EOF-based Time Series Reconstructor

Release 4.0.4

Dec 14, 2020

Contents

1 Installation Instructions **3**
- 1.1 Installation of QGIS Plugin . . . 3

2 User Guide **5**

3 Example **9**
- 3.1 Get the data . . . 9
- 3.2 Try out the tool . . . 10
- 3.3 Try out different settings . . . 10

4 API **13**

Python Module Index **17**

Index **19**

The EOF-based Time Series Reconstructor software package is a **QGIS plugin** that provides a EOF-based reconstruction tool for **filling gaps, caused by e.g. clouds, in geophysical time series, like ocean temperature**. It is developed specifically for **oceanography**.

The software is based on DINEOF: a FORTRAN tool in continuous development since 2008.

The software has now been developed in the open source environment to encourage further development of the tool.

The **instruction pages** can be found at <https://eof-reconstructor.readthedocs.io>.

The code **repository** can be found at https://bitbucket.org/kul-reseco/dineof.

PLEASE GIVE US CREDIT

When using the EOF-based Time Series Reconstructor, please use the following citation:

Alvera-Azcárte, A. Barth, A., Crabbé, A., Somers, B. (2020). EOF-based Time Series Reconstructor QGIS Plugin (Version x.x) [Software]. Available from https://bitbucket.org/kul-reseco/dineof.

ACKNOWLEDGEMENTS

The software and user guide are based on DINEOF 4.0 developed by Alvera-Azcárte and A. Barth (ULG Liège): available at https://github.com/aida-alvera/DINEOF.

The software also makes use of the QGISPluginSupport python packages developed by Benjamin Jakimow (HU Berlin): https://bitbucket.org/jakimowb/qgispluginsupport.

This revision into a QGIS plugin is funded primarily through BELSPO (the Belgian Science Policy Office) in the framework of the STEREO III Programme – Project LUMOS - SR/01/321.

Logo’s were created for free at https://logomakr.com.

SOFTWARE LICENSE

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License (COPYING.txt). If not see www.gnu.org/licenses.

For issues, bugs, proposals or remarks, visit the issue tracker.

CHAPTER 1

Installation Instructions

For issues, bugs, proposals or remarks, visit the issue tracker.

1.1 Installation of QGIS Plugin

The plugin is **not** available in the QGIS Python Plugins Repository, as it contains executables. For the same reason, it cannot be installed from zip.

However, there is an easy workaround. From the bitbucket page, download the latest stable distribution (*eof-reconstructor-x-qgis.zip*) and unpack it in the qgis plugin folder:

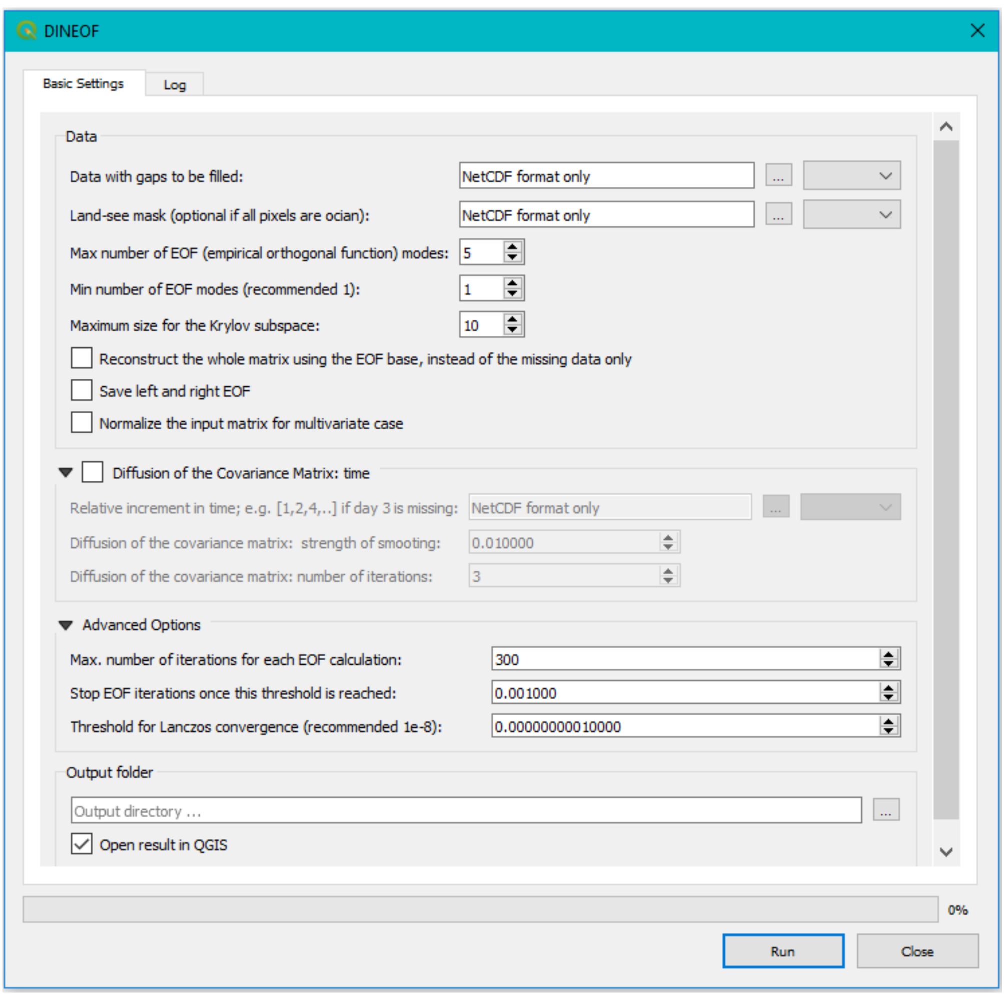
```
Windows: C:\Users\USER\AppData\Roaming\QGIS\QGIS3\profiles\default\python\plugins

Linux: /home/USER/.local/share/QGIS/QGIS3/profiles/default/python/plugins
```

Note: The plugin is build for QGIS Version 3.16 and up. The plugin has been tested on Windows 10.0, Ubuntu 16.04 and Raspbian GNU/Linux 10 (buster).

CHAPTER 2

User Guide



1. Select a NetCDF file with the data with gaps to be filled.
2. Optionally, select a NetCDF file with a land-see mask. If no mask is selected, all pixels are assumed to be ocean.
3. Set the maximum number of EOF modes to compute. Only the optimal number of EOFs + 3 will be computed for robust optimum detection. This number cannot be higher than the temporal size of the data minus 5. Start high and adjust down for subsequent reconstructions. If the max number of modes have been used, the algorithm probably needs more information.
4. Set the minimum number of EOF modes to compute. 1 is recommended.
5. Set the maximum size of the Krylov subspace. This number must be smaller or equal to the temporal size of teh data, and must be higher or equal to the maximum number of EOF modes plus 5.
6. Optionally, choose to reconstruct the entire image, and not only the missing data.
7. Optionally, save the left and right EOFs to file.
8. Optionally, normalize the input matrix for the multivariate case.
9. Optionally, choose a time vector for the diffusion of hte covariance matrix.
   1. The time file should also be in NetCDF file format, and specified as the relative increment in time between the individual images. For example [1,2,3,4,. . .] if all data is 1 day apart; [1,2,4,. . .] if day 3 is missing, . . .
   2. Optionally, set the strength of the filter.
   3. Optionally, set the number of iterations for the filter.
10. Set the advance functions, if you know what you are doing:
    1. You can set the maximum number of iterations for each EOF calculation. Use large umber and only increase if this number is reached for each EOF.
    2. You can change the threshold, for which EOF iterations stop, once rms(successive reconstruction) / std(existing data) is below this threshold.
    3. You can set the threshold for the Lanczos convergence. 1e-8 is a good number here.
11. Optionally, choose a folder for all results.

Once the processing is done, the the log tab in the widget shows as summary of the process. This summary can also be found in the file *log.txt*:

```
*********************************************************************
Numerical data read
You entered the values:
number of EOF modes you want to compute        5
maximal size for the Krylov subspace          10

You asked not to normalise the input matrices
You asked not to write the left and right EOFs used for the reconstruction
...

*********************************************************************
Now some statistics about your data:

Number of mask land points:                                    38315
       Dimension of file  1:                   301  x   201  x    10
                       Mean:                                   18.80
         Standard deviation:                                    0.63
```

(continues on next page)

(continued from previous page)

```
Size of the matrix used in DINEOF:                     22186 x     10

Missing data:                   100636   out of    221860 ( 45.36%)

Number of cross validation points                              2258
*******************************************************************

Time (in seconds) for 1 EOF mode calculation in DINEOF    0.0156

# EOF modes asked:   5        Convergence level required: 0.1E-02

EOF mode    Expected Error    Iterations made   Convergence achieved
________    ______________    _______________   ____________________

     1             0.4966               300           0.1237E-02
     2             0.4755                96           0.9935E-03
     3             0.5418               205           0.9989E-03
     4             0.7587               300           0.1262E-02
     5             0.8973               234           0.9967E-03

Minimum reached in cross-validation
Number of optimal EOF modes:            2

Make last reconstruction, including data put aside for cross-validation

     2             0.4755                96           0.9935E-03

DINEOF finished!

number of eigenvalues retained for the reconstruction   2
        expected error calculated by cross-validation   0.4755
           total time (in seconds) in lanczos process   11.3438

SCE filled matrix (valex eliminated) = 173496.98260663458

mean    18.818932248247631
Standard deviation  0.88403924852814730       0.63180127935943464        18.
↪796893489562994

...done!
```

The actual result, will be saved in the output folder, in a NetCDF file wit the same name as your original data.

CHAPTER 3

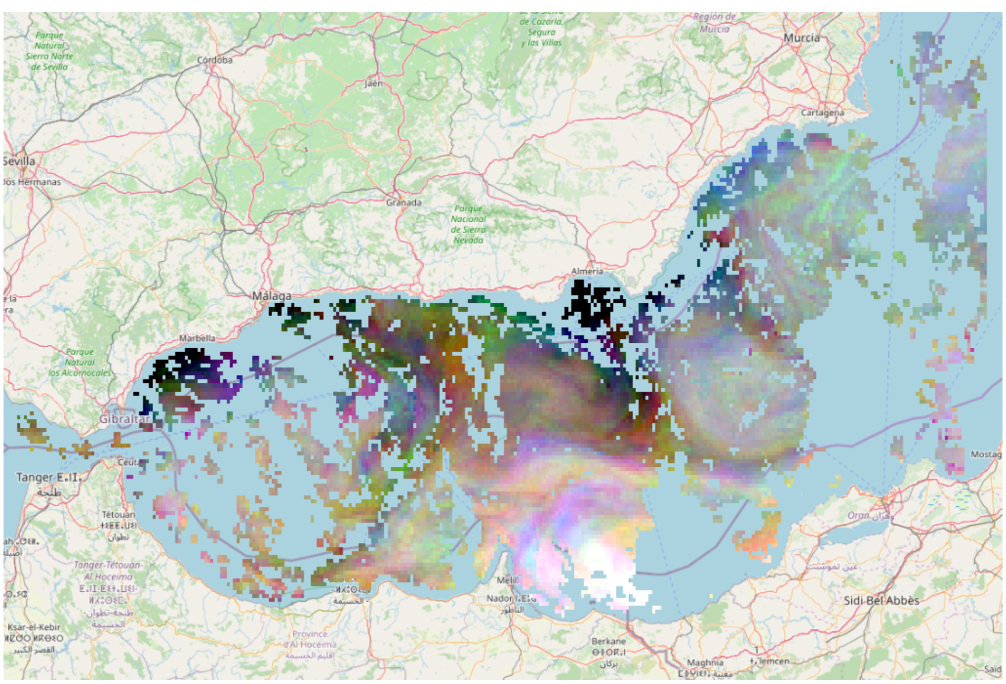
Example

For issues, bugs, proposals or remarks, visit the issue tracker.

3.1 Get the data

You can download the example data on bitbucket.

Have a look at the sea surface temperature (*SST*) layer in QGIS. Set the CRS (WGS 84) and add for example OpenStreetMap (OpenLayers Plugin) as a background layer:

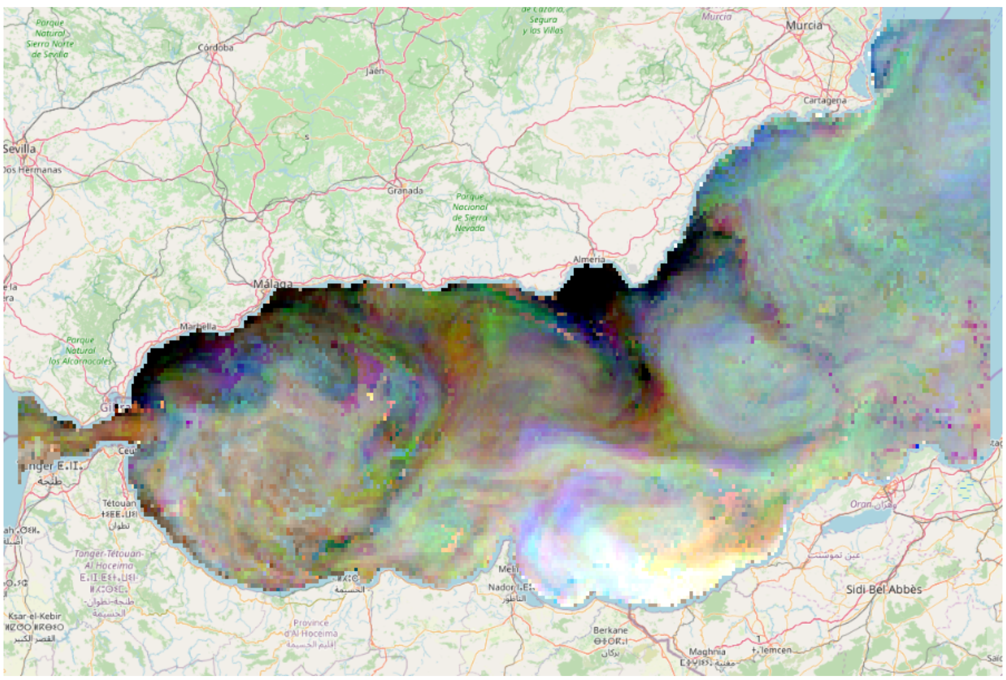


Beware that this is a false colour composite of the first 3 days as red, green and blue.

3.2 Try out the tool

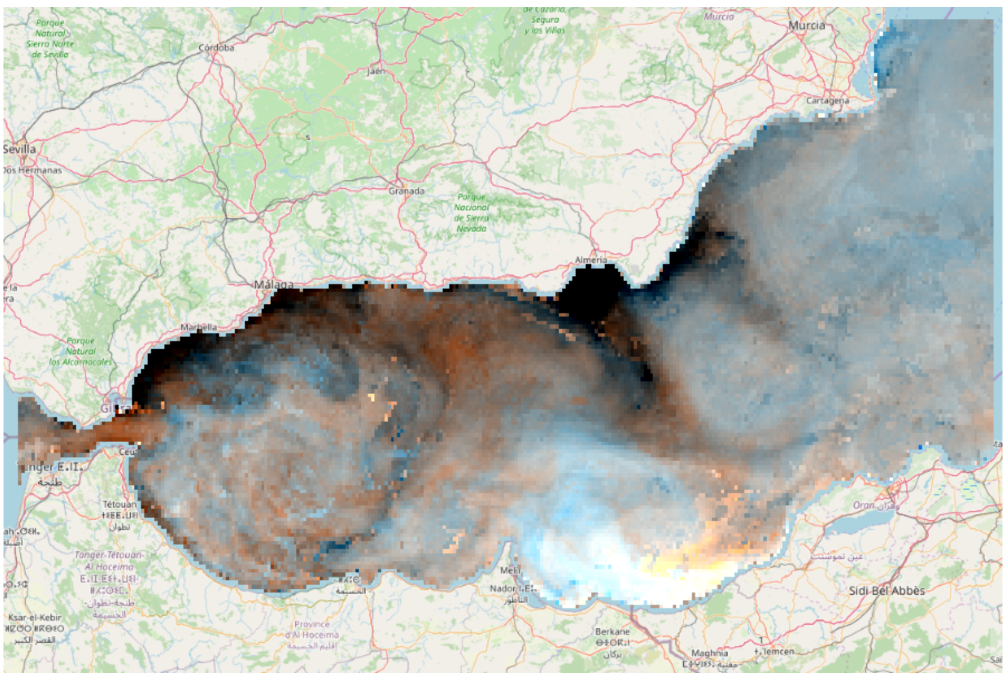
1. Start by selecting the data with gaps: browse to the file you just downloaded and select it. Then select the correct data set from the drop down menu: it is *SST* (see surface temperature).
2. Next, select the same dataset for the land-sea mask. This time select *mask* from the drop down.
3. All the other settings, we can leave as they are.
4. Optionally, choose a folder for all results, otherwise an output folder will be created next to your downloaded file.

The result will open automatically in QGIS. Beware that this is a false colour composite of the first 3 days as red, green and blue.



3.3 Try out different settings

Now you can try out different settings. For example, try to run the algorithm with full reconstruction instead of only reconstructing the missing data. The result will look slightly different:


Murcia
Córdoba
Jaén
Cartagena
Sevilla
Dos Hermanas
Granada
Almería
Málaga
Marbella
Tétouan
Oran
Sidi Bel Abbès
Berkane
Maghnia
Ksar-el-Kebir

CHAPTER 4

API

Source code: https://bitbucket.org/kul-reseco/dineof/src.

For issues, bugs, proposals or remarks, visit the issue tracker.

Date : April 2020
Copyright : © 2020 by Ann Crabbé
Email : acrabbe.foss@gmail.com
Acknowledgements : Based on DINEOF 4.0 (ULG Liege) by Alida Alvera-Azcárte and Alexander Barth.

This file is part of the EOF-based Time Series Reconstructor QGIS plugin package.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License (COPYING.txt). If not see www.gnu.org/licenses.

---

class dineof.core.dineof.**Dineof**(*data_path: str*, *output_folder: str*, *eof_max: int*, *krylov_max: int*, *mask_path: str = None*, *eof_min: int = 1*, *eof_max_it: int = 300*, *eof_it_stop: float = 0.001*, *lanczos: float = 1e-08*, *time_path: str = None*, *time_alpha: float = 0.01*, *time_it: int = 3*, *reconstruct_all: int = 1*, *save_eof_base: int = 0*, *normalize_input: int = 0*, *cv_seed: int = 243435*, *cv_clouds: str = None*)

Bases: `object`

EOF-based method to fill in missing data from geophysical fields, such as clouds in sea surface temperature.

Parameters

- **data_path** – path to netCDF data with gaps to be filled
- **output_folder** – path to output folder where all results will be stored
- **eof_max** – the max number of EOF modes to compute; only the optimal number of EOFs + 3 will be computed for robust optimum detection must be <= temporal size of data - 5; start high and adjust down for subsequent reconstructions; if the max number of modes have been used, the algorithm probably needs more information
- **krylov_max** – the maximum size for the Krylov subspace; must be: >= eof_max + 5 and <= temporal size of data
- **mask_path** – path to land-sea mask; netCDF format (optional) without mask, all points are considered as ocean
- **eof_min** – the min number of EOF modes to computes; default = 1 (recommended)
- **eof_max_it** – set the maximum iterations for each EOF calculation; default = 300; eof decomposition <-> truncated reconstruction and replacement of missing data; use large number and only increase if this number is reached for each EOF
- **eof_it_stop** – stop EOF iterations once: rms(successive reconstruction) / std(existing data) < eof_it_stop
- **lanczos** – threshold for Lanczos convergence; default = 1.e-8 (recommended)
- **time_path** – path to time file; netCDF format; used for the diffusion of the covariance matrix specified as the relative increment in time between your individual images; example: [1,2,3,4,…] if all data is 1 day apart; [1,2,4,…] if day 3 is missing
- **time_alpha** – sets strength of the filter; 0 to deactivate
- **time_it** – number of iterations for the filter
- **reconstruct_all** – 1: reconstruct all points; 0: reconstruct only missing data
- **save_eof_base** – 1: save left and right EOFs; 0: not
- **normalize_input** – 1: normalize the input matrix for multivariate case; 0: not
- **cv_seed** – initialize random number generator, for selection of cross-validation points; repeating an experiment with the same seed will yield the exact same result while another seed will yield a slight variation
- **cv_clouds** – path to cloud file; netCDF format; mask with manual selection of cross-validation points cv_seed will be ignored

For more information on how DINEOF works, please refer to:

Alvera-Azcárate, A., Barth, A., Rixen, M., Beckers, J.M., 2005, Reconstruction of incomplete oceanographic data sets using empirical orthogonal functions: application to the Adriatic Sea surface temperature, Ocean Modelling, 9-4, p. 325-346

Beckers, J.M., Rixen, M., 2003, EOF calculations and data filling from incomplete oceanographic data sets, Journal of Atmospheric and Oceanic Technology, 20-12, p. 1839-1856

The multivariate application of DINEOF is explained in:

Alvera-Azcárate, A., Barth, A., Beckers, J.M., Weisberg, R.H., 2007, Multivariate reconstruction of missing data in sea surface temperature, chlorophyll, and wind satellite fields, Journal of Geophysical Research. Oceans, 112-C3, p. C03008

The error calculation using an optimal interpolation approach is explained in:

Beckers, J.M., Barth, A., Alvera-Azcárate, A., 2006, DINEOF reconstruction of clouded images including error maps – application to the Sea-Surface Temperature around Corsican Island, Ocean Science, 2, 183–199

For more information about the Lanczos solver:

Toumazou, V., Cretaux, J.F., 2001, Using a Lanczos Eigensolver in the Computation of Empirical Orthogonal Functions, Monthly Weather Review, 129-5, p. 1243–1250

`execute`()

`write_init_file`(*path*)

Parameters **`path`** – the path of the init file

Returns Write the input variables as text string to path

Python Module Index

d

dineof.core.dineof, 13

Index

D

Dineof (*class in dineof.core.dineof*), 13
dineof.core.dineof (*module*), 13

E

execute() (*dineof.core.dineof.Dineof method*), 15

W

write_init_file() (*dineof.core.dineof.Dineof method*), 15