ESAIM: PROCEEDINGS AND SURVEYS, 2018, Vol. 60, p. 163-179
Jean-François Coeurjolly & Adeline Leclercq-Samson

GAUSSIAN PROCESSES FOR COMPUTER EXPERIMENTS

François Bachoc[1], Emile Contal[2], Hassan Maatouk[3] and Didier Rullière[4]

Abstract. This paper collects the contributions which were presented during the session devoted to Gaussian processes at the Journées MAS 2016. First, an introduction to Gaussian processes is provided, and some current research questions are discussed. Then, an application of Gaussian process modeling under linear inequality constraints to financial data is presented. Also, an original procedure for handling large data sets is described. Finally, the case of Gaussian process based iterative optimization is discussed.

Résumé. Cet article réunit les contributions qui ont étés présentées lors de la session des Journées MAS 2016 dédiée aux processus gaussiens. Tout d'abord, une introduction aux processus gaussiens est donnée, et certaines questions de recherche actuelles sont discutées. Ensuite, une application d'un modèle de processus gaussien sous contraintes de type inégalité, à des données financières, est présentée. Puis, une procédure originale, pour gérer des volumes de données importants, est présentée. Enfin, le cas de l'optimisation séquentielle par processus gaussiens est discuté.

Introduction

Gaussian process models [46, 56], also called Kriging, consist in inferring the values of a realization of a Gaussian random field given (possibly noisy) observations at a finite set of observation points. They have become a popular method for a large range of applications, such as geostatistics [39], numerical code approximation [7, 50, 51] and calibration [8, 45] or global optimization [29].

In many application cases, a deterministic function is under consideration, that one treats as a realization from a Gaussian process. Thus, Gaussian processes can be interpreted as a Bayesian prior over unknown functions. In many applications, the function values are outputs from a computer model, and the function inputs are the corresponding simulation parameters. In these situations, the terminology "Gaussian processes for computer experiments" is largely employed.

Considering Gaussian processes, in comparison to other random field models, as a Bayesian prior for a deterministic function is often beneficial in terms of conceptual and computational simplicity. Indeed, Gaussian processes are always well defined whenever a covariance function can be defined, which is the case for many input spaces, either Euclidean spaces or more general spaces, like function spaces [41]. In addition, the full conditional distribution of a Gaussian process, given the observations, is Gaussian, explicit, and can be sampled relatively conveniently, see Section 1. As a consequence, Gaussian processes now constitute a popular and widely studied methodology for the analysis of computer experiments.

[1] Institut de Mathématiques de Toulouse, Université Paul Sabatier
[2] Centre de Mathématique et de Leurs Applications, ENS Cachan
[3] INRIA Centre de Recherche Rennes-Bretagne Atlantique
[4] ISFA, laboratoire SAF, EA2429, Université Lyon 1

Article published online by EDP Sciences and available at http://www.esaim-proc.org or https://doi.org/10.1051/proc/201760163

In this paper, we review the basic foundations of Gaussian processes, briefly discuss the current areas of research, and present three contributions that were presented at the Journées MAS 2016. The rest of the paper is organized as follows. In Section 1, we provide the general review on Gaussian processes and mention some research areas. In Section 2, we present novel contributions on constrained Gaussian processes. In Section 3 we present some new kriging predictors, suited to large datasets. In Section 4, we address Gaussian-process based stochastic optimization.

The contributions which are presented in Sections 2, 3 and 4 correspond to the references [14, 35], [49] and [12, 13].

1. Gaussian process regression: framework and current research questions

1.1. Distribution for a Gaussian process

Throughout this paper, D denotes a set, which we call the input space of a Gaussian process. Next, we give the definition of a Gaussian process.

Definition 1. *A stochastic process Y on D is a Gaussian process if, for any $n \in \mathbb{N}$, for any $x_1, ..., x_n \in D$, the random vector $(Y(x_1), ..., Y(x_n))$ is a Gaussian vector.*

Since Gaussian vectors are characterized by their mean vector and covariance matrices, it is natural to extend these objects for Gaussian processes. This is the object of the next definition (which actually holds more generally for stochastic processes).

Definition 2. *Let Y be a stochastic process on D such that $\mathrm{E}\left[Y(x)^2\right] < +\infty$ for all $x \in D$. Then the function $m : D \to \mathbb{R}$, with $m(x) = \mathrm{E}\left[Y(x)\right]$ is called the mean function of Y and the function $k : D \times D \to \mathbb{R}$, with $k(x_1, x_2) = \mathrm{Cov}\left[Y(x_1), Y(x_2)\right]$ is called the covariance function of Y.*

One benefit of Gaussian processes is that, whenever a valid covariance function on D can be chosen, a corresponding Gaussian process exists automatically, as shown in the following proposition.

Proposition 1. *Let m be any function from D to $\mathbb{R}$. Let k be a function from $D \times D$ to $\mathbb{R}$ for which, for any $n \in \mathbb{N}$ and for any $x_1, ..., x_n \in D$, the matrix $(k(x_i, x_j))_{1 \leq i,j \leq n}$ is symmetric and non-negative. Then, there exists a Gaussian process Y on D with mean function m and covariance function k.*

The previous proposition can be proved by using Kolmogorov's extension theorem, see for instance [10]. From this proposition, we see that the only non-trivial quantity to define, in order to create a Gaussian process, is the covariance function, since this function must respect the constraint of yielding symmetric non-negative matrices, as in the previous proposition. Fortunately, many functions k are shown to respect this constraint, for many possible input spaces D, see for instance [1, 46].

In terms of modeling, it is common practice to let the mean function drive the large-scale variations of the Gaussian process trajectories, and to let the covariance function drive the small-scale variations. In particular, when D is a subset of a Euclidean space, there exists many results relating the smoothness of the covariance function to the smoothness of the Gaussian process, see [2, 51]. Quite informally (see the references before for formal statements), one can keep in mind that if the covariance function is $2l$ times differentiable, then the Gaussian process has derivatives up to order l in the mean square sense. Furthermore, if the covariance function is "a bit more" than $2l$ times differentiable, then the Gaussian process trajectories are l times differentiable.

1.2. Conditional distribution for a Gaussian process

In the following, we use classical vectorial notations: for any functions $f : D \to \mathbb{R}$, $g : D \times D \to \mathbb{R}$ and for any vectors $A = (a_1, \ldots, a_n) \in D^n$ and $B = (b_1, \ldots, b_m) \in D^m$, we denote by $f(A)$ the $n \times 1$ real valued vector with components $f(a_i)$ and by $g(A, B)$ the $n \times m$ real valued matrix with components $g(a_i, b_j)$, $i = 1, \ldots, n$, $j = 1, \ldots, m$.

With such notations, the conditional distribution of the Gaussian process Y given observations of it is provided in the following Proposition, which follows from the Gaussian conditioning theorem (see e.g. Equation A.6 in [46]).

Proposition 2. *Let Y be a Gaussian process with mean function m and covariance function k. Let $n \in \mathbb{N}$. Let $\epsilon = (\epsilon_1, ..., \epsilon_n)$ be a Gaussian vector with mean vector 0, covariance matrix Σ, and which is independent of Y. Then, for any $X = (x_1, \dots, x_n) \in D^n$, for any $q \in \mathbb{N}$, for any $V = (v_1, ..., v_q) \in D^q$, conditionally to $(Y(x_1) + \epsilon_1, \dots, Y(x_n) + \epsilon_n) = (y_1, \dots, y_n) = y$, $(Y(v_1), ..., Y(v_q))$ is a Gaussian vector with mean vector*

$$m(V) + k(V, X)[k(X, X) + \Sigma]^{-1}(y - m(X))$$

and covariance matrix

$$k(V, V) - k(V, X)[k(X, X) + \Sigma]^{-1}k(X, V).$$

The previous proposition can be reformulated by saying that, given the $n \times 1$ vector of observations $Y(X) + \epsilon = y$, Y is a Gaussian process with mean and covariance function:

$$\begin{cases} \mathrm{E}\left[Y(x)|Y(X) + \epsilon = y\right] = m(x) + k(x, X)[k(X, X) + \Sigma]^{-1}(y - m(X)), \\ \mathrm{Cov}\left[Y(x), Y(x')|Y(X) + \epsilon = y\right] = k(x, x') - k(x, X)[k(X, X) + \Sigma]^{-1}k(X, x'). \end{cases}$$

1.3. **Covariance parameter estimation for a Gaussian process**

In the large majority of the cases, the mean and covariance functions of Y are estimated from the same observation vector $Y(X) + \epsilon$ as for the obtention of the conditional distribution of Y in the Proposition 2.

Here, to simplify the exposition, we shall assume that the mean function of Y is known to be zero, and that the covariance matrix Σ of ϵ is known. Most classically, the covariance function of Y is assumed to belong to a parametric space $\{k_\theta; \theta \in \Theta\}$, where Θ is a finite-dimensional set and where k_θ is a covariance function. For instance, when $D \subset \mathbb{R}$, we can have $\theta = (\sigma^2, \ell) \in (0, \infty)^2$, and $k_\theta(x_1, x_2) = \sigma^2 e^{-\ell|x_1 - x_2|}$, in which case it is assumed that the Gaussian process is a transformation (scaling of the values and of the inputs) of the Ornstein-Uhlenbeck process.

Often, an estimator $\hat{\theta}(y)$, obtained from the observation vector y which is a realization of $Y(X) + \epsilon$, is obtained, and Proposition 2 is used directly with k replaced by $k_{\hat{\theta}}$ and m replaced by 0. This is known as the "plug-in approach" [56]. The most standard estimator $\hat{\theta}$ is the maximum likelihood estimator, defined as

$$\hat{\theta}(y) \in \underset{\theta \in \Theta}{\arg\min} \left(\log(\det(k_\theta(X, X) + \Sigma)) + y^t(k_\theta(X, X) + \Sigma)^{-1}y\right),$$

see for instance [46, 56]. Another class of estimators $\hat{\theta}$ appearing in the literature is given by cross validation estimators [4–6, 46, 66].

1.4. **Current research questions**

There remain important theoretical and practical needs for developments on Gaussian processes.

From a theoretical standpoint, although it yields good performances in practice, Gaussian process-based prediction of unknown functions is significantly less understood than other standard techniques for function prediction. In particular, when considering the setting where D is a fixed compact space and n goes to infinity, it is not clear, to the best of the authors' knowledge, if $\mathrm{E}\left[Y(x)|y\right]$, where $y = f(X) + \epsilon$, converges to $f(x)$ as $n \to \infty$, for any fixed continuous function f. Some partial results are nevertheless given in [24, 63]. In contrasts, it is clear that other popular methods, like nearest-neighboor regression or kernel smoothing, can asymptotically recover continuous functions as $n \to \infty$.

Another aspect of Gaussian processes for which open questions remain is covariance parameter estimation. When D is compact and fixed as $n \to \infty$, it is known that some covariance parameters can not be consistently

estimated [56]. For the parameters that could be consistently estimated, asymptotic results for maximum likelihood are currently available only in special cases [32, 33, 64, 65]. In constrasts, when D is allowed to grow as $n \to \infty$, with volume proportional to n, general results can be given for maximum likelihood [5, 18, 19, 38].

From a practical standpoint, computing the conditional moments of Proposition 2 and computing the likelihood criterion require to inverse matrices of size $n \times n$. This task is generally admitted to entail a complexity of $O(n^3)$ in time and $O(n^2)$ in space. This becomes problematic when n goes beyond $10^3 - 10^4$, and there is currently a large body of literature proposing alternative computation procedures for larger values of n [17, 30, 31, 55].

Finally, Gaussian processes can be inadapted as prior distributions on some types of unknown functions. In particular, when it is known that the unknown function satisfies conditions like boundedness, monotony or convexity, Gaussian processes can become inadapted if used naively. Some alternative procedures are proposed for instance in [20] and references therein. Similarly, the Gaussian distribution can be inadapted as a prior distribution for some types of values, and alternatives to Gaussian processes are proposed for instance in [62].

2. Constrained Gaussian processes

The content of this section corresponds to the articles [14, 35].

2.1. Introduction and related work

In several application situations, physical systems (computer model output) are known to satisfy linear inequality constraints (such as boundedness, monotonicity and convexity) with respect to some or all input variables. For example, in Banking and Finance, the discount factor is known to be monotone (non-increasing) with respect to time-to-maturities [14]. In computer experiment framework, Gaussian Process (GP) is a popular model [50]. This is because it satisfies several nice properties: uncertainty can be quantified, a GP conditionally to given data (equality constraints) is also a GP [16] and the partial derivatives of a GP are also Gaussian processes [16, 44]. Recently, a new methodology to explicitly incorporate linear inequality constraints into a GP model has been proposed in [28]. It is based on a modification of the covariance function in Gaussian processes. In [23, 47] respectively, the problem is to incorporate monotonicity constraints into a Gaussian process model without (resp. with) noisy data. The main idea in their strategies is to put derivative informations in a special places to force the process to be monotone. By this methodology the monotonicity constraint is not respected in the entire domain. Our aim is to show how one can incorporate inequality constraints into a Gaussian process model, where the inequality constraints are guaranteed in the entire domain. The difficulty of the problem comes from the fact that, when incorporating an infinite number of inequality constraints into a GP, the resulting process is not a GP in general. In this section, we show that the model developed in [35] is capable of incorporating an infinite number of inequality constraints into a GP model

$$Y^N(\boldsymbol{x}) = \sum_{j=0}^{N} \xi_j \phi_j(\boldsymbol{x}), \quad \boldsymbol{x} \in \mathbb{R}^d,$$

where $\xi = (\xi_0, \ldots, \xi_N)^t$ is a zero-mean Gaussian vector with covariance function Γ^N and ϕ_j are some basis functions. The basis functions $\phi_1, ..., \phi_N$ are fixed by the user, and the covariance matrix of $\epsilon_1, ..., \epsilon_N$ is assumed to be known throughout the section (except in the application in Section 2.5).

2.2. Monotonicity in one-dimensional case

In this section, let $\mathcal{C}^1$ be the space of functions that admit derivatives up to order 1. Let C be the set of monotone (non-decreasing) functions defined as

$$C = \{f \in \mathcal{C}^1(\mathbb{R}) \ : \ f'(x) \geq 0, \ x \in \mathbb{R}\}.$$

We are interested in constraining a GP Y so that its sample paths belong to C. Without loss of generality, the input variable x is supposed to be in $[0,1]$. For simplicity, the original GP Y is supposed to be zero-mean with known covariance function K. First, we discretize the input set for example uniformly to $N+1$ knots $u_0,\dots,u_N$ but the methodology can be adapted easily to non-uniform subdivision. Then, we consider a set of basis functions

$$\phi_j(x) = \int_0^x h_j(x), \quad x \in [0,1],$$

where $h_j,\ j=0,\dots,N$ are the hat functions associated to the knots u_j: $h_j(x) = h((x-u_j)/\Delta_N)$, where $\Delta_N = 1/N$ and $h(x) = (1-|x|)\,\mathbf{1}_{(|x|\leq 1)},\ x \in \mathbb{R}$. First, we remark that the value of the derivative of any basis function at any knot is equal to Kronecker's Delta function ($\phi_j'(u_k) = \delta_{j,k}$), where $\delta_{j,k}$ is equal to one if $j=k$ and zero otherwise. Second, the hat functions are non-negative and then the basis functions ϕ_j are non-decreasing in $[0,1]$. In that case, the proposed model can be reformulated as follows

$$Y^N(x) = Y(0) + \sum_{j=0}^{N} Y'(u_j)\phi_j(x) = \zeta + \sum_{j=0}^{N} \xi_j \phi_j(x), \quad x \in [0,1], \tag{1}$$

where $\zeta = Y(0)$ and $\xi_j = Y'(u_j),\ j=0,\dots,N$.

Proposition 3. *With the notation introduced before, the finite-dimensional approximation Y^N of the original Gaussian processes Y verifies the following properties*

(1) *$(Y^N(x))_{x\in[0,1]}$ is a finite-dimensional GP with covariance function*

$$K_N(x,x') = \left(1, \phi(x)^t\right) \Gamma^N \left(1, \phi(x')^t\right)^t,$$

where $\phi(x) = (\phi_0(x),\dots,\phi_N(x))^t$ and Γ^N is the covariance matrix of the Gaussian vector $(Y(0), Y'(u_0),\dots,Y'(u_N))^t$ which is equal to

$$\Gamma^N = \begin{bmatrix} K(0,0) & \frac{\partial K}{\partial x'}(0,u_j) \\ \frac{\partial K}{\partial x}(u_i,0) & \Gamma^N_{i,j} \end{bmatrix}_{0\leq i,j\leq N},$$

with $\Gamma^N_{i,j} = \frac{\partial^2 K}{\partial x \partial x'}(u_i,u_j),\ i,j=0,\dots,N$ and K the covariance function of the original GP Y.

(2) *Y^N converges uniformly pathwise to Y when N tends to infinity.*

(3) *Y^N is monotone (non-decreasing) if and only if the $(N+1)$ random coefficients $Y'(u_j)$ are non-negative.*

Proof. The proof of the proposition can be found in [35]. □

From this proposition, one can deduce that the infinite number of inequality constraints is reduced to a finite minimum number of constraints on the coefficients $Y'(u_j),\ j=0,\dots,N$. Therefore, the problem is reduced to the simulation of the Gaussian vector $(Y(0), Y'(u_j),\dots,Y'(u_N))^t$ truncated in the convex set formed by the following two constraints (interpolation and inequality constraints):

$$\begin{aligned} &Y^N(X) = y, \\ &(\zeta,\xi) \in C_\xi = \left\{(\zeta,\xi) \in \mathbb{R}^{N+2} \ :\ \xi_j \geq 0,\ j=0,\dots,N\right\}. \end{aligned}$$

Simulated trajectories: From Proposition 3 (Item (3)), the simulation of the finite-dimensional approximation of Gaussian processes Y^N conditionally to given data and monotonicity constraints is equivalent to simulating the Gaussian vector (ζ,ξ) restricted to $I_\xi \cap C_\xi$

$$\begin{aligned} I_\xi &= \left\{(\zeta,\xi) \in \mathbb{R}^{N+2} \ :\ A(\zeta,\xi) = y\right\}, \\ C_\xi &= \left\{(\zeta,\xi) \in \mathbb{R}^{N+2} \ :\ \xi_j \geq 0,\ j=0,\dots,N\right\}, \end{aligned}$$

where the $n \times (N+2)$ matrix A is defined as

$$A_{i,j} = \begin{cases} 1 & \text{for } i = 1, \dots, n \text{ and } j = 1, \\ \phi_{j-2}\left(x^{(i)}\right) & \text{for } i = 1, \dots, n \text{ and } j = 2, \dots, N+2. \end{cases}$$

The sampling scheme can be summarized in two steps: first of all, we compute the conditional distribution of the Gaussian vector (ζ, ξ) with respect to interpolation conditions

$$(\zeta, \xi) \mid A(\zeta, \xi) = y \sim \mathcal{N}\left((A\Gamma^N)^t (A\Gamma^N A^t)^{-1} y, \Gamma^N - (A\Gamma^N)^t (A\Gamma^N A^t)^{-1} A\Gamma^N\right). \tag{2}$$

We simulate the Gaussian vector (ζ, ξ) with the above distribution (2). Using an improved rejection sampling [34], we select only random coefficients in the convex set C_ξ. The sample paths of the conditional Gaussian process are then generated by (1), so that they satisfy both interpolation conditions and monotonicity constraints in the entire domain (see the R package 'constrKriging' developed in [36] for more details).

Remark 1 (Boundedness constraints)**.** *For boundedness constraints, the finite-dimensional approximation Y^N of Y can be reformulated as*

$$Y^N(x) = \sum_{j=0}^{N} Y(u_j) h_j(x), \quad x \in [0,1],$$

where $h_j,\ j = 0, \dots, N$ are the hat functions centered at the j^{th} knot. In that case, the covariance matrix of the coefficients $(Y(u_j))_j$ is equal to Γ^N, where $\Gamma^N_{i,j} = K(u_i, u_j)$. Additionally, Y^N is bounded between two constants a, b (i.e., $Y^N(x) \in [a,b]$) if and only if $Y(u_j) \in [a,b]$ for all $j = 0, \dots, N$.

Remark 2 (Convexity constraints)**.** *For convexity constraints, the finite-dimensional approximation Y^N of Y can be written as*

$$Y^N(x) = Y(0) + xY'(0) + \sum_{j=0^N} Y''(u_j) \varphi_j(x), \quad x \in [0,1],$$

where $\varphi_j,\ j = 0, \dots, N$ are the two times primitive of the hat functions (i.e., $\varphi_j(x) = \int_0^x (\int_0^t h_j(u)du)dt$). In that case, the covariance matrix of the coefficients $Y''(u_j)$ is equal to

$$\Gamma^N_{i,j} = \mathrm{Cov}\left[Y''(u_i), Y''(u_j)\right] = \frac{\partial^2}{\partial x \partial x'} K(u_i, u_j).$$

Additionally, Y^N is convex if and only if $Y''(u_j) \geq 0$ for all $j = 0, \dots, N$.

2.3. **Isotonicity in two dimensions**

We now assume that the input is $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$ and without loss of generality is in the unit square. The monotonicity (non-decreasing) constraints with respect to the two input variables is defined as

$$x_1 \leq x_1' \quad \text{and} \quad x_2 \leq x_2' \quad \Rightarrow \quad f(x_1, x_2) \leq f(x_1', x_2').$$

As in the one-dimensional case, we construct the basis functions such that monotonicity constraints are *equivalent* to constraints on the coefficients. The finite-dimensional approximation of GPs Y^N is defined as

$$Y^N(x_1, x_2) \quad = \quad \sum_{i,j=0}^{N} Y(u_i, u_j) h_i(x_1) h_j(x_2) = \sum_{i,j=0}^{N} \xi_{i,j} h_i(x_1) h_j(x_2), \tag{3}$$

where h_j are the hat functions defined in Section 2.2.

Proposition 4. *With the notations introduced before, $(Y^N(\boldsymbol{x}))_{\boldsymbol{x} \in [0,1]^2}$ verifies the following properties:*

- Y^N *is a finite-dimensional GP with covariance function* $K_N(\boldsymbol{x},\boldsymbol{x}') = \Phi(\boldsymbol{x})^t \Gamma^N \Phi(\boldsymbol{x}')$, *where* $\Phi(\boldsymbol{x})^t = (h_i(x_1)h_j(x_2))_{i,j}$, $\Gamma^N_{(i,j),(i',j')} = K\left((u_i,u_j),(u_{i'},u_{j'})\right)$ *and* K *is the covariance function of the original GP* Y.
- Y^N *converges uniformly to* Y *when* N *tends to infinity.*
- Y^N *is non-decreasing with respect to the two input variables if and only if the* $(N+1)^2$ *random coefficients* $\xi_{i,j},\ i,j = 0,\dots,N$ *verify the following linear constraints:*
  (1) $\xi_{i-1,j} \le \xi_{i,j}$ and $\xi_{i,j-1} \le \xi_{i,j},\ i,j = 1,\dots,N$.
  (2) $\xi_{i-1,0} \le \xi_{i,0},\ i = 1,\dots,N$.
  (3) $\xi_{0,j-1} \le \xi_{0,j},\ j = 1,\dots,N$.

From the last property, the problem is equivalent to simulate the Gaussian vector $\xi = (\xi_{i,j})_{i,j}$ restricted to the convex set $I_\xi \cap C_\xi$, where

$$I_\xi = \left\{ \xi \in \mathbb{R}^{(N+1)^2} \ : \ Y^N\left(x_1^{(i)}, x_2^{(i)}\right) = \sum_{i,j=0}^{N} \xi_{i,j} h_i\left(x_1^{(i)}\right) h_j\left(x_2^{(i)}\right) = y_i \right\},$$

$$C_\xi = \left\{ \xi \in \mathbb{R}^{(N+1)^2} \text{such that } \xi_{i,j} \text{ verify the constraints 1. 2. and 3.} \right\}.$$

Remark 3 (Isotonicity in two dimensions with respect to one variable)**.** If the function is non-decreasing with respect to the first variable only, then

$$Y^N(x_1,x_2) = \sum_{i,j=0}^{N} Y(u_i,u_j) h_i(x_1) h_j(x_2) = \sum_{i,j=0}^{N} \xi_{i,j} h_i(x_1) h_j(x_2), \tag{4}$$

is non-decreasing with respect to x_1 *if and only if* the random coefficients satisfy $\xi_{i-1,j} \le \xi_{i,j},\ i = 1,\dots,N$ and $j = 0,\dots,N$.

2.4. **Constrained Kriging in financial term-structures**

The suggested model (1) has been applied to finance and economic domain to estimate discount factors and default probabilities [14]. In this section, we focus on discount factors. The real data are represented by the following linear equality constraints

$$AY^N(X) = b, \tag{5}$$

where A and b are some given matrix and vector respectively and X is the input vector of observations.

2.5. **Curve construction at a single and several quotation dates**

We now illustrate the constrained GP method described above in a one and two-dimensional setting. In one-dimensional case, the construction is based on market quotes as of 30/12/2011 [15]. The Matérn 5/2 covariance function has been used

$$k(x,x') = \sigma^2 \left(1 + \frac{\sqrt{5} \mid x - x' \mid}{\theta} + \frac{5(x-x')^2}{3\theta^2}\right) \exp\left(-\frac{\sqrt{5}|x-x'|}{\theta}\right), \tag{6}$$

where the covariance parameters (σ and θ) have been estimated using the suited cross validation method described in [14] and [37]. We get $\hat{\theta} = 30$ and $\hat{\sigma} = 0.93$.

In the left panel of Figure 1, we generate 100 sample paths taken from model (1) conditionally to linear equality constraints (5) and monotonicity (non-increasing) constraints. Note that the simulated curves (gray lines) are non-increasing in the entire domain. The black solid line represents the posterior maximum of the constrained Gaussian process. The black dashed-lines represent the 95% point-wise confidence intervals quantified by simulation. The red dash-dotted points are associated to the best-fitted Nelson-Siegel curves [42]

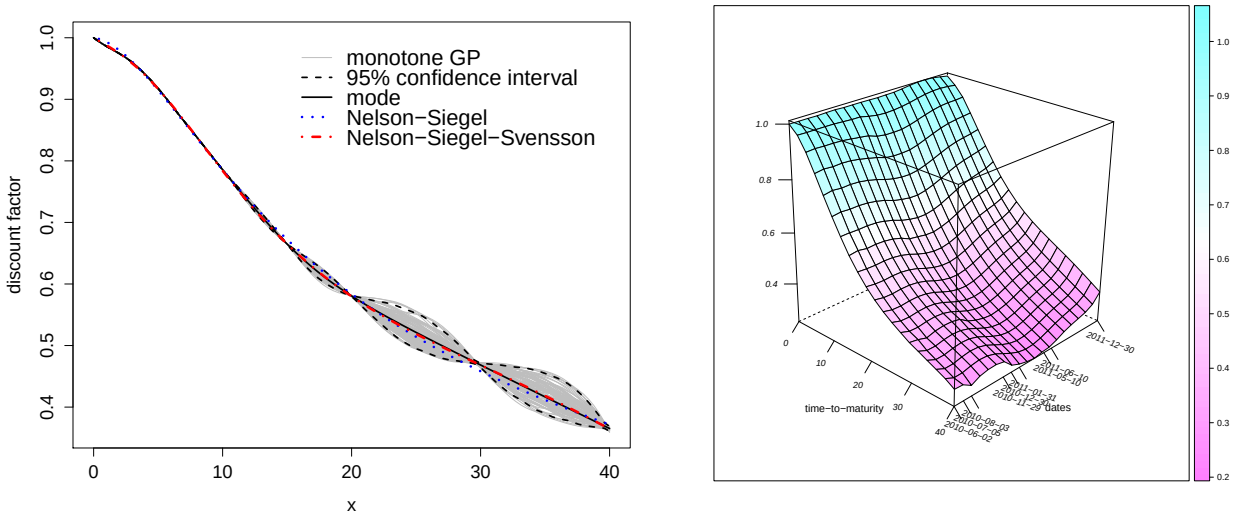

FIGURE 1. Simulated paths (gray lines) taken from the conditional GP with non-increasing constraints and market-fit constraints Swap vs Euribor 6M market quotes as of 30/12/2011 (left). Discount factors as a function of time-to-maturities and quotation dates (right).

and the blue dotted points are associated to the best-fitted Svensson curves [57]. In the right panel of Figure 1, discount factors as a function of time-to-maturities and quotation dates are shown using model (3). The monotonicity (non-increasing) constraint is respected to the first (time-to-maturities) variable only.

3. NESTED KRIGING FOR LARGE DATASETS

We present in this section some results from a joint work with F. Bachoc, C. Chevalier, N. Durrande and D. Rullière [49].

As discussed in Section 1.4, it is usually admitted that obtaining a Kriging prediction (in the sense of Proposition 2), at one point given n observations of the process Y, has a complexity of $O(n^3)$ in time and $O(n^2)$ in space. We propose here a new method aiming at reducing these complexities in order to deal with large datasets.

Classical methods of the literature are dedicated to this problem, as *inducing points* [26,43], *low rank approximations* [54], *Gaussian Markov Random Fields* [48], *compactly supported covariance functions* and *covariance tapering* [22,31,53]. These methods suffer from either the loss of interpolation properties or either difficulties to capture small or large scale dependencies. Some methods aggregate submodels or "experts" based on subsets of the data, as *mixture of experts* [21,27,58], or *consensus methods* [60,61]. As they often ignore some covariances between submodels or experts, one can show that they suffer from inconsistencies and accuracy losses.

3.1. **Proposed aggregation**

The proposed method is based on the idea of aggregating submodels that are cheaper to construct. Compared to [49], we focus here on the simplified context of Kriging submodels relying on centered Gaussian Processes, but results can be adapted to more general settings, such as non-Gaussian processes, or other types of submodels.

Let us split the input points vector X into p distinct subvectors X_i, $i = 1, \ldots, p$. Consider a new input point $x \in D$ where we want to predict $Y(x)$. Now consider Gaussian process regression submodels M_i, each based on

a subset of the data X_i, $i \in \mathcal{A}$, where $\mathcal{A} = \{1, \dots, p\}$ is the set of submodels indexes:

$$M_i(x) = \mathrm{E}\left[Y(x)|Y(X_i)\right] = k(x, X_i)k(X_i, X_i)^{-1}Y(X_i)\,. \tag{7}$$

The p submodels are gathered into a $p\times 1$ vector $M(x) = (M_1(x)\dots, M_p(x))^t$. For a given covariance function k, the random column vector $(M_1(x), \dots, M_p(x), Y(x))^t$ is centered with finite first two moments and both the $p \times 1$ covariance vector $k_M(x) = \mathrm{Cov}\left[M(x), Y(x)\right]$ and the $p \times p$ covariance matrix $K_M(x) = \mathrm{Cov}\left[M(x), M(x)\right]$ can be obtained with basic linear algebra:

$$\begin{cases} \big(k_M(x)\big)_i = \mathrm{Cov}\left[M_i(x), Y(x)\right] = k(x, X_i)k(X_i, X_i)^{-1}k(X_i, x)\,, \\ \big(K_M(x)\big)_{i,j} = \mathrm{Cov}\left[M_i(x), M_j(x)\right] = k(x, X_i)k(X_i, X_i)^{-1}k(X_i, X_j)k(X_j, X_j)^{-1}k(X_j, x)\,. \end{cases} \tag{8}$$

The main idea for aggregating the submodels $M_i(x), \dots, M_p(x)$ is to take into account their cross-covariances, as well as their covariances with the unknown output $Y(x)$. It differs in this sense to most consensus aggregation techniques [60, 61] and to classical machine learning aggregations [21, 58].

The proposed aggregation $M_{\mathcal{A}}(x)$ and its associated mean square error $v_{\mathcal{A}}(x)$ are defined as

$$\begin{cases} M_{\mathcal{A}}(x) = \mathrm{E}\left[Y(x)|M_i(x), i \in \mathcal{A}\right] = k_M(x)^t K_M(x)^{-1} M(x)\,, \\ v_{\mathcal{A}}(x) = \mathrm{V}\left[Y(x)|M_i(x), i \in \mathcal{A}\right] = k(x, x) - k_M(x)^t K_M(x)^{-1} k_M(x)\,. \end{cases} \tag{9}$$

3.2. **Properties**

Among basic properties in the specific Gaussian Kriging case, this aggregation is optimal and interpolating: it is optimal in the sense that $M_{\mathcal{A}}(x)$ is the best linear unbiased estimator (BLUE) of $Y(x)$ that writes $\sum_{i\in\mathcal{A}} \alpha_i(x)M_i(x)$, with mean squared error $v_{\mathcal{A}}(x) = \mathrm{E}\left[(Y(x) - M_{\mathcal{A}}(x))^2\right]$. Furthermore, the aggregation is interpolating: if one of the submodels M_j interpolates a point x_i, $M_j(x_i) = Y(x_i)$, then the aggregated model is also interpolating at this point, $M_{\mathcal{A}}(x_i) = Y(x_i)$ and $v_{\mathcal{A}}(x_i) = 0$. Note that some usual methods dealing with large datasets, as inducing points [26, 43], do not satisfy this interpolation property.

In the example of Figure 2 we give two Kriging predictors, one predictor $M_1(.)$ based on four observations, one other $M_2(.)$ based on three other observations. There is no difficulty to obtain the required quantities $k_M(x)$, $K_M(x)$, and thus the aggregated predictor. One can observe that the aggregate predictor $M_{\mathcal{A}}(x)$ is very close to the one of the full model, $M_{\text{full}}(x)$, which is the classical Kriging predictor based on all seven observations.

From a theoretical point of view, more properties can be derived. Some developments show that the aggregation method can be seen as an approximation of the full model, but can also be seen as an exact method relying on a slightly modified process. Partly relying on this fact, bounds for the difference $|M_{\mathcal{A}}(x) - M_{\text{full}}(x)|$ can be derived. It can also be shown that if the knowledge of all submodels at the prediction point x allows to retrieve all initial observations, then the aggregation $M_{\mathcal{A}}(.)$ corresponds exactly to the full model $M_{\text{full}}(.)$. The detail of all these properties is given in [49].

An important consistency result, justifying the use of covariances, is the following one, which is proved in [9].

Proposition 5 (Consistency)**.** *Let D be a compact subset of $\mathbb{R}^d$. Let Y be a Gaussian process on D with mean zero and continuous covariance function k. Let $(x_{ni})_{1\le i\le n, n\in\mathbb{N}}$ be a triangular array of observation points so that $x_{ni} \in D$ for all $1 \le i \le n, n \in \mathbb{N}$ and so that for all $x \in D$, $\lim_{n\to\infty} \min_{i=1,\dots,n} ||x_{ni} - x|| = 0$.*

For $n \in \mathbb{N}$, let $\mathcal{A}_n = \{1, ..., p_n\}$ be the set of submodel indexes and let $M_1(x), ..., M_{p_n}(x)$ be any collection of p_n Kriging predictors based on respective design points $X_1, \dots, X_{p_n}$. Assume that each component of

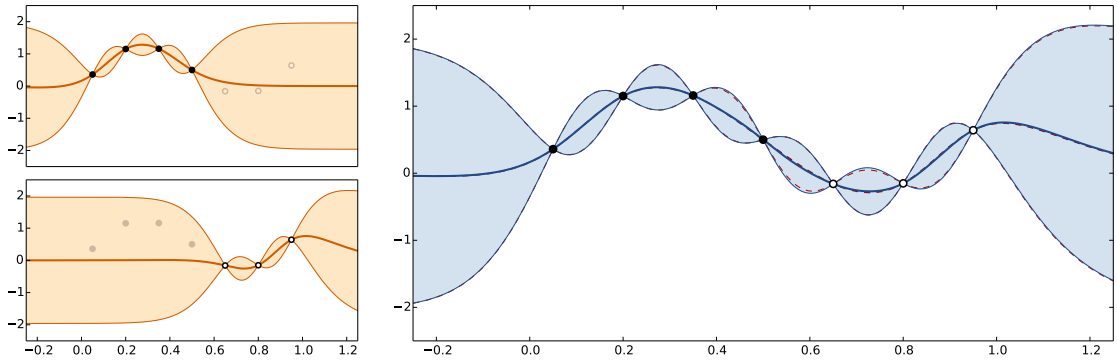

FIGURE 2. Example of aggregation of two Gaussian process regression models. For each model, we represent the predicted mean and 95% confidence intervals. Left panels: submodels to aggregate. Right panel: aggregated model (solid lines) and full model (dashed lines).

$X = (x_{n1}, ..., x_{nn})$ is a component of at least one X_i, $1 \leq i \leq p_n$. Then we have

$$\sup_{x \in D} \mathbb{E}\left(\left(Y(x) - M_{\mathcal{A}_n}(x) \right)^2 \right) \to_{n \to \infty} 0. \tag{10}$$

In the literature, many aggregation methods do not use covariances between submodels, but only prediction variances of each submodel. This is the case for many consensus aggregation and for other aggregation techniques as Product of Experts, Generalized Product of Experts, Bayesian Committee Machine, Robust Bayesian Committee Machine (see [21]). For these methods and under quite general assumptions, one can show that it is possible to find a triangular array of observations that becomes dense into D, but is however such that

$$\liminf_{n \to \infty} \mathrm{E}\left[\left(Y(x) - \bar{M}_{\mathcal{A}_n}(x) \right)^2 \right] > 0. \tag{11}$$

where $\bar{M}_{\mathcal{A}_n}$ is the considered alternative aggregation technique. It results that, contrary to the proposed aggregation, many classical methods can be shown to be inconsistent. For a large number of observations, this is clearly an important advantage of the proposed method.

Concerning the complexity, one can show that starting from n observations, a reachable complexity for q prediction points is $O(n)$ in space and $O(qn^2)$ in time, for example when aggregating $\sqrt{n}$ submodels of $\sqrt{n}$ observations each. This complexity is to be compared to $O(n^2)$ in space and $O(n^3 + qn)$ in time for the full model. Thus, while requiring an additional cost compared to the cheapest methods of the literature, the method is still tractable for large number of observations, say up to 10^6, provided that the number of prediction points is small compared to the number of observations, $q \ll n$.

As aggregated models can themselves be aggregated, the method can be built along more complex tree structures. However, while this can reduce the complexity in time, the general computational complexity order remains $O(qn^2)$, only the factor multiplying this order being modified. This however opens some perspectives for further large reduction of the complexity of the algorithm.

3.3. **Numerical illustration with known covariance**

Consider test functions that are given by samples over $[0, 1]$ of a centered Gaussian process Y with a one dimensional Matérn 5/2 kernel, with known parameters σ^2 and θ (see (6) for a definition of the Matérn kernel).

The Figure 3 shows the boxplots of some performance criteria for 50 replications of the experiments. Each experiment consists in predicting the Gaussian process sample path on a grid of 201 points, based on 30 observations. 10 submodels are build with three observations each.

We consider the following criteria to quantify the distance between the aggregated model and the full model: the mean square error (MSE) that measures the accuracy of the aggregated mean compared to the full model, the mean variance error (MVE) that measures the accuracy of the predicted variance and the mean negative log probability (MNLP) that measures the overall distribution fit, see [59]. The other methods that are considered in this benchmarck are the (generalized) product of Experts: PoE, GPoE, the (robust) Bayesian Committee Machine: BCM, RBCM, and the smallest Prediction Variance (SPV), see [21, 49].

The setting differs slightly from the one in [49], but the conclusion remains identical: it appears in Figure 3 that the proposed approach gives the best approximation of the full model for the three considered criteria.

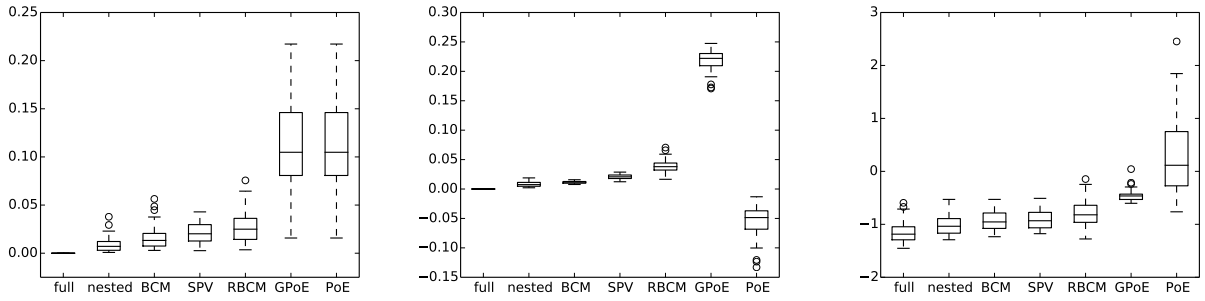

FIGURE 3. Quality assessment of the aggregated models for 50 test functions. Each test function is a sample from a Gaussian process and in each case 30 observation points are sampled uniformly on $[0, 1]$. Left panel: MSE (should be small), center panel: MVE (should be close to 0), right panel: MNLP (should be small).

3.4. **Numerical illustration with unknown covariance**

In practice, covariance parameters of the kernel are unknown and have to be estimated. For kernels that write $k = \sigma^2 k_\theta$, we give a leave-one-out method for estimating the parameter θ, and then the parameter σ^2:

$$\widehat{\theta} \in \operatorname*{argmin}_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n} \left(f(x_i) - m_{\mathcal{A},\theta,-i}(x_i)\right)^2 \quad \text{and} \quad \widehat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} \frac{\left(f(x_i) - m_{\mathcal{A},\widehat{\theta},-i}(x_i)\right)^2}{v_{\mathcal{A},\widehat{\theta},-i}(x_i)}\,, \tag{12}$$

where $f(x_i)$ are the observed values of $Y(x_i)$ and $m_{\mathcal{A},\theta,-i}$ the leave-one-out mean prediction based on a parameter θ and a leaved observation i, which does not depend on σ. For usual Kriging methods, the computational cost for calculating one leave-one-out error $\left(f(x_i) - m_{\mathcal{A},\theta,-i}(x_i)\right)^2$ is $O(n^3)$. In our procedure, this cost can be $O(n^2)$. Thus for large datasets, even one leave-one-out error cannot be computed with usual Kriging methods, whereas it becomes tractable with our method. Furthermore, the average leave-one-out error can be estimated on a subset of successively excluded points $\mathcal{I}$, having cardinal $q \ll n$, thus leading to an average error of cost $O(qn^2)$.

This makes possible to build a dedicated stochastic gradient descent for calculating an estimator of θ, defined as the limit of a sequence of $(\theta_i)_{i\geq 1}$, starting from a given value θ_0. The sequences (a_i) and (δ_i) of increments

and step sizes, and the sequence of random directions (h_i) are given.

$$\theta_i = \theta_{i-1} - \frac{a_i}{2\delta_i}\left(\frac{1}{q}\sum_{j\in\mathcal{I}_i}\left(f(x_j) - m_{\mathcal{A},-j,\theta_{i-1}+\delta_i h_i}(x_j)\right)^2 - \frac{1}{q}\sum_{j\in\mathcal{I}_i}\left(f(x_j) - m_{\mathcal{A},-j,\theta_{i-1}-\delta_i h_i}(x_j)\right)^2\right) h_i. \quad (13)$$

Subsets $\mathcal{I}_i$ are sampled uniformly among subsets of $\{1,\dots,n\}$ of size q. Some more details and links to a publicly available code are given in [49].

On a real data set of size $n = 10^4$ in dimension $d = 6$, we get for $p = 90$ submodels a MSE criterion of 0.00418, which is better than the most competitive of the other methods in this benchmark, SPV and a variant of GPoE, having respective MSE equal to 0.00556 and 0.0244. Results are quite similar with MNLP criterion, -1.7 for our method, compared to -1.55 and 2.13 for the best two competitors. Other numerical results are omitted here, but we get the same conclusion when replicating the experiments with other test and learning sets, or other covariance kernels. Overall, we found that the accuracy of the proposed method, along with dedicated parameter estimation, outperforms state-of-the-art alternative aggregation methods.

4. Gaussian processes for stochastic optimization

The content of this section corresponds to the references [12, 13].

4.1. **Sequential stochastic optimization**

Optimizing an unknown, non-convex and potentially noisy function is at the center of many computer experiments. The goal of a sequential optimization procedure may be either seen as maximizing the sum of the outputs (or rewards) received at each iteration, that is to minimize the cumulative regret widely used in bandit problems, or as maximizing the best reward received so far, that is to minimize the simple regret. This task becomes challenging when we only have mild information about the unknown function. To tackle this challenge, a Bayesian approach has been shown to be empirically efficient [29, 40]. In this approach, we model the unknown function as a centered Gaussian Process $Y : D \to \mathbb{R}$ which allows to control the assumptions we put on the smoothness of the function by choosing different kernels, as described in Section 1. We consider that the observations from the unknown function are affected by an independent additive Gaussian noise. A sequential optimization algorithm iterates two steps: at iteration $t \in \mathbb{N}$, it first chooses $x_t \in D$ based on the previous observations $y_1, \dots, y_{t-1}$, and next queries the unknown function at x_t and observes $y_t = Y(x_t) + \epsilon_t$, where ϵ_t is an independent centered Gaussian of known variance η^2. The cumulative regret is then an unknown random variable, defined for each iteration $T \in \mathbb{N}$ as:

$$R_T = \sum_{t=1}^{T}\Big(\sup_{x\in D} Y(x) - Y(x_t)\Big) = T\sup_{x\in D} Y(x) - \sum_{t=1}^{T} Y(x_t)\,.$$

The simple regret, or optimization error, is similarly defined as:

$$S_T = \sup_{x\in D} Y(x) - \max_{1\le t\le T} Y(x_t)\,.$$

We note that $S_T \le R_T/T$, therefore an upper bound of R_T for a given optimization algorithm leads to an upper bound on the optimization error.

Several optimization strategies have been proposed in this respect. The Expected Improvement algorithm [29] is the one-step-ahead optimal rule. The convergence rate of the simple regret obtained by this strategy is analyzed in [11] for deterministic and fixed function in the RKHS space generated by the covariance of the Gaussian process. The Entropy Search algorithm [25] approximates the maximization of the information gain

Algorithm 1: GP-UCB(k, η, δ) on finite D

$X_0 \leftarrow \emptyset$; $Y_0 \leftarrow \emptyset$
for $t = 0, 1, \dots$ **do**
  $\beta_t \leftarrow 2\log\left(|D| t^2 \frac{\pi^2}{6\delta}\right)$
  for $x \in D$ **do**
    $\mu_t(x) \leftarrow k(X_t, x)\left[k(X_t, X_t) + \eta^2 I\right]^{-1} Y_t$
    $s_t^2(x) \leftarrow k(x, x) - k(X_t, x)\left[k(X_t, X_t) + \eta^2 I\right]^{-1} k(X_t, x)$
    $U_t(x) \leftarrow \mu_t(x) + \sqrt{\beta s_t^2(x)}$
  end
  $x_{t+1} \leftarrow \arg\max_{x \in D} U_t(x)$; $y_{t+1} \leftarrow$ **Query**(x_{t+1})
  $X_{t+1} \leftarrow [X_{t+1}; x_{t+1}]$; $Y_{t+1} \leftarrow [Y_{t+1}; y_{t+1}]$
end

on the optimum from each evaluation. This strategy typically displays low simple regret in practice. To the best of the authors' knowledge, no theoretical convergence rates are known. The GP-UCB algorithm [52] extends the popular UCB policy from bandits problem [3] to the Bayesian optimization setting. For a finite space D, this algorithm exhibits state-of-the-art regret bounds, for both the simple and cumulative regrets. At each iteration t, a high probabilistic upper confidence bound on the unknown function is built given the available observations. For a fixed tolerance parameter $0 < \delta < 1$, the algorithm defines $\beta_t = 2\log\left(|D| t^2 \frac{\pi^2}{6\delta}\right)$ and the upper confidence bounds:

$$\forall x \in D, \qquad U_t(x) = \mu_t(x) + \sqrt{\beta_t s_t^2(x)}\,,$$

where μ_t (resp. s_t^2) is the posterior expectation (resp. variance) given the observations, defined in Proposition 2. The selection of the next query then follows the maximization of the upper confidence bounds, which forms a tradeoff between exploitation (maximizing the predicted value) and exploration (maximizing the uncertainty):

$$x_{t+1} \in \arg\max_{x \in D} U_t(x)\,.$$

The GP-UCB algorithm is presented in Algorithm 1, its theoretical guarantees are described in the following section.

4.2. **Theoretical guarantees for finite input spaces**

Thanks to Proposition 2 we know that the posterior distribution of the Gaussian process is another Gaussian process. Therefore, with a union bound on the iterations $n \in \mathbb{N}$ and on the input points $x \in D$, we have the following concentration guarantee.

Proposition 6. *Fix $0 < \delta < 1$. Defines $\beta_t = 2\log\left(|D| t^2 \frac{\pi^2}{6\delta}\right)$ for all iteration $t \in \mathbb{N}$. Then for a centered Gaussian process Y, with probability at least $1 - \delta$:*

$$\forall t \in \mathbb{N}, \forall x \in D, \qquad \mu_t(x) - \sqrt{\beta_t s_t^2(x)} \leq Y(x) \leq \mu_t(x) + \sqrt{\beta_t s_t^2(x)}\,,$$

where μ_t and s_t^2 are as previously.

Using the UCB rule, that is choosing the points maximizing the upper confidence bound, we can bound the regret incurred at each iteration.

Proposition 7. *For all iteration $t \in \mathbb{N}$, selects $x_{t+1} \in \arg\max_{x \in D} U_t(x)$ as previously. Under the event of Proposition 6 the following holds,*

$$\forall t \in \mathbb{N}, \qquad \sup_{x \in D} Y(x) - Y(x_{t+1}) \leq 2\sqrt{\beta_t s_t^2(x_{t+1})}\,.$$

That is, for the cumulative regret R_T,

$$\forall T \in \mathbb{N}, \qquad R_T \leq 2\sqrt{\beta_T} \sum_{t=1}^{T} s_{t-1}(x_t)\,.$$

The previous inequality can be translated into explicit regret bounds according to the covariance of the Gaussian process, using the information-theoretical inequalities of [52]. We consider here three popular covariance functions for $D \subset \mathbb{R}^n$:

- the linear kernel, $k(x, x') = x^t x'$ modelling linear functions,
- the squared exponential kernel, $k(x, x') = \exp(-\|x - x'\|_2^2)$ modelling infinitely differentiable functions,
- the Matérn kernel of parameter $\nu > 1$, $k(x, x') = \int_D e^{iu^t(x-x')}(1 + \|u\|_2^2)^{-\nu - n/2} du$ modelling functions that are m times differentiable for the largest integer $m < \nu$.

We note that the Matérn kernel enjoys simple explicit forms when 2ν is an odd integer. For $\nu = 1/2$, the process is the Ornstein-Uhlenbeck process, for $\nu \to \infty$, the Matérn kernel converges to the squared exponential kernel.

Proposition 8 (Lemma 5.4 and Theorem 5 in [52])**.** *Let D be a finite subset of $\mathbb{R}^n$, under the event of Proposition 6, we have the following inequalities, with $\beta_T \leq \mathcal{O}(\log(T|D|))$:*

- *for the linear kernel, $R_T \leq \mathcal{O}\big(\sqrt{\beta_T n T \log T}\big)$,*
- *for the squared exponential kernel, $R_T \leq \mathcal{O}\Big(\sqrt{\beta_T T \log^{n+1} T}\Big)$,*
- *for the Matérn kernel of parameter $\nu > 1$, $R_T \leq \mathcal{O}\Big(\sqrt{\beta_T} T^{\frac{\nu + n(n+1)}{2\nu + n(n+1)}}\Big)$.*

The previous inequalities ensure that the GP-UCB strategy has a sub-linear cumulative regret with high probability for the considered covariance functions. This directly implies convergence rates of the optimization error. The next section extends these results for non-finite input spaces D.

4.3. **Theoretical guarantees for continuous metric spaces**

When the input space D is continuous, the union bound from Proposition 6 does not hold. We solve this issue by introducing successive discretizations of D. Points in these discretizations are well-spaced for a particular pseudo-metric defined from the covariance of Y. We calibrate their density such that the approximation error is of the order of the error of the UCB policy. We first define the canonical pseudo-metric of the Gaussian process by:

$$\forall x, x' \in D, \qquad d(x, x') = \sqrt{k(x, x) - 2k(x, x') + k(x', x')}\,.$$

We are able to derive bounds on the discretization error of well-spaced points with respect to d, as described in the following proposition.

Proposition 9. *Let D be a bounded subset of $\mathbb{R}^n$. Let X be a finite subset of D and a mapping $\pi : D \to X$ such that it exists $\epsilon > 0$ satisfying $\forall x \in D, d(x, \pi(x)) \leq \epsilon$. Then for all $\delta > 0$ and the covariance functions considered in Proposition 8, with probability at least $1 - \delta$,*

$$\forall x \in D, \qquad Y(x) - Y(\pi(x)) \leq \mathcal{O}\Big(\epsilon\sqrt{\log(1/\delta) + n\log(1/\epsilon)}\Big).$$

The proof of this result involves tight concentration inequalities on the extremes of the Gaussian process, detailed in [12]. The previous proposition allows to adapt the GP-UCB algorithm for continuous spaces. At

iteration t, we run the UCB policy on X_t a discretization of D with mapping π_t, that is we select $x_{t+1} \in \arg\max_{x \in X_t} U_t(x)$, with U_t as before and $\beta_t = 2\log\left(|X_t| t^2 \frac{\pi^2}{6\delta}\right)$. We calibrate the discretization such that:

$$\forall x \in D,\ d(x, \pi_t(x)) \leq t^{-1/2}\,.$$

In the light of the previous proposition, we obtain a discretization error of $\mathcal{O}\big(\sqrt{(n\log t)/t}\big)$. Over T iterations, the sum of the discretization errors is then bounded by $\mathcal{O}\big(\sqrt{nT\log T}\big)$. Now, since $d(x,x') \leq \|x - x'\|_2$ for the above covariance functions, the number of points required in X_t does not exceed $\mathcal{O}\big(t^{n/2}\big)$, that is $\beta_T \leq \mathcal{O}\big(n\log T\big)$. Summing both the approximation error and the error of the UCB policy, and following the steps of Proposition 8, we obtain the state-of-the-art regret bounds where the $\log|D|$ factor in β_T is replaced by n the dimension of D.

Acknowledgment

The authors are grateful to David Ginsbourger for his participation to the session during the Journées MAS. F. Bachoc and D. Rullière thank the other authors of the preprint presented in the section 3, namely Clément Chevalier and Nicolas Durrande. Furthermore, they acknowledge support from the Chair in Applied Mathematics OQUAIDO, which Âţ gather partners in technological research (BRGM, CEA, IFPEN, IRSN, Safran, Storengy) and academia (Ecole Centrale de Lyon, Mines Saint-Etienne, University of Grenoble, University of Nice, University of Toulouse) on the topic of advanced methods for computer experiments. H. Maatouk is grateful to his co-authors Areski Cousin and Didier Rullière, for the manuscript "Kriging of Financial Term-Structures" and to his PhD advisor Xavier Bay. Finally, all the authors are grateful to the two anonymous referees for their feedback, which led to an improvement of the manuscript.

References

[1] P. Abrahamsen. A review of Gaussian random fields and correlation functions. Technical report, Norwegian computing center, 1997.

[2] R.J. Adler. *An introduction to continuity, extrema, and related topics for general Gaussian processes*. Hayward, CA: Institute of mathematical statistics, 1990.

[3] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.

[4] F. Bachoc. Cross validation and maximum likelihood estimations of hyper-parameters of Gaussian processes with model mispecification. *Computational Statistics and Data Analysis*, 66:55–69, 2013.

[5] F. Bachoc. Asymptotic analysis of the role of spatial sampling for covariance parameter estimation of Gaussian processes. *Journal of Multivariate Analysis*, 125:1–35, 2014.

[6] F. Bachoc. Asymptotic analysis of covariance parameter estimation for Gaussian processes in the misspecified case. *Accepted in Bernoulli*, 2016.

[7] F. Bachoc, K. Ammar, and J.M. Martinez. Improvement of code behavior in a design of experiments by metamodeling. *Nuclear science and engineering*, 183(3):387–406, 1016.

[8] F. Bachoc, G. Bois, J. Garnier, and J.M Martinez. Calibration and improved prediction of computer models by universal Kriging. *Nuclear Science and Engineering*, 176(1):81–97, 2014.

[9] F. Bachoc, N. Durrande, D. Rullière, and C. Chevalier. Some properties of nested Kriging predictors. *arXiv preprint arXiv:1707.05708*, 2017.

[10] P. Billingsley. *Probability and Measure, 3rd edition*. Wiley, New York, 1995.

[11] A. D. Bull. Convergence rates of efficient global optimization algorithms. *The Journal of Machine Learning Research*, 12:2879–2904, 2011.

[12] E. Contal. *Statistical learning approaches for global optimization*. PhD thesis, Université Paris-Saclay, 2016.

[13] E. Contal and N. Vayatis. Stochastic process bandits: Upper confidence bounds algorithms via generic chaining. *arXiv preprint arXiv:1602.04976*, 2016.

[14] A. Cousin, H. Maatouk, and D. Rullière. Kriging of financial term-structures. *European Journal of Operational Research*, 255(2):631 – 648, 2016.

[15] A. Cousin and I. Niang. On the range of admissible term-structures. *arXiv preprint arXiv:1404.0340*, 2014.

[16] H. Cramer and R. Leadbetter. *Stationary and related stochastic processes: sample function properties and their applications*. Wiley series in probability and mathematical statistics. Tracts on probability and statistics. 1967.

[17] N. Cressie and G. Johannesson. Fixed rank Kriging for very large spatial data sets. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(1):209–226, 2008.
[18] N. Cressie and S.N. Lahiri. The asymptotic distribution of REML estimators. *Journal of Multivariate Analysis*, 45:217–233, 1993.
[19] N. Cressie and S.N. Lahiri. Asymptotics for REML estimation of spatial covariance parameters. *Journal of Statistical Planning and Inference*, 50:327–341, 1996.
[20] S. Da Veiga and A. Marrel. Gaussian process modeling with inequality constraints. *Ann. Fac. Sci. Toulouse*, 21(2):529–555, 2012.
[21] M. Deisenroth and J. Ng. Distributed Gaussian processes. *Proceedings of the 32nd International Conference on Machine Learning, Lille, France. JMLR: W&CP volume 37*, 2015.
[22] R. Furrer, M. G. Genton, and D. Nychka. Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 2006.
[23] S. Golchi, D.R. Bingham, H. Chipman, and D.A. Campbell. Monotone Emulation of Computer Experiments. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):370–392, 2015.
[24] T. Hangelbroek, F. J. Narcowich, and J. D. Ward. Kernel approximation on manifolds I: bounding the lebesgue constant. *SIAM J. Math. Anal.*, 42:1732–1760, 2010.
[25] P. Hennig and C. J. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13:1809–1837, 2012.
[26] J. Hensman, N. Fusi, and N. D Lawrence. Gaussian processes for big data. *Uncertainty in Artificial Intelligence*, pages 282–290, 2013.
[27] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.
[28] C. Jidling, N. Wahlström, A. Wills, and T B. Schön. Linearly constrained Gaussian processes. *arXiv preprint arXiv:1703.00787*, 2017.
[29] D.R. Jones, M. Schonlau, and W.J. Welch. Efficient global optimization of expensive black box functions. *Journal of Global Optimization*, 13:455–492, 1998.
[30] C. G. Kaufman, D. Bingham, S. Habib, K. Heitmann, and J. A. Frieman. Efficient emulators of computer experiments using compactly supported correlation functions, with an application to cosmology. *The Annals of Applied Statistics*, 5(4):2470–2492, 2011.
[31] C. G. Kaufman, M. J. Schervish, and D. W. Nychka. Covariance tapering for likelihood-based estimation in large spatial data sets. *Journal of the American Statistical Association*, 103(484):1545–1555, 2008.
[32] W-L. Loh. Fixed-domain asymptotics for a subclass of Matérn-type Gaussian random fields. *The Annals of Statistics*, 33:2344–2394, 2005.
[33] W-L. Loh and T-K. Lam. Estimating structured correlation matrices in smooth Gaussian random field models. *The Annals of Statistics*, 28:880–904, 2000.
[34] H. Maatouk and X. Bay. *A New Rejection Sampling Method for Truncated Multivariate Gaussian Random Variables Restricted to Convex Sets*, volume 163, pages 521–530. In: Cools R and Nuyens R (Eds). Springer International Publishing, Cham, 2016.
[35] H. Maatouk and X. Bay. Gaussian process emulators for computer experiments with inequality constraints. *Mathematical Geosciences*, 49(5):1–26, 2017.
[36] H. Maatouk and Y. Richet. constrKriging. *R package available online at https://github.com/maatouk/constrKriging*, 2015.
[37] H. Maatouk, O. Roustant, and Y. Richet. Cross-Validation Estimations of Hyper-Parameters of Gaussian Processes with Inequality Constraints. *Procedia Environmental Sciences*, 27:38 – 44, 2015. Spatial Statistics conference 2015.
[38] K. V. Mardia and R. J. Marshall. Maximum likelihood estimation of models for residual covariance in spatial regression. *Biometrika*, 71:135–146, 1984.
[39] G. Matheron. *La Théorie des Variables Régionalisées et ses Applications*. Fasicule 5 in Les Cahiers du Centre de Morphologie Mathématique de Fontainebleau. Ecole Nationale Supérieure des Mines de Paris, 1970.
[40] J. B. Močkus. *Bayesian approach to global optimization*. Mathematics and its applications. Kluwer Academic, 1989.
[41] M. Morris. Gaussian surrogates for computer models with time-varying inputs and ouptputs. *Technometrics*, 54:42–50, 2012.
[42] C.R. Nelson and A.F. Siegel. Parsimonious modeling of yield curves. *The Journal of Business*, 60(4):pp. 473–489, 1987.
[43] T. Nickson, T. Gunter, C. Lloyd, M. A. Osborne, and S. Roberts. Blitzkriging: Kronecker-structured stochastic Gaussian processes. *arXiv preprint arXiv:1510.07965*, 2015.
[44] E. Parzen. *Stochastic processes*. Holden-Day series in probability and statistics. Holden-Day, San Francisco, London, Amsterdam, 1962.
[45] R. Paulo, G. Garcia-Donato, and J. Palomo. Calibration of computer models with multivariate output. *Computational Statistics and Data Analysis*, 56:3959–3974, 2012.
[46] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Cambridge, 2006.
[47] J. Riihimaki and A. Vehtari. Gaussian processes with monotonicity information. In *AISTATS*, volume 9 of *JMLR Proceedings*, pages 645–652. JMLR.org, 2010.
[48] H. Rue and L. Held. *Gaussian Markov random fields, Theory and applications*. Chapman & Hall, 2005.

[49] D. Rullière, N. Durrande, F. Bachoc, and C. Chevalier. Nested Kriging estimations for datasets with large number of observations. *Statistics and Computing*, 2017.
[50] J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4:409–423, 1989.
[51] T.J. Santner, B.J. Williams, and W.I. Notz. *The Design and Analysis of Computer Experiments*. Springer, New York, 2003.
[52] N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Information-theoretic regret bounds for Gaussian process optimization it setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.
[53] M. L. Stein. Statistical properties of covariance tapers. *Journal of Computational and Graphical Statistics*, 22(4):866–885, 2013.
[54] M. L. Stein. Limitations on low rank approximations for covariance matrices of spatial data. *Spatial Statistics*, 8:1–19, 2014.
[55] M. L. Stein, J. Chen, and M. Anitescu. Stochastic approximation of score functions for Gaussian processes. *The Annals of Applied Statistics*, 7(2):1162–1191, 2013.
[56] M.L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, New York, 1999.
[57] L. Svensson. Estimating and interpreting forward interest rates: Sweden 1992-1994. Technical report, National Bureau of Economic Research, 1994.
[58] V. Tresp. A bayesian committee machine. *Neural Computation*, 12(11):2719–2741, 2000.
[59] C. K. I. Williams and C. E. Rasmussen. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
[60] R. L. Winkler. The consensus of subjective probability distributions. *Management Science*, 15(2):B–61, 1968.
[61] R. L. Winkler. Combining probability distributions from dependent information sources. *Management Science*, 27(4):479–488, 1981.
[62] G. Xu and M. G. Genton. Tukey g-and-h random fields. *Journal of the American Statistical Association*, to appear, 2016.
[63] S. J. Yakowitz and F. Szidarovszky. A comparison of Kriging with nonparametric regression methods. *Journal of Multivariate Analysis*, 16:21–53, 1985.
[64] Z. Ying. Asymptotic properties of a maximum likelihood estimator with data from a Gaussian process. *Journal of Multivariate Analysis*, 36:280–296, 1991.
[65] Z. Ying. Maximum likelihood estimation of parameters under a spatial sampling scheme. *The Annals of Statistics*, 21:1567–1590, 1993.
[66] H. Zhang and Y. Wang. Kriging and cross validation for massive spatial data. *Environmetrics*, 21:290–304, 2010.