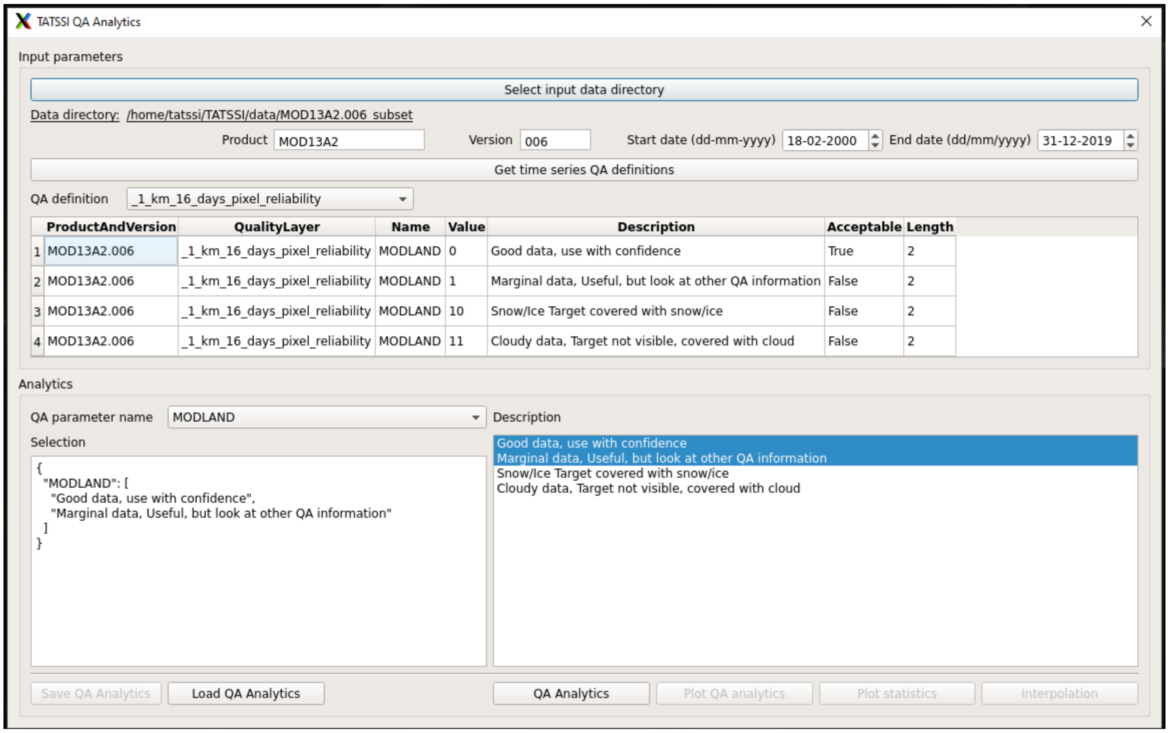


Figure 12: QA Analytics module applied to some MOD13A2.006 files from 2000 to 2019.

category allows selection) some internal code is generated and displayed in Selection (left panel); this internal code can be saved for future analyses, see Save QA Analytics.

4.1.9 QA Analytics

A click on this button triggers calculation of the percentage of data available and the max-gap length of the chosen **Product** as a function of the QA parameter categories previously selected. Observe that a click on this button also habilitates the buttons Save QA Analytics, Plot QA analytics, Plot statistics and Interpolation.

4.1.10 Save QA Analytics

Saves the current selection for the chosen QA definition and corresponding QA parameter name into a JSON file. This file can be used to apply the same QA settings to a different time series.

4.1.11 Load QA Analytics

Load QA analytics parameters previously saved.

4.1.12 Plot QA Analytics

Window pops up showing a map for the *% of data available* (left panel) and another with the *Max gap length* (right panel). If no QA parameter category has been selected then these maps will be empty. Note that when hovering over any of the maps, at the right bottom corner, y, x, (LAT, LON) coordinates, and [pixel value] are displayed. Should the user not be satisfied with the quality/amount of data available, recall that it is always possible to go one step back, select more/different QA parameter categories, and recalculate the QA Analytics.

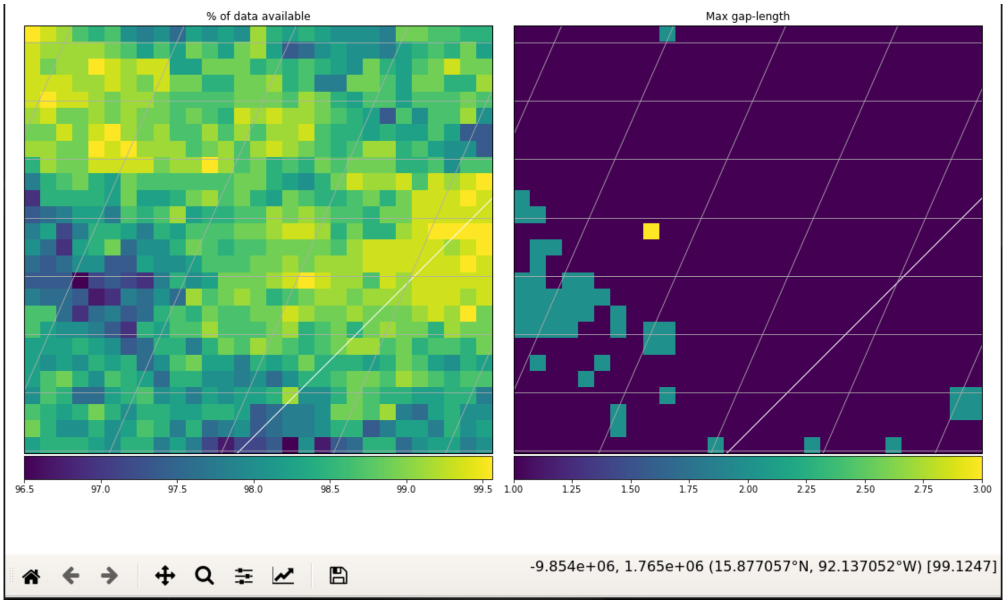


Figure 13: Plot QA Analytics applied to some MOD13A2.006 files from 2000 to 2019. This time series missed at most 4% of its observations when using the Good and Marginal data QA flags; the max-gap length ranges from 1 to 3 observations.

4.1.12.1 Plot QA graphical controller buttons

For Home, Left arrow, Right arrow, Crossroads, Magnifying glass, Configuring subplots and Save the figure see Graphical controller buttons

- Edit axis, curve and image parameters. Scroll menu pops up allowing to edit *% of data available*, *Max gap length*, *color palette left*, *color palette right*. When either of these is chosen then *Axes* and *Images* can be editted, see Graphical controller buttons for more details.

4.1.13 Plot statistics

Shows the histograms for the *% of data available* and the *Max-gap length* statistics.

4.1.14 Interpolation

This window presents 3 graphical panels, see Figure 14, and allows for selection of a few interpolation methods to fill time series gaps. The top panel shows 2 plots, the original image (left) and the image resulting from masking according to QA parameter categories selected above. The bottom panel will show a time series (Original data, dotted-line) and its interpolated versions.

4.1.14.1 Data variables

Scroll allowing selection of any of the times series associated with the chosen product. For instance, for a MOD13A2.006, this scroll will show 12 names, corresponding to each one of the 12 layers of this product.

4.1.14.2 Dates

Scroll allowing selection of any of the dates included in the time series under analysis. Both left and right top panels change as a function of the selected date. These panels can be editted and, for instance, different color palettes applied to each map, see Interpolation graphical controller buttons for more details.

Figure 14: Linear and spline interpolation applied to some MOD13A2.006_subset files from 2000 to 2019. Images in top panel are from 26-06-2010. Spectral color palette and black diamond marker style are shown in this plot.

4.1.14.3 Interpolation methods

Selecting at least one of these methods habilitates the button Interpolate. Note that the combined selection of any interpolation method and a click on any pixel of the maps from the top panel (the graphical controller button **Crossroads** allows for selection of pixels) adds a new curve to the bottom panel. This curve is the interpolated version of the masked time series resulting from the **QA Analytics**. Note that, by default, blue points are also added to this plot. These points indicate the observations (in the time series) kept by the user QA selection, that is, these are the points left in the time series after applying **QA Analytics**.

TATSSI stores time series objects in a Python xarray which allows for different interpolation methods using the interpolate_na class. These are some of these methods:

- linear. Standard linear interpolation is used as gap-filling method; See the numpy interpolation linear method for more details.
- nearest, quadratic and cubic. These are methods from the inter1d class.
- spline. Fills missing values following a standard spline fitting. See scipy.interpolate for more details.

4.1.14.4 Interpolation graphical controller buttons

For Home, Left arrow, Right arrow, Crossroads, Magnifying glass, Configure subplots and Save the figure see Graphical controller buttons.

- Edit axis, curves and image parameters. Scroll menu pops up allowing to edit three graphical panels. For details on edition of *Axes*, *Curves* and *Images* see Graphical controller buttons.

4.1.14.5 Interpolate

A click on this button will apply the chosen interpolation methods to the masked time series of satellite imagery. It is advised that a healthy amount of time be invested in exploring the effects of different interpolation methods on different/several pixels across the analyzed images. Note that a progress bar is habilitated to indicate the status of the interpolation process; the more interpolation methods are chosen the more time the interpolation process will last.

4.1.15 Example

Figure 12 shows the quality analysis of the MOD13A2.006_subset files when *Good data* and *Marginal data* are selected from Description box. Once the button *QA Analytics* is pressed then the buttons *Save QA Analytics*, *Plot QA Analytics*, *Plot statistics* and *Interpolation* are habilitated. Also as a result of pressing *QA Analytics*, TATSSI calculates the *% of data available* and *Max gap-length* in the time series MOD13A2.006_subset. These statistics are shown in Figure 13. When only the *Good data* flag is selected then the resulting time series will tend to have more gaps. The submodule *Interpolation* provides a few methods to fill in these gaps. Figure 14 shows the application of linear and spline interpolation to a pixel from the MOD13A2.006_subset.

5 Appendix

5.1 An example to add `metadata` to external files

This example, written in Python, works for files with the following name structure:

`productName_YYYY_DDD.Nadir_Reflectance_bandNumber.tif`

Basically, the user must get `date`, as given in name file, and create a character string which will be used as `metadata`:

```python
def add_dates(data_dir):
    """
    For every file in data dir, add the date as metadata
    based on its filename. This works only for files like:
    MCD43A4_2018345.Nadir_Reflectance_Band1.tif
    """
    # Get all the GeoTiff files in data dir
    fnames = os.path.join(data_dir, '*.tif')
    fnames = glob(fnames)
    fnames.sort()

    for fname in fnames:
        # In this particular case, date is YYYYDDD
        str_date = os.path.basename(fname)
        str_date = str_date.split('.')[0]
        str_date = str_date.split('_')[1]

        # Convert YYYYDDD to YYYY-MM-DD
        fmt = '%Y%j'
        _date = datetime.strptime(str_date, fmt)
        str_date = _date.strftime('%Y-%m-%d')

        add_date_to_metadata(fname, str_date)
        LOG.info(f"Metadata added for {fname}")
```

Here is the auxiliary function `add_date_to_metadata`:

```python
def add_date_to_metadata(fname, str_date):
    """
    Adds date to metadata as follows:
        RANGEBEGINNINGDATE=YYYY-MM-DD
    :param fname: Full path of files to add date to its metadata
    :param str_date: date in string format, layout: YYYY-MM-DD
    """

    # Open the file for update
    d = gdal.Open(fname, gdal.GA_Update)

    # Get metadata
    md = d.GetMetadata()

    # Add date
    md['RANGEBEGINNINGDATE'] = str_date

    # Set metadata
    d.SetMetadata(md)

    d = None
    del(d)
```

References

Garcia, D. (2010). Robust smoothing of gridded data in one and higher dimensions with missing values. *Computational statistics & data analysis*, 54(4):1167–1178.

Kendall, M. G. (1962). *Rank correlation methods*. Griffin, 4 edition.

Killick, R., Haynes, K., and Eckley, I. A. (2016). *changepoint: An R package for changepoint analysis*. R package version 2.2.2.

Mann, H. B. (1945). Nonparametric tests against trend. *Econometrica: Journal of the Econometric Society*, pages 245–259.