Gaussian Processes under Inequality Constraints:

Sequential Construction and Dimension Reduction

F. Bachoc, A. F. López-Lopera and O. Roustant

November 27, 2020

Institut de Mathématiques de Toulouse (IMT), France

Joint work...

F. Bachoc
IMT, France

A.F. López-Lopera
IMT-BRGM, France

O. Roustant
INSA-IMT, France

Sequential construction and dimension reduction of Gaussian processes under inequality constraints

François Bachoc*, Andrés F. López Lopera*,† and Olivier Roustant*
*Institut de Mathématiques de Toulouse, F-31062 Toulouse, France
†BRGM, DRP/R3C, 3 avenue Claude Guillemin, F-45060 Orléans cédex 2, France

September 10, 2020

Abstract

Accounting for inequality constraints, such as boundedness, monotonicity or convexity, is challenging when modeling costly-to-evaluate black box functions. In this regard, finite-dimensional Gaussian process (GP) models bring a valuable solution, as they guarantee that the inequality constraints are satisfied everywhere. Nevertheless, these models are currently restricted to small dimensional situations (up to dimension 5). Addressing this issue, we introduce the MaxMod algorithm that sequentially inserts one-dimensional knots or adds active variables, thereby performing at the same time dimension reduction and efficient knot allocation. We prove the convergence of this algorithm. In intermediary steps of the proof, we propose the notion of multi-affine extension and study its properties. We also prove the convergence of finite-dimensional GPs, when the knots are not dense in the input space, extending the recent literature. With simulated and real data, we demonstrate that the MaxMod algorithm remains efficient in higher dimension (at least in dimension 20), and has a smaller computational complexity than other constrained GP models from the state-of-the-art, to reach a given approximation error.

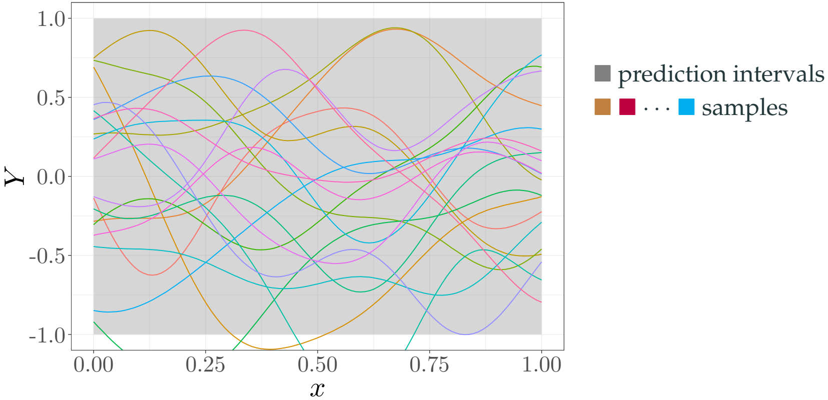
Source: https://arxiv.org/abs/2009.04188

Table of contents

Motivation

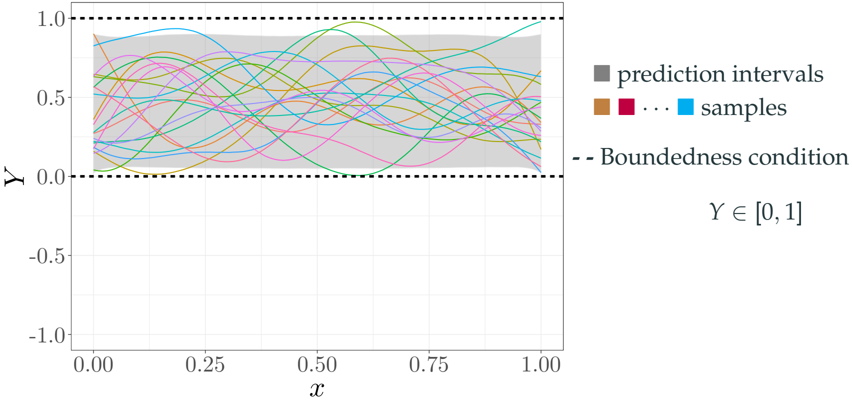
Motivation: Gaussian processes (GPs) under inequality constrains

GPs form a flexible **prior over functions** [Rasmussen and Williams, 2005]:



Motivation: Gaussian processes (GPs) under inequality constrains

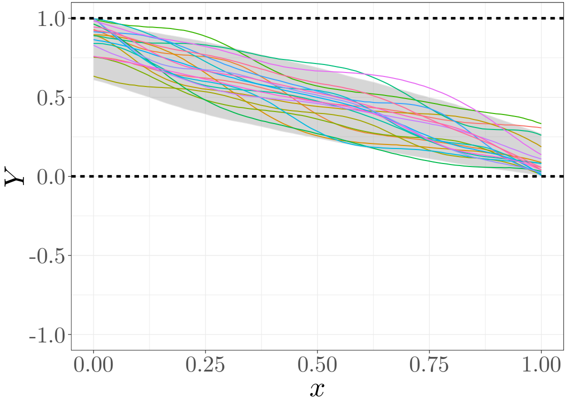
GPs form a flexible **prior over functions** [Rasmussen and Williams, 2005]:



$Y \in [0, 1]$

Motivation: Gaussian processes (GPs) under inequality constrains

GPs form a flexible **prior over functions** [Rasmussen and Williams, 2005]:



■ prediction intervals

■ ■ · · · ■ samples

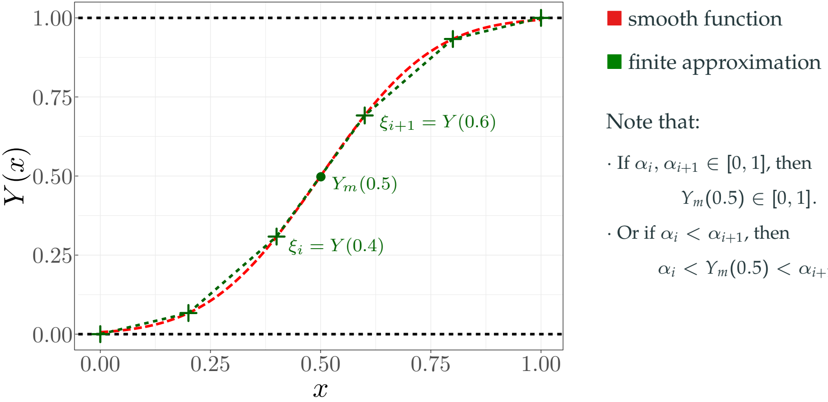
- - Boundedness condition

$$Y \in [0, 1]$$

↓ Monotonicity condition

$$Y(x') \leq Y(x), \ \ \forall \, x' \geq x.$$

Motivation: finite-dimensional approximation



Note that:

- If $\alpha_i, \alpha_{i+1} \in [0, 1]$, then

$$Y_m(0.5) \in [0, 1].$$

- Or if $\alpha_i < \alpha_{i+1}$, then

$$\alpha_i < Y_m(0.5) < \alpha_{i+1}.$$

Pro: imposing constraints over knots is enough [Maatouk and Bay, 2017]

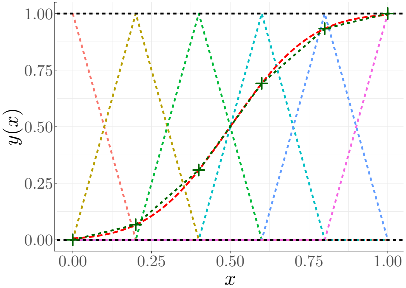
Motivation: finite-dimensional approximation

· Let the (constrained) finite-dimensional GP Y_m be defined as

$$Y_m(x) = \sum_{j=1}^{m} \alpha_j \phi_j(x), \text{ s.t. } \begin{cases} Y_m(x_i) + \varepsilon_i = y_i & \text{(regression conditions)}, \\ l \leq \Lambda \alpha \leq u & \text{(linear inequality conditions)}, \end{cases} \quad (1)$$

where $x_i \in [0, 1]$, $y_i \in \mathbb{R}$ for $i = 1, \ldots, n$; and

- $\alpha = [\alpha_1, \ldots, \alpha_m]^\top \sim \mathcal{N}(0, \Gamma_\theta)$ with covariance matrix Γ_θ,
- (Λ, l, u) define the inequality conditions,
- $\varepsilon_i \sim \mathcal{N}(0, \tau^2)$ with noise variance τ^2, and
- $\phi_j : [0, 1] \mapsto \mathbb{R}$ are (asymmetric) hat basis functions:



Motivation: finite-dimensional approximation

· Then, *uncertainty quantification* relies on simulating the **truncated vector** α [López-Lopera et al., 2018]:

$$\Lambda\alpha | \{\Phi\alpha + \varepsilon = y, l \leq \Lambda\alpha \leq u\} \sim \mathcal{TN}(\Lambda\mu, \Lambda\Sigma\Lambda^\top, l, u), \tag{2}$$

with conditional parameters μ and Σ given by

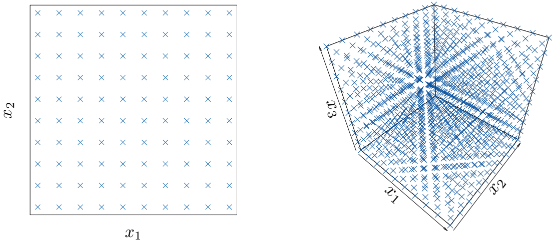
$$K = \Phi\Gamma\Phi^\top + \tau^2 I, \quad \mu = \Gamma\Phi^\top K^{-1}\boldsymbol{y}, \quad \Sigma = \Gamma - \Gamma\Phi^\top K^{-1}\Phi\Gamma. \tag{3}$$

∗ Eq. (2) is computed via *Monte Carlo* (MC) or *Markov Chain MC* (MCMC):

- e.g. *Hamiltonian Monte Carlo* (HMC) [Pakman and Paninski, 2014]

- **Con:** the cost of Y_m increases as d increases.

$$Y_m(x) = \sum_{j_1,\ldots,j_d=1}^{m_1,\ldots,m_d} \left[\prod_{p=1,\ldots,d} \phi_{j_p}^{(p)}(x_p) \right] \alpha_{j_1,\ldots,j_d}, \quad \text{s.t.} \quad \begin{cases} Y_m(x_i) + \varepsilon_i = y_i, \\ \alpha \in \mathcal{C}. \end{cases} \tag{4}$$



- This drawback can be mitigated by considering:
  - a *“smarter” construction of rectangular grids* of knots
  - and/or *further assumptions for complexity simplification*
    → e.g. *inactive variables*

The MaxMod algorithm

The maximum a posteriori (mode) function in 1D

· Let $\widehat{\alpha}$ be the mode that maximises the pdf of $\alpha | \{\Phi\alpha + \varepsilon = y, l \leq \Lambda\alpha \leq u\}$:

$$\widehat{\alpha} = \underset{\alpha \text{ s.t. } l \leq \Lambda\alpha \leq u}{\arg\max} \{-[\alpha - \mu]^\top \Sigma^{-1} [\alpha - \mu]\}, \tag{5}$$

with $\widehat{\alpha} = [\widehat{\alpha}_1, \ldots, \widehat{\alpha}_m]^\top$.

· The *MAP estimate* of Y_m is given by

$$\widehat{Y}_m(x) = \sum_{j=1}^{m} \widehat{\alpha}_j \phi_j(x). \tag{6}$$

INSTITUT DE MATHÉMATIQUES DE TOULOUSE

The maximum a posteriori (mode) function in 1D

· Let $\widehat{\alpha}$ be the mode that maximises the pdf of $\alpha | \{\Phi\alpha + \varepsilon = y, l \leq \Lambda\alpha \leq u\}$:

$$\widehat{\alpha} = \underset{\alpha \text{ s.t. } l \leq \Lambda\alpha \leq u}{\arg\max} \{-[\alpha - \mu]^\top \Sigma^{-1} [\alpha - \mu]\}, \quad (5)$$

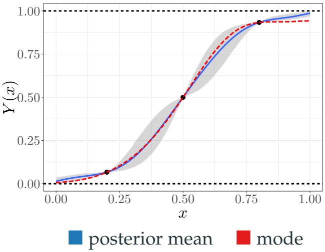
with $\widehat{\alpha} = [\widehat{\alpha}_1, \ldots, \widehat{\alpha}_m]^\top$.

· The *MAP estimate* of Y_m is given by

$$\widehat{Y}_m(x) = \sum_{j=1}^{m} \widehat{\alpha}_j \phi_j(x). \quad (6)$$

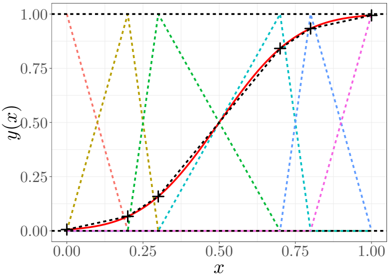
Pro:

- $\widehat{Y}_m$ can be used as a point estimate
- Easy and fast calculations
- Convergence to the spline solution as $m \to \infty$ [Bay et al., 2016]
- Starting point for MCMC



Asymmetric hat basis functions

· In practice, we modify the construction of the hat basis functions ϕ:



Pros:

- This construction allows the free location of the knots
- Constrained GP model's properties are preserved [see Bachoc et al., 2020, López-Lopera, 2018]

- Let $\widehat{Y}_S$ be the MAP function with an ordered set of knots:

$$S = \{t_0, \ldots, t_m\}, \quad \text{with} \quad 0 = t_0 < \cdots < t_m = 1.$$

- Here, we aim at adding a new knot t in S (where?)

The MaxMod algorithm in 1D

- Let $\widehat{Y}_S$ be the MAP function with an ordered set of knots:

$$S = \{t_0, \ldots, t_m\}, \quad \text{with} \quad 0 = t_0 < \cdots < t_m = 1.$$

- Here, we aim at adding a new knot t in S (where?)
- To do so, we aim at *maximising the total modification of the MAP*:

$$I_S(t) = \int_{[0,1]} \left(\widehat{Y}_{S \cup t}(x) - \widehat{Y}_S(x) \right)^2 dx. \tag{7}$$

- The integral in (7) has a closed-form expression.

- Let $\widehat{Y}_S$ be the MAP function with an ordered set of knots:

$$S = \{t_0, \ldots, t_m\}, \quad \text{with} \quad 0 = t_0 < \cdots < t_m = 1.$$

- Here, we aim at adding a new knot t in S (where?)
- To do so, we aim at *maximising the total modification of the MAP*:

$$I_S(t) = \int_{[0,1]} \left(\widehat{Y}_{S \cup t}(x) - \widehat{Y}_S(x) \right)^2 dx. \tag{7}$$

- The integral in (7) has a closed-form expression.

Algorithm MaxMod (maximum modification of the MAP) in 1D

Input parameters: the initial subdivision $S^{(0)} \in \mathcal{S}$.

Sequential procedure: for $\kappa \in \mathbb{N}$, do:

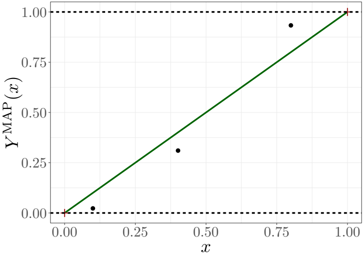
1: Set $t^\star_{\kappa+1} \in [0, 1]$ such that

$$I_{S^{(\kappa)}}(t^\star_{\kappa+1}) \geq \sup_{t \in [0,1]} I_{S^{(\kappa)}}(t)$$

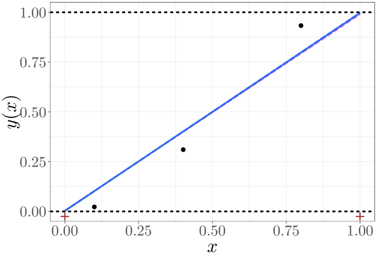
2: $S^{(\kappa+1)} = S^{(\kappa)} \ \cup \ t^\star_{\kappa+1}$.

1D example under boundedness and monotonicity constraints

MAP estimate



conditional sample-path



● training points + knots ■ MAP estimate

■ predictive mean ■ 90% confidence intervals

· Let $\widehat{Y}_{\mathcal{J},S}$ be the MAP function with $|\mathcal{J}|$ active variables and ordered sets of knots $S_{\mathcal{J}}$ for $\mathcal{J} \subseteq \{1, \ldots, D\}$.

· Then, the criterion to maximise is given by

$$I_{\mathcal{J},S}(i,t) = \begin{cases} \frac{1}{N_{S,\mathcal{J},i}} \int_{[0,1]^d} \left(\widehat{Y}_{\mathcal{J},\, S \cup_i t}(x) - \widehat{Y}_{\mathcal{J},\, S}(x)\right)^2 dx & \text{if } i \in \mathcal{J}, \\ \frac{1}{N_{S,\mathcal{J},i}} \int_{[0,1]^{d+1}} \left(\widehat{Y}_{\mathcal{J}\cup\{i\},\, S+i}(x) - \widehat{Y}_{\mathcal{J},\, S}(x)\right)^2 dx & \text{if } i \notin \mathcal{J}, \end{cases} \tag{8}$$

where $N_{S,\mathcal{J},i}$ is the increase of the number of basis functions.

· Let $\widehat{Y}_{\mathcal{J},S}$ be the MAP function with $|\mathcal{J}|$ active variables and ordered sets of knots $S_{\mathcal{J}}$ for $\mathcal{J} \subseteq \{1, \dots, D\}$.

· Then, the criterion to maximise is given by

$$I_{\mathcal{J},S}(i, t) = \begin{cases} \frac{1}{N_{S,\mathcal{J},i}} \int_{[0,1]^d} \left(\widehat{Y}_{\mathcal{J},S \cup_i t}(x) - \widehat{Y}_{\mathcal{J},S}(x)\right)^2 dx & \text{if } i \in \mathcal{J}, \\ \frac{1}{N_{S,\mathcal{J},i}} \int_{[0,1]^{d+1}} \left(\widehat{Y}_{\mathcal{J}\cup\{i\},S+i}(x) - \widehat{Y}_{\mathcal{J},S}(x)\right)^2 dx & \text{if } i \notin \mathcal{J}, \end{cases} \tag{8}$$

where $N_{S,\mathcal{J},i}$ is the increase of the number of basis functions.

Algorithm MaxMod in dimension d

Input parameters: the initial set of active variables $\mathcal{J}_0 \subseteq \{1, \dots, D\}$ and the initial subdivision $S^{(0)} \in \mathcal{S}_{\mathcal{J}_0}$.

Sequential procedure: for $\kappa \in \mathbb{N}$, do:

1: Set $i^\star_{\kappa+1} \in \{1, \dots, D\}$, $t^\star_{\kappa+1} \in [0, 1]$ such that

$$I_{\mathcal{J}_\kappa,S^{(\kappa)}}(i^\star_{\kappa+1}, t^\star_{\kappa+1}) \geq \sup_{\substack{i\in\{1,\dots,D\},\\ t\in[0,1]}} I_{\mathcal{J}_\kappa,S^{(\kappa)}}(i, t)$$

2: **if** $i^\star_{\kappa+1} \in \mathcal{J}_\kappa$ **then** $\mathcal{J}_{\kappa+1} = \mathcal{J}_\kappa$ and $S^{(\kappa+1)} = S^{(\kappa)} \cup_{i^\star_{\kappa+1}} t^\star_{\kappa+1}$.

3: **else** $\mathcal{J}_{\kappa+1} = \mathcal{J}_\kappa \cup \{i^\star_{\kappa+1}\}$ and $S^{(\kappa+1)} = S^{(\kappa)} + i^\star_{\kappa+1}$.

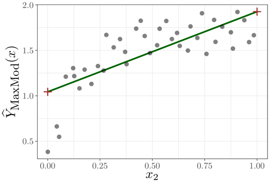
The MaxMod algorithm in higher dimensions

2D example under monotonicity constraints

Evolution of the MaxMod algorithm using $f(x) = \frac{1}{2}x_1 + \mathsf{arctan}(10x_2)$

2D example under monotonicity constraints

Evolution of the MaxMod algorithm using $f(x) = \frac{1}{2}x_1 + \mathrm{arctan}(10x_2)$

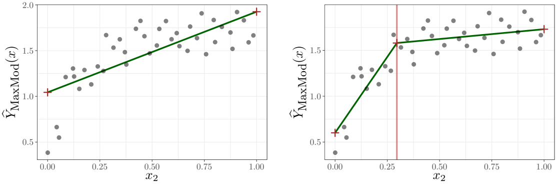


(a) iteration 0

• training points + knots ■ MAP estimate

2D example under monotonicity constraints

Evolution of the MaxMod algorithm using $f(x) = \frac{1}{2}x_1 + \text{arctan}(10x_2)$

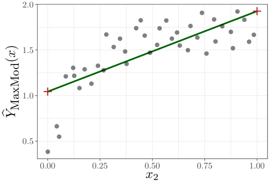


(a) iteration 0 (b) iteration 1

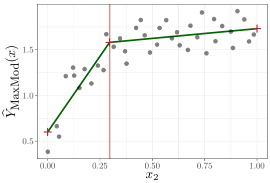
● training points + knots ■ MAP estimate

2D example under monotonicity constraints

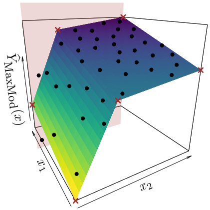
Evolution of the MaxMod algorithm using $f(x) = \frac{1}{2}x_1 + \text{arctan}(10x_2)$



(a) iteration 0



(b) iteration 1

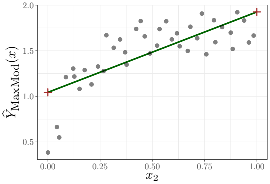


(c) iteration 2

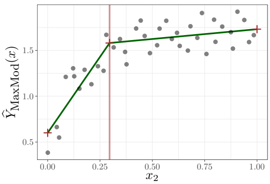
● training points + knots ■ MAP estimate

2D example under monotonicity constraints

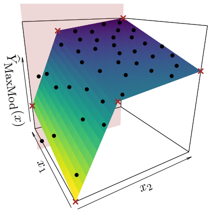
Evolution of the MaxMod algorithm using $f(x) = \frac{1}{2}x_1 + \arctan(10x_2)$



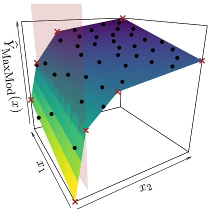
(a) iteration 0



(b) iteration 1



(c) iteration 2



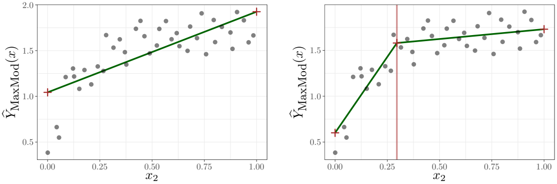
(d) iteration 3

● training points + knots ■ MAP estimate

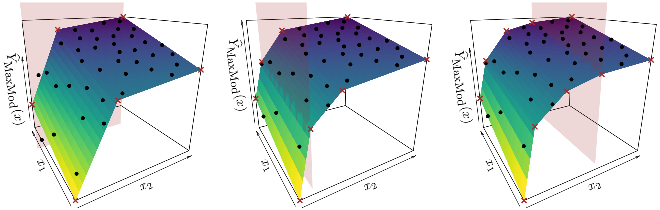
The MaxMod algorithm in higher dimensions

2D example under monotonicity constraints

Evolution of the MaxMod algorithm using $f(x) = \frac{1}{2}x_1 + \arctan(10x_2)$



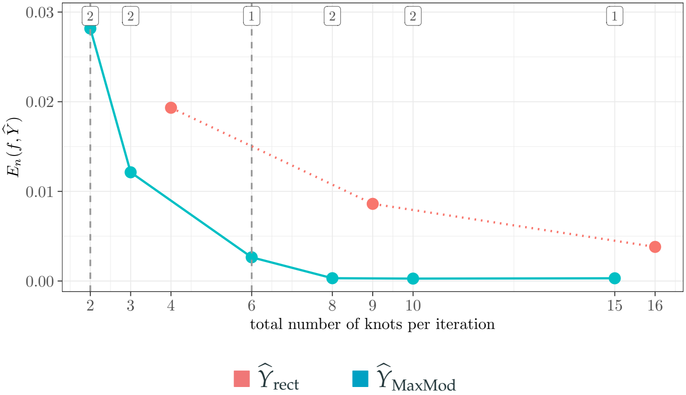
(a) iteration 0 (b) iteration 1



(c) iteration 2 (d) iteration 3 (e) iteration 4

● training points + knots ■ MAP estimate

The MaxMod algorithm: example in 2D



Evolution of the (normalized) bending energy E_n:

$$E_n(f, \widehat{Y}) = \frac{\int_{[0,1]^D} (f(x) - \widehat{Y}(x))^2 dx}{\int_{[0,1]^D} f^2(x) dx} \tag{9}$$

The MaxMod algorithm in higher dimensions

· In practice, we introduce a reward:

$$R_{\mathcal{J},S}(i,t) = \begin{cases} \Delta d(t, S_i) & \text{if } i \in \mathcal{J} \\ \Delta' & \text{if } i \notin \mathcal{J} \end{cases} \tag{10}$$

- Δ' is the reward for adding a new active variable.
- $\Delta d(t, S_i)$ is the reward for inserting a knot in an existing dimension times the distance to the closest existing knot.

- In practice, we introduce a reward:

$$R_{\mathcal{J},S}(i,t) = \begin{cases} \Delta d(t, S_i) & \text{if } i \in \mathcal{J} \\ \Delta' & \text{if } i \notin \mathcal{J} \end{cases} \tag{10}$$

  - Δ' is the reward for adding a new active variable.
  - $\Delta d(t, S_i)$ is the reward for inserting a knot in an existing dimension times the distance to the closest existing knot.

Algorithm Modified MaxMod in dimension d

Input parameters: $\Delta > 0$, $\Delta' > 0$, the initial set of active variables $\mathcal{J}_0 \subseteq \{1, \ldots, D\}$ and the initial subdivision $S^{(0)} \in \mathcal{S}_{\mathcal{J}_0}$.

Sequential procedure: For $\kappa \in \mathbb{N}$, do:

1: Set $i^\star_{\kappa+1} \in \{1, \ldots, D\}$, $t^\star_{\kappa+1} \in [0, 1]$ such that

$$I_{\mathcal{J}_\kappa, S^{(\kappa)}}(i^\star_{\kappa+1}, t^\star_{\kappa+1}) + R_{\mathcal{J}_\kappa, S^{(\kappa)}}(i^\star_{\kappa+1}, t^\star_{\kappa+1}) \geq \sup_{\substack{i \in \{1,\ldots,D\}, \\ t \in [0,1]}} \left(I_{\mathcal{J}_\kappa, S^{(\kappa)}}(i, t) + R_{\mathcal{J}_\kappa, S^{(\kappa)}}(i, t) \right).$$

2: **if** $i^\star_{\kappa+1} \in \mathcal{J}_\kappa$ **then** $\mathcal{J}_{\kappa+1} = \mathcal{J}_\kappa$ and $S^{(\kappa+1)} = S^{(\kappa)} \cup_{i^\star_{\kappa+1}} t^\star_{\kappa+1}$.

3: **else** $\mathcal{J}_{\kappa+1} = \mathcal{J}_\kappa \cup \{i^\star_{\kappa+1}\}$ and $S^{(\kappa+1)} = S^{(\kappa)} + i^\star_{\kappa+1}$.

Dimension reduction illustration under monotonicity constraints

· Here, we consider the target function:

$$f(x) = \sum_{i=1}^{d} \operatorname{arctan}\left(5\left[1 - \frac{i}{d+1}\right]x_i\right), \quad \text{with} \quad x \in [0,1]^d.$$

· In addition to $(x_1, \ldots, x_d)$, we include $D - d$ fake variables $(x_{d+1}, \ldots, x_D)$.

· We test the MaxMod for $D \in \{5, 10, 15, 20\}$, $d \in \{2, 3, 4, 5\}$.

· For each value of D, we evaluate f at a maximin LHD with $n = 10 \times D$.

· As a stopping rule, we check that:

$$I_{\mathcal{J}_\kappa, S^{(\kappa)}}(i,t) + R_{\mathcal{J}_\kappa, S^{(\kappa)}}(i,t) \leq 5 \times 10^{-5}, \quad \text{with} \quad \kappa \in \mathbb{N}.$$

· We fix $\Delta = \Delta' = 1 \times 10^{-9}$.

INSTITUT DE MATHÉMATIQUES DE TOULOUSE

Performance of the MaxMod algorithm for $D \in \{5, 10, 15, 20\}$ and $d \in \{2, 3, 4, 5\}$.

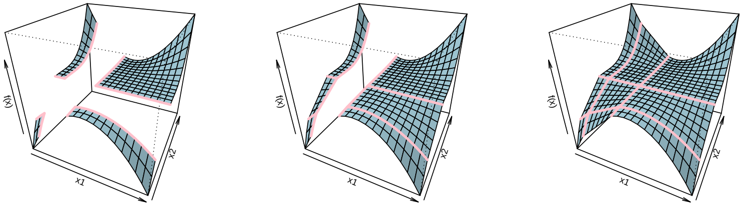
D	d	active dimensions	knots per dimension	$E_n(f, \widehat{Y})$
5	2	(1, 2)	(7, 4)	4.51×10^{-5}
	3	(1 ,2, 3)	(6, 6, 4)	4.09×10^{-4}
	4	(1, 2, 3, 4)	(4, 4, 3, 2)	9.05×10^{-4}
	5	(1, 2, 3, 4, 5)	(3, 4, 4, 3, 2)	1.19×10^{-3}
10	2	(1, 2)	(5, 3)	2.78×10^{-5}
	3	(1, 2, 3)	(5, 4, 3)	1.79×10^{-3}
	4	(1, 2, 3, 4)	(5, 3, 3, 2)	2.89×10^{-4}
	5	(1 ,2, 3, 4, 5)	(3, 4, 3, 3, 2)	4.31×10^{-4}
15	2	(1, 2)	(4, 3)	1.85×10^{-4}
	3	(1, 2, 3)	(4, 3, 3)	1.94×10^{-4}
	4	(1, 2, 3, 4)	(3, 3, 3, 2)	1.94×10^{-4}
	5	(1, 2, 3, 4, 5)	(3, 3, 3, 3, 2)	9.29×10^{-5}
20	2	(1, 2)	(5, 3)	9.37×10^{-5}
	3	(1, 2, 3)	(4, 4, 3)	1.40×10^{-4}
	4	(1, 2, 3, 4)	(4, 3, 3, 3)	1.97×10^{-4}
	5	(1, 2, 3, 4, 5)	(3, 3, 3, 3, 2)	2.83×10^{-4}

The multiaffine extension of a multivariate function

The multiaffine extension of a multivariate function

Definition

Let $F_1, \dots, F_d$ (general) closed subset of $[0, 1]$ containing 0 and 1.
Let f be a continuous function on $F = F_1 \times \dots \times F_d$.
Then, there exists a *unique continuous extension of f on $[0,1]^d$ such that any 1D marginal cut functions $u_i \mapsto f(u_i, t_{\sim i})$ is affine* on intervals of $[0, 1] \setminus F_i$.
Denoted $P_{F \to [0,1]^d}(f)$, it is obtained by sequential 1D affine interpolations.



Sequential construction of the multiaffine extension (2D case)

Properties

- The multiaffine extension is expressed with 2^d neighbours as

$$P_{F\to[0,1]^d}(f)(t) = \sum_{\epsilon_1,\dots,\epsilon_d\in\{-,+\}} \left(\prod_{j=1}^{d} \omega_{\epsilon_j}(t_j)\right) f(t_1^{\epsilon_1},\dots,t_d^{\epsilon_d}),$$

  where t_j^-, t_j^+ are the closest left and right neighbours of t_j in F_j,
  $\omega_+(t_j) = \frac{t_j - t_j^-}{t_j^+ - t_j^-}$ if $t_j \notin F_j$ and $\frac{1}{2}$ otherwise, and $\omega_-(t_j) = 1 - \omega_+(t_j)$.
- It preserves monotonicity and componentwise convexity.

Probabilistic interpretation

Consider the Brownian sheet, with kernel $k(x, x') = \min(x_1, x_1') \dots \min(x_d, x_d')$. Up to originating the Brownian sheet in the orthant $(\mathbb{R}^-)^d$, the multiaffine extension coincides on $[0, 1]^d \setminus F$ with the conditional expectation of the Brownian sheet on ∂F.

Indeed, if (Y_x) is the Brownian sheet, 1D sections are prop. to the Brownian:

$$k_{|\{x_2,\dots,x_d\}}(x, x') = x_2 \dots x_d \min(x_1, x_1')$$

Hence for $x_2 > 0, \dots, x_d > 0$, conditioning on values at x_1 gives an affine interpolator (mean of a Brownian bridge).

Technical issue: the Brownian sheet is *conditioned on a continuum*.
That probabilistic point of view has not been used in the proofs.

Convergence

Setting:

- Noise-free case from now on.
- Fixed data set from now on.
- $\mathcal{I}$: set of functions interpolating the data set.
- Fixed input dimension $d \in \mathbb{N}$ from now on.
- For variable $j \in \{1, \ldots, d\}$: sequence of one-dimensional knots $t_1^{(j)}, \ldots, t_{m_j}^{(j)}$ and $m_j \to \infty$. The sequence is dense in $[0, 1]$.
- The MAP estimate $\widehat{Y}_{m_1,\ldots,m_d} : [0, 1]^D \to \mathbb{R}$.
- Kernel k with corresponding RKHS $\mathcal{H}$ of functions from $[0, 1]^d$ to $\mathbb{R}$.
- Inequality set $\mathcal{C}$ of functions from $[0, 1]^d$ to $\mathbb{R}$.

Theorem (Bay, Grammont, Maatouk) Bay et al. [2016, 2017]

Under some technical conditions

$$\widehat{Y}_{m_1,\ldots,m_d} \to Y_{\text{opt}},$$

uniformly on $[0, 1]^d$, with

$$Y_{\text{opt}} = \underset{f \in \mathcal{H} \cap \mathcal{C} \cap \mathcal{I}}{\operatorname{argmin}} ||f||_{\mathcal{H}}.$$

Setting:

- For variable $j \in \{1, \ldots, d\}$: sequence of one-dimensional knots $t_1^{(j)}, \ldots, t_{m_j}^{(j)}$ and $m_j \to \infty$. The sequence has closure $F_j \subset [0,1]$.

First approach: can we still find a limit function from $[0,1]^d$ to $\mathbb{R}$?

$\longrightarrow$ Not successful to stay on $[0,1]^d$ here.

Instead: Work on $F := F_1 \times \ldots \times F_d$ and define

- $\mathcal{H}_F$ RKHS of k restricted to $F \times F$.
- $\mathcal{C}_F$: set of functions from F to $\mathbb{R}$ which multi-affine extensions satisfy inequality constraints.
- $\mathcal{I}_F$: set of functions from F to $\mathbb{R}$ which multi-affine extensions interpolate the data set.

When the sequence of knots is not dense

Theorem (Bachoc, López-Lopera, Roustant 2020)

Under some technical conditions

$$\widehat{Y}_{m_1,\ldots,m_d} \to Y_{\mathrm{opt,F}},$$

uniformly on F, with

$$Y_{\mathrm{opt,F}} = \underset{f \in \mathcal{H}_F \cap \mathcal{C}_F \cap \mathcal{I}_F}{\operatorname{argmin}} ||f||_{\mathcal{H}_F}.$$

As a consequence

$$\widehat{Y}_{m_1,\ldots,m_d} \to P_{F \to [0,1]^d}\left(Y_{\mathrm{opt,F}}\right),$$

uniformly on $[0,1]^d$.

- MAP $\widehat{Y}_{\text{MaxMod},m}$ at iteration m of MaxMod.

Theorem (Bachoc, López-Lopera, Roustant 2020)

Under some technical conditions, as $m \to \infty$,

$$\widehat{Y}_{\text{MaxMod},m} \to Y_{\text{opt}},$$

uniformly on $[0,1]^d$, with

$$Y_{\text{opt}} = \underset{f \in \mathcal{H} \cap \mathcal{C} \cap \mathcal{I}}{\text{argmin}} ||f||_{\mathcal{H}}.$$

INSTITUT DE MATHÉMATIQUES DE TOULOUSE

Proof arguments:

- Previous theorem $\longrightarrow$ let us show that sequence of knots is dense.
- As is common for algorithms maximizing acquisition functions (EGO,...), two ingredients:
  - $\rightarrow$ Show that acquisition function is small at points close to existing ones.
  - $\rightarrow$ Show that acquisition function is large at points away from existing ones.
- Here:
  - $\rightarrow$ Show that mode perturbation vanishes from $\widehat{Y}_{\text{MaxMod},m}$ to $\widehat{Y}_{\text{MaxMod},m+1}$ $\longrightarrow$ previous convergence result.
  - $\rightarrow$ Acquisition function is large at points away from existing ones $\longrightarrow$ the exploration reward.

Conclusions

Conclusions

- We introduced the MaxMod algorithm, that sequentially inserts knots or adds active variables to a constrained GP model.
- We showed numerically that it is tractable and accurate (at least in $D = 20$):
  - It typically needs less knots $\rightarrow$ smaller complexity.
  - It can be applied to high dimensions with moderate effective ones.
- A proof of convergence guarantees that MaxMod globally converges to an optimal infinite dimensional model:
  - The proof tackles the case where knots are not dense in the input domain.
  - The notion of a multi-affine extension is constructed.

References

F. Bachoc, A. F. López-Lopera, and O. Roustant. Sequential construction and dimension reduction of Gaussian processes under inequality constraints. *arXiv*, 2020.

X. Bay, L. Grammont, and H. Maatouk. Generalization of the Kimeldorf-Wahba correspondence for constrained interpolation. *Electronic Journal of Statistics*, 2016.

X. Bay, L. Grammont, and H. Maatouk. A new method for interpolating in a convex subset of a Hilbert space. *Computational Optimization and Applications*, 68(1):95–120, 2017.

A. F. López-Lopera. *Gaussian Process Modelling under Inequality Constraints*. PhD thesis, Mines Saint-Étienne, 2018.

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

H. Maatouk and X. Bay. Gaussian process emulators for computer experiments with inequality constraints. *Mathematical Geosciences*, 2017.

A. Pakman and L. Paninski. Exact Hamiltonian Monte Carlo for truncated multivariate Gaussians. *Journal of Computational and Graphical Statistics*, 2014.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, MA, 2005.