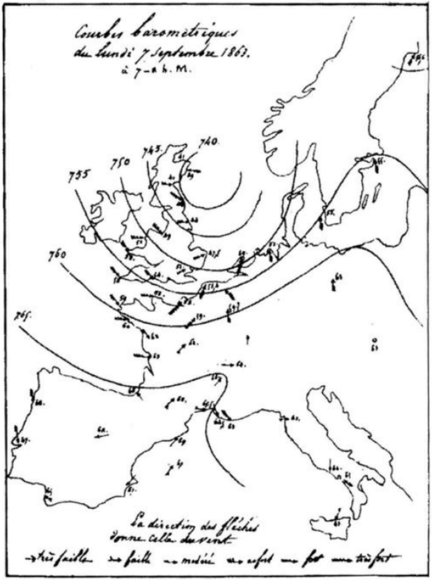
Diva Lecce 2016

Diva in 2 dimensions

Sylvain Watelet, Alexander Barth, Aida Alvera-Azcárate,
Mohamed Ouberdous, Charles Troupin, & Jean-Marie Beckers

Acknowledgements: SeaDataNet, EMODnet Chemistry,
EMODnet Biology, STARESO

Interpolation 150 years ago…



What is Diva?

Data
Interpolating
Variational
Analysis

What is Diva?

- a method to produce gridded fields
- a set of bash scripts and Fortran programs

What is not Diva?

- a plotting tool
- a *black-box*
- a numerical model

A little bit of history

Code development (1990-1996)

- Variational Inverse Method (VIM) (Brasseur, 1991, JMS, JGR)
- cross-validation (Brankart and Brasseur, 1996, JAOT)
- error computation (Brankart and Brasseur, 1998, JMS; Rixen et al., 2000, OM)

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

- set of bash scripts (`divamesh`, `divacalc`, ...)
- Fortran executables
- parameters optimization tools
- Matlab/Octave scripts for plotting

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

3D-analysis (2007-2008)

- superposition of 2D layers
- automated treatment and optimization
- stability constraint (Ouberdous et al.)

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

3D-analysis (2007-2008)

4D-analysis (2008-2009)

- start from ODV spreadsheet
- *detrending* (with J. Carstensen, DMU)
- NetCDF 4-D climatology files

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

3D-analysis (2007-2008)

4D-analysis (2008-2009)

Web tools

- On-line analysis (Barth et al., 2010, Adv. Geosci.) http://gher-diva.phys.ulg.ac.be/web-vis/diva.html
- Climatology viewer: http://gher-diva.phys.ulg.ac.be/web-vis/clim.html

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

3D-analysis (2007-2008)

4D-analysis (2008-2009)

Web tools

2011-2012

- multivariate approach
- data transformation tools
- 4-D graphical interface
- implementation of *source/decay* terms
- advanced error computation (Troupin et al., 2012, OM)

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

3D-analysis (2007-2008)

4D-analysis (2008-2009)

Web tools

2011-2012

2013-2015

- Modernisation of the code structure
- n-dimensional generalisation
- optimized and approximate error calculations (clever poor man)

A little bit of history

Code development (1990-1996)

2D-analysis (2006-2007)

3D-analysis (2007-2008)

4D-analysis (2008-2009)

Web tools

2011-2012

2013-2015

On-going:

- Analysis at a specific distance from the bottom
- Correlated observations errors (data weighting)
- ... ☕

A little bit of history

Code development	(1990-1996)
2D-analysis	(2006-2007)
3D-analysis	(2007-2008)
4D-analysis	(2008-2009)
Web tools	
2011-2012	
2013-2015	
On-going:	
General:	$\boxed{\text{user-driven developments}}$

Diva history

4.6.7

Released in October 2014.

New features

- Transformation of user relative length or advection fields files (ascii format) into the gher binary format, via a run of Diva (new script "asctobin")

New bug fixes

- Correction of time axis and climatology_bounds in Netcdf output files (diva3Dwrt.F,diva4Dwrt.F,dv4DYRwrt.F,dv3DncYRw.F)
- Correction of some attributes in 4D netcdf (databins, snr, cl, varbak) (dv3DncYRw.F, diva3Dsub)
- Update of driver files (also in Example4D)

4.6.6

Released in September 2014.

New features

- Check for severe errors in DIVA 3D/4D (script "godiva") + simple errors and warnings
- Possibility of binning the data before the parameters optimization (script "divabin" + program "binning_lines.f90")
- Variable correlation length, depending on depth (script "divarlvardepth" + program "rlvardepth.f90")

New bug fixes

- Correction of the example in 4D (datasource)
- Correction of the script divaguessformODV4
- Exact match needed between variable name in "varlist" and its real name in the data file.

4.6.5

Released in April 2014.

New features

New bug fixes

- "end of line" problems under Windows (file "datasource")
- Portability of scripts using the "sort" command

- http://modb.oce.ulg.ac.be/mediawiki/index.php/New_Diva_Features

Diva related tools

Diva: base tool (command line), 2D analysis

Godiva: automatic repetition of 2D analysis

Diva-on-web: 2D analysis with your data on our server

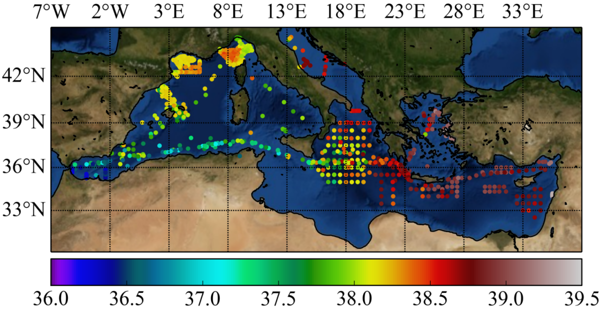
OceanBrowser: visualisation tool of 4D NetCDF files

divand: multi-dimension analysis (lon, lat, time, depth)

divaformatlab: wrapper to use in matlab

Clone-diva-x.x.x: virtual machine containing diva-x.x.x + other stuff (gfortran, netcdf,...)

Common problem

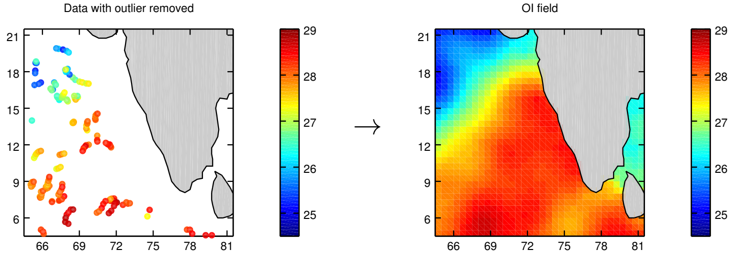


Appears when

- trying to produce maps
- calculate volume averages
- prepare initial conditions for models
- quality control of data
- ...

The gridding problem

Gridding is the determination of a field $\phi(\boldsymbol{r})$, on regular grid of positions $\boldsymbol{r}$ based on arbitrarily located observations. Often the vector $\boldsymbol{r}$ is on a 2D, 3D or even 4D space.



- The fewer observations are available, the harder the gridding problem is
- In oceanography, *in situ* observations are sparse
- Observations are inhomogeneously distributed in space and time (more observations in the coastal zones and in summer)
- The variability of the ocean is the sum of various processes occurring at different spatial and temporal scales.

The gridding problem

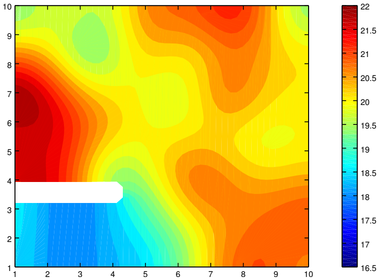


Figure 1: Example of oceanographic field.

- Figure 1 shows an idealized square domain with a barrier (*e.g.* a peninsula or a dike).
- This field is the true field that we want to reconstruct based on observations. Let's assume that the field represents temperature.
- The barrier suppresses the exchanges between each side of the barrier.
- The field varies smoothly over some length-scale

Sampling locations

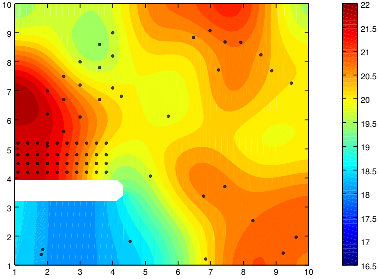


Figure 2: Sampling locations within the domain

- In regions where a measurement campaign has been carried out, a higher spatial coverage is achieved.
- Large gaps are also present.
- Based on the value of the field at the shown location, we will estimate the true field.

True field at the sampling locations

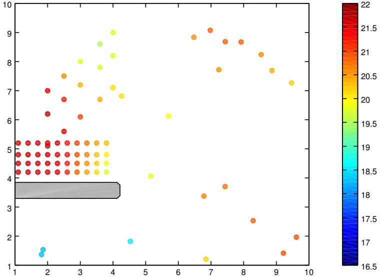


Figure 3: Value of the true field extract at the location of the observations.

- only the value of the observations is shown
- some information about the position of the structures and fronts is lost
- no method can provide exactly the true field.
- the more information about its structure and evolution we include in the analysis, to close we can get to the true field.

Observation errors

Observations are in general affected by different error sources and other "problems" that need to be taken into account:

1. Instrumental errors (limited precision or possible bias of the sensor)
2. Representative errors: the observations do not necessarily corresponds to the field we want to obtain. For example, we want to have a monthly average, but the observations are instantaneous (or averages over a very short period of time).
3. Synopticity errors: all observations are not taken at the same time.
4. Other errors sources: human errors (e.g. permutation of longitude and latitude), transmission errors, malfunctioning of the instrument, wrong decimal separators...

Quality control is an important step to exclude suspicious data from the analysis. But since this is a subjective decision, the data should never be deleted but flagged as suspicious or bad data.

Observation errors

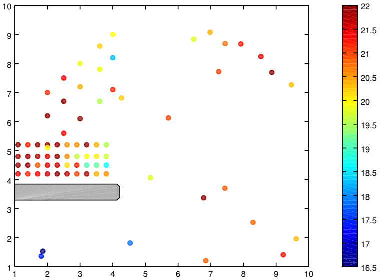
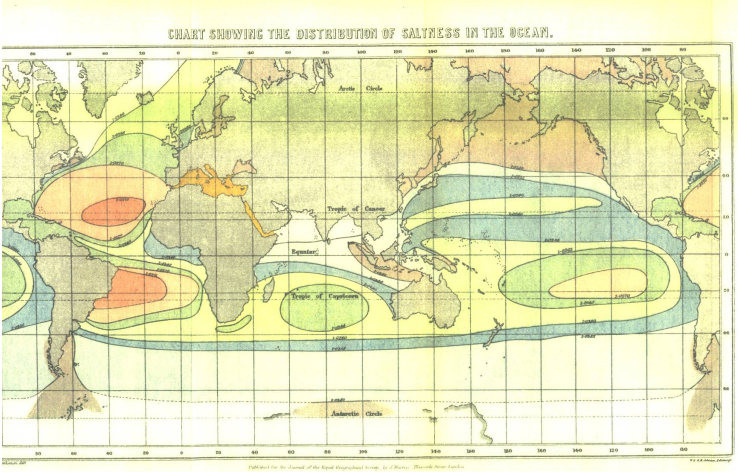


Figure 4: Observation with errors

In figure 4, a random perturbation was added to the observation shown in figure 3. This simulates the impact of the different error sources. To simplify matters, each observation was perturbed independently.

Solutions - Subjective methods



Solutions - Subjective methods

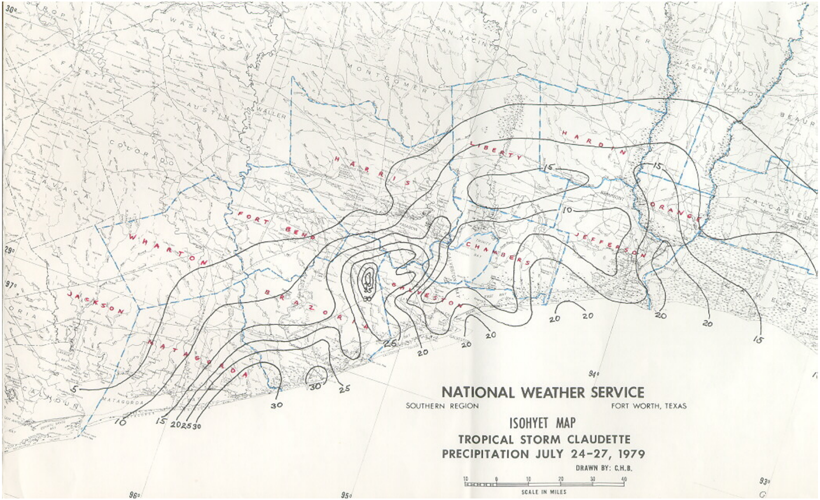


Figure 5: *Isohyet (lines of constant precipitation) drawn by hand (from* `http://www.srh.noaa.gov/hgx/hurricanes/1970s.htm`*)*

Interpolation or Analysis ?

Because observations have errors, it is always better to produce a field approximation and never a strict interpolation.

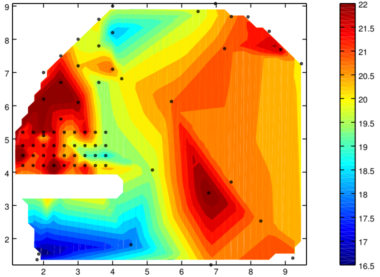


Figure 6: Gridded field using linear interpolation. This method is implemented in the function `griddata` *of Matlab and GNU Octave.*

- Figure 6 shows what would happen if the observations would have been interpolated linearly.
- The domain is decomposed into triangles where the vertices are the location of the data points based on the Delaunay triangulation.
- Within each triangle, the value is interpolated linearly.

Solutions - Objective methods

- Subjective method is not sufficiently ... objective.
- Data Assimilation: region and model dependent.

$\Rightarrow$ Objective analysis of data that are anomalies with respect to a background field $\varphi_b(\boldsymbol{r})$.

As opposed to the subjective method, objective analysis techniques aim to use mathematical formulations to infer the value of the field at unobserved locations based on the observation d_j. Most objective methods can be expressed as a linear combination of data anomalies d_j using weights w_j:

$$\varphi(\boldsymbol{r}) = \varphi_b(\boldsymbol{r}) \; + \sum_{j=1}^{N_d} w_j d_j \tag{1}$$

The field $\varphi(\boldsymbol{r})$ can be evaluated in any position $\boldsymbol{r}$, hence gridding is possible. The background field (or first guess) φ_b is defined *a priori* and anomalies calculated with respect to this reference field (for example a climatological average). There are several ways to define the weighting function w_j, which result in different gridding techniques.

Cressman method

Cressman weights depend only on the distance r between the location $\boldsymbol{r}$ where the value of the field should be estimated and the location of the observation $\boldsymbol{r}_j$:

$$r = |\boldsymbol{r} - \boldsymbol{r}_j| \tag{2}$$

The weights are then parameterized according to,

$$\begin{aligned} \tilde{w}(r) &= \tfrac{R^2-r^2}{R^2+r^2} \quad \text{for} \quad r < R \\ &= 0 \quad\quad\quad\; \text{for} \quad r \geq R \end{aligned} \tag{3}$$

The weights as a function of distance are shown in figure 7. Weights must be scaled by their sum to ensure no bias.

$$w_j = \tilde{w}_j / \sum_j \tilde{w}_j \tag{4}$$

Cressman method

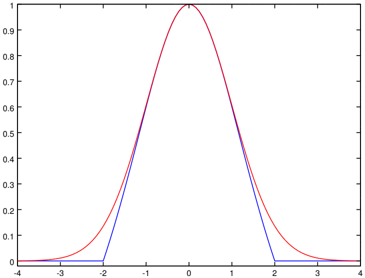


Figure 7: Cressman weights for $R = 2$ (blue) and Barnes weights for $R = 1$ (red).

The search radius R is the typical control parameter and defines the length-scale over which an observation is used. This length scale can be made to vary in space depending on data coverage and/or physical scales. This parameter is chosen by the users based on their knowledge of the domain and the problem.

Cressman method

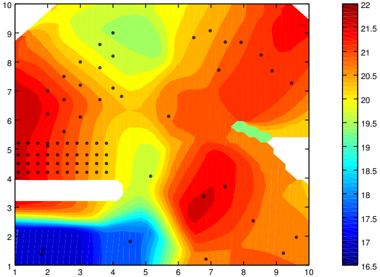


Figure 8: Gridded field by Cressman weighting

The Cressman weighting is a very simple and numerically quite efficient method. However, it suffers from some limitations which are apparent in figure 8.

- No estimate can be obtained at locations when no observation is located within the R.
- In regions with very few observations, the method can return a discontinuous field.
- The presence of barriers cannot be taken into account easily.
- All observations are assumed to have a similar error variance since the weighting is based only on distance.

Barnes method

As a variant of the Cressman weights, other weighting functions can be defined. In the Barnes scheme, the weights are defined using a Gaussian function:

$$\tilde{w}(d) = e^{-\frac{d^2}{2R^2}} \tag{5}$$

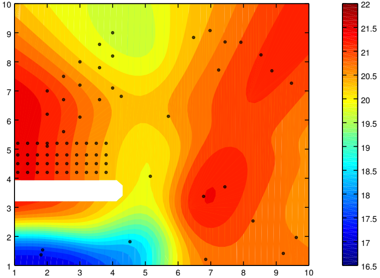


Figure 9: Gridded field using Barnes weights

Since the Barnes weights are never zero, in principle all observations are used for the gridding. An estimation can be obtained everywhere (which can be accurate or not). Artificial discontinuities are avoided using the Barnes weights (figure 9).

Estimation

- Observer 1: 14°
- Observer 2: 16°

Your best guess ?

Estimation

- Observer 1: 14°
- Observer 2: 16°

Your best guess ?

15°

Estimation

- Observer 1: 14°
- Observer 2: 16°

Your best guess ?

15°

But what if observer 1 uses digital thermometer and observer 2 his finger ?

Estimation

- Observer 1: 14°
- Observer 2: 16°

Your best guess ?

15°

But what if observer 1 uses digital thermometer and observer 2 his finger ?

Best guess probably near 14°.

Estimation

- Observer 1: 14°
- Observer 2: 16°

Your best guess ?

15°

But what if observer 1 uses digital thermometer and observer 2 his finger ?

Best guess probably near 14°.

Exploit knowledge of errors !

Optimal estimate

$$T_1 = T^t + \epsilon_1, \quad \langle \epsilon_1 \rangle = 0, \quad T_2 = T^t + \epsilon_2, \quad \langle \epsilon_2 \rangle = 0 \tag{6}$$

statistical average, denoted by $\langle \quad \rangle$ with unbiased estimates $\langle \epsilon_* \rangle = 0$
Linear estimate

$$T = w_1 T_1 \; + \; w_2 T_2 \; = \; (w_1 + w_2) T^t \; + \; (w_1 \epsilon_1 + w_2 \epsilon_2) \tag{7}$$

$$\langle T \rangle = (w_1 + w_2) T^t, \tag{8}$$

we obtain an unbiased estimate of the true state if we take
$w_1 + w_2 = 1$. This leaves one parameter free to chose: w_2

> Exploit knowledge on errors to find optimal value of w_2

Choice of weighting ?

$$T^a = (1 - w_2)T_1 + w_2 T_2 \; = \; T_1 + w_2(T_2 - T_1) \tag{9}$$

while in reality there is an error

$$T^a - T^t = (1 - w_2)\epsilon_1 + w_2\epsilon_2, \tag{10}$$

This error is zero on average but its variance is not zero:

$$\left\langle (T^a - T^t)^2 \right\rangle = (1-w_2)^2 \left\langle \epsilon_1^2 \right\rangle + w_2^2 \left\langle \epsilon_2^2 \right\rangle + 2(1-w_2)w_2 \left\langle \epsilon_1\epsilon_2 \right\rangle \tag{11}$$

The actual errors ϵ_1 and ϵ_2 are not known, but the error variance $\left\langle \epsilon_1^2 \right\rangle$ are. Often we can reasonably suppose that the errors ϵ_1 and ϵ_2 are uncorrelated $\left\langle \epsilon_1\epsilon_2 \right\rangle = 0$. The error variance $\left\langle \epsilon^2 \right\rangle$ of the analysis is

$$\left\langle \epsilon^2 \right\rangle = (1 - w_2)^2 \left\langle \epsilon_1^2 \right\rangle + w_2^2 \left\langle \epsilon_2^2 \right\rangle. \tag{12}$$

So what ?

Minimisation

$$\left\langle \epsilon^2 \right\rangle = (1 - w_2)^2 \left\langle \epsilon_1^2 \right\rangle + w_2^2 \left\langle \epsilon_2^2 \right\rangle . \tag{13}$$

Naturally, the best estimate for T is the one with the lowest expected error variance and we will use w_2, which minimizes the right-hand side:

$$w_2 = \frac{\left\langle \epsilon_1^2 \right\rangle}{\left\langle \epsilon_1^2 \right\rangle + \left\langle \epsilon_2^2 \right\rangle} \tag{14}$$

Best estimate

With (14) we obtain the minimal error variance

$$\langle \epsilon^2 \rangle = \frac{\langle \epsilon_1^2 \rangle \langle \epsilon_2^2 \rangle}{\langle \epsilon_1^2 \rangle + \langle \epsilon_2^2 \rangle} = \left(1 - \frac{\langle \epsilon_1^2 \rangle}{\langle \epsilon_1^2 \rangle + \langle \epsilon_2^2 \rangle} \right) \langle \epsilon_1^2 \rangle \,, \tag{15}$$

while the estimate of the temperature itself reads

$$T^a = T_1 \; + \; \left(\frac{\langle \epsilon_1^2 \rangle}{\langle \epsilon_1^2 \rangle + \langle \epsilon_2^2 \rangle} \right) (T_2 - T_1) \,. \tag{16}$$

Error variance on the combination of T_1 and T_2 is smaller than both $\langle \epsilon_1^2 \rangle$ and $\langle \epsilon_2^2 \rangle$.

Optimal Interpolation I

Same problem but data distributed in space and *a priori* information on background (with variance σ^2).
Weighting of background (zero value when working with anomalies and σ^2 local variance and covariances between points) information and data points (observed values and observational error variance)

- "Model forecast": Background field.
- Need for covariance of the background field between data points: each element i, j of $\mathbf{B}$ provides the covariance between points in location i and j. Covariance between a given point and all data points is stored in column vector $\mathbf{c}$ and the local variance at the analysis point is noted σ^2.
- Analysis ϕ of anomaly $\mathbf{y}$ with respect to background leads to spatial analysis at any desired location of covariance between any two points is known.

Optimal Interpolation II

$$\phi = \mathbf{c}^t (\mathbf{B} + \mathbf{R})^{-1} \mathbf{y} \tag{17}$$

with a local error variance of the analysis

$$\epsilon_a^2 = \sigma^2 - \mathbf{c}^t (\mathbf{B} + \mathbf{R})^{-1} \mathbf{c} \tag{18}$$

Note that inversion of matrix is needed (cost increases as the cube of number of data points).

Background covariance

Problem, how to specify background covariances (between all data points and between data points and the desired analysis location).

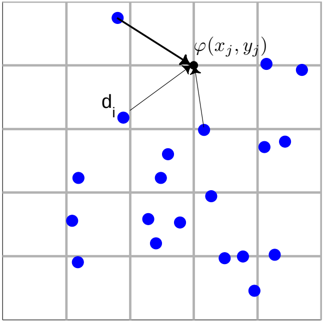
- c_i= covariance between location of the analysis and data location of point i = $C(x, x_i)$
- B_{ij}=covariance between location of data point i and location of point j= $C(x_i, x_j)$

Approaches

- Normally obtained via statistics on data. Seldom possible (noticable exception: satellite images).
- Standard OI: via functions $B_{ij} = f(r/L)$ where r is the distance between points i and j, but still function f needs to be determined. L is the so-called correlation length. Here statistics on all data couples as a function of distance. Example: $f = \sigma^2 \exp(-r^2/L^2)$.
- Via functionals (see Kernel of DIVA later)

DIVA: Data-Interpolating Variational Analysis

N_d data points d_i • $\rightarrow$ gridded field



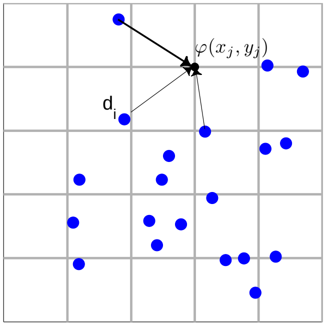
Formulation: minimize cost function $J[\varphi]$

$$\min J[\varphi] = \sum_{i=1}^{N} \mu_i \left[d_i - \varphi(x_i, y_i)\right]^2$$

$$+ \int_D \left(\boldsymbol{\nabla\nabla}\varphi : \boldsymbol{\nabla\nabla}\varphi + \alpha_1 \boldsymbol{\nabla}\varphi \cdot \boldsymbol{\nabla}\varphi + \alpha_0 \varphi^2\right) \mathrm{d}D$$

DIVA: Data-Interpolating Variational Analysis

N_d data points d_i • → gridded field



Formulation: minimize cost function $J[\varphi]$

$$\min J[\varphi] = \sum_{i=1}^{N} \mu_i \left[d_i - \varphi(x_i, y_i)\right]^2 \qquad \text{data–analysis misfit}$$

$$+ \quad \int_D \left(\boldsymbol{\nabla\nabla}\varphi : \boldsymbol{\nabla\nabla}\varphi + \alpha_1 \boldsymbol{\nabla}\varphi \cdot \boldsymbol{\nabla}\varphi + \alpha_0 \varphi^2\right) \mathrm{d}D \qquad \text{field regularity}$$

Analysis parameters are related to data

Non-dimensional version:

$$L = \text{length scale} \quad \rightarrow \quad \tilde{\boldsymbol{\nabla}} = L \boldsymbol{\nabla} \tag{19}$$

$$\rightarrow \quad D = L^2 \tilde{D} \tag{20}$$

Analysis parameters are related to data

Non-dimensional version:

$$L = \text{length scale} \quad \rightarrow \quad \tilde{\boldsymbol{\nabla}} = L\boldsymbol{\nabla} \tag{19}$$

$$\rightarrow \quad D = L^2 \tilde{D} \tag{20}$$

$$\begin{aligned}\tilde{J}[\varphi] &= \sum_{i=1}^{N} \mu_i L^2 [d_i - \varphi(x_i, y_i)]^2 \\ &+ \int_{\tilde{D}} \left(\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi : \tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi + \alpha_1 L^2 \tilde{\boldsymbol{\nabla}}\varphi \cdot \tilde{\boldsymbol{\nabla}}\varphi + \alpha_0 L^4 \varphi^2 \right) \mathrm{d}\tilde{D}\end{aligned}$$

Analysis parameters are related to data

Non-dimensional version:

$$L = \text{length scale} \quad \rightarrow \quad \tilde{\boldsymbol{\nabla}} = L\boldsymbol{\nabla} \tag{19}$$

$$\rightarrow \quad D = L^2 \tilde{D} \tag{20}$$

$$\tilde{J}[\varphi] = \sum_{i=1}^{N} \mu_i L^2 [d_i - \varphi(x_i, y_i)]^2 + \int_{\tilde{D}} \left(\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi : \tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi + \alpha_1 L^2 \tilde{\boldsymbol{\nabla}}\varphi \cdot \tilde{\boldsymbol{\nabla}}\varphi + \alpha_0 L^4 \varphi^2 \right) \mathrm{d}\tilde{D}$$

- $\alpha_0 \rightarrow L$ for which data anomaly term $\simeq$ regularity term: $\alpha_0 L^4 = 1$

Analysis parameters are related to data

Non-dimensional version:

$$L = \text{length scale} \quad \rightarrow \quad \tilde{\boldsymbol{\nabla}} = L\boldsymbol{\nabla} \tag{19}$$

$$\rightarrow \quad D = L^2\tilde{D} \tag{20}$$

$$\begin{aligned}\tilde{J}[\varphi] &= \sum_{i=1}^{N} \mu_i L^2 [d_i - \varphi(x_i, y_i)]^2 \\ &+ \int_{\tilde{D}} \left(\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi : \tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi + \alpha_1 L^2 \tilde{\boldsymbol{\nabla}}\varphi \cdot \tilde{\boldsymbol{\nabla}}\varphi + \alpha_0 L^4 \varphi^2 \right) \mathrm{d}\tilde{D}\end{aligned}$$

- $\alpha_0 \rightarrow L$ for which data anomaly term $\simeq$ regularity term: $\alpha_0 L^4 = 1$
- $\alpha_1 \rightarrow$ influence of gradients: $\alpha_1 L^2 = 2\xi, \qquad \xi = 1$

Analysis parameters are related to data

Non-dimensional version:

$$L = \text{length scale} \quad \rightarrow \quad \tilde{\boldsymbol{\nabla}} = L\boldsymbol{\nabla} \tag{19}$$

$$\rightarrow \quad D = L^2 \tilde{D} \tag{20}$$

$$\tilde{J}[\varphi] = \sum_{i=1}^{N} \mu_i L^2 [d_i - \varphi(x_i, y_i)]^2 + \int_{\tilde{D}} \left(\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi : \tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi + \alpha_1 L^2 \tilde{\boldsymbol{\nabla}}\varphi \cdot \tilde{\boldsymbol{\nabla}}\varphi + \alpha_0 L^4 \varphi^2 \right) \mathrm{d}\tilde{D}$$

- $\alpha_0 \rightarrow L$ for which data anomaly term $\simeq$ regularity term: $\alpha_0 L^4 = 1$
- $\alpha_1 \rightarrow$ influence of gradients: $\alpha_1 L^2 = 2\xi, \quad \xi = 1$
- $\mu_i L^2 \rightarrow$ weight on data: $\mu_i L^2 = 4\pi \dfrac{\text{signal}}{\text{noise}_i}$

Analysis parameters are related to data

Non-dimensional version:

$$L = \text{length scale} \quad \rightarrow \quad \tilde{\boldsymbol{\nabla}} = L\boldsymbol{\nabla} \tag{19}$$

$$\rightarrow \quad D = L^2 \tilde{D} \tag{20}$$

$$\begin{aligned}
\tilde{J}[\varphi] &= \sum_{i=1}^{N} \mu_i L^2 [d_i - \varphi(x_i, y_i)]^2 \\
&+ \int_{\tilde{D}} \left(\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi : \tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{\nabla}}\varphi + \alpha_1 L^2 \tilde{\boldsymbol{\nabla}}\varphi \cdot \tilde{\boldsymbol{\nabla}}\varphi + \alpha_0 L^4 \varphi^2 \right) \mathrm{d}\tilde{D}
\end{aligned}$$

Coefficients α_0, α_1 and μ_i related to

1. Correlation length L
2. Signal-to-noise λ
3. Observational noise standard deviation ϵ_i^2

Main analysis parameters

Correlation length L:

- Measure of the *influence* of data points
- Estimated by a least-square fit of the covariance function

Signal-to-noise ratio λ:

- Measure of the *confidence* in data
- Estimated with Generalized Cross Validation techniques

Correlation function

In variational analysis, the correlation function is not specified `a priori`, but its form is determined by the differential in equation (19). For an infinite domain, one can show that for $\alpha_0 = L^{-4}$ and $\alpha_1 = 2L^{-2}$ the correlation function is given by:

$$C(r) = \frac{r}{L} K_1 \left(\frac{r}{L} \right) \tag{21}$$

where r is the Euclidean distance, L is the correlation length and K_1 is the modified Bessel function.

Correlation function

Correlation functions can be examined by analysing a single point with high signal-to-noise ratio and no background field (Fig. 10). In this particular case, we performed an analysis with a point located at the center $(0, 0)$ of a square domain. The correlation length is equal to 1 and the signal-to-noise ratio is taken equal to 1000.

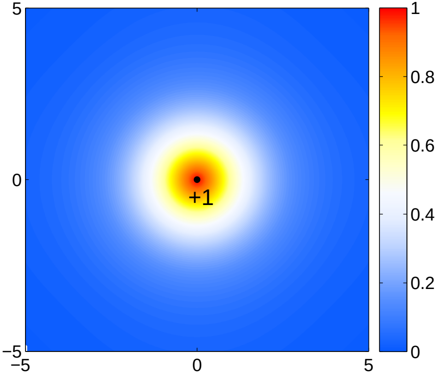


Figure 10: Analysis of a single data point with high signal-to-noise ration and no background field.

Correlation function

Figure 11 shows both the exact solution and the correlation function by a single point analysis. The two curves are close to each other, the differences between the two curves being only due to the boundaries.

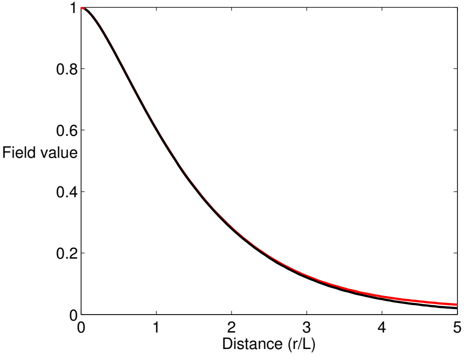
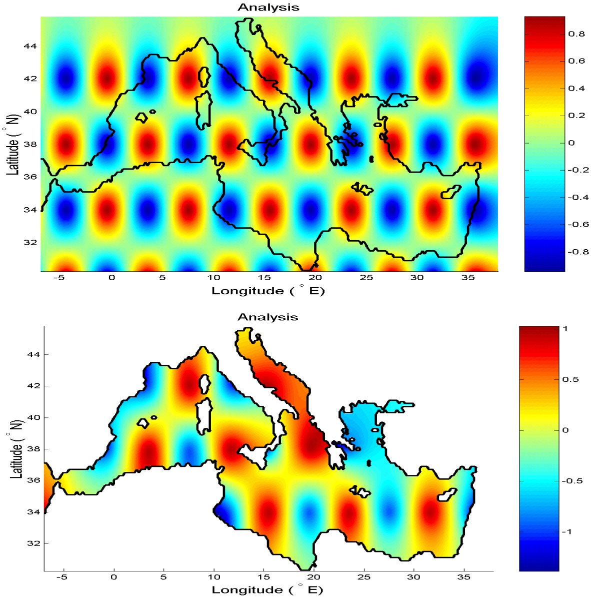


Figure 11: The theoretical Kernel function is given in red, while the black curve comes from the analysis of a single point with a unit value.

The Kernel function can be used to calibrate Diva parameters $(\alpha_0, \alpha_1, \mu)$ so as to fit observed covariance functions. This principle is used by the tool `divafit` which helps one to estimate the correlation length.

Illustration of covariance functions



Parameter calibration : L

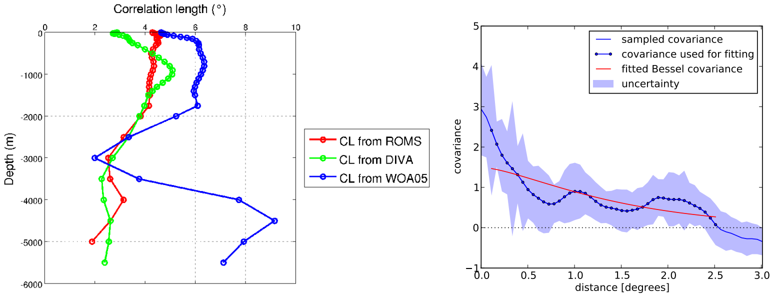


Figure 12: Spatial coherence of parameters: here correlation length obtained with covariance fitting (Troupin et al., 2010).

Parameter calibration : SNR I

Generalities

Let us consider the vector $\boldsymbol{d}$ containing the N data anomalies. Objective analysis of $\boldsymbol{d}$ leads to analysed field with minimal expected error variance. The analysis φ^a at any location $\boldsymbol{r}$ is given by

$$\varphi^a(\boldsymbol{r}) = \boldsymbol{c}(\mathbf{B} + \mathbf{R})^{-1}\mathbf{d} \tag{22}$$

where $\boldsymbol{c}$ is a vector containing the background covariance between the point in which the analysis is to be performed and all data point locations. The optimal interpolation is based on the *background covariance* matrix $\mathbf{B}$ and *error covariance* matrix $\mathbf{R}$ of the data. Let us call $\tilde{\boldsymbol{d}}$, the analysis vector at data points. The two vectors $\boldsymbol{d}$ and $\tilde{\boldsymbol{d}}$ can be related by the expression:

$$\tilde{\boldsymbol{d}} = \mathbf{A}\boldsymbol{d} \tag{23}$$

where the matrix $\mathbf{A}$, used to perform the analysis at the data points, is calculated according to

Parameter calibration : SNR II

Generalities

$$\mathbf{A} = \mathbf{B}(\mathbf{B}+\mathbf{R})^{-1}. \tag{24}$$

The *data-covariance* matrix is the statistical average $\langle \quad \rangle$ of data products:

$$\langle \boldsymbol{d}\, \boldsymbol{d}^{\mathrm{T}} \rangle = \mathbf{B} + \mathbf{R}, \tag{25}$$

where $^{\mathrm{T}}$ it the transposed matrix or vector. For uncorrelated observational errors, error-covariance matrix $\mathbf{R}$ is diagonal, with a variance ϵ_i^2 for point i, *i.e.*

$$\mathbf{R} = \mathrm{diag}(\epsilon_i^2)$$

In that case, we can show that the variance of expected misfit at point i is

Parameter calibration : SNR III

Generalities

$$\left\langle \left(d_i - \tilde{d}_i\right)^2 \right\rangle = \epsilon_i^2 (1 - A_{ii}). \tag{26}$$

In practice covariance matrices are known only imperfectly: their structure is often considered to be fixed, but with imperfectly known amplitude. In other words it is often assumed that

Parameter calibration : SNR

Generalities

$$\mathbf{B} = \sigma^2 \tilde{\mathbf{B}} \tag{27}$$

$$\mathbf{R} = \epsilon^2 \tilde{\mathbf{R}} \tag{28}$$

$$\mathbf{c} = \sigma^2 \tilde{\mathbf{c}} \tag{29}$$

with non-dimensional correlation matrixes $\tilde{\mathbf{B}}$, $\tilde{\mathbf{R}}$, $\tilde{\mathbf{c}}$

$$\phi = \tilde{\mathbf{c}}^t \left(\tilde{\mathbf{B}} + \frac{1}{\lambda} \tilde{\mathbf{R}} \right)^{-1} \mathbf{y} \tag{30}$$

with signal-to noise ratio

$$\lambda = \frac{\sigma^2}{\epsilon^2} \tag{31}$$

Also the error field is only depending on the ratio.

Parameter calibration : SNR I

Ordinary Cross Validation (OCV)

The objective is to optimize the parameter λ by searching for its value for which the analysis has a minimal error. For this reason, we need to find a proxy norm that we will be able to minimize. As the difference of the analysis with the true field is not available, we could try to work with the difference of the analysed field at the data points with respect to the original data field:

$$\theta_i^2 = (d_i - \tilde{d}_i)^2. \tag{32}$$

If we try to minimize this norm, we will get an infinite signal-to-noise ratio and a perfect data-analysis fit. This is because the analysis at the data point is directly influenced by the corresponding data.

Parameter calibration : SNR II

Ordinary Cross Validation (OCV)

- To avoid this inconvenience, the solution is to calculate the difference of the data value with respect to the analysed field in which the data under investigation was not taken into account. This is called the *Ordinary Cross Validation* and is, with a practical trick, implemented in `divacv`. To make this estimate robust, the analysis has to be repeated over a large number of data points, increasing the computing cost, so that OCV is generally too expensive to perform unless a trick as in `divacv` can be used (there the analysis can be done without actually disregarding a data point but by correcting the difference).
- Variants of OCV take out several points at once to calculate error estimates and repeat the exercise several times to make estimates robust. In Diva this options are available in `divacvrand`. A variant is `divacvclasses` in which all data from a given class (e.g. specific year) are set aside and the analysis compared to.

Parameter calibration : SNR

Generalised cross validation (GCV)

According to Craven and Wahba (1978), modifying the error estimate as follows:

$$\hat{\theta}_i^2 = \frac{(d_i - \tilde{d}_i)^2}{(1 - A_{ii})^2}. \tag{33}$$

allows one to keep the data during the analysis and will reduce the computing cost. In this formulation, the denominator penalizes more heavily data points in which the analysis is forced to be close the data and accounts therefore for the self-influence of the data point (which is absent in the case of pure cross-validation).

Parameter calibration : SNR

Computation of A_{ii}

When the matrix $\mathbf{A}$ is not explicitly calculated, A_{ii} can be obtained by performing an analysis with a vector $\boldsymbol{e}_i = (0\,0 \ldots 0\,1\,0 \ldots 0)$ (zero on all data locations, except at point i, where its value is one). This demands an analysis for every data point in which the estimator is constructed.
The associated computational cost can be reduced by replacing A_{ii} by the average value and assuming:

$$A_{ii} \simeq \frac{1}{N} \operatorname{trace} \mathbf{A}. \tag{34}$$

Parameter calibration : SNR

Computation of A_{ii}

To avoid calculating all A_{ii} and summing them up, we can use the following estimate (Girard, 1989):

$$\frac{1}{N}\,\mathrm{trace}\,\mathbf{A} \simeq \frac{\boldsymbol{z}^{\mathrm{T}}\mathbf{A}\boldsymbol{z}}{\boldsymbol{z}^{\mathrm{T}}\boldsymbol{z}} \tag{35}$$

where $\boldsymbol{z}$ is a vector of random variables of zero mean. For robustness, the trace estimate can be repeated several times with different random vectors, averaging of the different estimates. The number of estimates is the parameter provided to the module `GCVFAC` of Diva.

Parameter calibration : SNR

Generalized cross validator

In order to make the error estimator robust, we take the average over all data points and define the *generalized cross validator* as

$$\Theta^2 = \frac{1}{N}\sum_{i=1}^{N}\hat{\theta}_i^2.$$

Assuming temporarily $\epsilon_i^2 = \epsilon^2$, hence having all misfits with the same weight, the generalized cross validator is obtained:

$$\Theta^2 = \frac{\|\boldsymbol{d} - \tilde{\boldsymbol{d}}\|^2}{N\left(1 - \frac{1}{N}\operatorname{trace}\mathbf{A}\right)^2} = \frac{\|(\mathbf{I} - \mathbf{A})\boldsymbol{d}\|^2}{(1/N)\left(\operatorname{trace}\mathbf{I} - \mathbf{A}\right)^2} \tag{36}$$

The GCV consists in minimizing Θ^2 by changing the signal-to-noise ratio λ. Θ^2 is a global estimate of the analysis error variance.

Parameter calibration : SNR

Generalized cross validator

Defining the diagonal matrix $\mathbf{W} = \mathrm{diag}(w_i)$, the generalized cross validator then reads:

$$\Theta^2 = \frac{\left(\boldsymbol{d} - \tilde{\boldsymbol{d}}\right)^{\mathrm{T}} \mathbf{W} \left(\boldsymbol{d} - \tilde{\boldsymbol{d}}\right)}{N \left(1 - \frac{1}{N}\,\mathrm{trace}\,\mathbf{A}\right)^2} \tag{37}$$

where the weights should satisfy

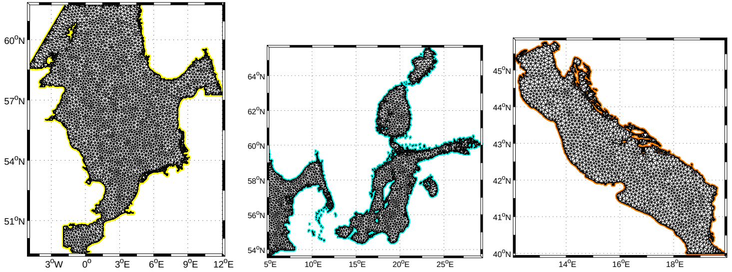
$$\sum_i \frac{1}{w_i} = N. \tag{38}$$

The $\tilde{w}_i$ are the values provided as optional fourth column in `data.dat`.

`divacv` uses cross validation with complete calculation of A_{ii}, whereas `divagcv` uses the approximation $A_{ii} N = \mathrm{trace}\,\mathbf{A}$.

Minimization with a finite-element method

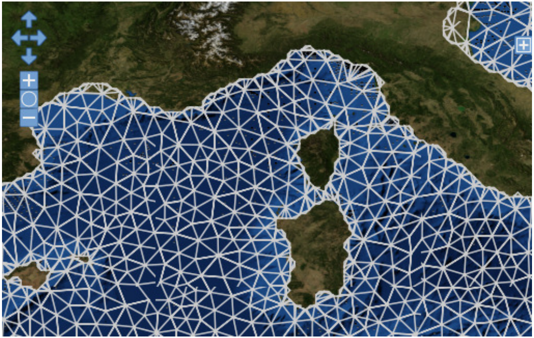
Field regularity → plate bending problem → finite-element solver



Advantages:

- boundaries taken into account
- numerical cost (almost independent on data number)
- no *a posteriori* masking (except if based on error level)

Minimization with a finite-element method



Solution by finite element method. Note decoupling of subbasins. (Each element is in fact composed by three sub-elements)

- The solution method relies on a finite-element resolution, the mesh being automatically generated using one of the tools provided within the software.
- The finite element mesh only covers actual sea so that land regions are excluded from the analysis and provide natural barriers for data-information propagation.

Minimization with a finite-element method

Triangular FE only covers sea: $$J[\varphi] = \sum_{e=1}^{N_e} J_e(\varphi_e) \tag{39}$$

In each element: $$\varphi_e(\mathbf{r_e}) = \mathbf{q_e}^T \mathbf{s}(\mathbf{r_e}) \quad \text{with} \begin{cases} \mathbf{s} & \rightarrow \text{shape functions} \\ \mathbf{q} & \rightarrow \text{connectors} \\ \mathbf{r_e} & \rightarrow \text{position} \end{cases} \tag{40}$$

(40) in (39) + variational principle

$$J_e(\mathbf{q_e}) = \mathbf{q_e}^T \mathbf{K_e} \mathbf{q_e} - 2\mathbf{q_e}^T \mathbf{g_e} + \sum_{i=1}^{N_{d_e}} \mu_i d_i \tag{41}$$

$$\text{where} \begin{cases} \mathbf{K_e} & \rightarrow \text{local stiffness matrix} \\ \mathbf{g} & \rightarrow \text{vector depending on local data} \end{cases}$$

Minimization with a finite-element method

On the whole domain:
$$J(\mathbf{q}) = \mathbf{q}^T\mathbf{K}\mathbf{q} - 2\mathbf{q}^T\mathbf{g} + \sum_{i=1}^{N_d} \mu_i d_i \qquad (39)$$

Minimum:
$$\mathbf{q} = \mathbf{K}^{-1}\mathbf{g} \qquad (40)$$

$$\mathbf{q} = \mathbf{K^{-1}}\,\mathbf{g} \qquad (41)$$

- Stiffness matrix
- Connectors (new unknowns)
- Charge vector

Mapping of data on FEM $\rightarrow$ transfer operator $\mathbf{T_2}$ $\rightarrow$ $\mathbf{g} = \mathbf{T_2}(\mathbf{r})\mathbf{d}$

Solution at any location $\rightarrow$ transfer operator $\mathbf{T_1}$ $\rightarrow$ $\varphi(\mathbf{r}) = \mathbf{T_1}(\mathbf{r})\mathbf{q}$

Results obtained at any location $\rightarrow \varphi = \mathbf{T_1}(\mathbf{r})\mathbf{K}^{-1}\mathbf{T_2}(\mathbf{r})\mathbf{d}$

DIVA as OI

DIVA is identical to the well known Optimal Interpolation

- if so-called reproducing kernel of the norm = covariance function of OI,
- if the noise is random, spatially uncorrelated and the signal/noise ratio parameter is identical with OI.

In this case, the OI solution = DIVA solution.

- Advantages of DIVA: regularization, fast finite-element solution, boundary effects taken into account.
- Difficulties: generalizations to 3D and multivariate versions are "hybrid" (some expensive O.I. components).

Major direct advantage of DIVA: matrix to invert is related to the finite-element mesh, NOT the number of data. Useful for large data sets (Rixen *et al.*, 2000). Equivalence allows to calculate error fields with DIVA even if formulation does not rely on error minimisation.

Comparison

Method	$\min(\epsilon^2)$	3D	Multivar	Ops/image	$\epsilon(\boldsymbol{r})$	a priori	C.V.	anisotropy
Cressman		$\star$	$\star$	$N_d N_a$		$w(r/L)$	(L)	$(\star)$
O.I.	$\star$	$\star$	$\star$	$N_d^3 + N_d N_a$	$\star$	$c(r/L)$	$L, \sigma^2/\mu^2$	$(\star)$
DIVA	$\star$	$(\star)$	$(\star)$	$N_e^{5/2}$	$\star$	$K(r/L)$	$L, \sigma^2/\mu^2$	$\star$
DINEOF	$(\star)$	$\star$	$\star$	$N_a^{5/4}$	$(\star)$	stat.	N	$\star$

N_d : number of data points

N_a : number of grid points for analysis

N_e : number of finite elements

N : number of EOFs

L : correlation length

σ^2/ϵ^2 : signal to noise ratio

$\star$: available feature

$(\star)$: available with some adaptations

Diva Cocktail Recipe

Ingredients:

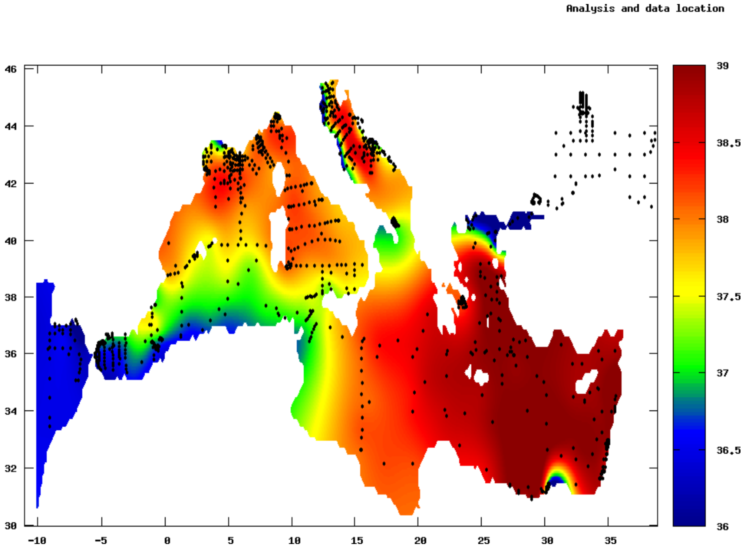
- 1 1/2 oz vodka
- 1/2 oz passion-fruit juice
- 1/2 oz lime juice
- 1 tbsp cherry juice
- fill with soda

Diva Cocktail Recipe

Ingredients:

- Smoothness
- Observation constraint
- Behaviour constraint

DIVA illustration



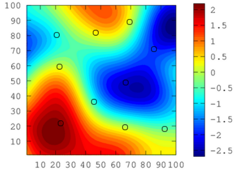
Want to use Diva?

Playing. . .

DIVA demo help!

Reference field

Field: Simple random field



Location of observations

Your name: charles
Correlation length-scale: 17

Choose the location of your observation (maximum 10)

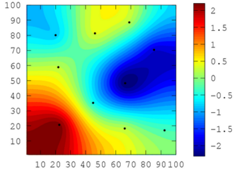
#	x	y	
1	22.492	20.667	[x]
2	46.201	81.273	[x]
3	21.884	58.848	[x]
4	66.261	48.242	[x]
5	65.957	18.242	[x]
6	85.410	70.364	[x]
7	44.985	35.212	[x]
8	20.061	80.061	[x]
9	68.997	88.545	[x]
10	92.401	17.030	[x]

Make analysis | Remove all | RMS = **0.4882289268**

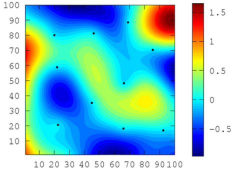
Top 5

Name	RMS
Chunksaah	0.165223169
bbl	0.1818438045
NicoG	0.1841920279
Charef	0.2056911308
fd	0.2111580456

Reconstruction



Difference

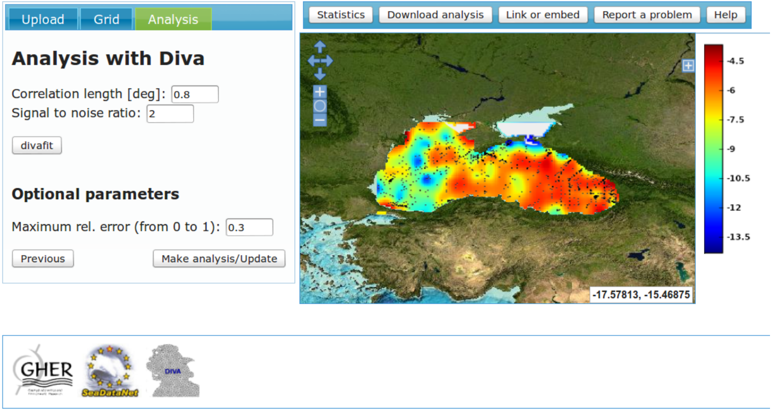


Contact: Alexander Barth, a.barth at ulg.ac.be, 2012.

http://data-assimilation.net/Tools/divand_demo/html/

Want to use Diva?

With your own data. . .



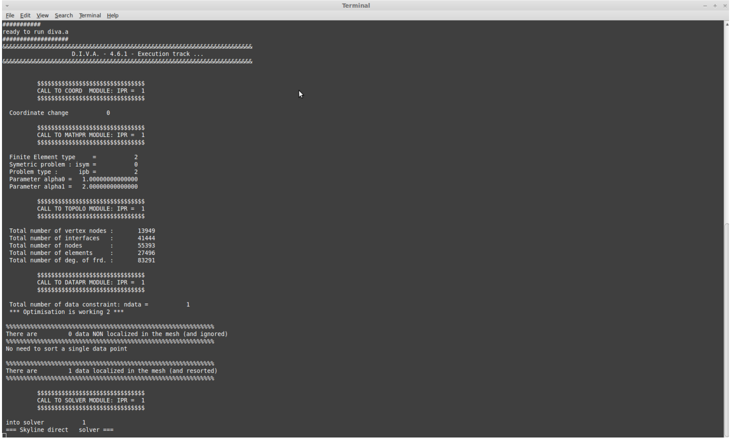
http://gher-diva.phys.ulg.ac.be/web-vis/diva.html or ODV or matlab wrapper

Want to use Diva?

For serious work:

2D version (for production), open source, GPL

nD version (for research), open source, GPL



http://modb.oce.ulg.ac.be/mediawiki/index.php/DIVA

Running Diva in 2D: input files

1. `data.dat`: contains the observations — x|y|value

```
36.5500 45.163 17.7138
33.7500 44.167 18.135
32.7500 44.167 18.51
36.2500 43.833 18.5892
33.2500 45.083 18.2326
32.7833 43.917 18.477
32.7500 43.500 18.59
37.2433 44.833 18.1555
36.5000 44.000 18.19
35.8333 43.750 18.62
34.2500 43.832 18.29
35.6500 44.000 18.75
38.0000 44.000 18.155
37.8200 44.368 17.1916
39.0000 42.500 18.23
33.1333 44.433 18.001
33.0500 44.433 18.09
33.2500 44.167 18.231
32.5333 44.833 18.014
38.0167 44.447 18.0568
```

Running Diva in 2D: input files

1. `data.dat`: contains the observations — x|y|value
2. `coast.cont`: delimits land and sea — (coastline or isobaths)

```
          7
        552
  27.4375000       40.3499985
  27.4500008       40.3375015
  27.4666672       40.3375015
  27.4833336       40.3375015
  27.5000000       40.3375015
  27.5166664       40.3375015
  27.5333328       40.3375015
  27.5499992       40.3375015
  27.5666676       40.3375015
  27.5791664       40.3499985
  27.5833340       40.3541679
  27.6000004       40.3541679
  27.6124992       40.3666649
  27.6166668       40.3708344
  27.6291676       40.3833351
  27.6291676       40.4000015
  27.6291676       40.4166679
  27.6291676       40.4333344
  27.6166668       40.4458351
  27.6124992       40.4500008
```

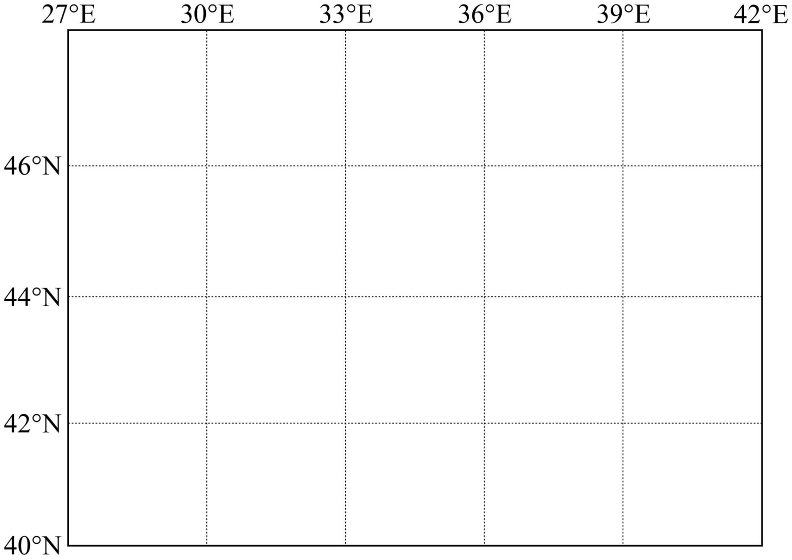
Running Diva in 2D: input files

1. `data.dat`: contains the observations — x|y|value
2. `coast.cont`: delimits land and sea — (coastline or isobaths)
3. `param.par`: analysis parameters — L, λ, resolution, …

```
# Lc: correlation length (in units coherent with your data)#
1.5
# icoordchange (=0 if no change of coordinates is to be performed; =1 if positions are in degree
2
# ispec: output files required#
0
# ireg: mode selected for background field: 0=null guess; 1=mean of data; 2=regression plan if a
2
# xori: x-coordinate of the first grid point of the output#
27
# xori: y-coordinate of the first grid point of the output#
40
# dx: step of output grid#
0.1
# dy: step of output grid#
0.1
# nx: number of grid points in the x-direction#
151
# ny: number of grid points in the y-direction#
76
# valex: exclusion value#
-99
# snr: signal to noise ratio of the whole dataset#
0.5
# varbak: variance of the background field. If zero, no error fields are produced. If one, relat
1.0
```

Workflow in 2D

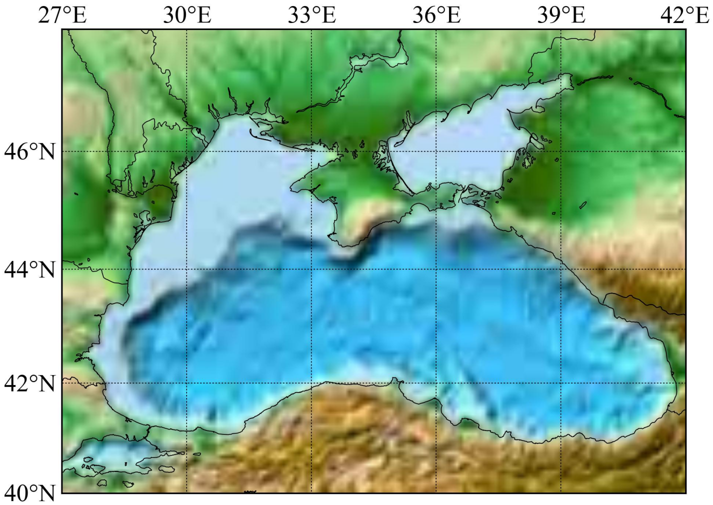
Select region of study



Workflow in 2D

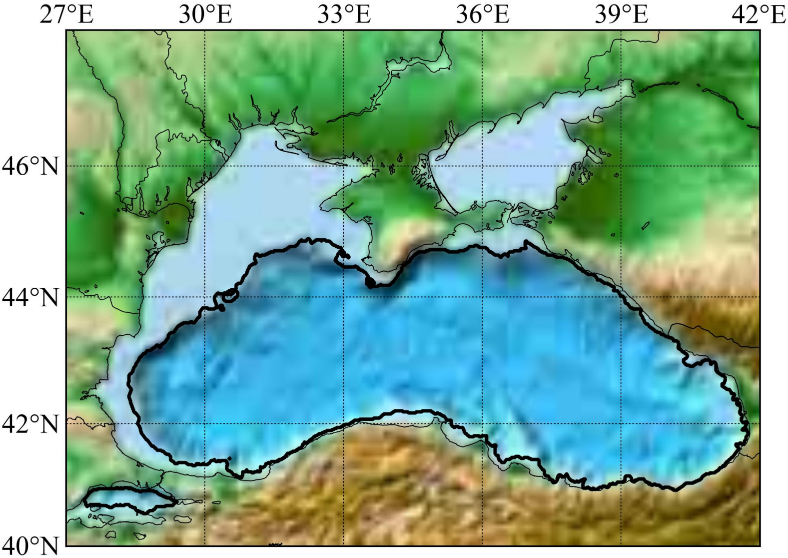
Extract topography, for example via

http://gher-diva.phys.ulg.ac.be/web-vis/diva.html



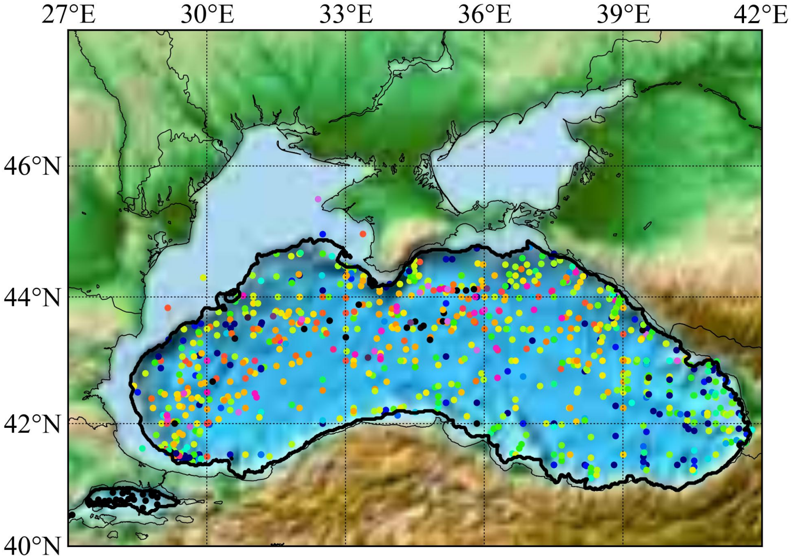
Workflow in 2D

Generate contour



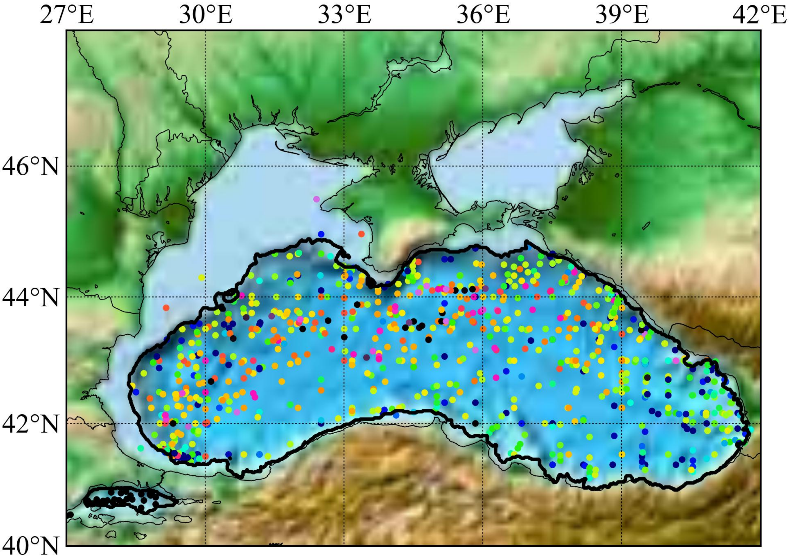
Workflow in 2D

Extract data



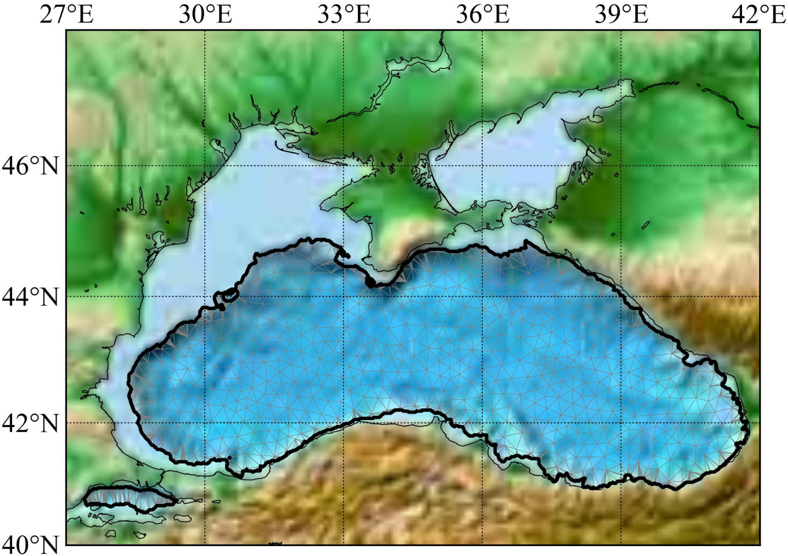
Workflow in 2D

Evaluate analysis parameters



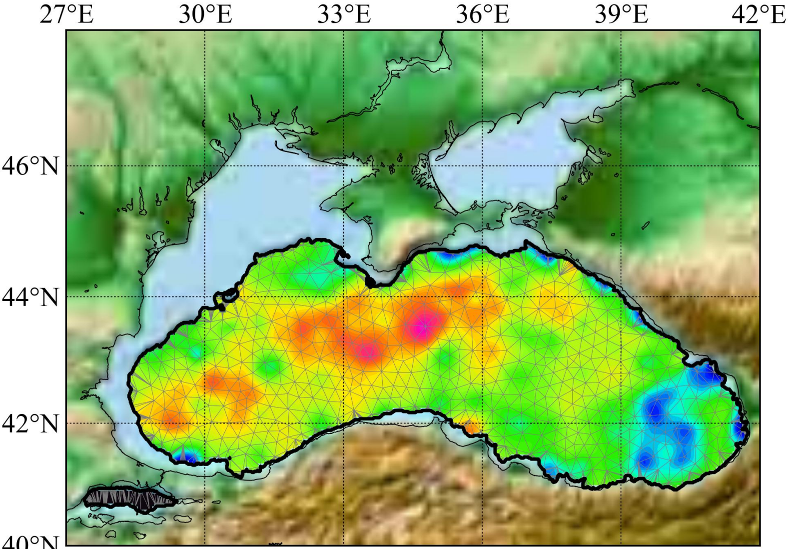
Workflow in 2D

Create finite-elementmesh



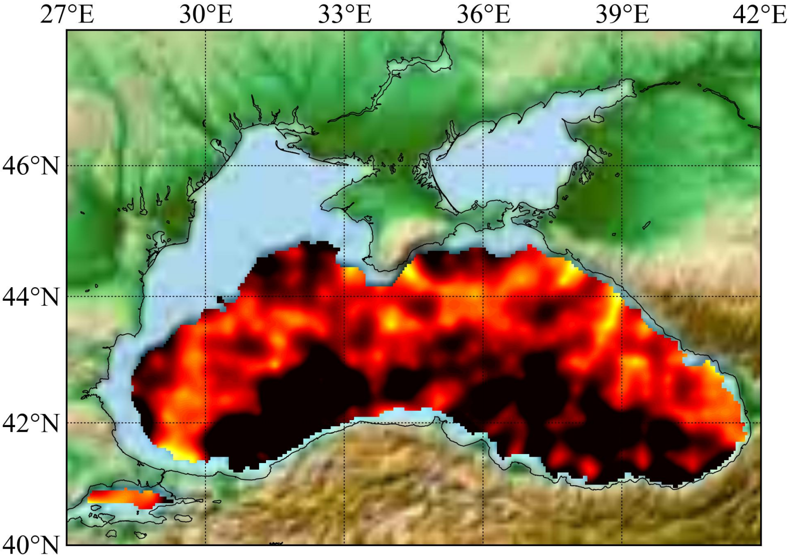
Workflow in 2D

Generate analysis



Workflow in 2D

Generate error field



When to use 2D version

- occasional use
- 2D fields like benthic properties
- for implementation of special features by your own (eg multiplicative bias correction, special background field creation based on habitats
- ...

otherwise: use 3D or 4D version directly

Next...

Diva in 4 dimensions