Sample covariance filtering

Benjamin Ménétrier

Last update: November 23, 2017

Documentation for the code "hdiag_nicas", distributed under the CeCILL-C license.

Contents

1 Framework

A localization matrix $\mathbf{L} \in \mathbb{R}^{n \times n}$ is applied to a multivariate state vector $\mathbf{x} \in \mathbb{R}^n$, built as the concatenation of p sub-vectors:

$$\mathbf{x} = \left(\begin{array}{c} \mathbf{x}_1 \\ \hline \vdots \\ \hline \mathbf{x}_p \end{array} \right) \tag{1}$$

where $\mathbf{x}_i$ is the sub-vector for variable i, of size n_i. Obviously, $n = \sum_{i=1}^{p} n_i$.

The localization matrix is split into blocks:

$$\mathbf{L} = \begin{pmatrix} \mathbf{L}_{11} & \cdots & \mathbf{L}_{1p} \\ \vdots & \ddots & \vdots \\ \mathbf{L}_{p1} & \cdots & \mathbf{L}_{pp} \end{pmatrix}$$

The problem is to design a positive semi-definite localization matrix:

- $\mathbf{L}_{ii}$ should be positive semi-definite,
- the constraint on $\mathbf{L}_{ij}$ for $i \neq j$ is less obious...

The solution is to build a square-root $\mathbf{U} \in \mathbb{R}^{n \times m}$ of the localization matrix $\mathbf{L}$, such that $\mathbf{L} = \mathbf{U}\mathbf{U}^{\mathrm{T}}$. The control vector $\mathbf{v}$ such that $\mathbf{x} = \mathbf{U}\mathbf{v}$ is defined in the control space of size m.

2 Univariate specific blocks

The simplest method is to design $\mathbf{U}$ as:

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{U}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{U}_p \end{pmatrix} \tag{2}$$

where $\mathbf{U}_i \in \mathbb{R}^{n_i \times m_i}$, m_i being the size of the control sub-vector related to variable i with $m = \sum_{i=1}^{p} m_i$. The resulting localization matrix is:

$$\mathbf{L} = \mathbf{U}\mathbf{U}^{\mathrm{T}} = \begin{pmatrix} \mathbf{U}_1\mathbf{U}_1^{\mathrm{T}} & 0 & \cdots & 0 \\ 0 & \mathbf{U}_2\mathbf{U}_2^{\mathrm{T}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{U}_p\mathbf{U}_p^{\mathrm{T}} \end{pmatrix} \tag{3}$$

In this case, auto-localization blocks are well defined:

$$\mathbf{L}_{ii} = \mathbf{U}_i \mathbf{U}_i^{\mathrm{T}} \tag{4}$$

but cross-localization blocks $\mathbf{L}_{ij}$ for $i \neq j$ are zero.

Advantages:

- Well-adapted for large differences between variables length-scales.
- Well-adapted for small cross-correlation (sampling noise is canceled).

Drawbacks:

- Ill-adapted for significant cross-correlations (signal is lost).

Cost: p applications of $\mathbf{U}_1$ to $\mathbf{U}_p$.

3 Multivariate specific blocks

3.1 State grid interpolation

A second method is to perform interpolations to and from a common state grid of size $\overline{n}$. $\mathbf{S}^{\mathrm{s}}_i \in \mathbb{R}^{n_i \times \overline{n}}$ is defined as the interpolation from the common state grid to the state grid of variable i. We gather these matrices into a global state interpolation matrix $\mathbf{S}^{\mathrm{s}}$:

$$\mathbf{S}^{\mathrm{s}} = \begin{pmatrix} \mathbf{S}^{\mathrm{s}}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{S}^{\mathrm{s}}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{S}^{\mathrm{s}}_p \end{pmatrix} \tag{5}$$

A specific block $\overline{\mathbf{U}}_i \in \mathbb{R}^{\overline{n} \times m}$ is applied to each variable i:

$$\mathbf{U} = \mathbf{S}^{\mathrm{s}} \begin{pmatrix} \overline{\mathbf{U}}_1 \\ \vdots \\ \overline{\mathbf{U}}_p \end{pmatrix} \tag{6}$$

The resulting localization matrix is:

$$\mathbf{L} = \mathbf{U}\mathbf{U}^{\mathrm{T}} = \mathbf{S}^{\mathrm{s}} \begin{pmatrix} \overline{\mathbf{U}}_1 \overline{\mathbf{U}}_1^{\mathrm{T}} & \cdots & \overline{\mathbf{U}}_1 \overline{\mathbf{U}}_p^{\mathrm{T}} \\ \vdots & \ddots & \vdots \\ \overline{\mathbf{U}}_p \overline{\mathbf{U}}_1^{\mathrm{T}} & \cdots & \overline{\mathbf{U}}_p \overline{\mathbf{U}}_p^{\mathrm{T}} \end{pmatrix} \mathbf{S}^{\mathrm{sT}} \tag{7}$$

In this case, each auto- or cross-localization block is given by:

$$\mathbf{L}_{ij} = \mathbf{S}^{\mathrm{s}}_i \overline{\mathbf{U}}_i \overline{\mathbf{U}}_j^{\mathrm{T}} \mathbf{S}^{\mathrm{sT}}_j \tag{8}$$

3.2 Control grid interpolation

An alternative to this method is to perform interpolations to and from a common control grid of size $\overline{m}$. $\mathbf{S}^{\mathrm{c}}_{i} \in \mathbb{R}^{m_i \times \overline{m}}$ is defined as the interpolation from the common vector grid to the control grid of variable i. We can write a global interpolation matrix $\mathbf{S}^{\mathrm{c}}$ as:

$$\mathbf{S}^{\mathrm{c}} = \begin{pmatrix} \mathbf{S}^{\mathrm{c}}_{1} & 0 & \cdots & 0 \\ 0 & \mathbf{S}^{\mathrm{c}}_{2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{S}^{\mathrm{c}}_{p} \end{pmatrix} \tag{9}$$

A specific block $\mathbf{U}_{i} \in \mathbb{R}^{n_i \times m_i}$ is applied to each variable:

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_{1} \\ \vdots \\ \mathbf{U}_{p} \end{pmatrix} \mathbf{S}^{\mathrm{c}} \tag{10}$$

The resulting localization matrix is $\mathbf{L} = \mathbf{U}\mathbf{U}^{\mathrm{T}}$ with each auto- or cross-localization block given by:

$$\mathbf{L}_{ij} = \mathbf{U}_{i}\mathbf{S}^{\mathrm{c}}_{i}\mathbf{S}^{\mathrm{cT}}_{j}\mathbf{U}^{\mathrm{T}}_{j} \tag{11}$$

3.3 Interpolation error

We assume that every interpolation $\mathbf{S}^{\mathrm{s}}$ or $\mathbf{S}^{\mathrm{c}}$ can be expanded as $\mathbf{S} = \mathbf{T} + \mathbf{E}$ where $\mathbf{T}$ is the "true" interpolation, i.e. the best we can get with a linear operator, and $\mathbf{E} = \mathbf{S} - \mathbf{T}$ the resulting error. Thus:

- For the state grid interpolation with specific blocks:

$$\begin{aligned} \mathbf{L}_{ij} &= \mathbf{S}^{\mathrm{s}}_{i}\overline{\mathbf{U}}_{i}\overline{\mathbf{U}}^{\mathrm{T}}_{j}\mathbf{S}^{\mathrm{sT}}_{j} \\ &= \mathbf{T}^{\mathrm{s}}_{i}\overline{\mathbf{U}}_{i}\overline{\mathbf{U}}^{\mathrm{T}}_{j}\mathbf{T}^{\mathrm{sT}}_{j} + \mathbf{E}^{\mathrm{s}}_{i}\overline{\mathbf{U}}_{i}\overline{\mathbf{U}}^{\mathrm{T}}_{j}\mathbf{T}^{\mathrm{sT}}_{j} + \mathbf{T}^{\mathrm{s}}_{i}\overline{\mathbf{U}}_{i}\overline{\mathbf{U}}^{\mathrm{T}}_{j}\mathbf{E}^{\mathrm{sT}}_{j} + \mathbf{E}^{\mathrm{s}}_{i}\overline{\mathbf{U}}_{i}\overline{\mathbf{U}}^{\mathrm{T}}_{j}\mathbf{E}^{\mathrm{sT}}_{j} \end{aligned} \tag{12}$$

  and similarly for the state grid interpolation with a common block.

- For the control grid interpolation:

$$\begin{aligned} \mathbf{L}_{ij} &= \mathbf{U}_{i}\mathbf{S}^{\mathrm{c}}_{i}\mathbf{S}^{\mathrm{cT}}_{j}\mathbf{U}^{\mathrm{T}}_{j} \\ &= \mathbf{U}_{i}\mathbf{T}^{\mathrm{c}}_{i}\mathbf{T}^{\mathrm{cT}}_{j}\mathbf{U}^{\mathrm{T}}_{j} + \mathbf{U}_{i}\mathbf{E}^{\mathrm{c}}_{i}\mathbf{T}^{\mathrm{cT}}_{j}\mathbf{U}^{\mathrm{T}}_{j} + \mathbf{U}_{i}\mathbf{T}^{\mathrm{c}}_{i}\mathbf{E}^{\mathrm{cT}}_{j}\mathbf{U}^{\mathrm{T}}_{j} + \mathbf{U}_{i}\mathbf{E}^{\mathrm{c}}_{i}\mathbf{E}^{\mathrm{cT}}_{j}\mathbf{U}^{\mathrm{T}}_{j} \end{aligned} \tag{13}$$

Assuming that interpolation errors are random with a short correlation length-scale, the control grid interpolation looks like a better option since a smoothing is always applied as the last operator. However, even with the state grid interpolation:

- either the interpolation error is smoothed in subsequent operators,
- or the interpolation error is itself applied on smoothed fields at high-resolution, leading to small interpolation errors.

As a conclusion, we don't think that interpolation error is a major issue that should favor one approach or the other.

3.4 Localization amplitude

We denote $\bar{\mathbf{u}}$ the k^{th} column of $\overline{\mathbf{U}}_i$ and $\bar{\mathbf{u}}'$ the k^{th} column of $\overline{\mathbf{U}}_j$. The k^{th} diagonal coefficient of $\overline{\mathbf{U}}_i\overline{\mathbf{U}}_j^{\mathrm{T}}$ is given by $\langle \bar{\mathbf{u}}, \bar{\mathbf{u}}' \rangle$, where $\langle \cdot, \cdot \rangle$ denotes the canonical inner product. The Cauchy-Schwartz inequality imposes that:

$$|\langle \bar{\mathbf{u}}, \bar{\mathbf{u}}' \rangle| \leq \sqrt{\langle \bar{\mathbf{u}}, \bar{\mathbf{u}} \rangle \langle \bar{\mathbf{u}}', \mathbf{u}' \rangle} \tag{14}$$

Thus, the cross-localization amplitude between variables i and j is necessarily smaller than the geometric mean of the auto-localization amplitudes for variables i and j. Besides, the more the vectors $\bar{\mathbf{u}}$ and $\bar{\mathbf{u}}'$ have different shapes, the smaller their inner product and the smaller the cross-localization amplitude.

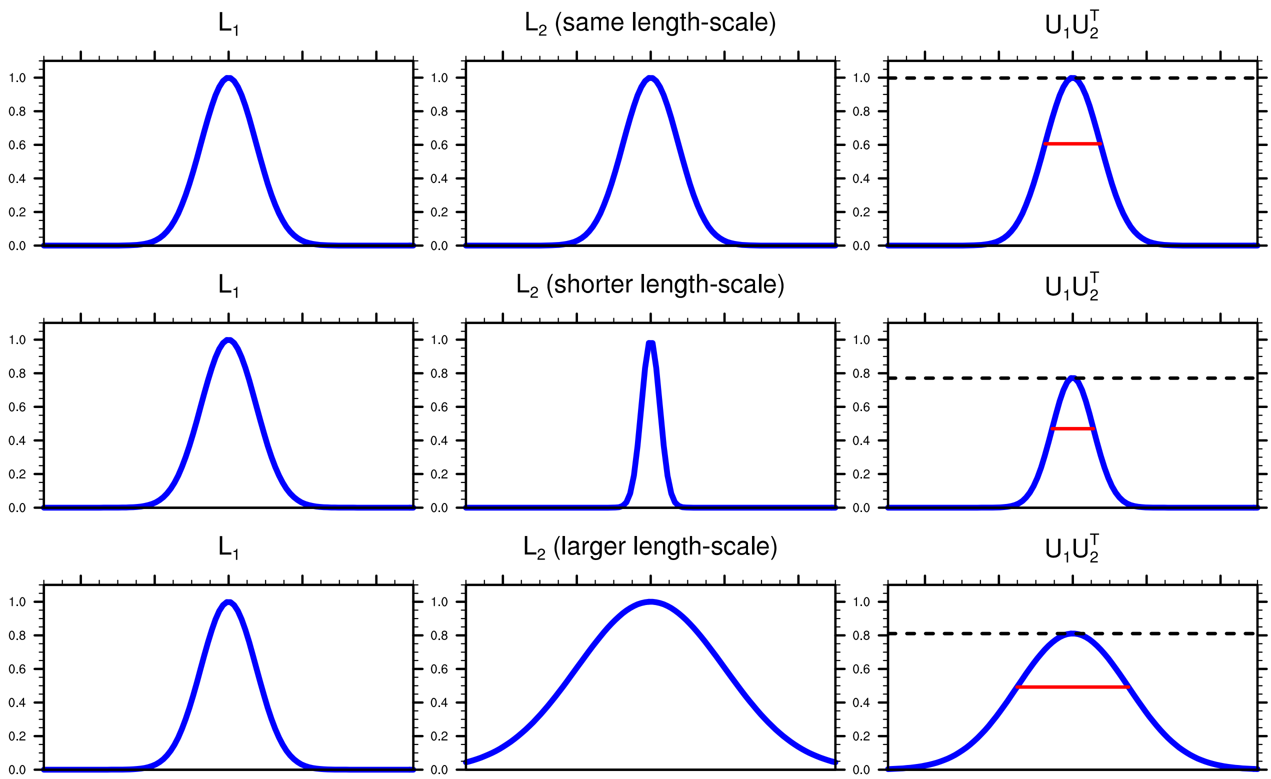

Figure 1: 1D illustration of the product of two localization square-root, with Gaussian localization functions.

Advantages:

- Well-adapted for large differences between variables length-scales.

Drawbacks:

- No control on the cross-localization amplitude.

Cost: p application of $\overline{\mathbf{U}}_i$.

3.5 Common block

A third method is to apply a common block $\overline{\mathbf{U}} \in \mathbb{R}^{\bar{n} \times m}$ to all variables, with a state grid interpolation:

$$\mathbf{U} = \mathbf{S}^{\mathrm{s}} \begin{pmatrix} \overline{\mathbf{U}} \\ \vdots \\ \overline{\mathbf{U}} \end{pmatrix} = \mathbf{S}^{\mathrm{s}} \begin{pmatrix} \mathbf{I}_{\bar{n}} \\ \vdots \\ \mathbf{I}_{\bar{n}} \end{pmatrix} \overline{\mathbf{U}} \tag{15}$$

where $\mathbf{I}_{\bar{n}} \in \mathbb{R}^{\bar{n} \times \bar{n}}$ is the identity matrix. It should be noted that to apply $\mathbf{U}$, the matrix $\overline{\mathbf{U}}$ needs to be applied only once. The resulting localization matrix is:

$$\mathbf{L} = \mathbf{U}\mathbf{U}^{\mathrm{T}} = \mathbf{S}^{\mathrm{s}} \begin{pmatrix} \overline{\mathbf{U}}\overline{\mathbf{U}}^{\mathrm{T}} & \cdots & \overline{\mathbf{U}}\overline{\mathbf{U}}^{\mathrm{T}} \\ \vdots & \ddots & \vdots \\ \overline{\mathbf{U}}\overline{\mathbf{U}}^{\mathrm{T}} & \cdots & \overline{\mathbf{U}}\overline{\mathbf{U}}^{\mathrm{T}} \end{pmatrix} \mathbf{S}^{\mathrm{sT}} \tag{16}$$

In this case, each auto- or cross-localization block is given by:

$$\mathbf{L}_{ij} = \mathbf{S}^{\mathrm{s}}_{i} \overline{\mathbf{U}}\overline{\mathbf{U}}^{\mathrm{T}} \mathbf{S}^{\mathrm{sT}}_{j} \tag{17}$$

It should be noted that this approach is inconsistent with the optimal linear filtering theory. Indeed, the cross-correlation between different variables, even at zero separation, is lower than 1 (i.e. the value for auto-correlations). Thus, the cross-localization should be lower than the auto-localization at zero separation, whereas $\left[\overline{\mathbf{U}}\overline{\mathbf{U}}^{\mathrm{T}}\right]_{ii}$ is applied in both case.

Advantages:

- Lower cost.

Drawbacks:

- Ill-adapted for large differences between variables length-scales.
- Ill-adapted for small cross-correlations (sampling noise is kept).

Cost: one application of $\overline{\mathbf{U}}$.

3.6 Common block, weighted

4 Applications

4.1 Application 1: 3D variables located on different points

The 3D variables can be located on different points of the grid (e.g. Arakawa C grid). In this case, the three approaches above can be used:

1. state grid interpolation, common block,
2. state grid interpolation, specific block,

3. control grid interpolation

The cost of each method is studied for three kinds of smoothers used to apply the square-root $\mathbf{U}$, depending on their input and output size:

- The first class of smoothers has the same grid on input and output (e.g. recursive filters and explicit/implicit diffusion methods). For this class, it is reasonable to assume that the cost of an interpolation is lower than the cost of the smoother itself. Thus, the first approach is the cheapest, and approaches 2 and 3 have the same cost
- The second class of smoothers includes the spectral methods. In this case, the control vector is in spectral space, whose truncation can be lowered to save memory requirements. Approaches 1 is then the cheapest, the comes approach 2. Approach 3 is not available since the control vector is not in grid-point space.
- The third class of smoothers is the one of NICAS, where the control vector size is given by the subgrid size, and can be significantly lower than the state size. With NICAS, the third approach could be simplified since the same subgrid (control space) could be used even if the full grids (state space) are different depending on the variable. The cost of the NICAS method being dominated by the interpolation from the subgrid to the full grid and its adjoint, approach 2 is necessarily more expensive than approach 3. The relative cost of approach on depends on how many variables must be interpolated on a common grid before applying a common block.

4.2 Application 2: 3D and 2D variables

If 3D and 2D variables are correlated, the cross-localization between them should be non-zero. The usual practice is to build an extended 3D grid with one more level at the bottom, and to copy the 2D variable on this extra level. The formalism described above can be used to understand this approach: the state grid interpolation is a simple copy of the 2D variable on the extra level, and a copy of the 3D variable on the remaining levels above.

Using a common block $\overline{\mathbf{U}}$ can lead to some issues when the 3D and 2D variables have very different length-scales (e.g. temperature and surface pressure). However, we can also use the state grid interpolation with two specific blocks $\overline{\mathbf{U}}_{3D}$ and $\overline{\mathbf{U}}_{2D}$, which are both defined on the extended grid.

The third approach with a control grid interpolation is not suitable to cross-localize 3D and 2D variables, since it is not possible to define cross-localizations on their specific grids.