Gaussian process modelling and applications

Andrés F. López-Lopera
DMI, Céramaths, Université Polytechnique Hauts-de-France

December 09, 2021

Outline

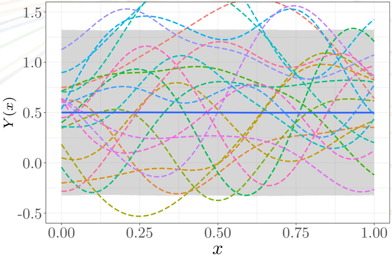
1. Gaussian processes
2. Physically-inspired Gaussian processes
3. Spatial Gaussian processes with functional inputs
4. Multi-fidelity Gaussian processes
5. Gaussian processes under inequality constraints

Gaussian processes

Gaussian processes as flexible priors

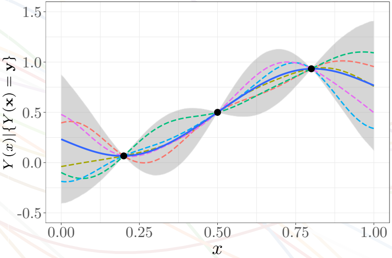
- They provide a well-founded (Bayesian) non-parametric framework

Gaussian prior



$Y \sim \mathcal{GP}(m, k_{\boldsymbol{\theta}})$

Gaussian posterior



$Y|\{Y(\boldsymbol{x}) = \boldsymbol{y}\} \sim \mathcal{GP}(m_{\text{cond}}, c_{\boldsymbol{\theta}})$

■ mean function ■ uncertainty ■ ■ · · · ■ GP samples

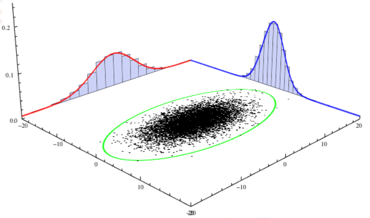
● interpolation points: $(\boldsymbol{x}, \boldsymbol{y}) = (x_i, y_i)_{i=1}^{n}$

Gaussian processes

· A GP $\{Y(\boldsymbol{x}), \boldsymbol{x} \in \mathbb{R}^d\}$ is a collection of random variables, any finite number of which have a joint Gaussian distribution [Rasmussen and Williams, 2005]:

$$\mathbf{Y} = [Y(x_1), Y(x_2), \cdots, Y(x_n)]^\top \sim \mathcal{N}(\boldsymbol{m}, \mathbf{K}), \quad (1)$$

with mean vector $\boldsymbol{m} \in \mathbb{R}^n$ and covariance matrix $\mathbf{K} \in \mathbb{R}^{n \times n}$. [1]



· The probability density function (pdf) of $\mathbf{Y}$ is given by

$$p(\mathbf{Y}) = \frac{1}{\sqrt{(2\pi)^n |\mathbf{K}|}} \exp\left(-\frac{1}{2}[\mathbf{Y} - \boldsymbol{m}]^\top \mathbf{K}^{-1}[\mathbf{Y} - \boldsymbol{m}]\right).$$

[1] **Remember:** A vector $\mathbf{Z} = [Z_1, \ldots, Z_d]^\top \in \mathbb{R}^d$ is said to be Gaussian if for any $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$, the real-valued random variable $Z_* = \alpha_1 Z_1 + \cdots + \alpha_n Z_d$ is Gaussian-distributed.

· Y is completely defined by its mean m and covariance (kernel) k functions:

$$Y \sim \mathcal{GP}(m, k), \tag{2}$$

where

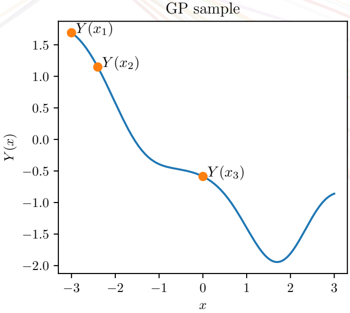
$$\begin{aligned} &\text{(trend)} & m(\boldsymbol{x}) &= \mathbb{E}\left\{Y(\boldsymbol{x})\right\}, \\ &\text{(covariance)} & k(\boldsymbol{x}, \boldsymbol{x}') &= \mathsf{cov}\left\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\right\}, \quad \text{for} \quad \boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^d. \end{aligned} \tag{3}$$

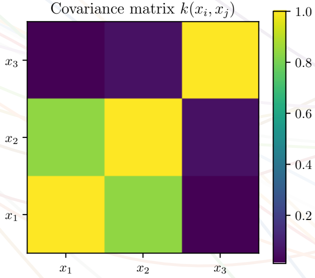
· The operator $\mathbb{E}$ denotes the expectation of random variables (r.v's), and the covariance operator is given by

$$\mathsf{cov}\left\{Y(\boldsymbol{x}), Y(\boldsymbol{x}')\right\} = \mathbb{E}\left\{[Y(\boldsymbol{x}) - m(\boldsymbol{x})][Y(\boldsymbol{x}') - m(\boldsymbol{x}')]\right\}.$$

Note. Independence between $Y(\boldsymbol{x}), Y(\boldsymbol{x}')$ implies that $k(\boldsymbol{x}, \boldsymbol{x}') = 0$

Squared Exponential (SE) kernel





· If $Y(\boldsymbol{x}), Y(\boldsymbol{x}')$ are correlated, then $k(\boldsymbol{x}, \boldsymbol{x}') \neq 0$

Kernel functions

- The covariance function (or kernel) must be:

$$\text{Symmetric:} \qquad k(\boldsymbol{x}_i, \boldsymbol{x}_j) = k(\boldsymbol{x}_j, \boldsymbol{x}_i)$$

$$\text{Positive Semi-Definite (p.s.d.):} \qquad \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$$

- The covariance function is said to be stationary if:

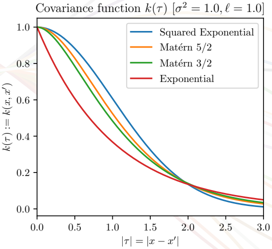
$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = k_S(\boldsymbol{x}_i - \boldsymbol{x}_j)$$

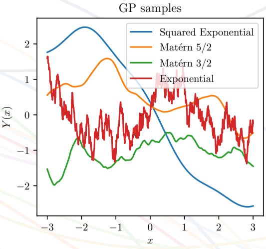
Common stationary kernel functions used in GP regression models in 1D.

Type of kernel	Expression for $k(x, x')$	Class
Squared Exponential	$\sigma^2 \exp\left\{-\frac{(x-x')^2}{2\ell^2}\right\}$	C^∞
Matérn 5/2	$\sigma^2 \left(1 + \frac{\sqrt{5}\lvert x-x'\rvert}{\ell} + \frac{5(x-x')^2}{3\ell^2}\right) \exp\left\{-\frac{\sqrt{5}\lvert x-x'\rvert}{\ell}\right\}$	C^2
Matérn 3/2	$\sigma^2 \left(1 + \frac{\sqrt{3}\lvert x-x'\rvert}{\ell}\right) \exp\left\{-\frac{\sqrt{3}\lvert x-x'\rvert}{\ell}\right\}$	C^1
Exponential	$\sigma^2 \exp\left\{-\frac{\lvert x-x'\rvert}{\ell}\right\}$	C

Kernel functions

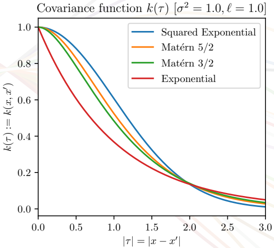
- In GPs, regularity assumptions are commonly encoded in kernels

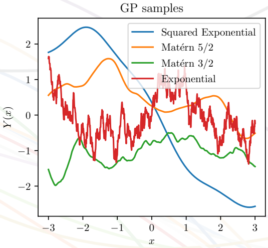


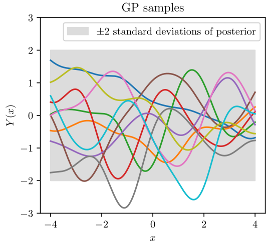


Kernel functions

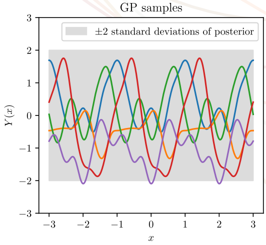
- In GPs, regularity assumptions are commonly encoded in kernels



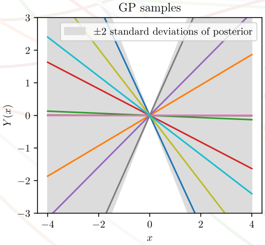




SE kernel



periodic kernel



linear kernel

Gaussian process regression

- Let $\{Y(\boldsymbol{x}), \boldsymbol{x} \in \mathbb{R}^d\}$ be a zero-mean GP with covariance function k
- In regression tasks, we aim at computing the conditional distribution:

$$Y | \{Y(\boldsymbol{x}_1) = y_1, \ldots, Y(\boldsymbol{x}_n) = y_n\},$$

for a set of observations $(\boldsymbol{x}_i, y_i)_{1 \leq i \leq n}$ for $n \in \mathbb{N}$

Gaussian process regression

· Let $\{Y(\boldsymbol{x}), \boldsymbol{x} \in \mathbb{R}^d\}$ be a zero-mean GP with covariance function k

· In regression tasks, we aim at computing the conditional distribution:

$$Y | \{Y(\boldsymbol{x}_1) = y_1, \ldots, Y(\boldsymbol{x}_n) = y_n\},$$

for a set of observations $(\boldsymbol{x}_i, y_i)_{1 \leq i \leq n}$ for $n \in \mathbb{N}$

· This conditional process is also GP-distributed with (conditional) mean and covariance functions given by

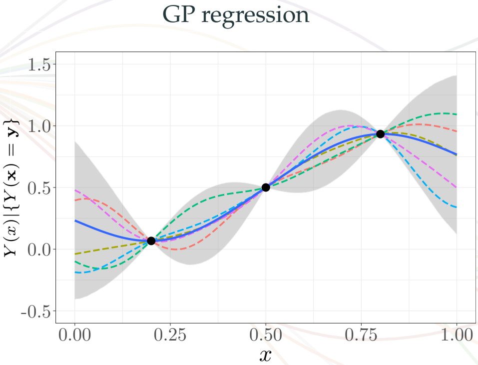
$$\mu(\boldsymbol{x}) = \boldsymbol{k}^\top(\boldsymbol{x})\, \mathbf{K}_n^{-1}\, \boldsymbol{y},$$
$$c(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}^\top(\boldsymbol{x})\, \mathbf{K}_n^{-1}\, \boldsymbol{k}(\boldsymbol{x}'),$$

with $\boldsymbol{k}(\boldsymbol{x}) = (\mathsf{cov}\,\{Y(\boldsymbol{x}), Y(\boldsymbol{x}_i)\})_{1 \leq i \leq n}$ and $\mathbf{K}_n = (\mathsf{cov}\,\left\{Y(\boldsymbol{x}_i), Y(\boldsymbol{x}_j)\right\})_{1 \leq i,j \leq n}$

· The conditional variance is given by

$$v(\boldsymbol{x}) = c(\boldsymbol{x}, \boldsymbol{x}) = k(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}^\top(\boldsymbol{x})\, \mathbf{K}_n^{-1}\, \boldsymbol{k}(\boldsymbol{x})$$

Gaussian process regression



conditional mean   confidence intervals $\cdots$ GP realisations

• training data: $(\boldsymbol{x}, \boldsymbol{y}) = (x_i, y_i)_{i=1}^n$

[Link]

Gaussian process regression with noisy observation

· For noisy observations, we have the conditional process:

$$Y|\{Y(\boldsymbol{x}_1) + \varepsilon_1 = y_1, \dots, Y(\boldsymbol{x}_n) + \varepsilon_n = y_n\},$$

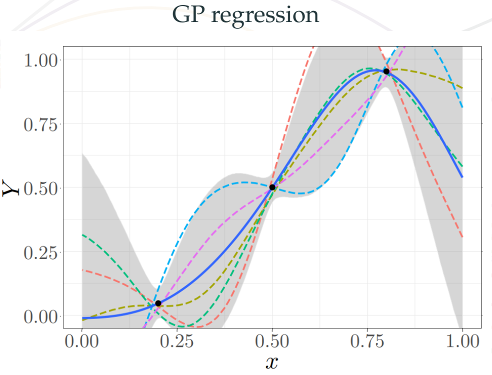
with additive noises $\varepsilon_i \sim \mathcal{N}\left(0, \tau^2\right)$, for $i = 1, \dots, n$, and noise variance τ^2

· This conditional process is also GP-distributed with conditional mean and covariance functions given by

$$\widetilde{\mu}(\boldsymbol{x}) = \boldsymbol{k}^\top(\boldsymbol{x})\, [\mathbf{K}_n + \tau^2 \mathbf{I}]^{-1}\, \boldsymbol{y},$$
$$\widetilde{c}(\boldsymbol{x}, \boldsymbol{x}') = k(\boldsymbol{x}, \boldsymbol{x}') - \boldsymbol{k}^\top(\boldsymbol{x})[\mathbf{K}_n + \tau^2 \mathbf{I}]^{-1}\, \boldsymbol{k}(\boldsymbol{x}'),$$

with $\boldsymbol{k}(\boldsymbol{x}) = (\mathrm{cov}\,\{Y(\boldsymbol{x}), Y(\boldsymbol{x}_i)\})_{1 \le i \le n}$ and $\mathbf{K}_n = (\mathrm{cov}\,\left\{Y(\boldsymbol{x}_i), Y(\boldsymbol{x}_j)\right\})_{1 \le i,j \le n}$

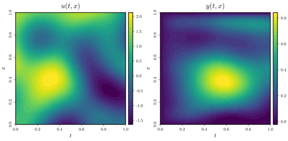
Gaussian process regression with noisy observation



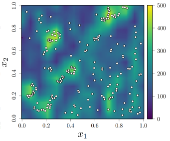
conditional mean · confidence intervals · · · · GP realisations

• training data: $(\boldsymbol{x}, \boldsymbol{y}) = (x_i, y_i)_{i=1}^n$

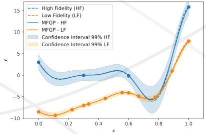
Applications



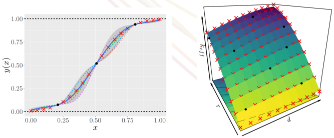
(a) Differential equations



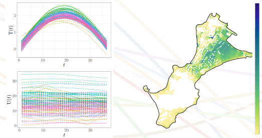
(b) Point processes



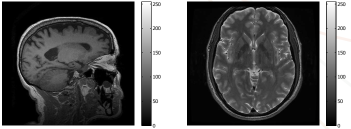
(c) Multi-fidelity



(d) Inequality constraints



(e) Functional data



(f) Image processing

Research projects based on GP modelling

Physically-inspired Gaussian processes

Linear operations of GPs

- Let $U \sim \mathcal{GP}(0, k_{u,u})$ with covariance function:

$$k_{u,u}(\boldsymbol{x}, \boldsymbol{x}') = \mathsf{cov}\left\{U(\boldsymbol{x}), U(\boldsymbol{x}')\right\} = \mathbb{E}\left\{U(\boldsymbol{x})U(\boldsymbol{x}')\right\}.$$

- For any linear operator $\mathcal{L}$ that commutes with the covariance,

$$Y = \mathcal{L} \circ U := \mathcal{L}(U), \tag{4}$$

we have that $Y \sim \mathcal{GP}(0, k_{y,y})$ with covariance function:

$$k_{y,y}(\boldsymbol{x}, \boldsymbol{x}') = \mathsf{cov}\left\{\mathcal{L} \circ U(\boldsymbol{x}), \mathcal{L} \circ U(\boldsymbol{x}')\right\} = \mathcal{L} \circ \mathcal{L}' \circ k_{u,u}(\boldsymbol{x}, \boldsymbol{x}').$$

- Furthermore, *cross-covariance* functions can be established:

$$k_{y,u}(\boldsymbol{x}, \boldsymbol{x}') = \mathsf{cov}\left\{\mathcal{L} \circ U(\boldsymbol{x}), U(\boldsymbol{x}')\right\} = \mathcal{L} \circ k_{u,u}(\boldsymbol{x}, \boldsymbol{x}'),$$
$$k_{u,y}(\boldsymbol{x}, \boldsymbol{x}') = \mathsf{cov}\left\{U(\boldsymbol{x}), \mathcal{L} \circ U(\boldsymbol{x}')\right\} = \mathcal{L}' \circ k_{u,u}(\boldsymbol{x}, \boldsymbol{x}').$$

Linear operations of GPs

· Hence, the joint process (U, Y) is also (centred) GP-distributed:

$$\begin{bmatrix} \mathbf{U} \\ \mathbf{Y} \end{bmatrix} = \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}_{u,u} & \mathbf{K}_{y,u}^{\top} \\ \mathbf{K}_{y,u} & \mathbf{K}_{y,y} \end{bmatrix}\right),$$

and, therefore, conditional GP formulas can be applied.

· e.g. U can be inferred using observations of Y:

$$\mathbf{U}|\{\mathbf{Y} = \boldsymbol{y}\} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{K}_{u,u} - \mathbf{K}_{y,u}^{\top}\mathbf{K}_{y,y}^{-1}\mathbf{K}_{y,u}\right).$$

Physically-inspired Gaussian processes: First-order ODE

· Consider the first-order ODE:

$$\frac{dY(t)}{dt} + \gamma Y(t) = SU(t), \tag{5}$$

with $\gamma \in \mathbb{R}^+, S \in \mathbb{R}^+$

· Assuming that $Y(0) = 0$, then we have

$$Y(t) = Sc(t) \int_0^t U(\tau) \exp(\gamma\tau) d\tau, \quad \text{avec} \quad c(t) = \exp(-\gamma t)$$

· If $U \sim \mathcal{GP}(0, k_{u,u})$, we can show that Y is also a GP with covariance function:

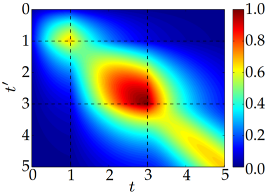
$$\begin{aligned} k_{y,y}(t,t') &:= \mathrm{cov}\left\{Y(t), Y(t')\right\} \left(= \mathbb{E}\left\{Y(t)Y(t')\right\}\right) \\ &= S^2 c(t)c(t') \int_0^t \exp(\gamma\tau) \int_0^{t'} \exp(\gamma\tau') \underbrace{k_{u,u}(\tau,\tau')}_{\mathbb{E}\{U(\tau),U(\tau')\}} d\tau' d\tau \end{aligned}$$

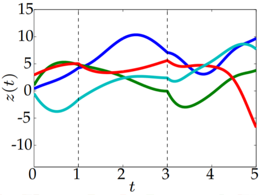
· Finally, we have that $U | \{Y(t_1) = y_1, \ldots, Y(t_n) = y_n\} \sim \mathcal{GP}(\mu_{u|y}, k_{u|y})$

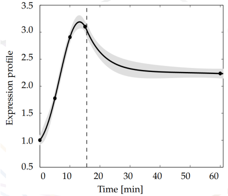
First-order ODE

M. Álvarez
Univ. of Sheffield, UK

Biology: prediction of protein concentrations







We considered the coupled system of ODEs:

$$\frac{dy_d(t)}{dt} + \gamma_d y_d(t) = B_d + \sum_{r=1}^{R} S_{r,d} u_r(t),$$

with $u_1, \ldots, u_R$ independent GPs.

- **A. F. López-Lopera and M. Álvarez:**
Switched latent force models for reverse-engineering transcriptional regulation in genes
IEEE/ACM Transaction on Computational Biology and Bioinformatics, 2019

Physically-inspired Gaussian processes: Spatio-temporal PDE

M. Álvarez
Univ. of Sheffield, UK

N. Durrande
Secondmind, UK

· We can consider the (linear) reaction-diffusion equation:

$$\frac{\partial Y(x,t)}{\partial t} = SU(x,t) - \lambda Y(x,t) + D\frac{\partial^2 Y(x,t)}{\partial x^2},$$

where

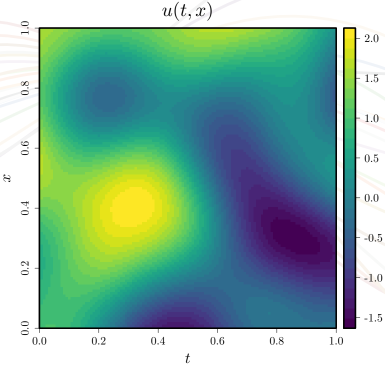
- Y: relative *gap protein concentration*;
- U: *messenger RNA* (mRNA);
- S, D, λ: *translation*, *decay* and *diffusion* rate constants, respectively.

· For simplicity, we assume homogeneous conditions:

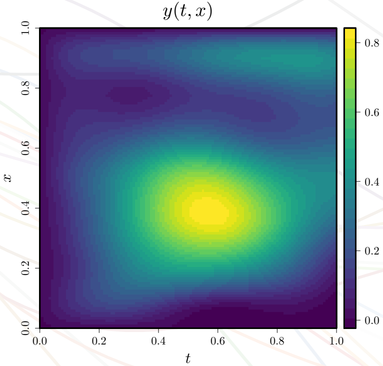
$$Y(x, t=0) = 0, \qquad Y(x=0,t) = Y(x=l,t) = 0, \text{ for } x \in [0,l], l \in \mathbb{R}^+.$$

- **A. F. López-Lopera, N. Durrande and M. Álvarez:**
Physically-inspired Gaussian process models for post-transcriptional regulation in Drosophila
IEEE/ACM Transaction on Computational Biology and Bioinformatics, 2021

Spatio-temporal PDE



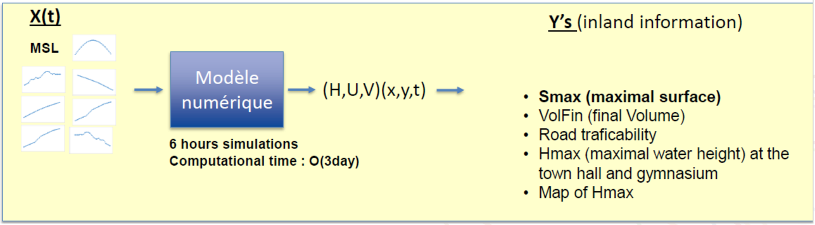
(a) mRNA



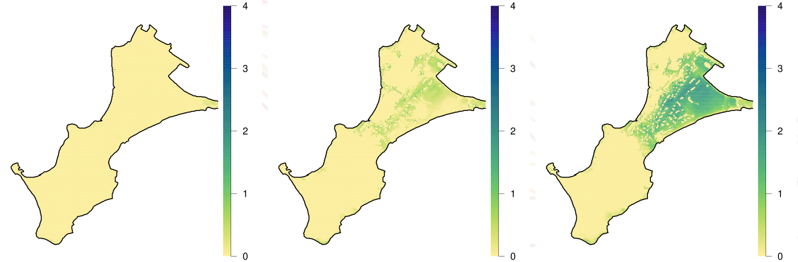
(b) Protein

Sample from the GP-mRNA model

Spatial Gaussian processes with functional inputs


X(t)
MSL
Modèle numérique
6 hours simulations
Computational time : O(3day)
(H,U,V)(x,y,t)
Y's (inland information)
• Smax (maximal surface)
• VolFin (final Volume)
• Road traficability
• Hmax (maximal water height) at the town hall and gymnasium
• Map of Hmax

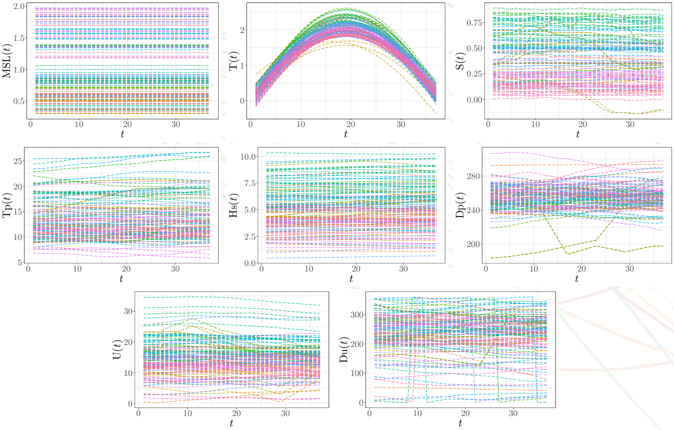
Spatial flood events: maximal inland water level (H_{max} [m])



Challenge 1:

· Each flood event takes $\sim$ **3 days of simulation**.

Drivers: hydro-meteorological conditions (tide, surge, wind speed, etc.)



Challenge 2:

· To consider inputs as functions (time-series) rather than scalars.

Motivation: Coastal flooding assessment

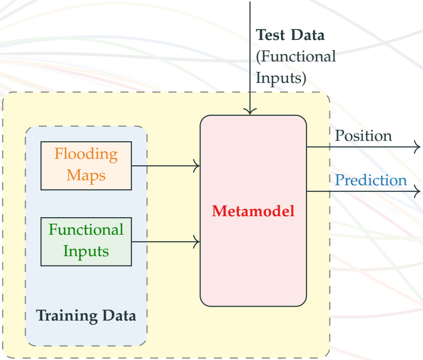
D. Idier
BRGM, Orléans

J. Rohmer
BRGM, Orléans

F. Bachoc
IMT, Toulouse

Goal:

to build a **metamodel** accounting for both spatial and functional data



· This will lead to faster (approximate) predictions.

Application: forecasting and early warning systems.

- **A. F. López-Lopera, D. Idier, J. Rohmer and F. Bachoc:**
Multi-output Gaussian processes with functional data: A study on coastal flood hazard assessment
Reliability Engineering and System Safety, 2021

Spatial Gaussian processes with functional inputs

· Let $\{Y(\boldsymbol{x}, \boldsymbol{\mathcal{F}}); \boldsymbol{x} \in \mathbb{R}^2, \boldsymbol{\mathcal{F}} \in \mathcal{F}(\mathcal{T}, \mathbb{R})^Q\}$ be a GP with spatial inputs $\boldsymbol{x} = (x_1, x_2)$ and functional inputs $\boldsymbol{\mathcal{F}} = (f_1, \cdots, f_Q)$.

· Then, $Y \sim \mathcal{GP}(0, k)$ with kernel k:

$$k((\boldsymbol{x}, \boldsymbol{\mathcal{F}}), (\boldsymbol{x}', \boldsymbol{\mathcal{F}}')) = \mathrm{cov}\left\{Y(\boldsymbol{x}, \boldsymbol{\mathcal{F}}), Y(\boldsymbol{x}', \boldsymbol{\mathcal{F}}')\right\} \tag{6}$$

· k must be defined according to the structure of data (i.e. regularity).

· Here, we consider that k is a *separable kernel*:

$$k((\boldsymbol{x}, \boldsymbol{\mathcal{F}}), (\boldsymbol{x}', \boldsymbol{\mathcal{F}}')) = k_s(\boldsymbol{x}, \boldsymbol{x}')k_f(\boldsymbol{\mathcal{F}}, \boldsymbol{\mathcal{F}}'), \tag{7}$$

with sub-kernels $k_s : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$ and $k_f : \mathcal{F}(\mathcal{T}, \mathbb{R})^Q \times \mathcal{F}(\mathcal{T}, \mathbb{R})^Q \to \mathbb{R}$.

· For k_f, we need a *measure of "dissimilarity"*, e.g.:

$$d(\mathcal{F}, \mathcal{F}') = \|\mathcal{F} - \mathcal{F}'\|_{\boldsymbol{\ell}} = \sqrt{\sum_{i=1}^{Q} \|f_i - f_i'\|_{\ell_i}^2}, \tag{8}$$

with the L^2-norm given by

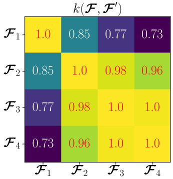
$$\|f_i - f_i'\|_{\ell_i}^2 = \frac{\int_{\mathcal{T}} (f_i(t) - f_i'(t))^2 dt}{{\ell_i}^2}. \tag{9}$$

· **Note:** f_i, f_i' are projected onto *basis functions* (e.g. PCA) for computing (9).

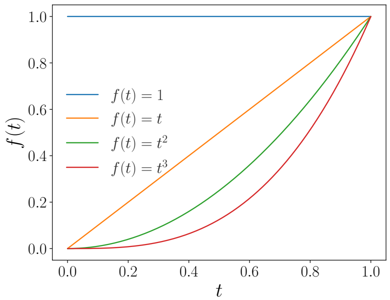
· Then, an example of a valid kernel is then given by

$$\text{(Squared Exponential)} \quad k_{f,\sigma^2,\boldsymbol{\ell}}(\mathcal{F}, \mathcal{F}') = \sigma^2 \exp\left\{-\frac{\|\mathcal{F} - \mathcal{F}'\|_{\boldsymbol{\ell}}^2}{2}\right\}. \tag{10}$$

Spatial Gaussian processes with functional inputs



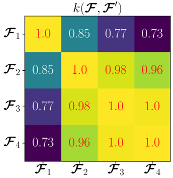
(a) Squared Exponential



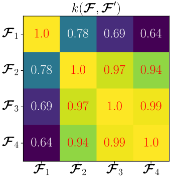
(e) target functions

Effect of the kernels considering $\mathcal{F}_1 = (f(t) = 1)$, $\mathcal{F}_2 = (f(t) = t)$, $\mathcal{F}_3 = (f(t) = t^2)$ and $\mathcal{F}_4 = (f(t) = t^3)$.

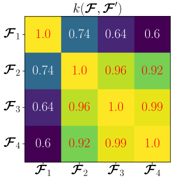
Spatial Gaussian processes with functional inputs



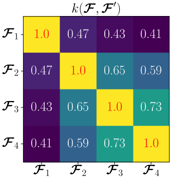
(a) Squared Exponential



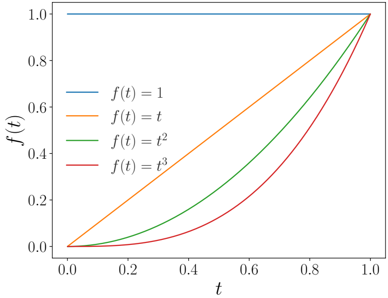
(b) Matérn 5/2



(c) Matérn 3/2



(d) Exponential



(e) target functions

Effect of the kernels considering $\boldsymbol{\mathcal{F}}_1 = (f(t) = 1)$, $\boldsymbol{\mathcal{F}}_2 = (f(t) = t)$, $\boldsymbol{\mathcal{F}}_3 = (f(t) = t^2)$ and $\boldsymbol{\mathcal{F}}_4 = (f(t) = t^3)$.

Link with multi-output Gaussian processes

- Note that Y can be written as a multi-output process Z:

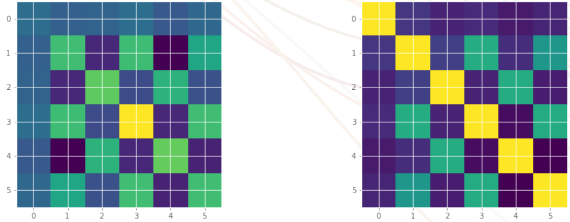
$$Z_i(\boldsymbol{x}) := Y(\boldsymbol{\mathcal{F}}_i, \boldsymbol{x}), \quad \text{for } i = 1, \cdots, R.$$

- In that case, k can be rewritten as:

$$k_{i,j}(\boldsymbol{x}, \boldsymbol{x}') = b_{i,j} \, k_s(\boldsymbol{x}, \boldsymbol{x}'), \tag{11}$$

with $b_{i,j} := k_f(\boldsymbol{\mathcal{F}}_i, \boldsymbol{\mathcal{F}}_j)$, for $i, j = 1, \ldots, R$.

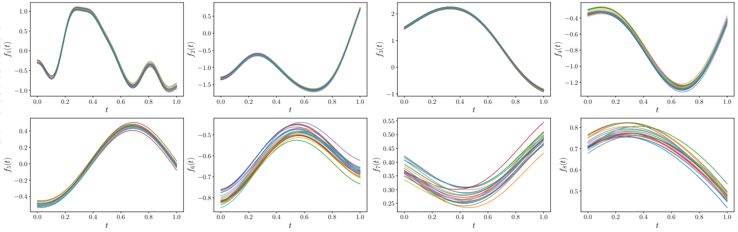
- k follows the structure of the *linear models of coregionalisation* (LMC):



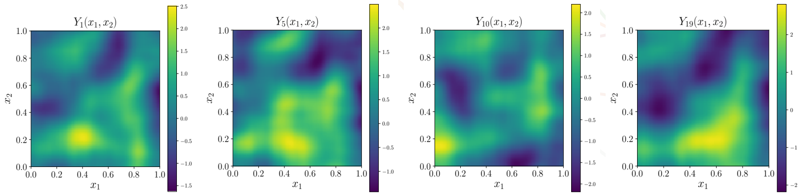
$\mathbf{B}$ $\qquad$ $k_f(\boldsymbol{\mathcal{F}}, \boldsymbol{\mathcal{F}}')$

Coregionalisation matrix

- Synthetic example with **8 functional inputs** and **20 spatial outputs**.



functional replicates



synthetic 100×100 maps

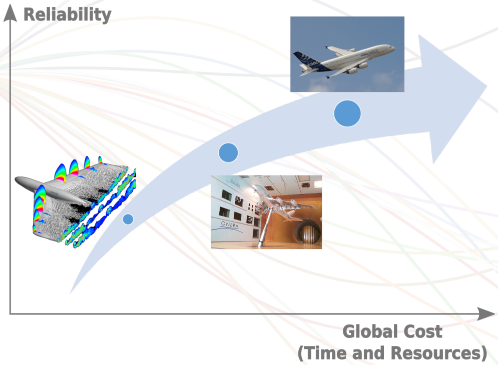
Extension to large datasets

- To scale the proposed GP framework to large datasets, we can:
  - via *Kronecker-products*
  - via *sparse-variational inference*
  - via *dimension reduction* of the output space

Multi-fidelity Gaussian processes

Multi-fidelity Gaussian processes

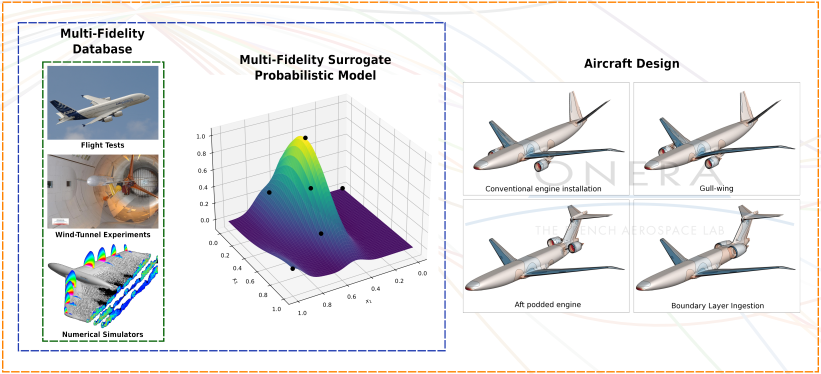
- Different data acquisition schemes may lead to different levels of fidelity



Acquisition Scheme	Level of Fidelity	Data Availability	Cost
flight tests	high	very low	expensive
wind-tunnel tests	upper-intermediate	intermediate	moderate
simulators	low or intermediate	high	cheap or moderate

Multi-fidelity Gaussian processes

· Data fusion (DF)-based frameworks aim at jointly treating data acquisition schemes while accounting for their corresponding levels of fidelity



Multi-fidelity DF-based architecture

Multi-fidelity model based on Gaussian processes

· We can consider the autoregressive model

$$Y_\ell(\boldsymbol{x}) = \rho_\ell(\boldsymbol{x}) Y_{\ell-1}(\boldsymbol{x}) + \delta_\ell(\boldsymbol{x}), \quad \text{for } \ell = 1, \ldots, L \tag{12}$$

where

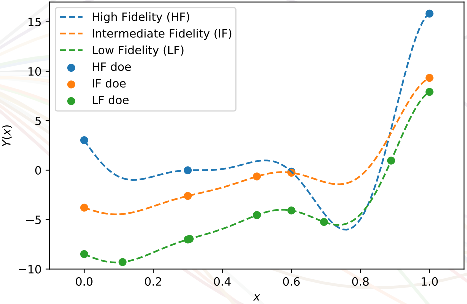
- $L+1$ is number of fidelity levels
- $Y_0 \sim \mathcal{GP}(m_0, k_0)$ and $\delta_\ell(\boldsymbol{x}) \sim \mathcal{GP}(m_\ell, k_\ell)$
- $\rho_\ell : \mathcal{D} \to \mathbb{R}$ is a scale factor between Z_ℓ and $Z_{\ell-1}$
- $\delta_\ell : \mathcal{D} \to \mathbb{R}$ is the discrepancy function tasked with capturing the differences between Z_ℓ and $Z_{\ell-1}$ beyond scaling

· If $Y_0, \delta_1, \ldots, \delta_L$ are independent GPs, then $Y_1, \ldots, Y_L$ are also GP-distributed

· Note that :

$$\operatorname{cov}\left\{Y_\ell(\boldsymbol{x}), Y_\ell(\boldsymbol{x}')\right\} = \rho_\ell(\boldsymbol{x})\rho_\ell(\boldsymbol{x}') \operatorname{cov}\left\{Y_{\ell-1}(\boldsymbol{x}), Y_{\ell-1}(\boldsymbol{x}')\right\} + \operatorname{cov}\left\{\delta_\ell(\boldsymbol{x}), \delta_\ell(\boldsymbol{x}')\right\}$$

- **R. Conde-Arenzana, A. F. López-Lopera, S. Mouton, N. Bartoli and T. Lefebvre:**
Multi-fidelity Gaussian process model for CFD and wind tunnel data fusion
Proceedings of the AeroBest conference, 2021

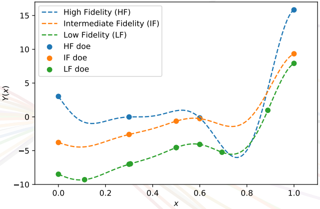


· For efficient implementations, we consider nested databases:

$$D_\ell \subseteq D_{\ell-1}, \tag{13}$$

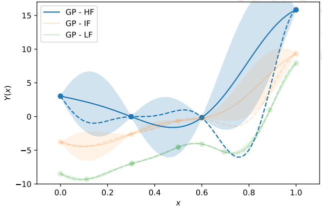
with $D_\ell = (\boldsymbol{x}_{\ell,i})_{0\leq \ell \leq L-1, 1\leq i\leq n_\ell}$.

Multi-fidelity model based on Gaussian processes

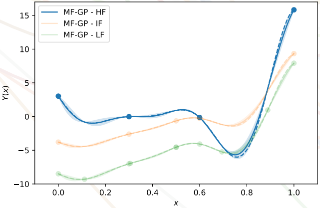


(a) True functions and design points

- The (nested) dataset contains:
  - 4 high-fidelity (HF) design points
  - 5 intermediate-fidelity (IF) design points
  - 8 low-fidelity (LF) design points



(b) Independent GP models



(c) Multi-fidelity GP model

1D multi-fidelity regression example with 3 levels of fidelity

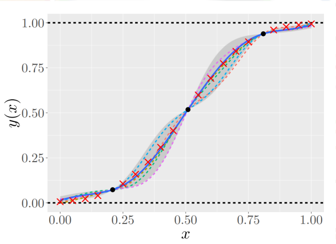
Research ideas:

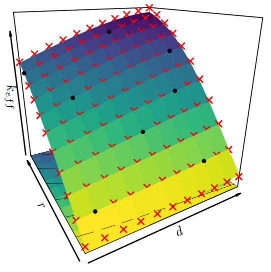
- Multi-fidelity Bayesian optimisation
- Multi-fidelity adaptive design of experiments
- Multi-fidelity transfer learning

Gaussian processes under inequality constraints

Finite-dimensional approximation of GPs

Risk assessment: nuclear safety





O. Roustant
INSA, Toulouse

F. Bachoc
IMT, Toulouse

N. Durrande
Secondmind, UK

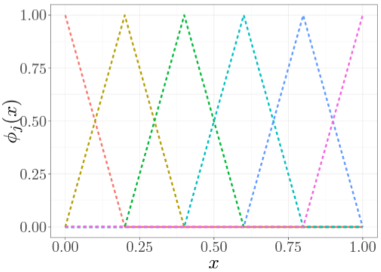
- A. F. López-Lopera, N. Durrande, F. Bachoc and O. Roustant:
Finite-dimensional Gaussian approximation with linear inequality constraints
SIAM/ASA Journal on Uncertainty Quantification, 2018

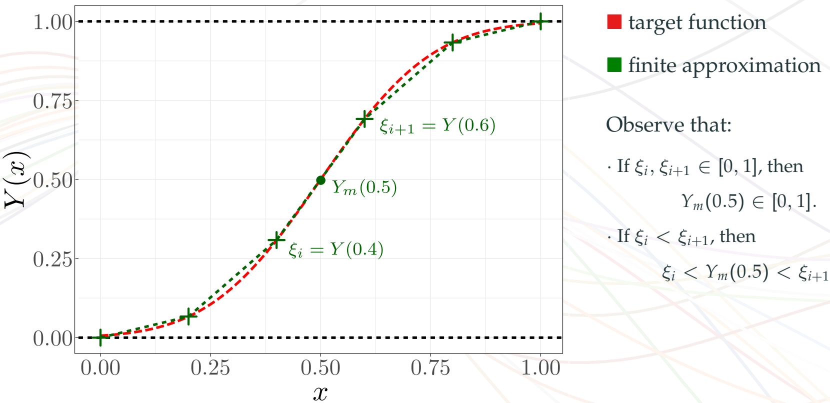
- F. Bachoc, A. Lagnoux and A. F. López-Lopera:
Maximum likelihood estimation for Gaussian processes under inequality constraints
Electronic Journal of Statistics, 2019

The model is based on a finite-dimensional approximation:

$$Y_m(x) = \sum_{j=1}^{m} \phi_j(x) Y(t_j), \text{ s.t. } \begin{cases} Y_m(x_i) = y_i, \text{ for } i = 1, \dots, n & \text{(interpolation conditions)} \\ Y_m \in \mathcal{E} & \text{(inequality constraints)} \end{cases}$$

with $\boldsymbol{\xi} = [Y(t_1), \dots, Y(t_m)] \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma})$, and $\phi_1, \dots, \phi_m$ hat basis functions





Observe that:

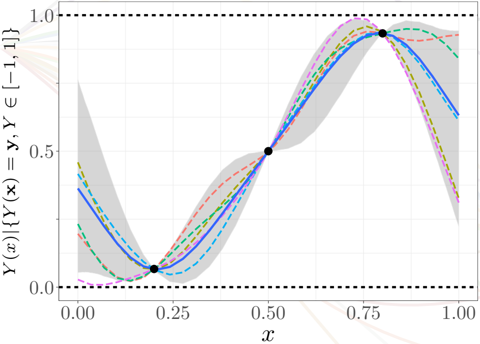
- If $\xi_i, \xi_{i+1} \in [0, 1]$, then
$$Y_m(0.5) \in [0, 1].$$
- If $\xi_i < \xi_{i+1}$, then
$$\xi_i < Y_m(0.5) < \xi_{i+1}.$$

Advantage: It is enough imposing the constraints over the knots

- Assuming that $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma})$, such that $\boldsymbol{l} \leq \boldsymbol{\Lambda}\boldsymbol{\xi} \leq \boldsymbol{u}$, we obtain:
$$\boldsymbol{\xi} \sim \mathcal{TN}(\mathbf{0}, \boldsymbol{\Lambda}\boldsymbol{\Gamma}\boldsymbol{\Lambda}^\top, \boldsymbol{l}, \boldsymbol{u})$$

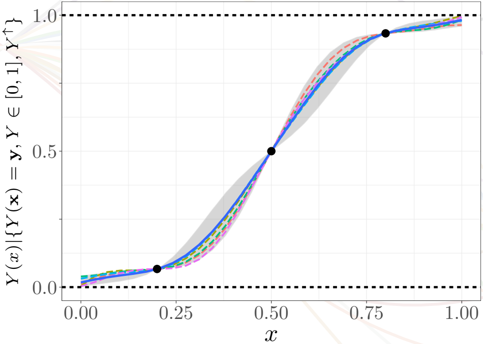
Gaussian processes under inequality constraints: numerical illustration

1D example with **boundedness** constraints via HMC



$$\underbrace{\begin{bmatrix}0\\0\\0\\\vdots\\0\\0\end{bmatrix}}_{\boldsymbol{l}} \leq \underbrace{\begin{bmatrix}1&0&0&\cdots&0&0\\0&1&0&\cdots&0&0\\0&0&1&\cdots&0&0\\\vdots&\vdots&\vdots&\ddots&\vdots&\vdots\\0&0&0&\cdots&1&0\\0&0&0&\cdots&0&1\end{bmatrix}}_{\boldsymbol{\Lambda}} \underbrace{\begin{bmatrix}\xi_1\\\xi_2\\\xi_3\\\vdots\\\xi_{m-1}\\\xi_m\end{bmatrix}}_{\boldsymbol{\xi}} \leq \underbrace{\begin{bmatrix}1\\1\\1\\\vdots\\1\\1\end{bmatrix}}_{\boldsymbol{u}}$$

1D example with **boundedness** & **monotonicity** constraints via HMC



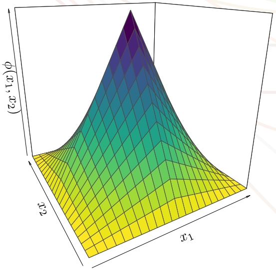
$$\underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ 0 \end{bmatrix}}_{\boldsymbol{l}} \leq \underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \\ 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix}}_{\boldsymbol{\Lambda}} \underbrace{\begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \vdots \\ \xi_{m-1} \\ \xi_m \end{bmatrix}}_{\boldsymbol{\xi}} \leq \underbrace{\begin{bmatrix} 1 \\ \infty \\ \infty \\ \vdots \\ \infty \\ 1 \end{bmatrix}}_{\boldsymbol{u}}$$

- The extension to d dimensions is obtained by **tensorisation**:

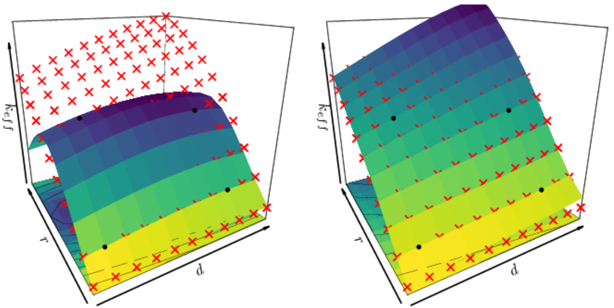
$$Y_m(\boldsymbol{x}) = \sum_{j_1,\ldots,j_d=1}^{m_1,\ldots,m_d} \left[\prod_{p=1,\ldots,d} \phi_{j_p}^{(p)}(x_p) \right] \xi_{j_1,\ldots,j_d}, \text{ s.t. } \begin{cases} Y_m(\boldsymbol{x}_i) + \varepsilon_i = y_i, \\ \boldsymbol{\xi} \in \mathcal{C}, \end{cases} \tag{14}$$

where $\boldsymbol{x}_i \in [0,1]^d$, $y_i \in \mathbb{R}$, $\varepsilon_i \sim \mathcal{N}(0, \tau^2)$, for $i = 1, \ldots, n$; and

- $\boldsymbol{\xi} = [\xi_{1,\ldots,1}, \ldots, \xi_{m_1,\ldots,m_d}]^\top \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Gamma}_\theta)$,
- $\mathcal{C}$ is a convex set of linear inequality constraints, and
- $\phi_{j_i}^{(i)} : [0,1] \mapsto \mathbb{R}$ are hat basis functions.



Nuclear safety application

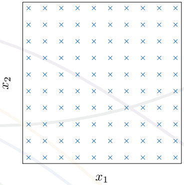


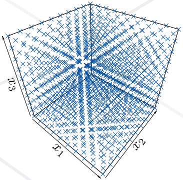
(a) $Q^2 = 0.07696$ (b) $Q^2 = 0.98225$

Nuclear safety application. (left) unconstrained GP and (right) GP under positivity and monotonicity constraints

Curse of dimensionality

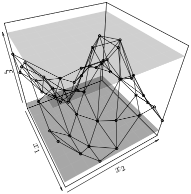
- The cost of Y_m increases as d (or $m = m_1 \times \cdots \times m_d$) increases

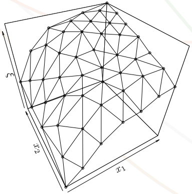




This downside can be partially mitigated in different ways...

- Using a **"smarter" construction of rectangular grids** of knots
- Considering **further assumptions for complexity simplification**:
  - e.g. inactive dimensions, additive conditions
- Using other types of designs of knots: e.g. **Delaunay triangulations**





Maximum likelihood estimation under constraints

· Consider $\{k_{\boldsymbol{\theta}}; \boldsymbol{\theta} \in \boldsymbol{\Theta}\}$, with $\boldsymbol{\Theta} \subset \mathbb{R}^p$, a parametric family of covariance functions where $\boldsymbol{\theta}$ defines the covariance parameters

· The maximum likelihood estimator, with log-likelihood function $\mathcal{L}_n(\boldsymbol{\theta}) = \log p_{\boldsymbol{\theta}}(\mathbf{Y}_n)$, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{MLE}} \in \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \ \mathcal{L}_n(\boldsymbol{\theta}).$$

· We studied the estimation of $\boldsymbol{\theta}$ accounting for inequality constraints:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{cMLE}} \in \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \ \mathcal{L}_{\mathcal{C},m}(\boldsymbol{\theta}),$$

with a conditional log-likelihood function given by:

$$\begin{aligned} \mathcal{L}_{\mathcal{C},m}(\boldsymbol{\theta}) &= \log p_{\boldsymbol{\theta}}(\mathbf{Y}_n | \boldsymbol{\xi} \in \mathcal{C}) \\ &= \log p_{\boldsymbol{\theta}}(\mathbf{Y}_n) + \log P_{\boldsymbol{\theta}}(\boldsymbol{\xi} \in \mathcal{C} | \boldsymbol{\Phi}\boldsymbol{\xi} = \mathbf{Y}_m) - \log P_{\boldsymbol{\theta}}(\boldsymbol{\xi} \in \mathcal{C}) \end{aligned}$$

Asymptotic consistency of the MLE & cMLE

· Let $\mathcal{E}_\kappa$ be one of the following convex set of functions (mild conditions)

$$\mathcal{E}_\kappa = \begin{cases} f : \mathbb{X} \to \mathbb{R}, f \text{ is } C^0 \text{ and } \forall \boldsymbol{x} \in \mathbb{X},\ \ell \leq f(\boldsymbol{x}) \leq u & \text{if } \kappa = 0, \\ f : \mathbb{X} \to \mathbb{R}, f \text{ is } C^1 \text{ and } \forall \boldsymbol{x} \in \mathbb{X},\ \forall i = 1, \cdots, d,\ \frac{\partial}{\partial x_i} f(\boldsymbol{x}) \geq 0 & \text{if } \kappa = 1, \\ f : \mathbb{X} \to \mathbb{R}, f \text{ is } C^2 \text{ and } \forall \boldsymbol{x} \in \mathbb{X},\ \frac{\partial^2}{\partial \boldsymbol{x}^2} f(\boldsymbol{x}) \text{ is a p.s.d. matrix} & \text{if } \kappa = 2. \end{cases}$$

corresponding to boundedness, monotonicity, and convexity constraints.

· Denote: $\boldsymbol{\theta}_0$ (true covariance parameters), $\widehat{\boldsymbol{\theta}}_n$ (MLE), $\widehat{\boldsymbol{\theta}}_{n,c}$ (cMLE).

Proposition (Asymptotic consistency of MLE)

Assume mild conditions. Let the MLE,

$$\widehat{\boldsymbol{\theta}}_n \in \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \mathcal{L}_n(\boldsymbol{\theta}).$$

Assume $\forall \varepsilon > 0$,

$$P(\|\widehat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \geq \varepsilon) \xrightarrow[n \to \infty]{} 0. \quad \textit{(Consistency of the unconditional ML)}$$

Let $\kappa \in \{0, 1, 2\}$*. Since* $P(Y \in \mathcal{E}_\kappa) > 0$*, then*

$$P(\|\widehat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \geq \varepsilon \mid Y \in \mathcal{E}_\kappa) \xrightarrow[n \to \infty]{} 0. \quad \textit{(Consistency of the conditional ML)}$$

Proof.

Accordint to Bayes' theorem, we have

$$P(\|\widehat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \geq \varepsilon |\, Y \in \mathcal{E}_\kappa) = \frac{P(\|\widehat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \geq \varepsilon,\ Y \in \mathcal{E}_\kappa)}{P(Y \in \mathcal{E}_\kappa)} \leq \frac{P(\|\widehat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \geq \varepsilon)}{P(Y \in \mathcal{E}_\kappa)}.$$

Since $P(Y \in \mathcal{E}_\kappa) > 0$, and $P(\|\widehat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \geq \varepsilon) \xrightarrow[n\to\infty]{} 0$, then the result follows.

Asymptotic consistency of the MLE & cMLE

Proposition (Consistency of the MLE and cMLE)

Assume $\forall \varepsilon > 0$ and $\forall M < \infty$, ***(Consistency of the unconditional ML)***

$$P\left(\sup_{\|\boldsymbol{\theta}-\boldsymbol{\theta}_0\| \geq \varepsilon} (\mathcal{L}_n(\boldsymbol{\theta}) - \mathcal{L}_n(\boldsymbol{\theta}_0)) \geq -M\right) \xrightarrow[n \to +\infty]{} 0.$$

Then, ***(Consistency of the conditional cML)***

$$P\left(\sup_{\|\boldsymbol{\theta}-\boldsymbol{\theta}_0\| \geq \varepsilon} (\mathcal{L}_{n,c}(\boldsymbol{\theta}) - \mathcal{L}_{n,c}(\boldsymbol{\theta}_0)) \geq -M \mid Y \in \mathcal{E}_\kappa\right) \xrightarrow[n \to +\infty]{} 0.$$

Consequently, both the MLE and cMLE are consistent estimators:

$$\widehat{\boldsymbol{\theta}}_n \in \operatorname{argmax}_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \mathcal{L}_n(\boldsymbol{\theta}) \xrightarrow[n \to +\infty]{P} \boldsymbol{\theta}_0, \quad \widehat{\boldsymbol{\theta}}_{n,c} \in \operatorname{argmax}_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \mathcal{L}_{n,c}(\boldsymbol{\theta}) \xrightarrow[n \to +\infty]{P \mid Y \in \mathcal{E}_\kappa} \boldsymbol{\theta}_0.$$

Proof.

- **A. F. López-Lopera, N. Durrande, F. Bachoc and O. Roustant:**
  Finite-dimensional Gaussian approximation with linear inequality constraints
  SIAM/ASA Journal on Uncertainty Quantification, 2018

Asymptotic normality of the MLE & cMLE

· For instance, we focus on estimating a single variance parameter σ_0^2, i.e.

$$k_{\sigma_0^2}(x, x') = \sigma_0^2 \, k_1(x, x'),$$

with fixed known correlation function k_1.

Theorem (Asymptotic normality of the MLE and cMLE)

· *Assume mild conditions. Then, the MLE $\widehat{\sigma}_n^2$ of σ_0^2 conditioned on $\{Y \in \mathcal{E}_\kappa\}$ is asymptotically Gaussian distributed. More precisely, for $\kappa = 0, 1, 2$,*

$$\sqrt{n}\left(\widehat{\sigma}_n^2 - \sigma_0^2\right) \xrightarrow[n \to +\infty]{\mathcal{L}|Y \in \mathcal{E}_\kappa} \mathcal{N}(0, 2\sigma_0^4).$$

· *Furthermore, the cMLE $\widehat{\sigma}_{n,c}^2$ of σ_0^2 conditioned on $\{Y \in \mathcal{E}_\kappa\}$ is also asymptotically Gaussian distributed:*

$$\sqrt{n}\left(\widehat{\sigma}_{n,c}^2 - \sigma_0^2\right) \xrightarrow[n \to +\infty]{\mathcal{L}|Y \in \mathcal{E}_\kappa} \mathcal{N}(0, 2\sigma_0^4).$$

· The results can be extended for Matérn models

Proof.

- **F. Bachoc, A. Lagnoux and A. F. López-Lopera:**
  Maximum likelihood estimation for Gaussian processes under inequality constraints, Electronic Journal of Statistics, 2019

Conclusions

- GPs provide a well-founded (Bayesian) non-parametric framework where regularity assumptions can be encoded in the kernel function.
- Various stochastic processes can be seen as GPs, e.g.:
  - The Wiener and Ornstein-Uhlenbeck processes
- Kernel functions can be defined for any measurable space, e.g.:
  - Euclidean space $\mathbb{R}^d$ (temporal, spatial or spatio-temporal spaces)
  - Function space $\mathcal{F}(\mathcal{T}, \mathbb{R})^Q$
- GP can be coupled to:
  - linear differential equations
  - autoregressive models
  - point process modelling
- GPs have been successfully applied in a wide range of applications:
  - biology, physics, engineering, neurosciences, ...

Appendix

Journal papers

[1] A. F. López-Lopera and M. Álvarez. “Switched latent force models for reverse-engineering transcriptional regulation in gene expression data”. In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2019).

[2] A. F. López-Lopera, N. Durrande, and M. Álvarez. “Physically-inspired GP models for post-transcriptional regulation in Drosophila”. In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* (2021).

[3] A. F. López-Lopera, D. Idier, J. Rohmer, and F. Bachoc. “Multioutput GPs with functional data: A study on coastal flood hazard assessment”. In: *Reliability Engineering & System Safety* (2021).

[4] D. Idier, A. Aurouet, F. Bachoc, J. Betancourt, F. Gamboa, S. Leroy, A. F. López-Lopera, Pedreros R., J. Rohmer, and A. Thibault. “A user-oriented local coastal flooding forecast and early warning system using metamodeling techniques”. In: *Journal of Marine Science and Engineering* (2021).

[5] A. F. López-Lopera, F. Bachoc, N. Durrande, and O. Roustant. “Finite-dimensional Gaussian approximation with linear inequality constraints”. In: *SIAM/ASA Journal on Uncertainty Quantification* (2018).

[6] F. Bachoc, A. Lagnoux, and A. F. López-Lopera. “Maximum likelihood estimation for GPs under inequality constraints”. In: *Electronic Journal of Statistics* (2019).

Conference papers

[7] R. Conde, A. F. López-Lopera, S. Mouton, N. Bartoli, and T. Lefebvre. “Multi-fidelity GPs for CFD and wind tunnel data fusion”. In: *AeroBest*. 2021.

[8] A. F. López-lopera, ST John, and N. Durrande. “GP-modulated Cox processes under linear inequality constraints”. In: *AISTATS*. 2019.

[9] A. F. López-Lopera, F. Bachoc, N. Durrande, J. Rohmer, D. Idier, and O. Roustant. “Approximating GP emulators with linear inequality constraints and noisy observations via MC and MCMC”. In: *MCQMC*. 2019.

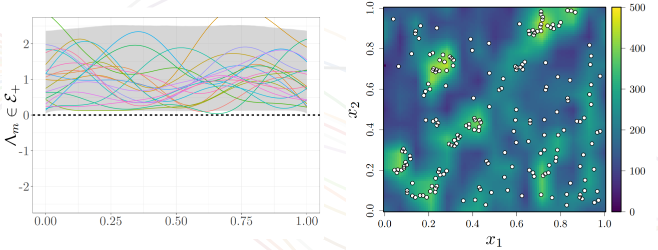
[10] A. F. López-Lopera, M. Álvarez, and A. Orozco. “Sparse linear models applied to power quality disturbance classification”. In: *CIARP*. 2017.

[11] H. Vargas, A. F. López-Lopera, A. Orozco, M. Álvarez, J. Hernández, and N. Malpica. “GPs for slice-based super-resolution MR images”. In: *ISVC*. 2015.

[12] A. F. López-Lopera, M. Álvarez, and A. Orozco. “Improving diffusion tensor estimation using adaptive and optimized filtering based on local similarity”. In: *IbPRIA*. 2015.

Other applications

Géostatistique : répartition spatiale des espèces d'arbres



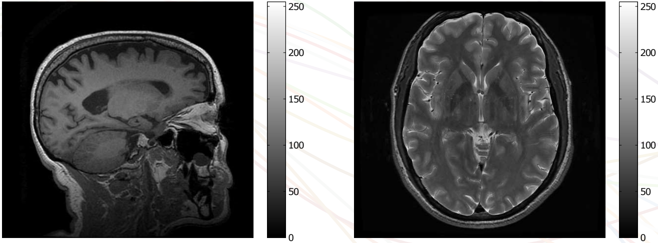
Nous avons considéré les processus de Cox en supposant un *prior* gaussien (sous la contrainte de la positivité) sur la fonction intensité :

$$\Lambda_m(x) = \sum_{j=1}^{m} \phi_j(x)\xi_j \quad \text{tel que} \quad \Lambda_m \in \mathcal{E}_+$$

- **A. F. López-Lopera, S. John and N. Durrande:**
Gaussian process modulated Cox processes under linear inequality constraints
International Conference on Artificial Intelligence and Statistics (AISTATS), 2019

Other applications

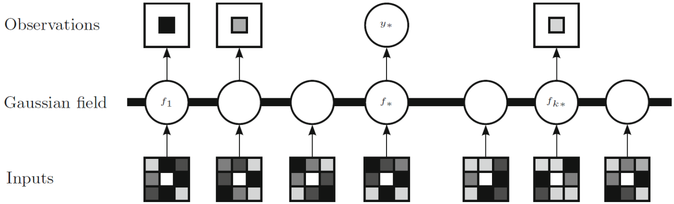
Neurosciences : l'imagerie par résonance magnétique



- **H. Vargas, A. López-Lopera, M.Álvarez, A. Orozco, J. Hernández and N. Malpica:**
Gaussian processes for slice-based super-resolution MR images
Lecture Notes in Computer Science (LNCC), 2015

Other applications

Neurosciences : l'imagerie par résonance magnétique



- H. Vargas, A. López-Lopera, M.Álvarez, A. Orozco, J. Hernández and N. Malpica:
Gaussian processes for slice-based super-resolution MR images
Lecture Notes in Computer Science (LNCC), 2015