Université de Liège

Empirical orthogonal functions for data with missing values

ESA Summer school 2010

Jean-Marie Beckers and GHER group Aida Alvera, Alexander Barth, Luc Vandenbulcke, Charles Troupin, Damien Sirjacobs, Mohamed Ouberdous

http://modb.oce.ulg.ac.be/GHER

Université de Liège
MARE-GHER Sart-Tilman B5
4000 Liège, Belgique

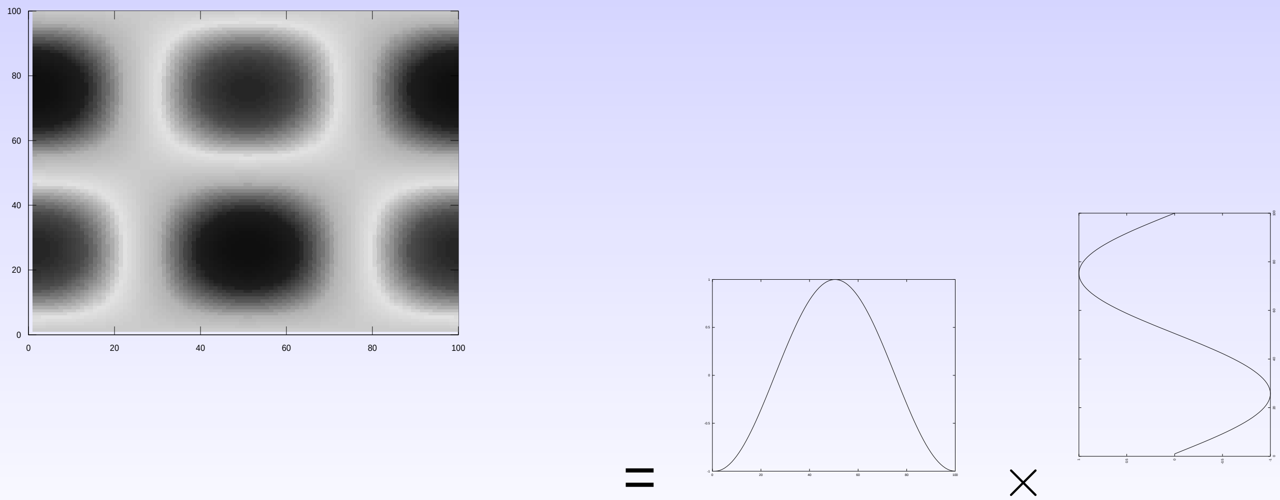
Outline

- ***EOFs***
- ***EOF with missing data***
- ***DINEOF***
- ***Examples***
- ***Summary***

- ***EOFs***
- ***EOF with missing data***
- ***DINEOF***
- ***Examples***
- ***Summary***

Generalisations and other names

- Empirical orthogonal functions (EOFs)
- Proper orthogonal decomposition (POD)
- Karhunen-Loève decompositions (best to start from for non-uniform data distribution)
- Proper orthogonal modes (POM)
- Principal component analysis (PCA)



Classical approach

- We assume that we have a matrix $\mathbf{X}$ containing the observations, which is arranged such that the element i, j of the matrix is called $(\mathbf{X})_{ij}$ and is given by the value of the field $f(\boldsymbol{r}, t)$ at location $\boldsymbol{r}_i$ and moment t_j :

$$(\mathbf{X})_{ij} = f(\boldsymbol{r}_i, t_j). \quad (1)$$

- The field f is an observational field and contains thus all errors (instrumental, unresolved structures, etc).
- We can then write the matrix as a succession of n column vectors

$$\mathbf{X} = \left(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n \right), \quad (2)$$

each of the column vectors $\mathbf{x}_j$ being the discrete state vector of size m at moment t_j.

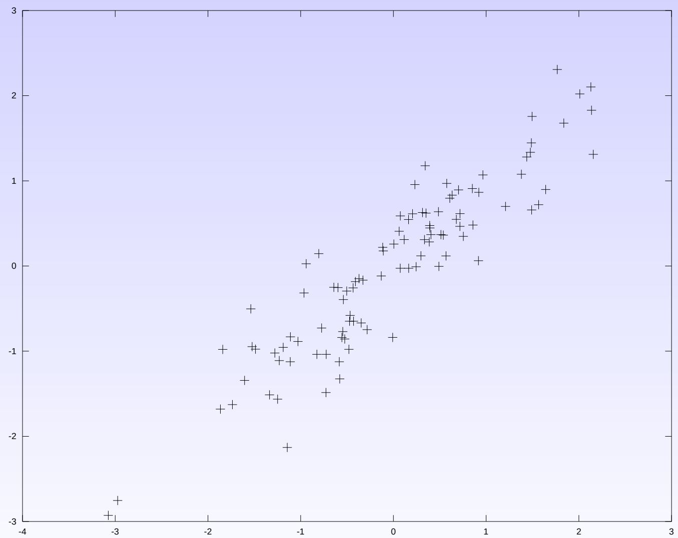
Defining a mode

Try to find a spatial structure $\mathbf{u}$ which represents at best the data:

- Maximize norm of $\mathbf{X}^{\mathrm{T}}\mathbf{u}$ with normalization constraint on $\mathbf{u}$.
- Try to find the direction in which the data have their largest component.

Find extrema of functional J with Lagrange multiplier

$$J = \mathbf{u}^{\mathrm{T}}\mathbf{X}\,\mathbf{X}^{\mathrm{T}}\mathbf{u} - \lambda\left(\mathbf{u}^{\mathrm{T}}\mathbf{u} - 1\right) \qquad (3)$$



Euler-Lagrange equation

- Variations in (3) on $\mathbf{u}$:

$$\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{u} = \lambda\mathbf{u} \tag{4}$$

- Variations on λ

$$\mathbf{u}^{\mathrm{T}}\mathbf{u} = 1 \tag{5}$$

EOFs are normalized eigenvectors of "covariance"[a] matrix $\mathbf{X}\,\mathbf{X}^{\mathrm{T}}$which is a symmetric positive defined matrix: real positive eigenvalues and orthogonal eigenvectors:

$$\lambda_i = \rho_i^2 \qquad \text{conventionally } \rho_i > 0 \text{ and } \rho_{i+1} \leq \rho_i \tag{6}$$

$$\mathbf{u_i}^{\mathrm{T}}\mathbf{u_j} = \delta_{ij} \tag{7}$$

Storing $\mathbf{u_i}$ as columns in $\mathbf{U}$ yields $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$

---

[a]" " because missing 1/n. Observe time summation to get spatial covariances

Temporal amplitudes

What is the temporal evolution $\mathbf{g_j}$ of the amplitudes of a mode $\mathbf{u_j}$?

Temporal amplitudes

What is the temporal evolution $\mathbf{g_j}$ of the amplitudes of a mode $\mathbf{u_j}$?
See it as minimizing the norm of $\mathbf{X} - \mathbf{u_j}\mathbf{g_j}^{\mathrm{T}}$

Temporal amplitudes

What is the temporal evolution $\mathbf{g_j}$ of the amplitudes of a mode $\mathbf{u_j}$?
See it as minimizing the norm of $\mathbf{X} - \mathbf{u_j}\mathbf{g_j}^{\mathrm{T}}$

$$\mathbf{g_j} = \mathbf{X}^{\mathrm{T}}\mathbf{u_j} \tag{8}$$

Projection of data of a given moment onto the spatial mode $\mathbf{u_j}$

Temporal modes

Using temporal covariances $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ instead of spatial covariances $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ yields temporal EOFs :

$$\mathbf{X}^{\mathrm{T}}\mathbf{X}\,\mathbf{v} = \mu\mathbf{v} \tag{9}$$

$$\mathbf{v}^{\mathrm{T}}\mathbf{v} = 1 \tag{10}$$

$$\mathbf{V}^{\mathrm{T}}\mathbf{V} = \mathbf{I} \tag{11}$$

Link between spatial and temporal modes? Yes, easily proven via SVD decomposition of a matrix

SVD

Singular value decomposition of a data matrix $\mathbf{X}$:

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}} = \sum_{k=1}^{q} \rho_k \mathbf{u}_k \mathbf{v}_k^{\mathrm{T}} \tag{12}$$

leads to

- spatial ($\mathbf{U}$) and temporal ($\mathbf{V}$) EOFs (orthonormal)
- singular values ρ_i (stored on the diagonal of $\mathbf{D}$)

They are also solution of

$$\begin{array}{c} \mathbf{X}\mathbf{v} = \rho\mathbf{u} \\ \mathbf{X}^{\mathrm{T}}\mathbf{u} = \rho\mathbf{v}, \end{array} \quad \Rightarrow \quad \begin{array}{c} \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{u} = \rho^2\mathbf{u} \\ \mathbf{X}^{\mathrm{T}}\mathbf{X}\,\mathbf{v} = \rho^2\mathbf{v} \end{array} \tag{13}$$

Spatial and temporal modes can be obtained via SVD decomposition!

Properties

- Conventionally (positive) singular values are ordered by decreasing value.
- A given mode j contributes to the explained variance as the squared norm of $\mathbf{u_j}\rho_j\mathbf{v_j}^{\mathrm{T}}$, ie: ρ_j^2.
- The total variance in the data (squared norm of $\mathbf{X}$) equals the sum of all squared singular values:

$$\text{trace}\left(\mathbf{X}\mathbf{X}^{\mathrm{T}}\right) = \text{trace}\left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right) = ||\mathbf{X}||_2^2 = \sum_{k=1}^{q} \rho_k^2, \qquad (14)$$

is a measure of the total variance (also sometimes abusively called energy) in the system. The ratio $f_k = \rho_k^2 / \sum_{k=1}^{q} \rho_k^2$ is thus a measure of the variance contained in mode k compared to the overall energy and one often says that mode k explains $100\, f_k\,\%$ of the variance and that the first K modes explain $100\, \sum_{k=1}^{K} \rho_k^2 / \sum_{k=1}^{q} \rho_k^2\,\%$ of the total variance.

Properties

- The first p modes define thus the best base using p base vectors in which the data can be expressed with minimum loss of information.
- Truncation can be used to filter data by rejecting modes.
- First modes often have physical meaning, the following less (due to orthogonality constraint).
- EOF very efficient for standing patterns, less for propagating features (try to generate synthetic data and look at SVD decomposition).
- Space and time can be interchanged.
- Calculate eigenvalues on smallest covariance matrix, eigenvalues are the same !
- No explicit information on "distance" or "time", only convariances are relevant.
- Data can be reordered without changing the EOFs.
- 2D spatial data can be packed into 1D arrays.

Truncation

$$\mathbf{X_N} = \mathbf{U_N}\mathbf{D_N}\mathbf{V_N}^{\mathrm{T}} = \sum_{k=1}^{N} \rho_k \mathbf{u}_k \mathbf{v}_k^{\mathrm{T}} \tag{15}$$

is the best approximation to $\mathbf{X}$ using only N spatial modes

The covariance matrix $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ can also be approximated by using the truncated representation of $\mathbf{X}$

$$\mathbf{X_N}\mathbf{X_N}^{\mathrm{T}} = \mathbf{U_N}\mathbf{D_N}\mathbf{V_N}^{\mathrm{T}}\mathbf{V_N}\mathbf{D_N}^{\mathrm{T}}\mathbf{U_N}^{\mathrm{T}} = \mathbf{U_N}\mathbf{D_N}\mathbf{D_N}^{\mathrm{T}}\mathbf{U_N}^{\mathrm{T}} = \tilde{\mathbf{U}}\tilde{\mathbf{U}}^{\mathrm{T}} \tag{16}$$

where $\tilde{\mathbf{U}}$ is matrix $\mathbf{U_N}$ where each column is multiplied by the associated singular value.

This reduced rank covariance matrix is specially usefull in Data Assimulation (see lesson 3)

Truncation rejects some data as noise, hence SVD can be used as filtering tool

Generalisations

- Values of the state vector can accommodate complex values, allowing complex EOF on 2D vectors (horizontal velocities, gradients ...), Fourrier transforms, Hilbert transforms, etc
- Multivariate EOFs $\mathbf{x}^{\mathrm{T}} = \left(\mathbf{T}^{\mathrm{T}}, \mathbf{S}^{\mathrm{T}}, \mathbf{p}^{\mathrm{T}}, ...\right)$ (needs proper non dimensional form)
- SSA, lagged covariances to detect autocorrelations in time

$$\mathbf{X} = \begin{pmatrix} x_1 & x_2 & x_3 & \ldots & x_n \\ x_2 & x_3 & x_4 & \ldots & x_1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ x_k & \ldots & & \ldots & x_{k-1} \end{pmatrix}, \tag{17}$$

- MSSA, temporally lagged spatial fields

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \ldots & \mathbf{x}_n \\ \mathbf{x}_2 & \mathbf{x}_3 & \mathbf{x}_4 & \ldots & \mathbf{x}_1 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \mathbf{x}_k & \ldots & & \ldots. & \mathbf{x}_{k-1} \end{pmatrix}, \tag{18}$$

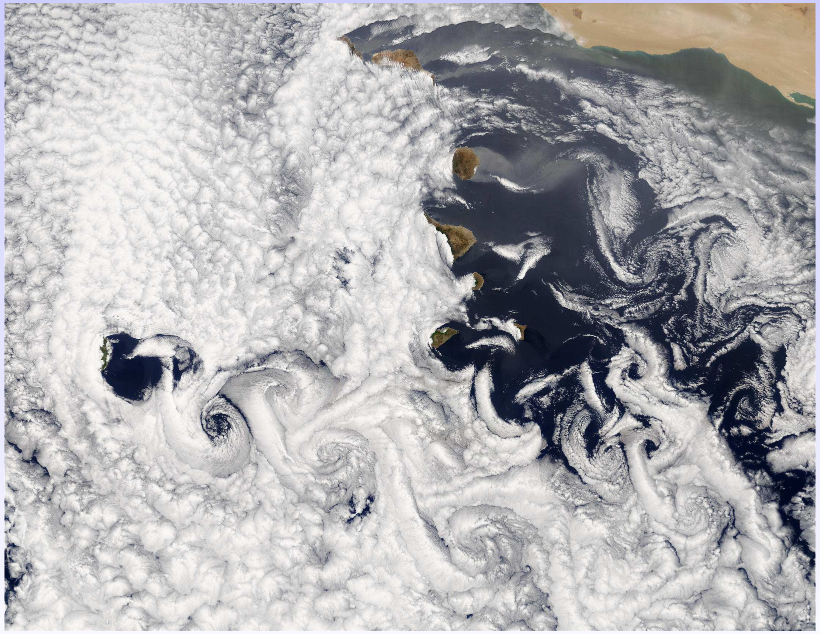
Applications

- Analysis of variability
- Data compression
- Filtering
- Data synthesizing
- Model intercomparison (time-space error distinctions, phase-amplitude error distinctions)
- Objective analysis (or optimal interpolation) of in-situ data: use of vertical EOFs and horizontal objective analysis of their amplitudes to reduces the number of data during the costly optimal interpolation
- Reducing the size of a problem by projecting (complex) reference model equations onto a small number of Principal Components
- Simplify forecasts by using any extrapolation method (neural networks, genetic algorithms etc) on temporal modes
- Ensemble preparation for assimilations

The Problem

EOFs and friends are powerful tools to analyse data and reduce the dimensionality of a problem. However, the decomposition assumes the data matrix **X** (or continuous function f) to be known.

Missing data must be imputed before performing EOF analysis



© ??

- *EOFs*
- ***EOF with missing data***
- *DINEOF*
- *Examples*
- *Summary*

Filling data in before EOF calculation

Data filling scene by scene: spatial interpolation as described in Lesson 1 .

- Distance weighting
- Optimal interpolation with presribed covariance function
- DIVA

Problems:

- Cost: each scene with $\sim 10^7$ data points and OI cost prohibitive. Hence suboptimal OI.
- Covariance used for spatial interpolation will generally be inconsistent with covariance (16).

Filling data in before EOF calculation

Data filling scene by scene: spatial interpolation as described in Lesson 1 .

- Distance weighting
- Optimal interpolation with presribed covariance function
- DIVA

Problems:

- Cost: each scene with $\sim 10^7$ data points and OI cost prohibitive. Hence suboptimal OI.
- Covariance used for spatial interpolation will generally be inconsistent with covariance (16).

Was there not the idea of EOFs being related to covariances?

Use EOF for filling in

IF we knew EOFs, we could use the covariance[a] matrix matrix **B** retrieved from EOFs series (16) for use within OI

$$\mathbf{B} = \frac{1}{n}\mathbf{U_N}\mathbf{D_N}\mathbf{D_N}^{\mathrm{T}}\mathbf{U_N}^{\mathrm{T}} \quad (19)$$

Truncation of data means that covariance matrix only represents "signal", the rejected data is "noise". Hence for OI we need to take into accound noise with its covariance matrix **R**, diagonal for simplicity.
OI interpolation $\mathbf{x}^a$ of a given scene with available data $\mathbf{y}$ and observation matrix[b] **H**:

$$\mathbf{x}^a = \mathbf{B}\mathbf{H}^{\mathrm{T}}\left(\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right)^{-1}\mathbf{y} \quad (20)$$

Problem: size of matrix to invert

[a] Here we need to work with real covariance matrices, hence introduction of 1/n

[b] Simple matrix with zeros and ones only at data location

THE matrix equation in Data Assimilation

Sherman-Morrison formula

$$\left(\mathbf{A} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\right)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}\left(\mathbf{I} + \mathbf{V}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{U}\right)^{-1}\mathbf{V}^{\mathrm{T}}\mathbf{A}^{-1} \qquad (21)$$

Proof left to your skills.

Final step

We need to calculate

$$\mathbf{x}^a = \mathbf{B}\mathbf{H}^{\mathrm{T}}\left(\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right)^{-1}\mathbf{y} \tag{22}$$

where

$$\mathbf{B} = \frac{1}{n}\mathbf{U_N}\mathbf{D_N}\mathbf{D_N}^{\mathrm{T}}\mathbf{U_N}^{\mathrm{T}} \tag{23}$$

Define $\mathbf{L}_p = \frac{1}{\sqrt{n}}\mathbf{H}\mathbf{U_N}\mathbf{D_N}$ and $\mathbf{L} = \frac{1}{\sqrt{n}}\mathbf{U_N}\mathbf{D_N}$ so that

$$\mathbf{x}^a = \mathbf{L}\mathbf{L_p}^{\mathrm{T}}\left(\mathbf{L_p}\mathbf{L_p}^{\mathrm{T}} + \mathbf{R}\right)^{-1}\mathbf{y} \tag{24}$$

Application of (21):

$$\mathbf{x}^a = \mathbf{L}\left(\mathbf{L_p}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{L_p} + \mathbf{I}\right)^{-1}\mathbf{L_p}^{\mathrm{T}}\mathbf{R}^{-1}\mathbf{y} \tag{25}$$

Now size of the matrix to invert: N number of modes retained instead of P number of observations !

Interpretation with uniform noise $\mathbf{R} = \mu^2 \mathbf{I}$

$$\mathbf{x}^a = \mathbf{L}\left(\mathbf{L_p}^{\mathrm{T}}\mathbf{L_p} + \mu^2\mathbf{I}\right)^{-1}\mathbf{L_p}^{\mathrm{T}}\,\mathbf{y} \tag{26}$$

Combination of spatial modes contained in **L** which fit best the data in a least square sense in the presence of noise.
Note that the error covariance of the analysis is also available from OI theory:

$$\mathbf{P}^a = \left(\mathbf{I} - \mathbf{B}\mathbf{H}^{\mathrm{T}}\left(\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right)^{-1}\mathbf{H}\right)\mathbf{B} \tag{27}$$

which leads to

$$\mathbf{P}^a = \mathbf{L}\mathbf{L}^{\mathrm{T}} - \mathbf{L}\left(\mathbf{L_p}^{\mathrm{T}}\mathbf{L_p} + \mu^2\mathbf{I}\right)^{-1}\mathbf{L_p}^{\mathrm{T}}\mathbf{L_p}\mathbf{L}^{\mathrm{T}} \tag{28}$$

or

$$\mathbf{P}^a = \mu^2\mathbf{L}\left(\mathbf{L_p}^{\mathrm{T}}\mathbf{L_p} + \mu^2\mathbf{I}\right)^{-1}\mathbf{L}^{\mathrm{T}} \tag{29}$$

Chicken or egg question

Yet even with simplified OI, circular dependance since EOF needed for interpolation and interpolated field needed to calculate EOFs

- iterate: first guess on EOFs, filling, recalculate EOFs, refill ...
- EOFs from other sources: models, reorthogonalized eigenvector of data-covariance matrix (using only present data)

In all cases, even with simplified OI, still expensive. Also destroying space-time symmetry of original SVD.

- *EOFs*
- *EOF with missing data*
- ***DINEOF***
- *Examples*
- *Summary*

Data INterpolating EOF: DINEOF interpolation

- Large scale EOFs should not be influenced by local changes in the values of a few points
- Large scale EOFs can thus be estimated by using a first guess of missing data
- Then, once the larger scale EOFs and their amplitudes are estimated, they can serve to calculate the value of the field at the missing points by

$$(\mathbf{X}_a)_{ij} = \left(\mathbf{U_N D_N V_N}^{\mathrm{T}}\right)_{ij} = \sum_{k=1}^{N} \rho_k \left(\mathbf{u}_k\right)_i \left(\mathbf{v}_k^{\mathrm{T}}\right)_j, \qquad (30)$$

- EOFs themselves can be re-evaluated and the process can be repeated until convergence.
- What is the optimal number N of relevant EOFs to be retained to recompose the signal at the missing data points ?

Optimal number N of EOFs: Cross validation technique

- Set aside a random set of valid data (random points or random clouds).
- Use the EOF interpolation and calculate an error estimate based on the rms distance between the interpolated field at these points and the data set aside there
- Start with 1 EOF, fill in, calculate rms error, continue by filling in with a second EOF until convergence, calculate rms error ...
- Provides reconstruction error as a function of number of EOF retained.
- The optimal truncation is the one that minimises the diagnosed error.

Once the optimal number know, perform a last iteration with this number of EOFs reinjecting the data set aside for cross-validation.

DINEOF

- Implementation much more rapid than OI version.
- Problem: no natural error maps in DINEOFs

Error maps:

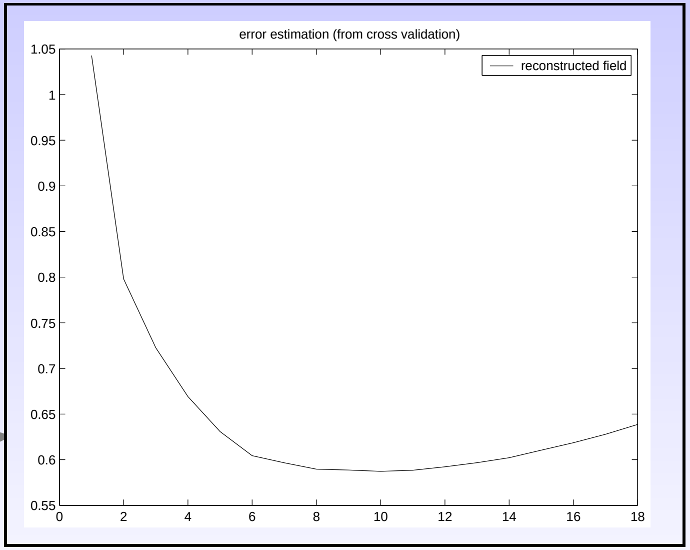
- Use error estimate from OI version (29)
- Numerical verification that difference OI-DINEOF is smaller than this error
- Can also be exploited to detect strange pixels in original data (outlier detection using spatial covariances)

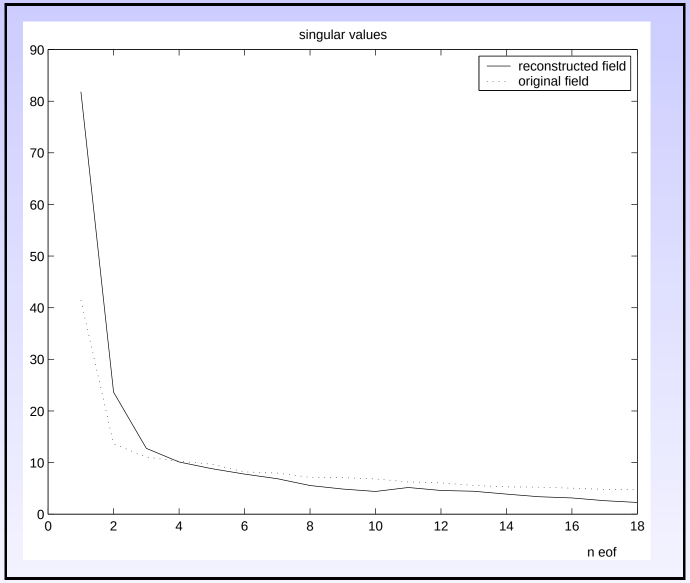
CANNOT see things under clouds which are due to patterns never seen before (EOF are exploiting past pattern)
Operational use (daily 10 day period based on last six month EOFs):
`http://gher-diva.phys.ulg.ac.be/DINEOF/dineof.html`

- EOFs
- EOF with missing data
- DINEOF
- **Examples**
- Summary

Real application Adriatic Sea

- AVHRR from `http://satftp.soest.hawaii.edu/adriatic/Adriatic/html/`
- Images with more than 90% cloud coverage were excluded
- 105 night images retained ($n = 105$) from Julian day 130 to 294
- Average cloud coverage is 52%
- Maximum number of data pixel/image is 94755 (m)

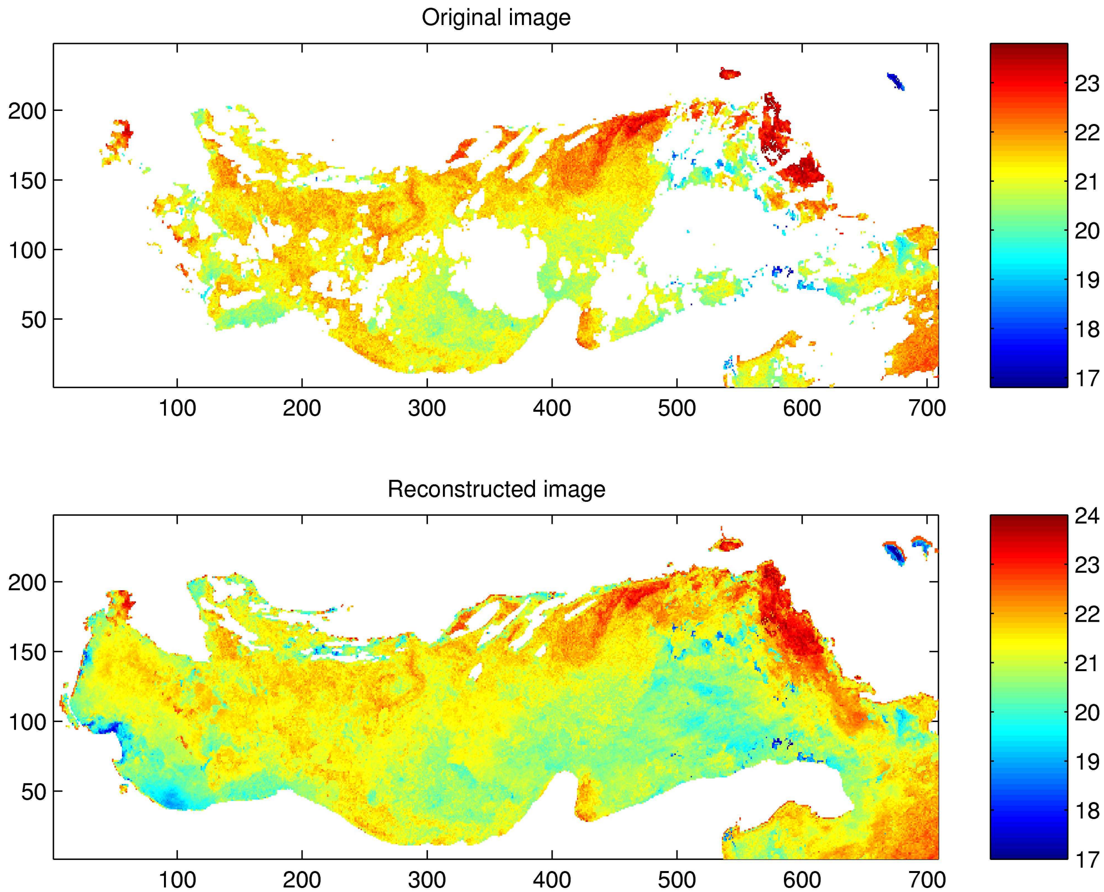




Cross validation ($N = 10$) and convergence

Université de Liège

Université de Liège

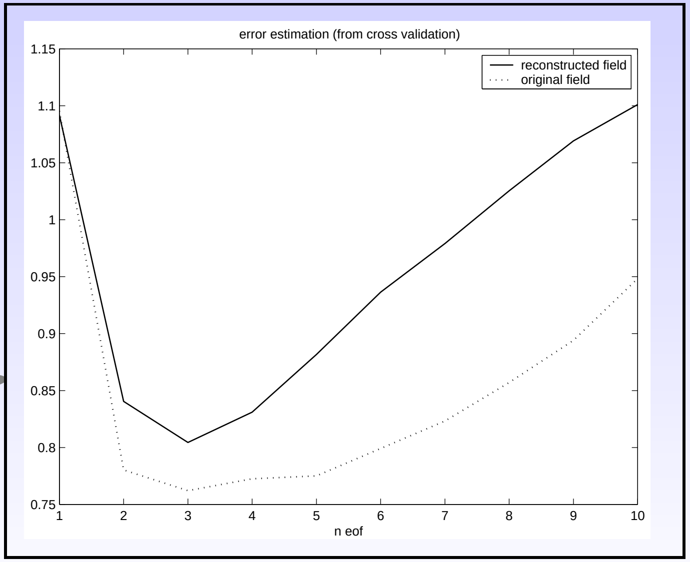


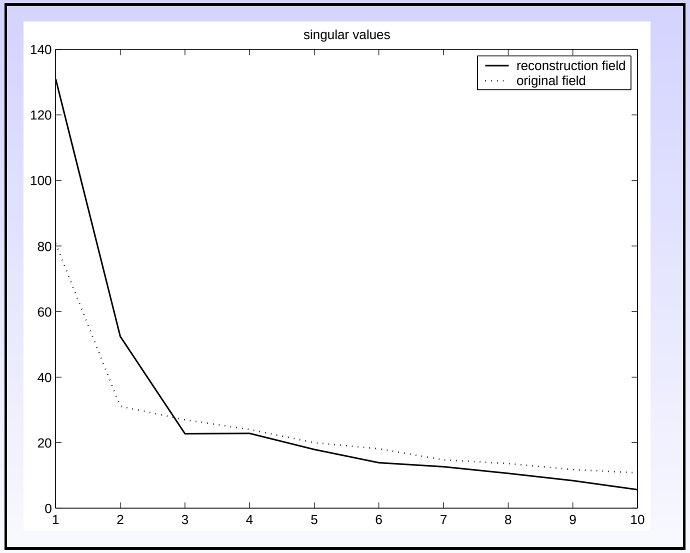
Validation

Validation: a second experiment

- 20 images that contain a 18% average cloud coverage
- On those images, additional virtual clouds were added to obtain a 58% cloud coverage
- The same procedure was applied
- Results are compared to the real data that were virtually clouded

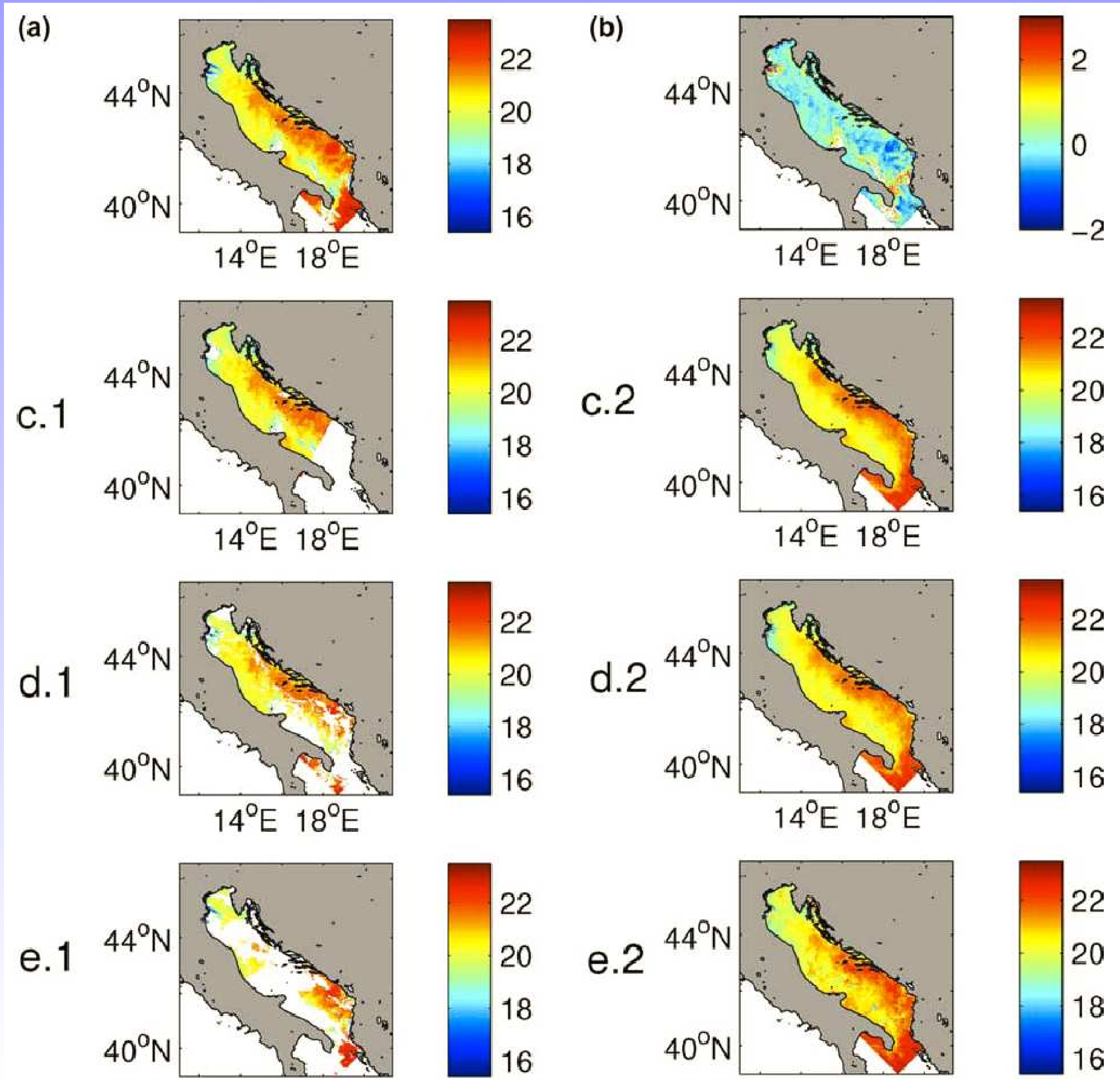
⇒ rms error of 0.22 ° was found.



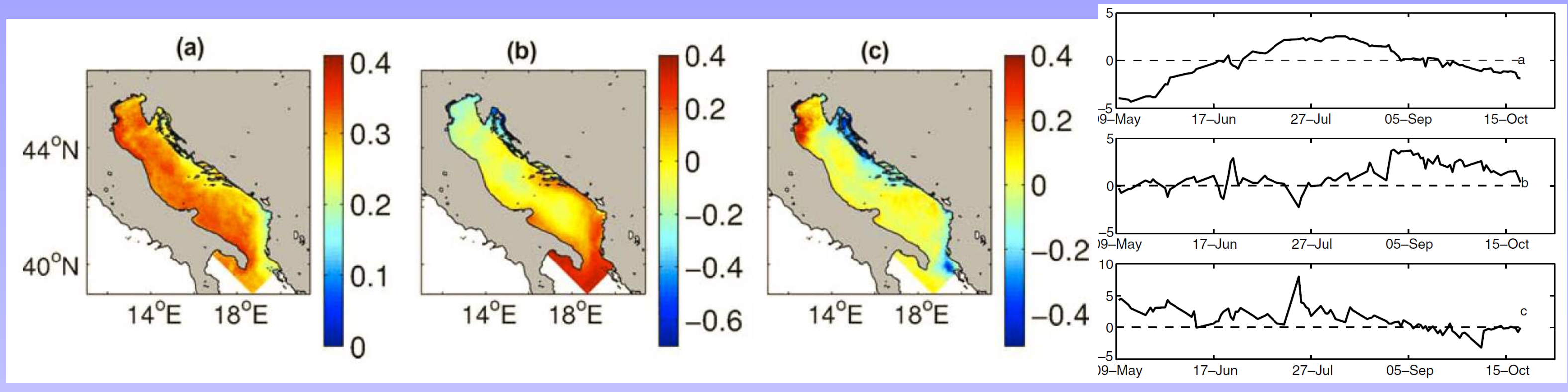


Université de Liège

Quality under virtual clouds



Modes



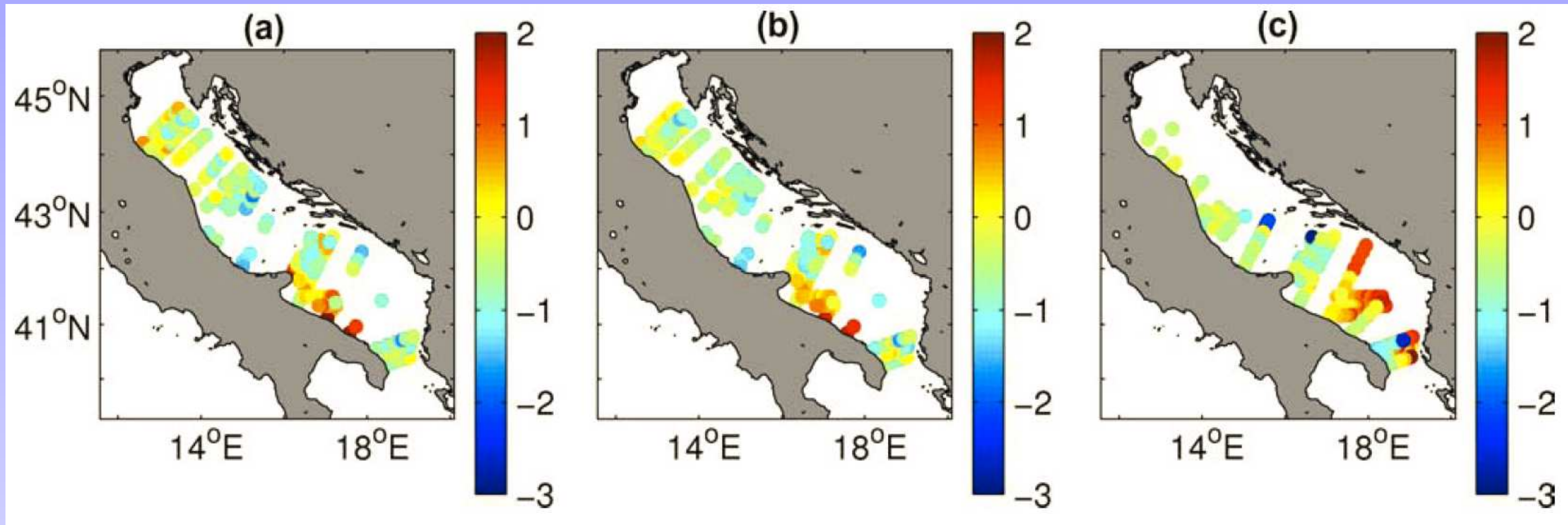
First three modes

1) Seasonal cycle with lower variations in the eastern part (87%)
2) WAC outflow and north-south difference (7%)
3) Po plume (2%)

Observe modulation of Eastern Adriatic Current from the Ionian Sea

Validation with in-situ

MEDAR in-situ data in the Adriatic Sea



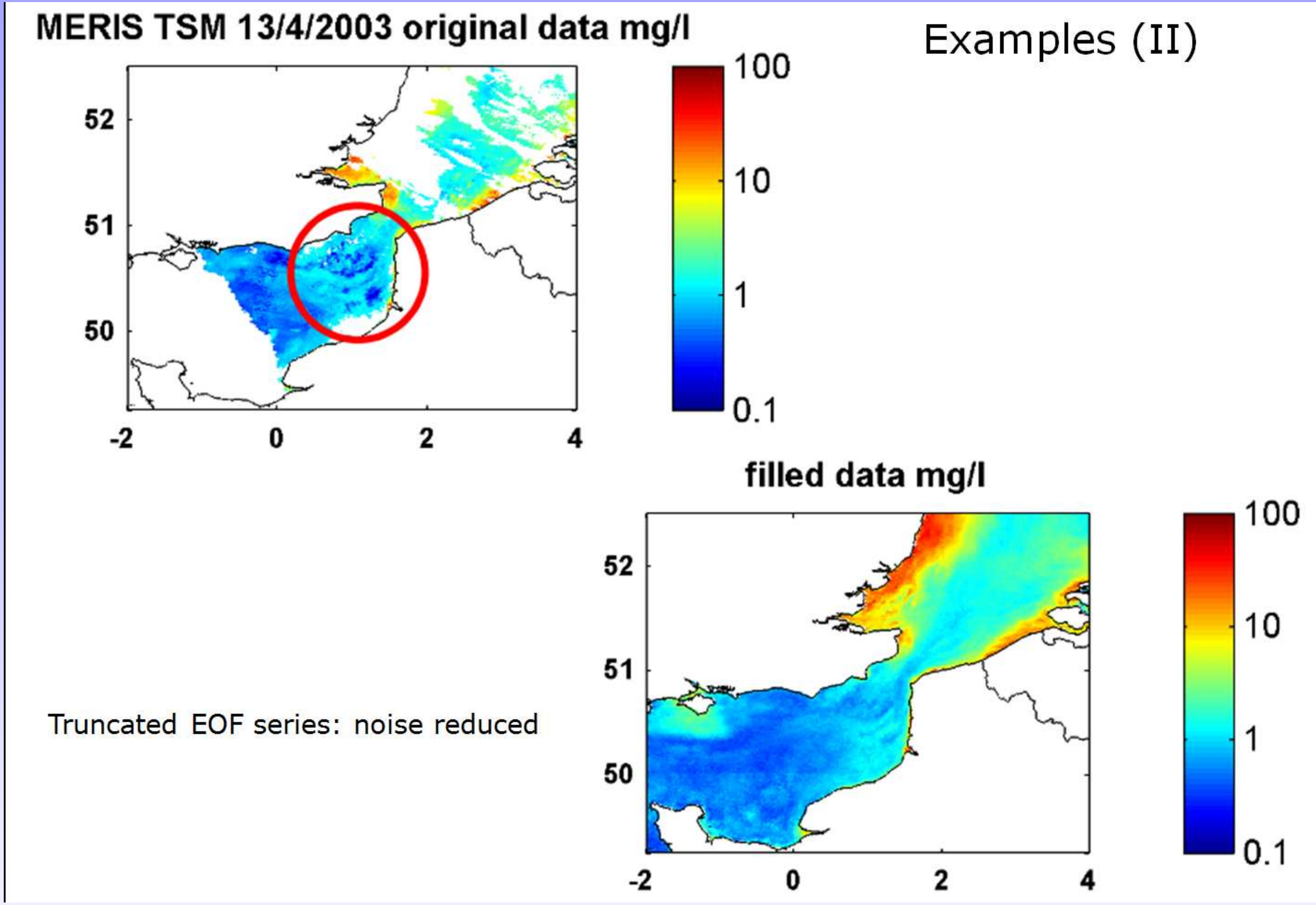
a) difference unclouded points before EOF reconstruction (rms 0.71°)

b) difference unclouded points after EOF reconstruction (rms 0.69°)

c) difference clouded points after EOF reconstruction (rms 0.95°)

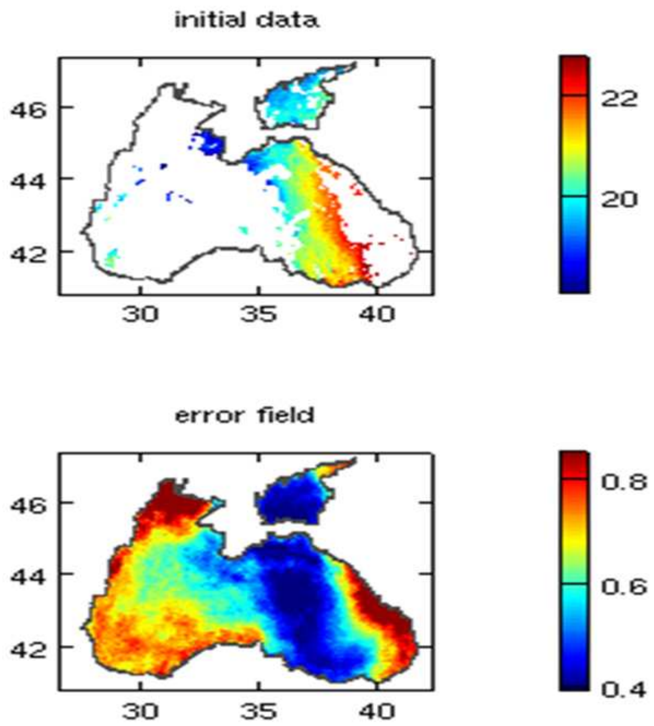
Standard OI has larger errors (1.8°, 2.4 °) and OI with EOF-based covariance similar errors.

Filtering effects

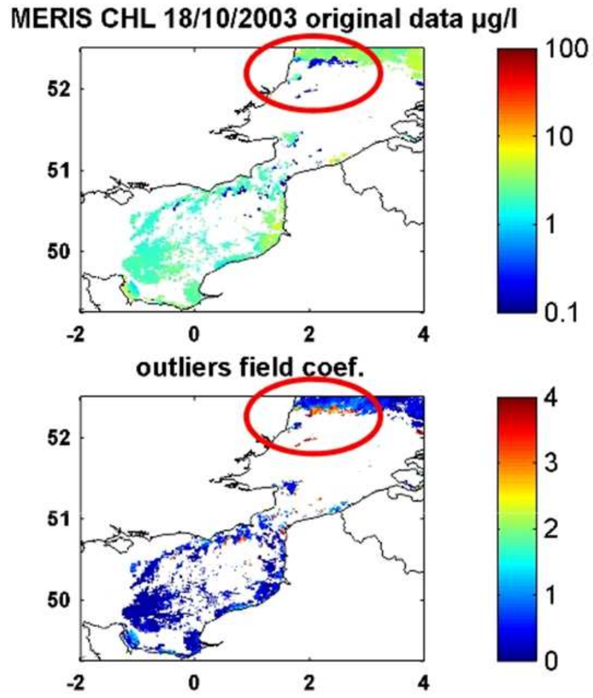


Outlier detection

Error maps will be calculated for TSM reconstructions using:
i) the EOF basis from DINEOF as background covariance
ii) the location of valid data

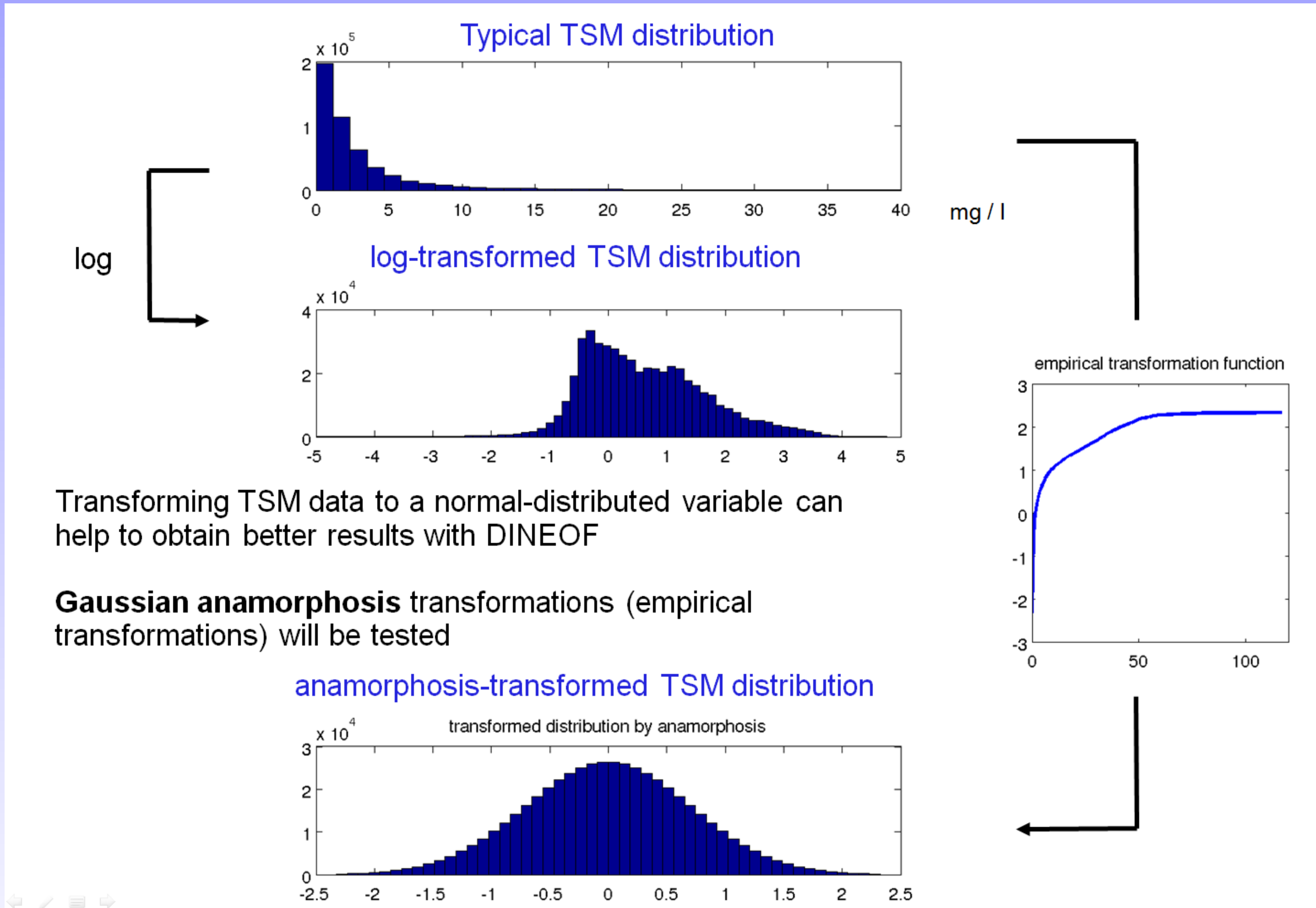


Outliers (pixels with value larger than the statistically expected misfit calculated during the analysis) will be objectively identified and removed from initial data



Exploits spatial information, contrary to most QC procedures in inversion.

Data transformations



Can be used to enforce positive values.

- *EOFs*
- *EOF with missing data*
- *DINEOF*
- *Examples*
- ***Summary***

EOFs

- Data based modes
- Efficient synthesising of data with a few modes
- Special form of truncated covariance matrix
- Use SVD decomposition libraries if available

DINEOF

- Self-coherent interpolation method and EOF calculation
- No calibration or *a priori* information needed
- Cross validation possibility
- Better representation of fundamental EOFs structures for assimilation and prediction
- Computationally efficient when $m \gg n$

Questions? More in papers joinded.

- J.-M. Beckers and M. Rixen. EOF calculations from incomplete oceanographic data sets. *Journal of Atmospheric and Ocean Technologies*, 20:1839–1856, 2003.
- A. Alvera-Azcárate, A. Barth, M. Rixen, and J.-M. Beckers. Reconstruction of incomplete oceanographic data sets using Empirical Orthogonal Functions. Application to the Adriatic sea. *Ocean Modelling*, 9:325–346, 2005.
- J.-M. Beckers, A. Barth and A. Alvera-Azcárate, DINEOF reconstruction of clouded images including error maps. Application to the Sea-Surface Temperature around Corsican Island. *Ocean Science*, 2:183–199, 2006.
- A. Alvera-Azcárate, A. Barth, J.-M. Beckers and R. Weisberg Multivariate Reconstruction of Missing Data in Sea Surface Temperature, Chlorophyll and Wind Satellite Fields. *J. Geophys. Res.*, 112:c03008, 2007.
- Alvera-Azcárate, A., Barth, A., Sirjacobs, D. and Beckers, J.-M., Enhancing temporal correlations in EOF expansions for the reconstruction of missing data using DINEOF. *Ocean Science*, 5:475–485, 2009.

Université de Liège

Karhunen-Loève decomposition

- Given a (complex) function $f(\boldsymbol{r}, t)$ of position $\boldsymbol{r}$ and time t [a]
- find a normalised spatial function $u(\boldsymbol{r})$ that is on (time) average closest to $f(\boldsymbol{r}, t)$
- so that function $f(\boldsymbol{r}, t)$ can be approximated by this spatial function multiplied by a amplitude that evolves in time. $f(\boldsymbol{r}, t) \sim g(t)u(\boldsymbol{r})$

$$I(u) \equiv \int \left| \int f(\boldsymbol{r}, t)\, u^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r} \right|^2 \mathrm{d}t \tag{31}$$

Maximize $I(u)$ with a normalisation constraint on u

where $u^\star$ is the complex conjugate of u.

[a] Or any other parameter, even a random parameter, i.e. f being a realisation of an experiment. We assume f to be an anomaly compared to a reference situation

Variational approach

$$I(u) \equiv \int \left| \int f(\boldsymbol{r},t)\, u^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r} \right|^2 \mathrm{d}t = \int \left[\int f(\boldsymbol{r},t)\, u^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r} \int f^\star(\boldsymbol{r}',t)\, u(\boldsymbol{r}') \mathrm{d}\boldsymbol{r}' \right] \mathrm{d}t \tag{32}$$

Find extremas of $J(u, \lambda)$

$$J(u,\lambda) \equiv \int\int\int f(\boldsymbol{r},t)\, u^\star(\boldsymbol{r})\, f^\star(\boldsymbol{r}',t)\, u(\boldsymbol{r}')\, \mathrm{d}\boldsymbol{r} \mathrm{d}\boldsymbol{r}' \mathrm{d}t - \lambda \left[\int u(\boldsymbol{r}) u^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r} - 1 \right] \tag{33}$$

λ being the (real) Lagrange multiplier associated with the normalisation constraint

$$\int u(\boldsymbol{r}) u^\star(\boldsymbol{r})\, \mathrm{d}\boldsymbol{r} = 1 \tag{34}$$

Variation of λ will lead to constraint (34) and variations of u by standard variational approach .

Variations on u

$$J(u+\delta u,\lambda) - J(u,\lambda) =$$

$$2\Re \int\int\int \left[f(\boldsymbol{r},t) f^\star(\boldsymbol{r}',t) \delta u^\star(\boldsymbol{r}) u(\boldsymbol{r}') \right] \mathrm{d}\boldsymbol{r}\mathrm{d}\boldsymbol{r}'\mathrm{d}t$$

$$-2\lambda\Re \int u(\boldsymbol{r})\, \delta u^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r}$$

$$+\mathcal{O}(\delta u)^2 =$$

$$2\Re \int \left\{ \int\int f(\boldsymbol{r},t) f^\star(\boldsymbol{r}',t)\, u(\boldsymbol{r}')\, \mathrm{d}\boldsymbol{r}'\mathrm{d}t - \lambda u(\boldsymbol{r}) \right\} \delta u^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r}$$

$$+\mathcal{O}(\delta u)^2$$

Extremum if $J(u+\delta u,\lambda) - J(u,\lambda) = 0$ for any arbitray change δu

Euler Lagrange Equations

$$\boxed{\int \int f(\boldsymbol{r}, t) f^\star(\boldsymbol{r}', t) \mathrm{d}t \, u(\boldsymbol{r}') \, \mathrm{d}\boldsymbol{r}' = \lambda \, u(\boldsymbol{r})} \tag{35}$$

with the normalisation constraint (from $\delta\lambda$):

$$\int u(\boldsymbol{r}) u^\star(\boldsymbol{r}) \, \mathrm{d}\boldsymbol{r} = 1 \tag{36}$$

u is an eigenfunction of the Kernel (Hermitien in our case) which is the covariance function $\int f(\boldsymbol{r}, t) f^\star(\boldsymbol{r}', t) \, \mathrm{d}t$ and λ the eigenvalue. There several solutions and we note u_k the k^{th} eigenfunction (with eigenvalue λ_k).

Properties

- Eigenfunctions with different eigenvalues are orthogonal in the sense

$$\int u_k(\boldsymbol{r}) u_l^\star(\boldsymbol{r})\, \mathrm{d}\boldsymbol{r} = \delta_{kl} \tag{37}$$

- There exist a countably infinite number of eigenvalues
-

$$\boxed{\lambda_k = \int \left| \int f(\boldsymbol{r},t)\, u_k^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r} \right|^2 \mathrm{d}t \geq 0} \tag{38}$$

- Function f can be developped in terms of u_k

$$f(\boldsymbol{r},t) = \sum_k g_k^\star(t)\, u_k(\boldsymbol{r}), \quad g_k^\star(t) = \int u_k^\star(\boldsymbol{r}) f(\boldsymbol{r},t)\, \mathrm{d}\boldsymbol{r} \tag{39}$$

Series

Since $\lambda_k \geq 0$, we note $\rho_k = \sqrt{\lambda_k}$ (later called singular values) and sort them so that

$$\rho_1 \geq \rho_2 \geq \rho_3 \geq ... \geq \rho_k \geq \rho_{k+1} \geq ... \qquad (40)$$

$$f(\boldsymbol{r}, t) = \sum_{k=1}^{\infty} g_k^\star(t)\, u_k(\boldsymbol{r}), \qquad g_k^\star(t) = \int u_k^\star(\boldsymbol{r}) f(\boldsymbol{r}, t)\, \mathrm{d}\boldsymbol{r} \qquad (41)$$

If we normalize the temporal functions also

$$v_k(t) = \frac{1}{\sqrt{\int g_k^\star(t') g_k(t') \mathrm{d}t'}} g_k(t) = \frac{1}{\sqrt{\lambda_k}} g_k(t) \qquad (42)$$

so that

$$\int v_k^\star(t') v_k(t')\, \mathrm{d}t' = 1 \qquad (43)$$

Université de Liège

Series

We realise that

$$f(\boldsymbol{r},t) = \sum_{k=1}^{\infty} \rho_k \, v_k^\star(t) \, u_k(\boldsymbol{r}) \qquad (44)$$

$$\lambda_k = \rho_k^2 = \int \left| \int f(\boldsymbol{r},t) \, u_k^\star(\boldsymbol{r}) \mathrm{d}\boldsymbol{r} \right|^2 \mathrm{d}t \qquad (45)$$

Total variance

$$\int \int f(\boldsymbol{r},t) f^\star(\boldsymbol{r},t) \, \mathrm{d}\boldsymbol{r}\mathrm{d}t = \sum_{k=1}^{\infty} \rho_k^2 \qquad (46)$$

ρ_k^2: part of the variance explained by mode k.

Practical use

Most of the time, function f is not known but measured of modelled by a numerical model.

> Discretise the eigenvalue problem assuming f to be given at discrete points

Alternative: use directly discrete sets of data and try to find basis vectors (of finite dimension) the best fit the discrete data set on average

Classical approach, uniformely discrete data distribution

- We assume that we have a matrix $\mathbf{X}$ containing the observations, which is arranged such that the element i, j of the matrix is called $(\mathbf{X})_{ij}$ and is given by the value of the field $f(\boldsymbol{r}, t)$ at location $\boldsymbol{r}_i$ and moment t_j :

$$(\mathbf{X})_{ij} = f(\boldsymbol{r}_i, t_j). \tag{47}$$

- The field f is an observational field and contains thus all errors (instrumental, unresolved structures, etc).
- We can then write the matrix as a succession of n column vectors

$$\mathbf{X} = \left(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\right), \tag{48}$$

each of the column vectors $\mathbf{x}_j$ being the discrete state vector of size m at moment t_j.

$\mathbf{u}$: discrete eigenvector (of size m)

Discretised eigenvalue problem

$$\int f(\boldsymbol{r},t)\,u^\star(\boldsymbol{r})\mathrm{d}\boldsymbol{r} \quad \rightarrow \mathbf{u}^{\mathrm{T}}\mathbf{X} \tag{49}$$

where u^{T} is the transposed conjugate of the column vector $\mathbf{u}$

$$\int u(\boldsymbol{r})u^\star(\boldsymbol{r})\,\mathrm{d}\boldsymbol{r} \quad \rightarrow \mathbf{u}^{\mathrm{T}}\mathbf{u} \tag{50}$$

$$\boxed{\int\int f(\boldsymbol{r},t)f^\star(\boldsymbol{r}',t)\,u(\boldsymbol{r}')\,\mathrm{d}\boldsymbol{r}'\mathrm{d}t = \lambda\,u(\boldsymbol{r}) \quad \rightarrow \mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{u} = \lambda\mathbf{u}} \tag{51}$$

$$g_k^\star(t) = \int u_k^\star(\boldsymbol{r})f(\boldsymbol{r},t)\,\mathrm{d}\boldsymbol{r} \rightarrow \mathbf{g}_k^{\mathrm{T}} = \mathbf{u}_k^{\mathrm{T}}\mathbf{X} \tag{52}$$

$$v_k(t) = \frac{1}{\sqrt{\lambda_k}}g_k(t) \rightarrow \rho_k\mathbf{v}_k = \mathbf{X}^{\mathrm{T}}\mathbf{u}_k \tag{53}$$

and $\mathbf{X}\,\mathbf{v}_k = \rho_k\mathbf{u}_k$ and $\mathbf{X}^{\mathrm{T}}\mathbf{X}\,\mathbf{v}_k = \rho_k^2\mathbf{v}_k$

EOF's

$$\mathbf{X}\mathbf{X}^{\mathrm{T}}\,\mathbf{u} = \rho^2\mathbf{u}, \quad \text{size } m \tag{54}$$

$$\mathbf{X}^{\mathrm{T}}\mathbf{X}\,\mathbf{v} = \rho^2\mathbf{v}, \quad \text{size } n \tag{55}$$

showing that the spatial modes $\mathbf{u}$ are the eigenvectors of the time-averaged covariance matrix $\mathbf{X}\mathbf{X}^{\mathrm{T}}$ while $\mathbf{v}$ are the eigenvectors of the spatially-averaged covariance matrix $\mathbf{X}^{\mathrm{T}}\mathbf{X}$.
Eigenvectors are normalised, and by virtue of eqs. (54) and (55) (Hermitien eigenvalue problem), orthogonal:

$$\mathbf{u}_k^{\mathrm{T}}\mathbf{u}_l = \delta_{kl}, \quad \mathbf{v}_k^{\mathrm{T}}\mathbf{v}_l = \delta_{kl}, \tag{56}$$

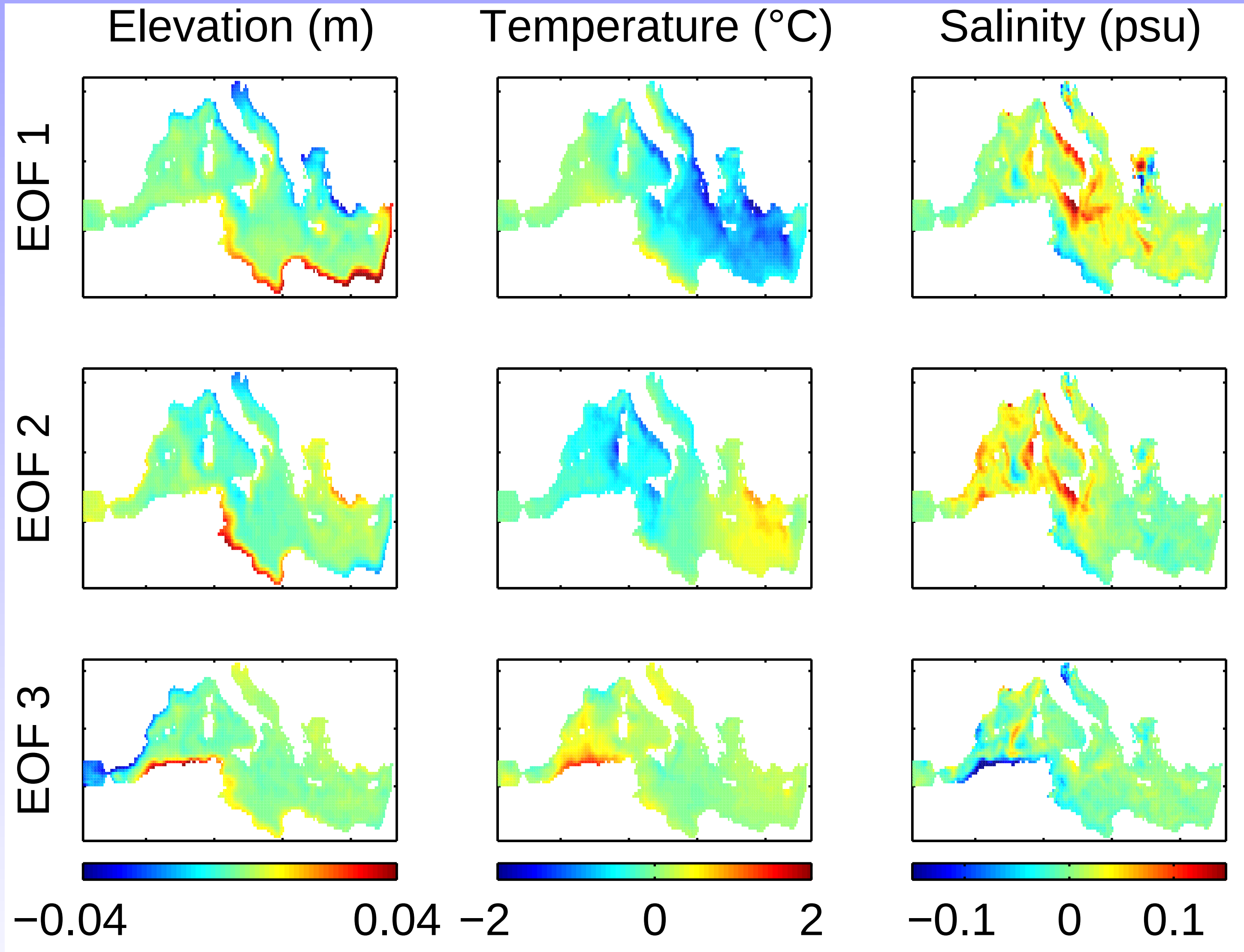
SVD

Only $q = \min(m, n)$ non-zero singular values and series decomposition (44) is now a matrix decomposition:

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^{\mathrm{T}} = \sum_{k=1}^{q} \rho_k \mathbf{u}_k \mathbf{v}_k^{\mathrm{T}} \tag{57}$$

where we have, as for matrix **X**, defined matrices **U** and **V** so that they have as columns k the eigenvectors $\mathbf{u}_k$ and $\mathbf{v}_k$ respectively, corresponding to the singular value ρ_k. Matrix **D** is then a rectangular matrix whose sole non-zero values are on its diagonal and $(\mathbf{D})_{kl} = \rho_k \delta_{kl}$.
This is the Singular Value Decomposition (SVD) of matrix **X**.

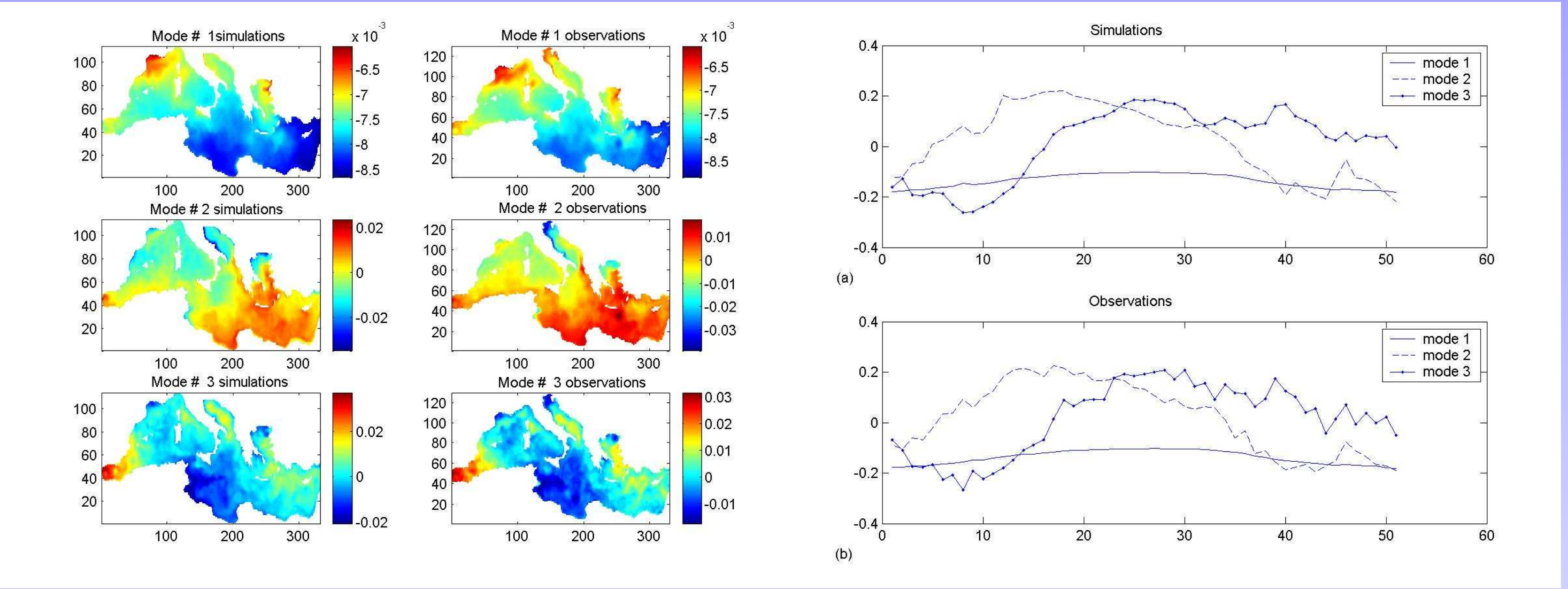
Multivariate Analysis: exemple



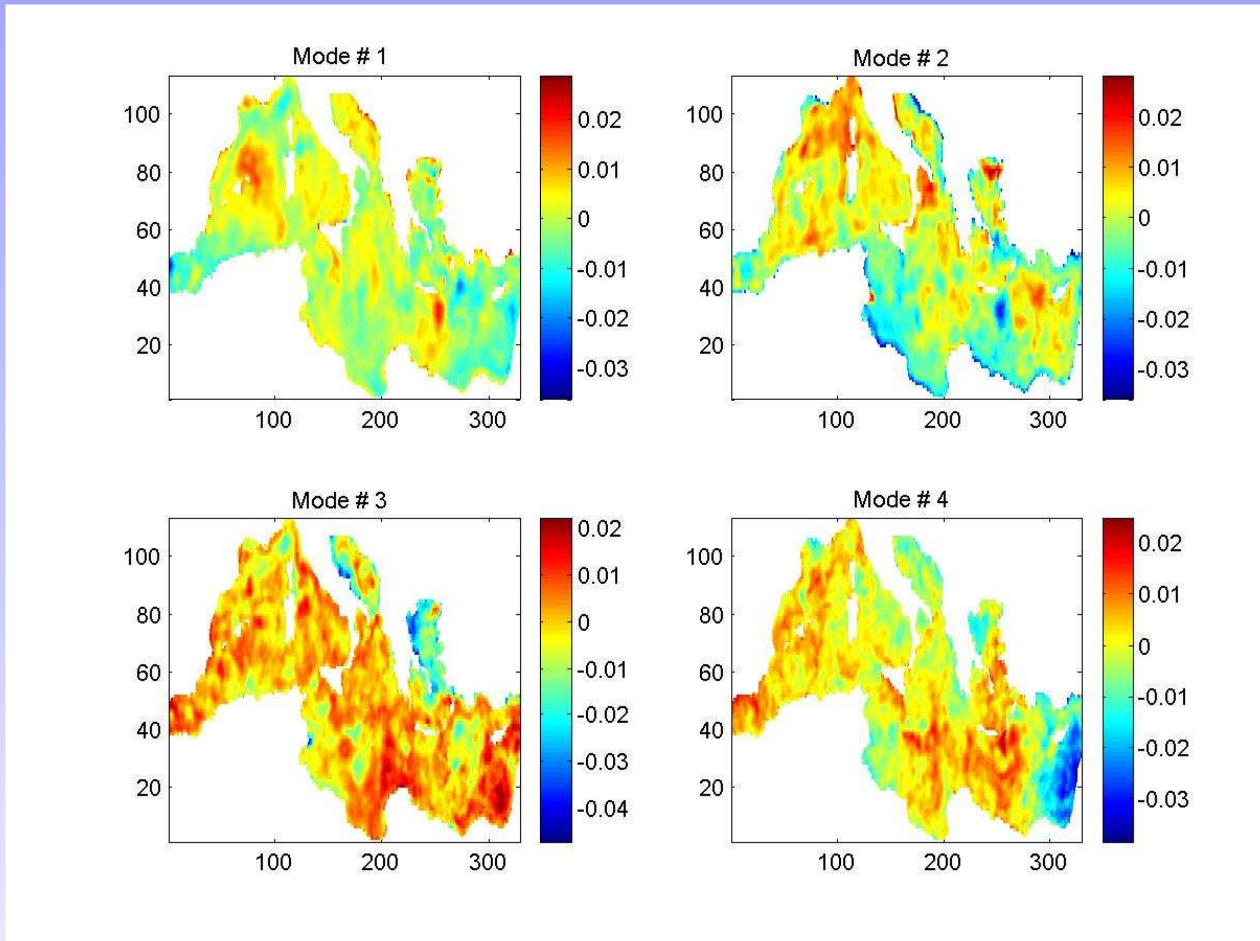
(60 modes to explain 80% variance)

MEDMEX intercomparison

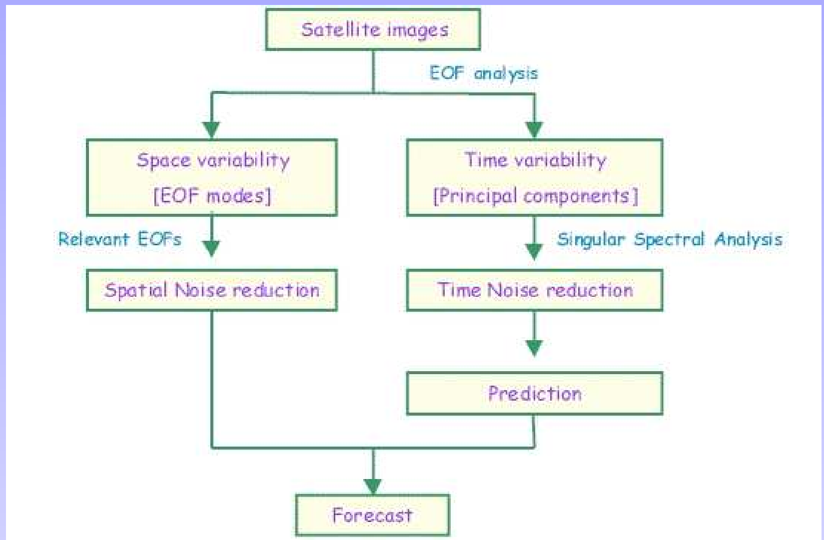
MFSTEP reanalysis

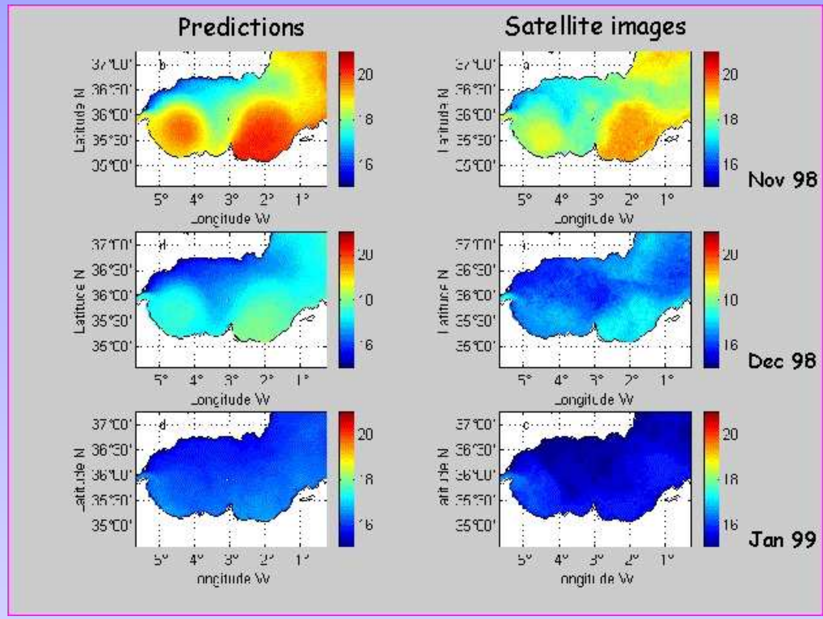


MFSTEP error modes



SOFT predictor





SVD decomposition for non-uniform data distribution

$$\int\int f(\boldsymbol{r},t)f^\star(\boldsymbol{r}',t)\mathrm{d}t\, u(\boldsymbol{r}')\,\mathrm{d}\boldsymbol{r}' = \lambda\, u(\boldsymbol{r}) \tag{58}$$

If data are not uniformely distributed, one can recast the problem into the (uniform) discretised space $\boldsymbol{\xi}$ and discretised time η if the position of the points to not change in time: i.e. $\boldsymbol{r} = \boldsymbol{r}(\boldsymbol{\xi})$ and if time stepping is identical for all spatial points. i.e. $t = t(\eta)$. Then integrals are modified by a change of variables and Jacobiens $\mathcal{J} = \left|\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\xi}}\right|$ and $\mathcal{K} = \frac{\partial t}{\partial \eta}$

$$\int\int f(\boldsymbol{\xi},\eta)f^\star(\boldsymbol{\xi}',\eta)\mathcal{K}(t)\mathrm{d}\eta\, u(\boldsymbol{\xi}')\,\mathcal{J}(\boldsymbol{\xi}')\mathrm{d}\boldsymbol{\xi}' = \lambda\, u(\boldsymbol{\xi}) \tag{59}$$

Asuming well behaved coordinate transformations (and suitable choice of signs) $\mathcal{K} > 0$ and $\mathcal{J} > 0$ so that one can recast the problem as follows

SVD decomposition for non-uniform data distribution

$$\int\int f(\boldsymbol{\xi},\eta) f^{\star}(\boldsymbol{\xi}',\eta)\mathcal{K}(t)\mathrm{d}\eta\, u(\boldsymbol{\xi}')\, \mathcal{J}(\boldsymbol{\xi}')\mathrm{d}\boldsymbol{\xi}' = \lambda\, u(\boldsymbol{\xi}) \qquad (60)$$

$$\int\int \sqrt{\mathcal{J}(\boldsymbol{\xi})\mathcal{K}(t)} f(\boldsymbol{\xi},\eta)\, f^{\star}(\boldsymbol{\xi}',\eta)\sqrt{\mathcal{J}(\boldsymbol{\xi}')\mathcal{K}(t)}\,\mathrm{d}\eta\sqrt{\mathcal{J}(\boldsymbol{\xi}')}u(\boldsymbol{\xi}')\,\mathrm{d}\boldsymbol{\xi}' = \lambda\sqrt{\mathcal{J}(\boldsymbol{\xi})}u(\boldsymbol{\xi}) \qquad (61)$$

Defining $\tilde{f} = \sqrt{\mathcal{J}\mathcal{K}}\, f$ and $\tilde{u} = \sqrt{\mathcal{J}}\, u$ one recovers the standard form which can then be discretised and solved by SVD decomposition. In practice it amounts at modifying the data matrix **X** by pre- and postmultiplying by a diagonal matrix whose elements are the square-roots of the jacobiens (or griz size). After SVD decomposition, the "real" EOF can be recoverd by multiplying the eignevectors by a diagonal matrix whose elements are the inverse of the square roots of the Jacobian.

Back to EOFs