Patched Gaussian Processes

Tom Perrin, Leo Roux & Thibault Lambert

Option Géostatistique, Mines Paris - PSL

tom.perrin@etu.minesparis.psl.eu
leo.roux@etu.minesparis.psl.eu
thibault.lambert@etu.minesparis.psl.eu

MINES PARIS | PSL

TABLE OF CONTENTS

GAUSSIAN PROCESS

Gaussian Process : Random surface $W(s)$ over a domain $\mathcal{D}$ such that :

$$\forall \{s_1, \dots, s_n\} \in \mathcal{D}^n, W := (W(s_1), \dots, W(s_n)) \sim \mathcal{N}(\mu, C).$$

- **Mean function :** $\mu(s) := \mathbb{E}[W(s)] \longrightarrow \mu := [\mu(s_1), \dots, \mu(s_N)]^T$
- **Covariance function :** $C(s, s') := \mathbb{C}\mathrm{ov}[W(s), W(s')] \longrightarrow C := [C(s_i, s_j)]_{1 \leqslant i,j \leqslant N}.$

Law of $\boldsymbol{W}$:

$$p(W) = \frac{1}{\sqrt{(2\pi)^d \mathsf{det}(C)}} \exp\left(-\frac{1}{2}(W - \mu)^T C^{-1} (W - \mu)\right).$$

KRIGING

Observations : $\mathcal{D} = \{(x_i, y_i) \mid i \in \{1, \dots, N\}\}$.

$$\textbf{Locations } \boldsymbol{x} := [x_1, \dots, x_N]^T \longrightarrow \textbf{Responses } \boldsymbol{y} := [y_1, \dots, y_N]^T.$$

Kriging : Stochastic predictions based on these observations.

- **Reponses $\boldsymbol{y}$** seen as realizations of a **gaussian process $\boldsymbol{f}$** with noise :

$$y_i = f(x_i) + \varepsilon_i, \text{ where } \varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2).$$

- **Prediction $\boldsymbol{f_* := f(x^*)}$** at new location x^* $\longrightarrow$ **Joint law of $\boldsymbol{(f_*, y)}$**:

$$p(f_*, y) \sim \mathcal{N}\left(0, \begin{bmatrix} c_{**} & c_{*x} \\ c_{x*} & \sigma^2 I + C_{xx} \end{bmatrix}\right).$$

where $c_{**} = C(x^*, x^*)$, $c_{x*} = [C(x_1, x^*), \dots, C(x_N, x^*)]^T$, $C_{xx} = [C(x_i, x_j)]_{1 \leqslant i,j \leqslant N}$.

KRIGING

Predictive distribution :

$$p(f_* \mid y) \sim \mathcal{N}(\underbrace{c_{x*}^T(\sigma^2 I + C_{xx})^{-1} y}_{\text{predictive mean}}, \underbrace{c_{**} - c_{x*}^T(\sigma^2 I + C_{xx})^{-1} c_{x*}}_{\text{predictive variance}})$$

- **Predicted value :** $f(x^*) \approx c_{x*}^T(\sigma^2 I + C_{xx})^{-1} y.$
- **Uncertainty :** $c_{**} - c_{x*}^T(\sigma^2 I + C_{xx})^{-1} c_{x*}.$

BAYESIAN APPROACH

Hierarchical model :

$$y_i = X(x_i) \cdot \beta + f_\theta(x_i) + \varepsilon_i.$$

- **Regression term $\boldsymbol{X(x_i) \cdot \beta}$:** Captures effects of known explanatory variables X.
- **Spatial effect $\boldsymbol{f_\theta(x_i)}$:** Realization of a gaussian process $f_\theta \sim \mathcal{N}(0, C_\theta)$.
- **Noise $\boldsymbol{\varepsilon_i}$:** $\varepsilon_i \overset{iid}{\sim} \mathcal{N}(0, \tau^2)$.

BAYESIAN APPROACH

Bayesian Inference : Estimation of the **hidden state** $\boldsymbol{f_\theta}$ and **parameters** $\phi := \{\beta, \theta, \tau^2\}$.

- **Prior distributions :** We assume prior knowledge $p(\beta)$, $p(\theta)$ and $p(\tau^2)$.
- **Likelihood :** $p(y \mid f_\theta, \beta, \tau^2) = \mathcal{N}(X\beta + f_\theta, \tau^2 I)$.
- **Latent GP prior :** $p(f_\theta \mid \theta) = \mathcal{N}(0, C_\theta)$.

Joint posterior distribution :

$$p(f_\theta, \beta, \theta, \tau^2 \mid y) \propto \underbrace{p(y \mid f_\theta, \beta, \tau^2)}_{\text{data likelihood}} \times \underbrace{p(f_\theta \mid \theta)}_{\text{spatial link}} \times \underbrace{p(\beta)p(\theta)p(\tau^2)}_{\text{parameter priors}}.$$

TABLE OF CONTENTS

COMPUTATIONAL COSTS

Both the predictive mean and variance require **solving linear systems** involving $\Sigma = \sigma^2 I + C_{xx}$.

- **Time complexity :** Matrix inversion $\longrightarrow \mathcal{O}(\boldsymbol{N^3})$.
- **Space complexity :** Storing covariance matrix $\longrightarrow \mathcal{O}(\boldsymbol{N^2})$.

Solutions :

- Acting on the **covariance matrix** $\boldsymbol{\Sigma}$: compact support, covariance tapering, markovian models $\longrightarrow$ **Sparse matrix**.
- Acting on the **amount of observations** $\boldsymbol{N}$...

INDEPENDANT LOCAL KRIGING

Idea : Split the observations $\mathcal{D}$ into K subsets $\mathcal{D}_1, \dots, \mathcal{D}_K$, with $\mathcal{D}_k := \{(x_i, y_i) \mid x_i \in \Omega_k\}$.

Region $\Omega_k \longrightarrow$ Local gaussian process $f_k \longrightarrow$ Associated covariance function $C_k(\cdot,\cdot)$.

- **Stationary process :** $C_k(\cdot,\cdot) = C(\cdot,\cdot)$.
- Otherwise : Different covariance functions C_k.

$$y_{k,i} = f_k(x_i) + \varepsilon_{k,i}.$$

Two questions :

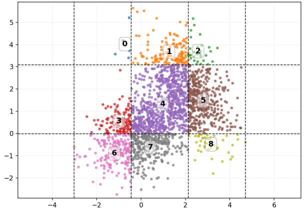
- How to split the observations efficiently ?
- How to deal with shared boundaries $\Gamma_{k,\ell} := \overline{\Omega}_k \cap \overline{\Omega}_\ell$?

■ SPLITTING THE OBSERVATIONS

Grid partitioning : Splitting the data using an uniform grid to cover the space.

⊕ Very easy implementation.

⊖ High density variance between regions.

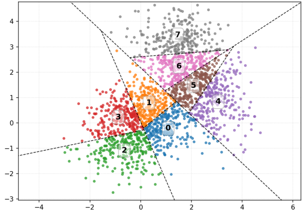


Grid partitioning

PCA partitioning : Splitting the data based on principal component projections values.

⊕ Balanced number of points ($N_k \approx$ cst).

⊖ Complex boundary definition.



PCA partitioning

BORDER DISCONTINUITY

Issue :

$$f_k(x) \neq f_\ell(x) \text{ at frontier } \Gamma_{k,\ell}.$$

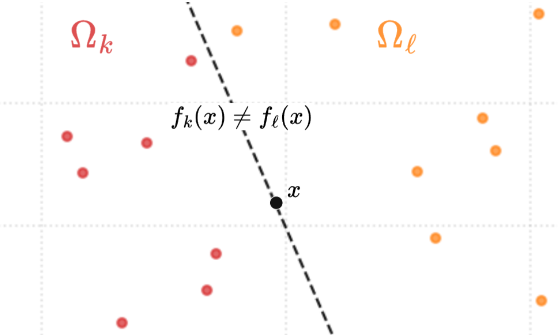
Solution :

$$\text{Set } \delta_{k,\ell} := f_k - f_\ell = 0 \text{ at } \Gamma_{k,\ell}.$$

In practice : pseudo-observations $\delta_{k,\ell}(x) = 0$ at the frontiers.

$$\mathbb{E}\left[f_*^{(k)} \mid y\right] \longrightarrow \mathbb{E}\left[f_*^{(k)} \mid y, \delta = 0\right].$$

$$\mathbb{V}\left[f_*^{(k)} \mid y\right] \longrightarrow \mathbb{V}\left[f_*^{(k)} \mid y, \delta = 0\right].$$



Potential discontinuity at frontier

TABLE OF CONTENTS

TABLE OF CONTENTS

PSEUDO-OBSERVATIONS

For two neighbor regions Ω_k, Ω_ℓ, place B observations on the border $\Gamma_{k,\ell}$:

- **Pseudo-locations :** $x^{(k,\ell)} := \left(x_1^{(k,\ell)}, \dots, x_B^{(k,\ell)}\right)^T$.
- **Pseudo-values :** $\delta_{k,\ell} := \left(\delta_{k,\ell}(x_1^{(k,\ell)}), \dots, \delta_{k,\ell}(x_B^{(k,\ell)})\right)^T$.

➤ **Observations :** $y := \left(y_1^T, \dots, y_K^T\right)^T$.

➤ **Pseudo-observations :** $\delta := \left(\delta_{1,1}^T, \dots, \delta_{1,K}^T, \dots, \delta_{K,K}^T\right)^T$.

PATCHWORK KRIGING

Predict $f_*^{(k)} := f_k(x^*)$ at $x^* \in \Omega_k \longrightarrow$ **Joint law of $(\boldsymbol{f_*^{(k)}}, \boldsymbol{y}, \boldsymbol{\delta})$** :

$$\begin{bmatrix} f_*^{(k)} \\ y \\ \delta \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} c_{**} & c_{*\mathcal{D}}^{(k)} & c_{*\delta}^{(k)} \\ c_{\mathcal{D}*}^{(k)} & C_{\mathcal{D}\mathcal{D}} & C_{\mathcal{D}\delta} \\ c_{\delta*}^{(k)} & C_{\delta\mathcal{D}} & C_{\delta\delta} \end{bmatrix} \right).$$

- On the border between regions k and l $\delta_{k,l} = f_k - f_l$
- Under the asumption f_k and f_l are independent gaussian processes, $\delta_{k,l}$ is one too
- Specifically, we can easily compute its covariance with $f_j(x)$, for a certain $x \in \mathcal{D}_j$

$$\mathbb{C}\text{ov}(\delta_{k,l}(x_1), f_j(x_2)) = \mathbb{C}\text{ov}(f_k(x_1), f_j(x_2)) - \mathbb{C}\text{ov}(f_l(x_1), f_j(x_2))$$

PATCHWORK KRIGING

Definitions of the covariance blocks :

$$
\begin{aligned}
c_{**} &:= \mathbb{C}\mathrm{ov}(f_*^{(k)}, f_*^{(k)}),\\
c_{*\mathcal{D}}^{(k)} &:= \left(\mathbb{C}\mathrm{ov}(f_*^{(k)}, y_1), \dots, \mathbb{C}\mathrm{ov}(f_*^{(k)}, y_K)\right),\\
c_{*\delta}^{(k)} &:= \left(\mathbb{C}\mathrm{ov}(f_*^{(k)}, \delta_{1,1}), \dots, \mathbb{C}\mathrm{ov}(f_*^{(k)}, \delta_{K,K})\right),
\end{aligned}
$$

$$
C_{\mathcal{D}\mathcal{D}} := \begin{bmatrix} \mathbb{C}\mathrm{ov}(y_1, y_1) & \dots & \mathbb{C}\mathrm{ov}(y_1, y_K) \\ \dots & \ddots & \dots \\ \mathbb{C}\mathrm{ov}(y_K, y_1) & \dots & \mathbb{C}\mathrm{ov}(y_K, y_K) \end{bmatrix}, \quad C_{\mathcal{D}\delta} := \begin{bmatrix} \mathbb{C}\mathrm{ov}(y_1, \delta_{1,1}) & \dots & \mathbb{C}\mathrm{ov}(y_1, \delta_{K,K}) \\ \dots & \ddots & \dots \\ \mathbb{C}\mathrm{ov}(y_K, \delta_{1,1}) & \dots & \mathbb{C}\mathrm{ov}(y_K, \delta_{K,K}) \end{bmatrix} \text{ and}
$$

$$
C_{\delta\delta} := \begin{bmatrix} \mathbb{C}\mathrm{ov}(\delta_{1,1}, \delta_{1,1}) & \dots & \mathbb{C}\mathrm{ov}(\delta_{1,1}, \delta_{K,K}) \\ \dots & \ddots & \dots \\ \mathbb{C}\mathrm{ov}(\delta_{K,K}, \delta_{1,1}) & \dots & \mathbb{C}\mathrm{ov}(\delta_{K,K}, \delta_{K,K}) \end{bmatrix}.
$$

PATCHWORK KRIGING

Predictive mean :

$$\mathbb{E}\left[f_*^{(k)} \mid y, \delta = 0\right] = \left(c_{*\mathfrak{D}}^{(k)} - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right) \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right)^{-1} y.$$

Predictive variance :

$$\begin{aligned}\mathbb{V}\left[f_*^{(k)} \mid y, \delta = 0\right] =& c_{**} - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} c_{\delta *}^{(k)} \\ & - \left(c_{*\mathfrak{D}}^{(k)} - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right) \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right)^{-1} \left(c_{\mathfrak{D}*}^{(k)} - c_{\delta *}^{(k)} C_{\delta\delta}^{-1} C_{\mathfrak{D}\delta}\right)\end{aligned}$$

Noting $Q := \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right)^{-1}, v := L^{-1} C_{\delta\mathfrak{D}}$ and $w_* := L^{-1} c_{\delta *}^{(k)}$ (where $C_{\delta\delta} = LL^T$),

$$\boxed{\begin{aligned}\mathbb{E}\left[f_*^{(k)} \mid y, \delta = 0\right] &= (c_{*\mathfrak{D}}^{(k)} - w_*^T v) Q y. \\ \mathbb{V}\left[f_*^{(k)} \mid y, \delta = 0\right] &= c_{**} - w_*^T w_* - (c_{*\mathfrak{D}}^{(k)} - w_*^T v) Q (c_{*\mathfrak{D}}^{(k)} - w_*^T v)^T.\end{aligned}}$$

PROOF

For two gaussian vectors $\mathbf{x}_1 \sim \mathcal{N}(\mu_1, \Sigma_1)$ and $\mathbf{x}_2 \sim \mathcal{N}(\mu_2, \Sigma_2)$, the conditional distribution $p(\mathbf{x}_1 \mid \mathbf{x}_2)$ is gaussian and verifies :

$$\mathbb{E}\left[\mathbf{x}_1 \mid \mathbf{x}_2\right] = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{x}_2 - \mu_2).$$

$$\mathbb{V}\left[\mathbf{x}_1 \mid \mathbf{x}_2\right] = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}.$$

Applying this to $\mathbf{x}_1 = f_*^{(k)}$ and $\mathbf{x}_2 = (y, \delta)$ gives :

$$\mathbb{E}\left[f_*^{(k)} \mid y, \delta\right] = \left[c_{*\mathcal{D}}^{(k)}, c_{*\delta}^{(k)}\right] \begin{bmatrix} C_{\mathcal{D}\mathcal{D}} & C_{\mathcal{D}\delta} \\ C_{\delta\mathcal{D}} & C_{\delta\delta} \end{bmatrix}^{-1} \begin{bmatrix} y \\ \delta \end{bmatrix}$$

and

$$\mathbb{V}\left[f_*^{(k)} \mid y, \delta\right] = c_{**} - \left[c_{*\mathcal{D}}^{(k)}, c_{*\delta}^{(k)}\right] \begin{bmatrix} C_{\mathcal{D}\mathcal{D}} & C_{\mathcal{D}\delta} \\ C_{\delta\mathcal{D}} & C_{\delta\delta} \end{bmatrix}^{-1} \begin{bmatrix} c_{\mathcal{D}*}^{(k)} \\ c_{\delta*}^{(k)} \end{bmatrix}.$$

PROOF

For invertible matrix A, B, D, the following inversion formula holds true :

$$\begin{bmatrix} A & B \\ B^T & D \end{bmatrix}^{-1} = \begin{bmatrix} \left(A - BD^{-1}B^T\right)^{-1} & -\left(A - BD^{-1}B^T\right)^{-1} BD^{-1} \\ -D^{-1}B^T \left(A - BD^{-1}B^T\right)^{-1} & \left(D - B^T A^{-1} B\right)^{-1} \end{bmatrix}.$$

Applying this to $A = C_{\mathfrak{D}\mathfrak{D}}, B = C_{\mathfrak{D}\delta}, D = C_{\delta\delta}$ gives, for the predictive mean :

$$\mathbb{E}\left[f_*^{(k)} \mid y, \delta\right] = \begin{bmatrix} c_{*\mathfrak{D}}^{(k)} \\ c_{*\delta}^{(k)} \end{bmatrix}^T \begin{bmatrix} \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} & -\left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1} \\ -C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}} \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} & \left(C_{\delta\delta} - C_{\delta\mathfrak{D}}C_{\mathfrak{D}\mathfrak{D}}^{-1}C_{\mathfrak{D}\delta}\right)^{-1} \end{bmatrix} \begin{bmatrix} y \\ \delta \end{bmatrix}$$

$$\begin{aligned} &= c_{*\mathfrak{D}}^{(k)} \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} y - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} C_{\mathfrak{D}\delta}^T \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} y \\ &\quad - c_{*\mathfrak{D}}^{(k)} \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}\delta + c_{*\delta}^{(k)} \left(C_{\delta\delta} - C_{\delta\mathfrak{D}}C_{\mathfrak{D}\mathfrak{D}}^{-1}C_{\mathfrak{D}\delta}\right)^{-1} \delta. \end{aligned}$$

Taking $\delta = 0$ finally leads to :

$$\mathbb{E}\left[f_*^{(k)} \mid y, \delta = 0\right] = \left(c_{*\mathfrak{D}}^{(k)} - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right) \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta}C_{\delta\delta}^{-1}C_{\delta\mathfrak{D}}\right)^{-1} y.$$

PROOF

Concerning the predictive variance, the inversion formula also gives :

$$
\begin{aligned}
&\mathbb{V}\left[f_*^{(k)} \mid y, \delta\right] \\
&= c_{**} - \begin{bmatrix} c_{*\mathcal{D}}^{(k)} \\ c_{*\delta}^{(k)} \end{bmatrix}^T \begin{bmatrix} \left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} & -\left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} \\ -C_{\delta\delta}^{-1} C_{\delta\mathcal{D}} \left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} & \left(C_{\delta\delta} - C_{\delta\mathcal{D}} C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta}\right)^{-1} \end{bmatrix} \begin{bmatrix} c_{\mathcal{D}*}^{(k)} \\ c_{\delta*}^{(k)} \end{bmatrix} \\
&= c_{**} - c_{*\mathcal{D}}^{(k)} \left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} c_{\mathcal{D}*}^{(k)} + c_{*\mathcal{D}}^{(k)} \left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} c_{\delta*}^{(k)} \\
&\quad + c_{*\delta}^{(k)} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}} \left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} c_{\mathcal{D}*}^{(k)} - c_{*\delta}^{(k)} \left(C_{\delta\delta} - C_{\delta\mathcal{D}} C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta}\right)^{-1} c_{\delta*}^{(k)}.
\end{aligned}
$$

This finally yields to :

$$
\mathbb{V}\left[f_*^{(k)} \mid y, \delta = 0\right] = c_{**} - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} c_{\delta*}^{(k)} - \left(c_{*\mathcal{D}}^{(k)} - c_{*\delta}^{(k)} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right) \left(C_{\mathcal{D}\mathcal{D}} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathcal{D}}\right)^{-1} \left(c_{\mathcal{D}*}^{(k)} - C_{\mathcal{D}\delta} C_{\delta\delta}^{-1} c_{\delta*}^{(k)}\right).
$$

CHOICE OF THE PARAMETERS

Covariance function :

- Different for each region $\longrightarrow C_k(\cdot,\cdot)$ for $k \in \{1,\dots,K\}$.
- Obtained by minimizing the negative log-likelihood :

$$NL(\theta) := -\log p(y, \delta = 0 \mid \theta)$$
$$= \frac{N}{2}\log(2\pi) + \frac{1}{2}\log\begin{vmatrix} C_{\mathfrak{D}\mathfrak{D}} & C_{\mathfrak{D}\delta} \\ C_{\delta\mathfrak{D}} & C_{\delta\delta} \end{vmatrix} + \frac{1}{2}\begin{bmatrix} y \\ 0 \end{bmatrix}^T \begin{bmatrix} C_{\mathfrak{D}\mathfrak{D}} & C_{\mathfrak{D}\delta} \\ C_{\delta\mathfrak{D}} & C_{\delta\delta} \end{bmatrix}^{-1} \begin{bmatrix} y \\ 0 \end{bmatrix}.$$

Hyperparameters :

- Number of regions $\boldsymbol{K} = 2^D$ (where D is the depth of the PCA partitioning).
  - ➤ $K \nearrow \Longrightarrow$ Less accurate predictions.
- Number of pseudo-observations $\boldsymbol{B}$ per frontier.
  - ➤ $B \nearrow \Longrightarrow$ More accurate predictions.

ALGORITHM

Steps :

1. Partition the domain : $\Omega = \bigcup_{k=1}^{K} \Omega_k$.
2. Create the pseudo-observations $\delta_{k,\ell}$ on each $\Gamma_{k,\ell}$.
3. Use MLE to choose each covariance function $C_k(\cdot,\cdot)$.
4. Compute $\mathbb{E}\left[f_*^{(k)} \mid y, \delta = 0\right]$ and $\mathbb{V}\left[f_*^{(k)} \mid y, \delta = 0\right]$:
   - Comptute $Q = \left(C_{\mathfrak{D}\mathfrak{D}} - C_{\mathfrak{D}\delta} C_{\delta\delta}^{-1} C_{\delta\mathfrak{D}}\right)^{-1}$.
   - Compute Cholesky decomposition of $C_{\delta\delta} = LL^T$
   - Deduce $v = L^{-1} C_{\delta\mathfrak{D}}$ and $w_* = L^{-1} c_{\delta *}^{(k)}$.
5. Be happy ☺

TABLE OF CONTENTS

INTUITIVE IDEA

- It seems logical to assume that far away observations have little impact on the Kriging.
- We now consider:

$$f_\theta \sim \mathcal{N}(0, K_\theta)$$

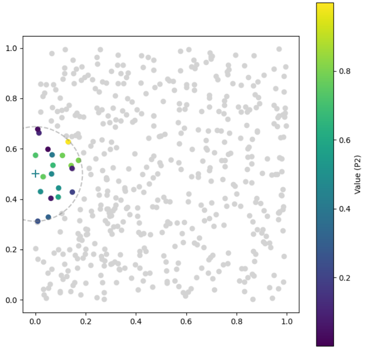
$$f(x^*) = \sum_{j=1}^{n} A_j y_j + \varepsilon$$

- With $A_j = \delta_{N(x^*)}$ Where $N(x^*)$ is the set of x^* nearest neighbors.
- Restraining the Kriging to the m nearest neighbors of each prediction location x^* leads to a sub-optimal prediction.
- Our Kriging variance is now $c_{**} - c_{N(x^*)*}^T (C_{N(x^*)N(x^*)})^{-1} c_{N(x^*)*}$. for the given x^*.

■ NEAREST-NEIGHBORS

Nearest-Neighbors are the points that have the highest spatial correlation with the prediction location x^*:

- In the case of stationary covariance functions, these are simply the closest points to x^*.
- For non-stationary covariance functions, the nearest neighbors are the maxima of the covariance function with



CHOOSING M

- First idea : running cross-validation for different values of m and choosing the one that minimizes the prediction error.

- In Bayesian Inference, m is treated as a parameter and updated in the MCMC algorithm.

$$p(f_\theta, \beta, \theta, \tau^2, m \mid y) \propto p(y \mid f_\theta, \beta, \tau^2) \times p(f_\theta \mid \theta) \times p(\beta)p(\theta)p(\tau^2)p(m)$$

TABLE OF CONTENTS

■ COMPLEXITY ANALYSIS I

For Method I (**Patchwork Kriging**), the expense is located in the computation of:

$$(C_{\delta\delta} - C_{\mathcal{D}\delta}^T C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta})^{-1}$$

Reminder

- N observations in a space of dimension d
- Split into K regions of $M = \frac{N}{K}$ points, with B pseudo-observations at each boundary

COMPLEXITY ANALYSIS I

For Method I (**Patchwork Kriging**), the expense is located in the computation of:

$$(C_{\delta\delta} - C_{\mathcal{D}\delta}^T C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta})^{-1}$$

Reminder

- N observations in a space of dimension d
- Split into K regions of $M = \frac{N}{K}$ points, with B pseudo-observations at each boundary

Complexity

- $C_{\mathcal{D}\mathcal{D}}^{-1}$ block diagonal $\rightarrow \boldsymbol{\mathcal{O}(KM^3)}$
- $C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta} \rightarrow \boldsymbol{\mathcal{O}(KBM^2)}$
- Inverting $C_{\delta\delta} - C_{\mathcal{D}\delta}^T C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta}$. As it is very sparse, we can use efficient methods to invert sparse matrix (Chan and George (1980)) $\longrightarrow \boldsymbol{\mathcal{O}(d^3B^3K)}$

COMPLEXITY ANALYSIS I

For Method I (**Patchwork Kriging**), the expense is located in the computation of:

$$(C_{\delta\delta} - C_{\mathcal{D}\delta}^T C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta})^{-1}$$

Reminder

- N observations in a space of dimension d
- Split into K regions of $M = \frac{N}{K}$ points, with B pseudo-observations at each boundary

Complexity

- $C_{\mathcal{D}\mathcal{D}}^{-1}$ block diagonal $\rightarrow \mathcal{O}(\boldsymbol{KM^3})$
- $C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta} \rightarrow \mathcal{O}(\boldsymbol{KBM^2})$
- Inverting $C_{\delta\delta} - C_{\mathcal{D}\delta}^T C_{\mathcal{D}\mathcal{D}}^{-1} C_{\mathcal{D}\delta}$. As it is very sparse, we can use efficient methods to invert sparse matrix (Chan and George (1980)) $\longrightarrow \mathcal{O}(\boldsymbol{d^3B^3K})$

$$\longrightarrow \mathcal{O}(\boldsymbol{KM^3 + KBM^2 + d^3B^3K}) \approx \mathcal{O}(\boldsymbol{\frac{N^3}{K^2} + d^3B^3K})$$

ACCURACY

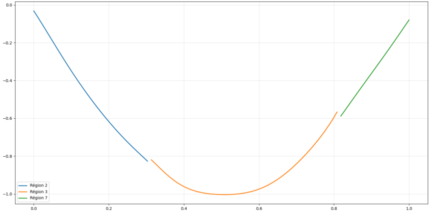
For Method I (**Patchwork Kriging**), the accuracy concerns :

- Quality of local reconstruction
- Quality of "stiches" (junctions of regions)

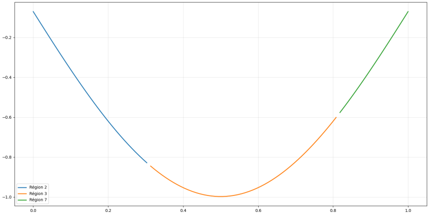
ACCURACY

For Method I (**Patchwork Kriging**), the accuracy concerns :

- Quality of local reconstruction
- Quality of "stiches" (junctions of regions)



B = 0



B = 5

RESULTS

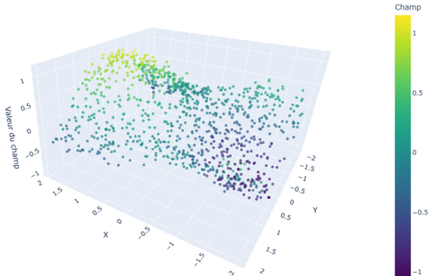
In Park and Apley (2018), Tests have been made to understand the extent of the Patchwork Kriging Technique

- Result are the most accurate for **long-range, stationary** covariance functions.
- Stitching accuracy is **conditioned by** B, but stops for $B \geq 8$

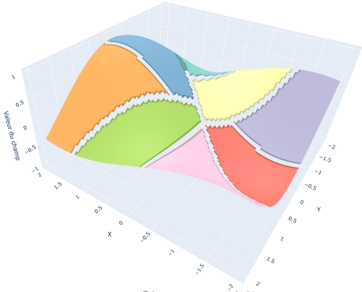
■ RESULTS

In Park and Apley (2018), Tests have been made to understand the extent of the Patchwork Kriging Technique

- Result are the most accurate for **long-range, stationary** covariance functions.
- Stitching accuracy is **conditioned by** B, but stops for $B \geq 8$



Observations bruitées



Champ prédit

RESULTS

In Park and Apley (2018), Tests have been made to understand the extent of the Patchwork Kriging Technique

- Result are the most accurate for **long-range, stationary** covariance functions.
- Stitching accuracy is **conditioned by** B, but stops for $B \geq 8$

High dimensionality ?

- Recall the complexity $\longrightarrow \boldsymbol{\mathcal{O}(\frac{N^3}{K^2} + d^3B^3K)}$
- Experimentally, for $d > 10$, dimension is a determining factor for computation time

$\rightarrow$ Trade-of quality/speed : keep $\dfrac{N}{K}$ and dB as low as possible.

COMPLEXITY ANALYSIS II

For Method II (**NN-Kriging**), the expense is located in the computation of

$$c_{**} - c_{N(x^*)*}^T (C_{N(x^*)N(x^*)})^{-1} c_{N(x^*)*}.$$

Reminder

- N observations of a space of dimension d
- for each new location, a neighbourhood of size m is computed

COMPLEXITY ANALYSIS II

For Method II (**NN-Kriging**), the expense is located in the computation of

$$c_{**} - c_{N(x^*)*}^T (C_{N(x^*)N(x^*)})^{-1} c_{N(x^*)*}.$$

Reminder

- N observations of a space of dimension d
- for each new location, a neighbourhood of size m is computed

Complexity

- Computing a new location$(C_{N(x^*)N(x^*)})^{-1} \rightarrow \mathbf{\mathfrak{O}(m^3)}$
- Computing n new location$(C_{N(x^*)N(x^*)})^{-1} \rightarrow \mathbf{\mathfrak{O}(nm^3)}$

RESULTS II

In Datta et al. (2016), Tests have been made to mesure performances of NNGP

		NNGP		
	True	$m = 5$	$m = 10$	Full Gaussian Process
$\beta_{1,0}$	1	0.81 (0.22, 1.43)	0.64 (−0.05, 1.45)	0.82 (−0.11, 1.71)
$\beta_{1,1}$	−5	−4.94 (−5.02, −4.85)	−4.95 (−5.04, −4.86)	−4.95 (−5.04, −4.86)
$\beta_{2,0}$	1	1.03 (0.15, 2.02)	1.31 (0.26, 2.37)	0.95 (−0.27, 2.18)
$\beta_{2,1}$	5	5.03 (4.91, 5.15)	5.02 (4.89, 5.14)	5.01 (4.89, 5.13)
τ_1^2	0.1	0.07 (0.02, 0.25)	0.06 (0.02, 0.25)	0.07 (0.02, 0.24)
τ_2^2	0.1	0.10 (0.02, 0.87)	0.08 (0.02, 0.70)	0.09 (0.03, 1.33)
Time (in minutes)	–	18.82	75.62	369.10

results of multivariate simulations for several values of m

Conclusion

Merci de votre attention !

"On ne peut pas inverser une matrice de taille 20000 une fois, mais on peut inverser 20000 matrice de taille 20 une fois... enfin je crois"

— *L. Lambert*

REFERENCES I

Chan, W. M. and George, A. (1980). A linear time implementation of the reverse cuthill-mckee algorithm. *BIT Numerical Mathematics*, 20:8–14.

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016). On nearest-neighbor Gaussian process models for massive spatial data. *Wiley Interdisciplinary Reviews Computational Statistics*, 8(5):162–171.

Park, C. and Apley, D. (2018). Patchwork Kriging for large-scale Gaussian process regression. *Journal of Machine Learning Research*.