Metamodeling with Gaussian processes

–

Lecture notes and exercises

O. Roustant
INSA Toulouse, 2023

Contents

Chapter 1

Introduction

This textbook is a brief introduction to *Metamodeling*, also known as *Surrogate modeling*, with a focus on Gaussian process modeling and its application to prediction, optimization, inversion and uncertainty quantification. We have chosen to present a synthesis of selected notions, rather than being exhaustive or providing full developments. Fortunately, some of these developments will be addressed during the class, most often in the form of exercises. Other exercises aim at giving skills for problem solving. For complements, we provide a short list of reference books or journal publications, that can serve as entry points in the literature.

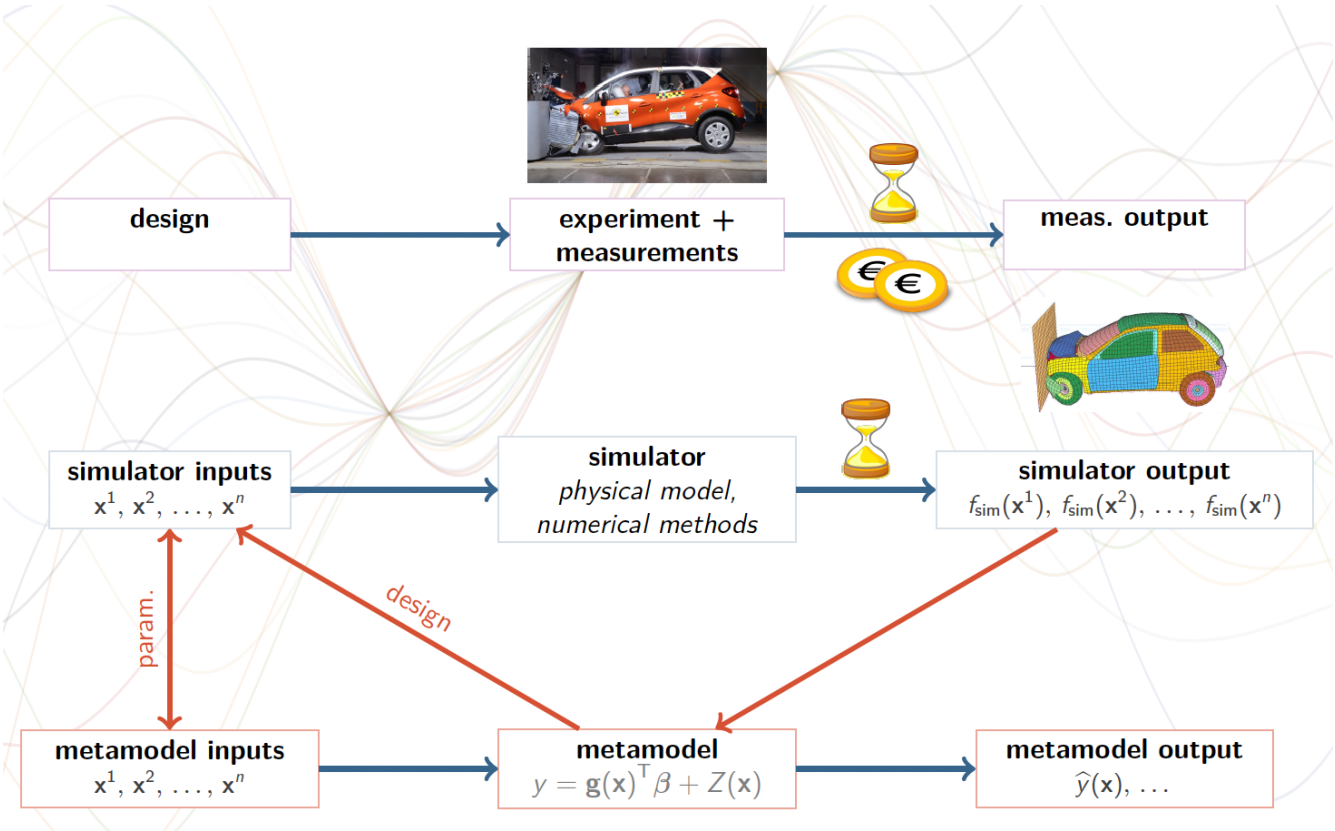

Figure 1.1: Illustration of metamodeling.

Figure 1.1 illustrates the principle of metamodeling for an industrial application. In this case study, the aim is to investigate the safety of a future model of vehicle. We have at our disposal a *numerical model* based on physical equations, noted f_{sim}. With this model, we can *virtually* investigate the safety of new prototypes, depending on several input variables. However, in many real problems, a single run of f_{sim} is time consuming, and the *computational budget is limited*. In particular, we cannot answer questions directly on f_{sim}, such as:

- to find the minimum of f_{sim} (optimization)
- to find the values of inputs such that f_{sim} is below some threshold (inversion)
- to quantify the influence of the inputs on the safety variable (sensitivity analysis)

A solution is to build a fast model, called *metamodel* or *surrogate model*, on which we can rely to solve the questions above. In optimization, the couple (numerical model, metamodel) is used in a sequential strategy. The metamodel is used to choose the new runs (*design of experiments*) of f_{sim}, hopefully in promising areas or in unvisited ones. Then the new runs can be used to update the metamodel, and improve its accuracy for the next step.

A metamodel can be any statistical model. We will focus here on Gaussian processes, which has several appealing features. It can be built with few data, and gives a measure of uncertainty in unvisited area. Furthermore, it is parameterized by two functions, and in particular a *kernel*, which provides a lot of flexibility and allows to incorporate expert or physical information.

Applications of metamodeling include the analysis of time-consuming numerical models (*computer experiments*) as well as the tuning and explicability of machine learning algorithms. Some reference books: Fang et al. (2005); Rasmussen and Williams (2006); Santner et al. (2018).

Finally, I would like to conclude this brief introduction by special thanks to X. Bay, M. Binois, Y. Deville and N. Durrande for several nice illustrations presented in this textbook.

Chapter 2

Gaussian processes

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space on which all the (real-valued) random variables will be defined. We denote by $L^2(\mathbb{P})$ the Hilbert space of square integrable random variables (defined on Ω).

2.1 Random processes

Definition. For a given set $\mathbb{X}$, a *random process* (RP) is the family of random variables $Y(x) : \Omega \to \mathbb{R}$, indexed by $x \in \mathbb{X}$. We will denote $Y := (Y(x))_{x \in \mathbb{X}}$.

Historically, the word *stochastic process* refers to temporal RP ($\mathbb{X} \subseteq \mathbb{R}$), whereas the word *random field* is often used for spatial RP ($\mathbb{X} \subseteq \mathbb{R}^p$, with $p \geq 2$). Notice however that $\mathbb{X}$ is not limited to a subset of $\mathbb{R}^p$ but can be a discrete set, a set of trees, manifolds, sets, probability distributions, etc.

Trajectory, realization or sample path. Let Y be a RP. For a fixed $w \in \Omega$, a *trajectory or realization or sample path* of Y is the function $x \mapsto Y(x)(w)$.

Second-order random process, mean, kernel. We say that Y is a *second-order RP* when all the random variables $Y(x)$ belong to $L^2(\mathbb{P})$. By Cauchy-Schwartz inequality, this implies that first moments (expectation) as well as second moments (covariances) are well-defined. We call:

- *mean function* or simply *mean* of Y the function $x \in \mathbb{X} \mapsto \mathbb{E}(Y(x))$.
- *covariance function* or *kernel* the function $(x, x') \in \mathbb{X} \times \mathbb{X} \mapsto \mathbb{C}\mathrm{ov}(Y(x), Y(x'))$.

Similarly, the *variance* of Y denotes the function $x \in \mathbb{X} \mapsto k(x, x) = \mathbb{V}\mathrm{ar}(Y(x))$.

Warning. The mean of Y is a function, whose value at x is the integral over all realizations of $Y(x)$: $\mathbb{E}(Y(x)) = \int_\Omega Y(x)(w) d\mathbb{P}(w)$. This has nothing to do with the *random* variable $\int_\mathbb{X} Y(x) d\mu(x)$, for some measure μ on $\mathbb{X}$, which is even not always defined.

Stationarity. Let $\mathbb{X}$ be a vector space, and let Y be a second-order RP on $\mathbb{X}$.

- Y is *strongly stationary* if for all locations $x_1, \ldots, x_n \in \mathbb{X}$, the law of $(Y(x_1+h), \ldots, Y(x_n+h))$ does not depend on h.

- Y is *weakly stationary* if for all locations $x_1, \dots, x_n \in \mathbb{X}$, the first two moments of the law of $(Y(x_1 + h), \dots, Y(x_n + h))$ do not depend on h. This is equivalent to say that the mean of Y is constant, and the kernel of Y depends only on the difference between locations:

$$\mathbb{E}(Y(x)) = m, \qquad k(x, x') = c(x - x')$$

with $m = \mathbb{E}(Y(x_0))$ (for some $x_0 \in \mathbb{X}$) and $c(h) = k(x, x - h)$.

Obviously, strong stationarity implies weak stationarity.

2.2 Gaussian processes

Definition. A random process Y defined on $\mathbb{X}$ is a *Gaussian process* (GP) if for all locations $x_1, \dots, x_n \in \mathbb{X}$ ($n \geq 1$), the random vector $(Y(x_1), \dots, Y(x_n))$ is a Gaussian vector. By definition of Gaussian vectors, the law of such random vectors is fully characterized by the mean m and the kernel k of Y. We will denote $Y \sim GP(m, k)$.

Direct consequences. Let Y be a GP. Then the properties of Gaussian vectors imply the following results:

- the notions of strong stationarity and weak stationarity coincide. Thus, we can omit 'strong' or 'weak', and simply speak of *stationary* GP.
- independence between $Y(x)$ and $Y(x')$ corresponds to a zero in the covariance matrix of $(Y(x), Y(x'))$. Similarly, conditional independence corresponds to zeros in precision matrices. This latter property is exploited in *Gaussian Markov random fields*.
- a GP is stable by linear mapping: formally, if Y is a GP, and L is a linear mapping operating on the sample paths of Y, then LY is a GP. This includes the case of linear differential operators. Examples and developments follow.
- a GP conditional on interpolation constraints $Y(x_i) = y_i, i = 1, \dots, n$ is still a GP. This is the basis of Gaussian process regression, developed in Section 4. By stability of GPs under linearity, this property is true for linear constraints (and not only interpolation ones).

Gaussian processes and linear operations If $Y \sim GP(0, k)$ and L is a linear function acting on the sample paths[1] of Y, then $LY \sim GP(0, k_L)$ where $k_L(s, t) = L_s L_t k(s, t)$. Here, the notation L_s (resp. L_t) means that we apply L on the function $s \mapsto k(s, t)$ (resp. $t \mapsto k(s, t)$).

The fact that LY is Gaussian can be proved by the linear combination property: since L is linear, a linear combination from LY can be rewritten as a linear combination from Y. The expression of k_L comes, formally, from the bilinearity of covariance (with slight abuses of notations):

$$\mathbb{C}\text{ov}(LY(s), LY(t)) = L_t(\mathbb{C}\text{ov}(LY(s), Y(t)) = L_s L_t \mathbb{C}\text{ov}(Y(s), Y(t))$$

[1] To be rigourous, one has to take care of the definition of LY. An example is given in the exercises.

An example of application is given by differential operators. In the 1-dimensional case, we have, formally that, if Y is a GP, then $(Y'(x))_{x\in\mathbb{X}}$ is a GP, with kernel [2]:

$$k_{Y'}(s,t) = \mathbb{C}\text{ov}\left(\frac{\partial Y(s)}{\partial s}, \frac{\partial Y(t)}{\partial t}\right) = \frac{\partial}{\partial t}\mathbb{C}\text{ov}\left(\frac{\partial Y(s)}{\partial s}, Y(t)\right) = \frac{\partial}{\partial s}\frac{\partial}{\partial t}\mathbb{C}\text{ov}(Y(s), Y(t)) = \frac{\partial^2 k}{\partial s \partial t}(s,t)$$

Simulation of a GP. A simulation of $Y \sim GP(m,k)$ is possible on a set of discrete locations $X = \{x_1, \ldots, x_n\}$. Indeed, then the law of $(Y(x_1), \ldots, Y(x_n))^\top$ is $\mathcal{N}(m(X), k(X,X))$ where

- $m(X)$ is the vector of size n whose component i is equal to $m(x_i)$
- $k(X,X)$ is the matrix of size n whose coefficient (i,j) is equal to $k(x_i,x_j)$

Obtaining a realization of Y at X, it is thus equivalent to simulating from $\mathcal{N}(m(X), k(X,X))$.

[2] When k is regular enough, it is indeed possible to give a sense to $Y'(x)$ and justify the whole computation.

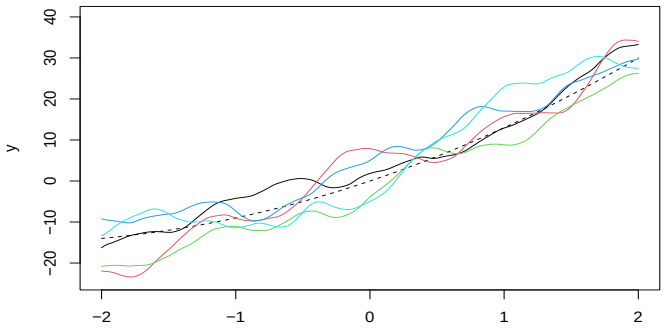

Figure 2.1: Five sample paths of a Gaussian process on $\mathbb{X} = [-2, 2]$ with an increasing mean (dotted line) and a Matérn kernel.

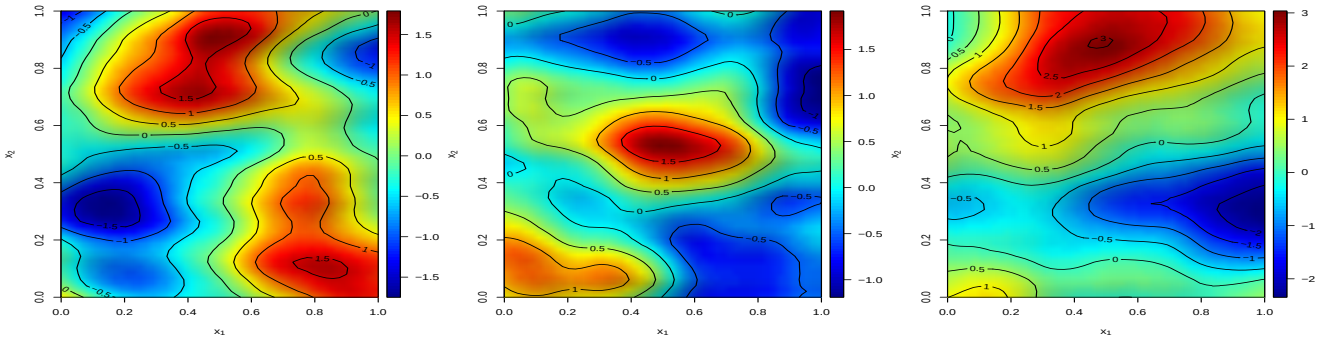

Figure 2.2: Realizations of a Gaussian process on $\mathbb{X} = [0, 1]^2$ with a kernel of the form $k(x, x'; \ell) = k_1(x_1, x_1'; 2\ell)k_2(x_2, x_2'; \ell)$ where k_1 is a one-dimensional Matérn kernel and ℓ is a parameter.

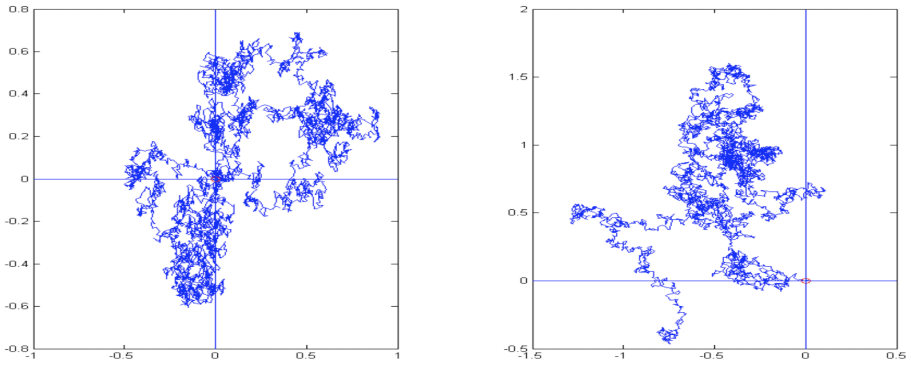

Figure 2.3: Two realizations of a 2D Brownian motion $Y(x) = \begin{pmatrix} Y_1(x) \\ Y_2(x) \end{pmatrix}$, whose components Y_1, Y_2 are independent, centered, non-stationary GPs on $\mathbb{X} = \mathbb{R}_+$ with kernel $k(x, x') = \min(x, x')$.

2.3 Exercises

Exercise 1 (Building a GP with invariance properties) *Let $Y_0 \sim GP(0, k_0)$ on $\mathbb{R}$. Define*

$$Y(x) := Y_0(x) - Y_0(-x)$$

Check that Y has odd *sample paths. Prove that Y is a GP by considering a linear combination extracted from Y. Check that Y is centered, and compute its kernel in function of k_0.*

Exercise 2 (Paving the way for GP regression with derivatives) *In this exercise we consider only formal computations, and assume that the mathematical objects involving derivatives can be defined properly, and that the corresponding operations are justified.*

Let Y be a centered GP on $\mathbb{R}$ with kernel k. Consider $Z = \begin{pmatrix} Y(x) \\ Y(x_1) \\ Y'(x_1) \end{pmatrix}$, with $x, x_1 \in \mathbb{R}$. Explain briefly why Z is a centered Gaussian vector, and compute its covariance matrix.

Exercise 3 (An example of physics-informed GP) *Let us consider the heat equation*

$$\frac{\partial u}{\partial t}(x,t) - \alpha \frac{\partial^2 u}{\partial x^2}(x,t) = 0$$

with initial condition $u(x,0) = \phi(x)$, with $\alpha > 0$. Denoting $S(x,t) = (4\pi\alpha t)^{-1/2} \exp\left(-\frac{x^2}{4\alpha t}\right)$, one can show by applying the Fourier transform to the equation that, under suitable conditions, the solution on $\mathbb{R}^2$ is:

$$u(x,t) = \int_{\mathbb{R}} S(x-y,t)\phi(y)dy. \tag{2.1}$$

We now consider that the initial function ϕ is unknown. As a prior information, we assume that $(\phi(x))_{x\in\mathbb{R}}$ is a centered Gaussian process with kernel k_ϕ. What can you say of the mapping $L : \phi \mapsto \int_{\mathbb{R}} S(x-y,t)\phi(y)dy$? Then explain briefly why $(u(x,t))_{(x,t)\in\mathbb{R}^2}$, given by Equation (2.1), should be a Gaussian process on $\mathbb{R}^2$. Compute formally its mean and its kernel in function of k_ϕ and S (assuming that all the integrals are well-defined).

Remark. *The advantage of using the Gaussian process regression rather than the solution of the heat equation is not clear in general, since the computation of the kernel seems more complicate than the solution itself (Equation 2.1). Fortunately, for some k_ϕ, the kernel k_u is given explicitely, which gives a clear advantage to GP regression. For instance, when $k_\phi(y,y') = \exp\left(-\frac{1}{2}\frac{(y-y')^2}{\theta^2}\right)$ is the square exponential kernel, then k_u has the form :*

$$k_u\left(\begin{pmatrix} x \\ t \end{pmatrix}, \begin{pmatrix} x' \\ t' \end{pmatrix}\right) = \frac{\sigma_u^2}{\sqrt{2\pi}\sqrt{\theta^2 + 2\alpha(t+t')}} \exp\left(-\frac{1}{2}\frac{(x-x')^2}{\theta^2 + 2\alpha(t+t')}\right).$$

Chapter 3

Kernels and reproducing kernel Hilbert spaces

3.1 Kernels

A *kernel* was defined as the covariance function of a random process, and thus quantifies the "proximity" between the output values $Y(x_1), Y(x_2)$ at two locations x_1, x_2. Mathematically, kernels extend the notion of positive semi-definite (psd) matrices to the continuous setting, and are equivalent to *psd functions*, as defined now.

Positive semi-definite functions. A function $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ is said to be *positive semi-definite* (psd) if for all finite set $X = \{x_1, \dots, x_n\}$, the matrix $k(X, X) = (k(x_i, x_j))_{1 \leq i,j \leq n}$ is psd. Equivalently, for all $n \geq 1$, all $x_1, \dots, x_n \in \mathbb{X}$, and all $\alpha_1, \dots, \alpha_n \in \mathbb{R}$, k is psd if and only if:

$$\sum_{i,j} \alpha_i \alpha_j k(x_i, x_j) \geq 0$$

Similarly the notion of (strict) positive definiteness can be defined, and is connected to the notion of Tchebychev systems (Karlin and Studden, 1966). However, it is often preferred to add a small noise to the data. Then the associated covariance matrices are $k(X, X) + \tau^2 I_n$, with $\tau > 0$ (see the next chapter), which are invertible even if $k(X, X)$ is only psd.

Kernels, covariance functions and psd functions. An important result is that the notions of covariance function and psd functions coincide: if k is the covariance of a (second-order) random process, then k is a psd function. Conversely, if k is a psd, we can build a second-order centered RP with covariance k; moreover there is a unique[1] centered GP with kernel k. Thus, the word *kernel* will denote either a covariance function or a psd function.

All kernels are scalar products in a feature space. If k is a kernel, then there exists a Hilbert space $(\mathcal{H}, \langle ., . \rangle)$, called *feature space* in machine learning, and a function $\Phi : \mathbb{X} \to \mathcal{H}$, often called *kernel embedding* in this context, such that $k(x, x') = \langle \Phi(x), \Phi(x') \rangle$ for all $x, x' \in \mathbb{X}$.

[1]There exist several centered GPs with kernel k: for instance, if $Y \sim GP(0, k)$, then $-Y \sim GP(0, k)$. But all of them have the same law since their law depends on the expectation, here the null function, and the covariance function, here k. Unicity must be understood in this sense, i.e. with a common unique law.

Indeed, we can choose for $\mathcal{H}$ the Hilbert space $L^2(\mathbb{P})$ and $\Phi(x) = Z(x)$, where Z is a centered RP with kernel k. Note that the representation is not unique: we can also use for $\mathcal{H}$ the RKHS associated to k, and $\Phi(x) = k(x, .)$ (see Section 3.2).
This shows that, up to a mapping all kernels are scalar products. The scalar product kernel is thus the prototype of kernels. However, this representation is not useful to provide a new kernel because it involves the kernel itself! In machine learning, this property is known as "kernel trick", since it allows creating non-linear methods with linear ones thanks to a mapping Φ that does not need to be explicit: only the kernel k is required.

General operations on kernels. Here are operations on kernels, valid for all set $\mathbb{X}$.

- If k is a kernel, then $\sigma^2 k$ is a kernel for all $\sigma \in \mathbb{R}$.
- (Stability by sum) If k_1, k_2 are two kernels on $\mathbb{X}^2$, then $k_1 + k_2$ is a kernel. Similarly if k_1, k_2 are kernels on $(\mathbb{X}_1)^2, (\mathbb{X}_2)^2$ respectively, the tensor sum[2] $k_1 \oplus k_2$ is a kernel on $(\mathbb{X}_1 \times \mathbb{X}_2)^2$.
- (Stability by product) If k_1, k_2 are two kernels on $\mathbb{X}^2$, then $k_1 k_2$ is a kernel. Similarly, if k_1, k_2 are kernels on $(\mathbb{X}_1)^2, (\mathbb{X}_2)^2$ respectively, the tensor product $k_1 \otimes k_2$ is a kernel on $(\mathbb{X}_1 \times \mathbb{X}_2)^2$.
- (*warping* or *embedding*) If k is a kernel on $\mathbb{X}^2$, and $f : \mathbb{U} \to \mathbb{X}$ is a function, then $k_f(u, u') := k(f(u), f(u'))$ is a kernel on $\mathbb{U}^2$.

Comments. We see that the set of kernels is a convex cone, stable by multiplication. The tensor product operation is widely used to define kernels on d-dimensional spaces from 1-dimensional kernels. Notice that the warping property is valid for *any* function f (in particular f may be non bijective). This property is very useful to transport a kernel defined on a known space to a target space, or to deal with non-stationarities.

We now provide other properties, valid for some set $\mathbb{X}$.

Mercer representation of kernels. It is well known that a real psd matrix admits a spectral (or eigen) decomposition. This result can be extended to psd functions at the price of additional assumptions. It is known as *Mercer representation*. For instance (Steinwart and Christmann, 2008, p. 150), assume that $\mathbb{X}$ is a *compact* metric space, and k is *continuous* on $\mathbb{X} \times \mathbb{X}$. Let ν be a finite measure supported by $\mathbb{X}$. Then there exists a Hilbert basis $(\phi_n)_{n\geq 0}$ of $L^2(\mathbb{X}, \nu) = \{f : \mathbb{X} \to \mathbb{R}, \, s.t. \, \int_T f(x)^2 d\nu(x) < +\infty\}$ (eigenfunctions) and a sequence of non-negative real numbers $(\lambda_n)_{n\geq 0}$ (eigenvalues) tending to zero, with $\lambda_0 \geq \lambda_1 \geq ...$, such that

$$k(x, x') = \sum_{n\geq 0} \lambda_n \phi_n(x) \phi_n(x')$$

where the convergence is uniform on $\mathbb{X} \times \mathbb{X}$. The $\phi_n' s$ are eigenfunctions of the Hilbert-Schmidt operator defined on $L^2(\mathbb{X}, \nu)$ by

$$Tf(x) = \int_T k(x, x') f(x') d\nu(x')$$

and thus verify $T\phi_n(x) = \lambda_n \phi_n(x)$ for all $n \geq 0$.

[2]The tensor sum is $k_1 \oplus k_2 \left(\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \begin{pmatrix} x_1' \\ x_2' \end{pmatrix} \right) = k_1(x_1, x_1') + k_2(x_2, x_2')$. Similar definition for the product.

Kernel name	Kernel form	Spectral measure
cosine	$\cos(2\pi h)$	Dirac δ_1
sinc	$\frac{\sin(\pi h)}{\pi h}$	Uniform
Squared exponential	$k(h) = \exp\left(-\frac{1}{2}\frac{h^2}{\ell^2}\right)$	Gaussian
Exponential	$\exp\left(-\frac{\lvert h\rvert}{\ell}\right)$	Student $t_{1/2}$
Matérn 3/2	$\left(1+\sqrt{3}\frac{\lvert h\rvert}{\ell}\right)\exp\left(-\sqrt{3}\frac{\lvert h\rvert}{\ell}\right)$	Student $t_{3/2}$
Matérn 5/2	$\left(1+\sqrt{5}\frac{\lvert h\rvert}{\ell}+\frac{5}{3}\frac{h^2}{\ell^2}\right)\exp\left(-\sqrt{5}\frac{\lvert h\rvert}{\ell}\right)$	Student $t_{5/2}$

Table 3.1: Examples of kernels of 1-dimensional stationary GPs on $\mathbb{X} = \mathbb{R}$. Here $h = x - x'$.

A kernel of a stationary GP on a general Hilbert space. If $\mathbb{X} = \mathcal{H}$ is a Hilbert space, with norm $\|.\|$, then the function

$$k(x, x') = e^{-\|x-x'\|^2} \tag{3.1}$$

is a kernel, called *squared exponential* kernel (or, which may be confusing, *Gaussian* kernel). Furthermore, the converse is true: if k is a kernel defined by (3.1), then the norm $\|.\|$ is Euclidean[3].

Notice that in practice, this kernel is often slightly modified in order to include scale parameters: $k(x, x'; \sigma^2, \ell) = \sigma^2 \exp\left(-\frac{\|x-x'\|^2}{\ell^2}\right)$. The result is unchanged.

Kernels of stationary GPs on $\mathbb{R}^d$ (Bochner's theorem). The kernel of a real-valued stationary GP on $\mathbb{R}^d$ is the Fourier transform of a probability distribution

$$k(x, x') = \int_{\mathbb{R}^d} \cos(2\pi\langle x - x', t\rangle) d\mu(t) \tag{3.2}$$

where $\langle ., .\rangle$ is the usual scalar product on $\mathbb{R}^d$. The probability measure μ is called *spectral measure*. Bochner's theorem thus provides a characterization of stationary GPs, parameterized by their spectral measure. Choosing a spectral measure can lead to explicit expressions of kernels. Examples of 1-dimensional kernels built this way are given in Table 3.1.

More on kernels. Theory on psd functions can be found in the books of Wendland (2004) and Berg et al. (1984). Complementary developments, in a machine learning perspective, are presented in (Rasmussen and Williams, 2006).

3.2 RKHS

RKHS have been developed in the fifty's in the seminal work of Aronszajn (1950). We refer to (Berlinet and Thomas-Agnan, 2004) for a recent presentation.

[3] a norm is Euclidean if it is associated to a scalar product: $\|x\|^2 = \langle x, x\rangle$

Definition. A reproducing kernel Hilbert space (RKHS) is a Hilbert space of real-valued functions[4] $\mathcal{H}$ defined on a set $\mathbb{X}$, for which evaluations $h \mapsto h(x)$ are continuous, for all x in $\mathbb{X}$.
By Riesz theorem, there exists a unique $k_x \in \mathcal{H}$ such that for all x in $\mathbb{X}$:

$$\langle h, k_x \rangle = h(x) \qquad \textit{(reproducing property)}$$

Now define the function $k : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ by $k(x, y) = k_y(x)$ and denote $k(x, .) : y \mapsto k(x, y)$. Replacing h by k_y in the equality above, we obtain

$$k(x, y) = \langle k_y, k_x \rangle = \langle k(y, .), k(x, .) \rangle$$

The word *reproducing* means that the value of $k(x, y)$ (resp. $f(x)$) is obtained as a scaler product from itself by using the functions $k(x, .)$ and $k(y, .)$ (resp. $k(x, .)$ and f).

Equivalence between kernels and RKHS (Moore-Aronszajn theorem). One can check that the function k defined above is psd. Thus, if $\mathcal{H}$ is a RKHS, we obtain a kernel, called *reproducing kernel*. Conversely, if k is a kernel, one can construct a unique RKHS $\mathcal{H}_k$ with reproducing kernel k. This RKHS is given by

$$\mathcal{H}_k = \overline{\text{span}(k(x, .), x \in \mathbb{X})}$$

with inner product defined on the $k(x, .)$'s by $\langle k(x, .), k(y, .) \rangle := k(x, y)$, and extended to $\mathcal{H}_k$ by linearity and continuity. The rigorous proof constitutes the *Moore-Aronszajn theorem*.
In summary, there is an equivalence between RKHS and kernels.

Equivalence between RKHS and random processes (Loève isometry). RKHS and random processes are strongly connected by the so-called *Loève representation theorem*. As $\mathcal{H}_k$ is spanned by the $k(x, .)$, the idea is to consider $\bar{\mathcal{L}}(Z) = \overline{\text{span}(Z(t),\ t \in \mathbb{X})}$, for a centered second order random process $Z = (Z(x))_{x \in \mathbb{X}}$ with covariance function k. As a closed subspace of $L^2(\mathbb{P})$, $\bar{\mathcal{L}}(Z)$ is a Hilbert space. Furthermore, $\langle k(x, .),\ k(y, .) \rangle = k(x, y) = \langle Z(x),\ Z(y) \rangle$, and it results that $\bar{\mathcal{L}}(Z)$ is isometric to $\mathcal{H}_k$ through the map defined on the $k(x, .)$'s by

$$\phi : \begin{array}{c} \mathcal{H}_k \to \bar{\mathcal{L}}(Z) \\ k(x, .) \mapsto Z(x) \end{array}$$

and extended by linearity and continuity. This important result serves as a dictionary to translate a functional problem into a probabilistic one, and vice-versa.

Examples. Finite-dimensional Hilbert spaces are all RKHS. Furthermore, if $(\varphi_n)_{n=1}^N$ is an orthonormal basis of $\mathcal{H}$, then its kernel is written $k(x, x') = \sum_{n=1}^N \varphi_n(x)\varphi_n(x')$ for all x, x' in $\mathbb{X}$.
Infinite dimensional RKHS may contain functions with a minimal *regularity*. When $\mathbb{X} = \mathbb{R}^d$, the Sobolev space $H^m(\mathbb{X})$ (with its usual norm) is a RKHS if and only if $m > d/2$. In particular, for $d = 1$, we see that $H^1(\mathbb{R}), H^2(\mathbb{R}), \ldots$ are RKHS. Furthermore, if $\mathbb{X}$ is a bounded interval, $L^2(\mathbb{X})$ is *not* a RKHS. A list of explicit kernels of Sobolev spaces with various norms, is given at the end of (Berlinet and Thomas-Agnan, 2004).

[4] We focus here on real-valued functions, but the theory is similar for complex-valued ones.

3.3 Exercises

Exercise 4 (A covariance function is a psd function) *Prove that if k is a covariance function of a random process, then k is a psd function.*

Exercise 5 (General operations on kernels) *Prove the results of the paragraph* General operations on kernels. *Furthermore, let Y, Y_1, Y_2 be centered GPs associated to k, k_1, k_2 respectively. We can assume that Y_1 and Y_2 are independent. Find a centered random process Z corresponding to the target kernel (sum of kernels, product of kernels, warped kernel), built in function of (some of) Y, Y_1, Y_2. Is Z a GP? Summarize your findings in the table below.*

kernel	associated RP	is this RP a GP?
$\sigma^2 k$		
$k_1 + k_2$		
$k_1 k_2$		
k_f		

Exercise 6 (Bochner's theorem, direct sense) *Using that for all u in $\mathbb{R}$, $\cos(u) = Re\,(e^{iu})$, prove that the function $(x, x') \mapsto \cos(2\pi\langle x - x', t\rangle)$ is psd. Deduce that k defined by (3.2) is a kernel.*

Exercise 7 (The square exponential kernel on a Hilbert space) *Here we propose a simple proof that if $(\mathcal{H}, \langle ., . \rangle)$ is a Hilbert space, then $k(x, x') = e^{-\|x - x'\|^2}$ is a kernel, with $\|x\|^2 = \langle x, x\rangle$.*

1. *Show that $k(x, x')$ has the form $k(x, x') = g(x)g(x')e^{2\langle x, x'\rangle}$*
2. *Show that $(x, x') \mapsto g(x)g(x')$ is a kernel (for any function g).*
3. *Show that if k_0 is a kernel, then e^{k_0} is a kernel. (Hint: Use the series expansion of* exp*)*
4. *Conclude.*

Exercise 8 (A kernel on sets of $\mathbb{R}^d$) *Let $\mathbb{X}$ the set of Lebesgue measurable sets A of $\mathbb{R}^d$. For $A \in \mathbb{X}$, denote by $Vol(A) = \int 1_A(x)dx$ its volume. Let $\mathcal{H} = L^2(\mathbb{R}^d)$ be the Hilbert space of square integrable functions on $\mathbb{R}^d$ with its usual scalar product $\langle f, g\rangle = \int_{\mathbb{R}^d} f(x)g(x)dx$. Prove (in two lines!) that*

$$k(A, B) = e^{-\|1_A - 1_B\|^2}$$

is a kernel on $\mathbb{X} \times \mathbb{X}$. Check that $k(A, B) = e^{-Vol(A \triangle B)}$, where $A \triangle B = (A \cup B) \backslash (A \cap B)$ is the symmetric difference of A, B. For an example of application, see Fellmann et al. (2023).

Exercise 9 (Kernels of finite-dimensional RKHS) *Let $\mathcal{H}$ be a finite dimensional Hilbert space with an orthonormal basis $(\varphi_n)_{n=1}^N$. Prove that $\mathcal{H}$ is a RKHS with kernel given by $k(x, x') = \sum_{n=1}^N \varphi_n(x)\varphi_n(x')$ for all x, x' in $\mathbb{X}$.*
Now consider a (general) basis of functions $(\psi_n)_{n=1}^N$, with Gram matrix $G = (\langle \psi_i, \psi_j\rangle)_{1 \le i,j \le n}$. Deduce from the orthornormal case that $k(x, x') = \psi^\top G^{-1} \psi$, with $\psi = (\psi_1, \ldots, \psi_n)^\top$.

Exercise 10 (An example of Sobolev-type RKHS) *Consider the anchored Sobolev space:*

$$H_0^1(0,1) = \{h \in L^2(0,1),\, h(0) = 0 \text{ and } h' \in L^2(0,1)\}$$

with norm $\|h\|^2 = \int_0^1 h'^2(x)dx$. *Here the derivative is defined in the weak sense:* h *admits a weak derivative if there exists a function noted* h' *such that for all* C^∞ *compactly supported function* g *(test function) we have* $\int_0^1 h(x)g'(x)dx = -\int_0^1 h'(x)g(x)dx$. *Furthermore, recall that in one dimension, the elements of Sobolev spaces are absolutely continuous functions, thus verify*

$$h(x) = h(0) + \int_0^x h'(y)dy$$

Compare this property with the reproducing property of a RKHS, and deduce that $H_0^1(0,1)$ *is a RKHS with kernel* $k(x,x') = \min(x,x')$. *What is the associated Gaussian process?*

Chapter 4

Gaussian process regression

In this section, we have a set of observations $y_1, \ldots, y_n$ at associated locations $x_1, \ldots, x_n$ in some space $\mathbb{X}$, coming from an unknown function f_{sim}. The question is to build a function (metamodel) f that interpolates the data, i.e. $f(x_i) = y_i$ $(i = 1, \ldots, n)$, or, at least approximates the data, if the observations are noisy. Among the numerous existing techniques, we focus on the Gaussian process regression, which has two main advantages in the metamodeling context. Firstly, it is a probabilistic method, which allows quantifying uncertainty in unvisited area. Secondly, it is parameterized by two functions (mean and kernel), which gives a lot of flexibility and allows incorporating expert and physical knowledge.

The roots of GP regression are geostatistics, with the work of Krige (1951), further developed by Matheron (1963), in dimension 2 or 3. Its extension to higher dimensions has been done at the end of the 80's, motivated by questions arisen in analyzing big computer codes (Sacks et al., 1989). Interestingly, GP regression can be interpreted as a functional approximation problem in RKHS. Knowing the three facets of GP regression is useful. For instance, the Universal Kriging variance formula coming from geostatistics maybe a good tradeoff for uncertainty quantification, as it partially accounts for parameter uncertainty without requiring a time-consuming Bayesian inference technique. And the RKHS approach may be the most convenient way to perform GP regression when f_{sim} is a non-linear partial differential equation.

4.1 The Gaussian process approach

The idea of GP regression is to assume that the unknown function f_{sim} is a sample path of a Gaussian process $Y \sim GP(m, k)$. The approximation that we want to build is then obtained by conditioning on the observations $Y(x_i) = y_i, i = 1, \ldots, n$. Using the properties of Gaussian vectors, this conditional process is still a GP, with closed-form expressions. We know distinguish two cases.

Noise-free observations. The conditional process $Y(x)$ knowing $Y(x_i) = y_i$ $(i = 1, \ldots, n)$ is a GP with mean m_{c} and kernel k_{c}, given by:

$$m_{\text{c}}(x) = m(x) + k(x, X)k(X, X)^{-1}(y - m(X)) \tag{4.1}$$

$$k_{\text{c}}(x, x') = k(x, x') - k(x, X)k(X, X)^{-1}k(X, x') \tag{4.2}$$

where $y = (y_1, \ldots, y_n)^\top$, $m(X) = (m(x_1), \ldots, m(x_n))^\top$, $k(x, X) = (k(x_1, x), \ldots, k(x_n, x))$, $k(X, x) = k(x, X)^\top$ and $k(X, X) = (k(x_i, x_j))_{1 \leq i,j \leq n}$.

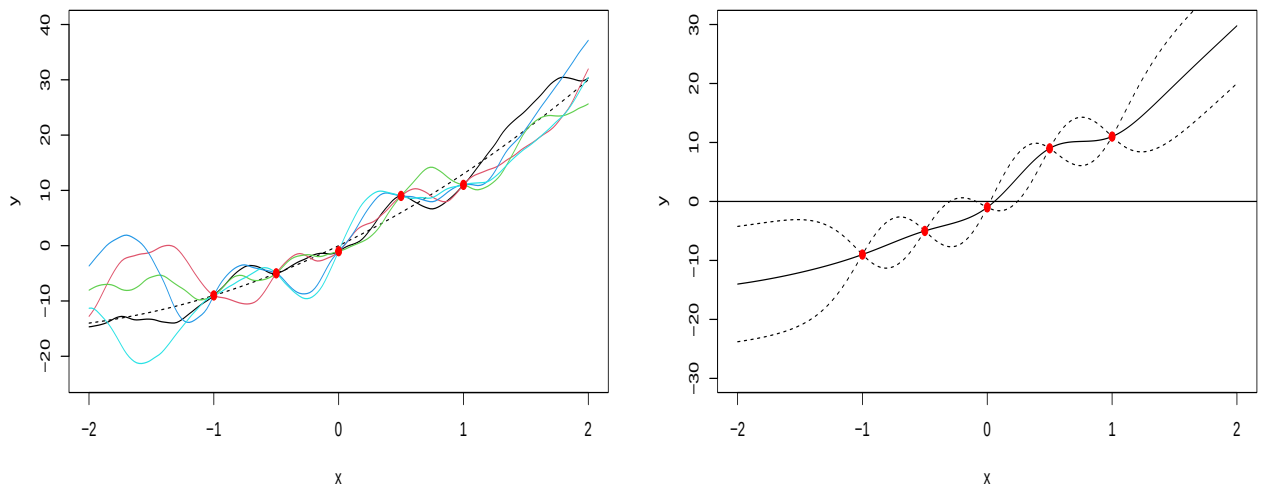

Figure 4.1: Gaussian process of Figure 2.2 conditional on $\{Y(x_i) = y_i, i = 1, \ldots, n\}$ for $n = 5$ points. Left: simulations. Right: mean $m_c(x)$ and 95% prediction intervals $\left[m_c(x) \pm 1.96\sqrt{k_c(x,x)}\right]$.

As expected, one can check that m_c interpolates the data: $m_c(x_i) = y_i$, and the uncertainty is zero at design points: $k_c(x_i, x_i) = 0$; notice that we also have: $k_c(x_i, x) = 0$ for all $x \in \mathbb{X}$. Furthermore, we inherit properties from Gaussian vector conditioning: m_c is affine with respect to the observations y, and k_c does not depend on them.

Noisy observations. When the observations are noisy, we can write

$$y_i = Y(x_i) + \epsilon_i$$

Let us further assume that the ϵ_i's are $\mathcal{N}(0, \tau_i^2)$, mutually independent, and independent of Y. The prediction is then a *filtering* problem: to predict the underlying value $Y(x)$ conditional on noisy observations $y_1, \ldots, y_n$. Then, the conditional process $Y(x)$ knowing $Y(x_i) + \epsilon_i = y_i$ $(i = 1, \ldots, n)$ is a GP with mean m_c and kernel k_c, given by:

$$\begin{aligned} m_c(x) &= m(x) + k(x, X)\left[k(X, X) + \Delta\right]^{-1}(y - m(X)) \quad (4.3) \\ k_c(x, x') &= k(x, x') - k(x, X)\left[k(X, X) + \Delta\right]^{-1} k(X, x') \quad (4.4) \end{aligned}$$

where Δ is the diagonal matrix with $\Delta_{i,i} = \tau_i^2$ $(i = 1, \ldots, n)$. In other words, the formula are similar to the noise-free case, the only difference being that $k(X, X)$ is replaced by $k(X, X) + \Delta$. Notice however that m_c is no more interpolating the data, and the uncertainty at observed locations is not zero, due to the presence of noise.

4.2 The geostastical approach: Kriging

Simple Kriging. Let Y be a centered[1] second-order random process process. In geostatistics, the prediction of $Y(x)$ knowing $Y(x_1), \ldots, Y(x_n)$ is computed by the *Best Linear Unbiased Predictor (BLUP)*. We look for a predictor defined linearly on the observations at locations $x_1, \ldots, x_n$

$$\hat{Y}(x) := w_0(x) + w_1(x)Y(x_1) + \cdots + w_n(x)Y(x_n)$$

where $w(x) := (w_0(x), \ldots, w_n(x)) \in \mathbb{R}^{n+1}$. The aim is to find $w(x)$ that minimizes $\text{MSE} := \mathbb{E}\left[\left(Y(x) - \hat{Y}(x)\right)^2\right]$, subject to $\mathbb{E}\left[\hat{Y}(x)\right] = \mathbb{E}[Y(x)]$. This interpolation method is called *Kriging*, in the honor of Daniel Krige, who used this technique to predict the gold content in mines.

By definition we recognize that the BLUP is equal to the orthogonal projection of $Y(x)$ onto $\text{span}\{1, Y(x_1), \ldots, Y(x_n)\}$, or equivalently to the linear conditional expectation

$$\hat{Y}(x) = \mathbb{E}_L\left[Y(x) | Y(x_1), \ldots, Y(x_n)\right]$$

Now, for Gaussian vectors, the (non-linear) regression coincides with the linear one. Thus, we get exactly the same formula than for GPR, called *simple Kriging* formula:

$$\begin{aligned} m_{SK}(x) &= \mathbb{E}\left[Y(x) | \{Y(x_i) = y_i, i = 1, \ldots, n\}\right] = m_{\text{c}}(x) && (4.5) \\ s^2_{SK}(x) &= \mathbb{V}\text{ar}\left[Y(x) | \{Y(x_i) = y_i, i = 1, \ldots, n\}\right] = k_{\text{c}}(x, x) && (4.6) \end{aligned}$$

Here $m_{SK}(x)$ is equal to $\hat{Y}(x)$ when $Y(x_i)$ is fixed to y_i, and $s^2_{SK}(x)$ is equal to the value of $\text{MSE}(x)$ at the optimum of $w(x)$.

Thus the geostatistical framework gives a natural extension of GP regression in a non-Gaussian setting. However, then the interpretation is in term of best *linear* predictor, and the conditional law of $Y(x) | \{Y(x_i) = y_i, i = 1, \ldots, n\}$ is known only through its first two moments.

Ordinary and Universal Kriging. In simple Kriging, the parameters of Y are assumed to be known. In Universal Kriging, we assume that $Y(x)$ has a mean of the form $m(x) = f(x)^\top \beta$, where $f(x)$ is a vector of known functions, and β is a vector of unknown parameters. Then, it can be shown (Cressie, 1992) that the BLUP has the same form as for Simple Kriging, up to centering by $f(x)^\top \hat{\beta}$, where $\hat{\beta}$ is the general least square (GLS) estimate of β, $\hat{\beta} = (F^\top K^{-1} F)^{-1} F^\top y$ (with $K = k(X, X)$). The Kriging variance is greater than SK variance. They are given by:

$$\begin{aligned} m_{\text{UK}}(x) &= f(x)^\top \hat{\beta} + k(x, X)k(X, X)^{-1}(y - f(x)^\top \hat{\beta}) && (4.7) \\ s^2_{\text{UK}}(x) &= s^2_{\text{SK}}(x) + (f(x)^\top - k(x, X)K^{-1}F)^\top (F^\top K^{-1} F)^{-1}(f(x)^\top - k(x, X)K^{-1}F) && (4.8) \end{aligned}$$

The expressions of $m_{\text{UK}}(x)$ and $s^2_{\text{UK}}(x)$ are called *Universal Kriging* (UK) formulas. When the mean function is constant, $m(x) = \beta \in \mathbb{R}$, they are called *Ordinary Kriging* (OK) formulas.

The UK formulas have a Bayesian interpretation, detailed in Section 4.4. They may be preferred to SK formulas for uncertainty quantification, as they account for trend parameters uncertainty.

[1] The approach is immediately adapted to the case where Y has a known mean, by considering $Y(x) - \mathbb{E}(Y(x))$.

4.3 The functional approach: approximation in RKHS.

The Gaussian process regression can be reinterpreted as an approximation problem in RKHS. Given a kernel k, let $\mathcal{H}$ be the corresponding RKHS. Then, GPR for noisy observations actually corresponds to the penalized problem

$$\min_{h\in\mathcal{H}} J_\lambda(h), \qquad \text{with} \qquad J_\lambda(h) = \sum_{1\leq i\leq n} (y_i - h(x_i))^2 + \lambda \|h\|^2_{\mathcal{H}}$$

where $x_1, \ldots, x_n$ are a set of observations, and $y_1, \ldots, y_n$ the corresponding response values. The parameter λ is a positive real number, interpreted as a *regularization* parameter.

Reduction to finite dimension (representer theorem) It happens that, because $\mathcal{H}$ is chosen as a RKHS, this infinite dimensional optimization problem collapses to finite dimension. Indeed, let F be the finite dimensional subspace of $\mathcal{H}$ spanned by $k(x_1, .), \ldots, k(x_n, .)$. Using the decomposition $\mathcal{H} = F + F^\perp$, we can write $h = f + g$ with $f \in F$ and $g \in F^\perp$. Now, applying the reproducing property leads to $g(x_i) = \langle g, k(x_i, .)\rangle = 0$, and we obtain:

$$J_\lambda(h) = J_\lambda(f) + \lambda \|g\|^2_{\mathcal{H}}$$

The optimization problem is then separable and can be done independently along f and g, which implies $g = 0$. Finally, the optimum is obtained for $h = f \in F$. This strong result is an example of the *representer theorem* (Kimeldorf and Wahba, 1970).

Resolution of the problem. Link with splines and GP regression. Now, as $h \in F$, we can write $h(x) = k(x, X)\alpha$, where $k(x, X) = (k(x_1, x), \ldots, k(x_n, x))$ and $\alpha = (\alpha_1, \ldots, \alpha_n)^\top \in \mathbb{R}^n$. Denoting $K = (k(x_i, x_j))_{1\leq i,j\leq n}$, we obtain after some algebra that the criterion $J_\lambda(h)$ can be written as a second order polynomial in α:

$$J_\lambda(f) = (y - K\alpha)^T(y - K\alpha) + \lambda\alpha^T K\alpha \tag{4.9}$$

where $y = (y_1, \ldots, y_n)^\top$. Minimizing with respect to α leads to:

$$h(x) = k(x, X)(K + \lambda I_n)^{-1} y \tag{4.10}$$

This expression corresponds to the formula for *smoothing splines*, and is also equal to the conditional expectation in GP regression with noisy observations, in the probabilistic framework. When λ tends to zero, we get the formula for interpolation splines and usual Kriging prediction, which can be viewed as a minimal norm interpolator (see Exercise 15).

4.4 Hyperparameters inference

In applications, we consider GPs with a linear trend, which can be written in the form

$$Y(x) = m(x) + Z(x)$$

where:

- $m(x) = \beta_1 f_1(x) + \cdots + \beta_p f_p(x)$ is a linear trend (the f_i's are known functions)
- Z is a centered GP with kernel $k(x, y; \Theta)$.

Here β and Θ are vectors of unknown parameters, often called *hyperparameters*. We now describe two standard ways to estimate these hyperparameters.

Maximum likelihood. Maximum likelihood estimation (MLE) consists in finding the parameters that maximize the density of probability at observations. Here, the law of $(Y(x_1), \ldots, Y(x_n))$ is $\mathcal{N}(F\beta; k(X, X; \Theta))$, where $X = \{x_1, \ldots, x_n\}$ and F is the $n \times p$ matrix whose row i contains $f_1(x_i), \ldots, f_p(x_i)$. The pdf value at observations $y = (y_1, \ldots, y_n)^\top$ is written:

$$L(y; \beta, \Theta) = \frac{1}{(2\pi)^{n/2} |k(X, X; \Theta)|^{1/2}} \exp\left(-\frac{1}{2}(y - F\beta)^\top k(X, X; \Theta)^{-1}(y - F\beta)\right)$$

Thus β, Θ are estimated by maximizing $L(y; \beta, \Theta)$. Contrarily to linear regression, this (non-convex) optimization problem has not an explicit solution, and must be solved numerically[2].

Bayesian inference. Link with Universal Kriging. In Bayesian statistics, the parameters $\theta = (\beta, \Theta)$ themselves are assumed to be random, with a *prior* distribution. Bayesian inference consists in computing the whole *posterior distribution* of θ conditional on the observations. When the prior admits a density f_θ (with respect to the Lebesgue measure), applying the Bayes rules shows that the posterior admits the density

$$f_{\theta\,|(Y(x_1),\ldots,Y(x_n))=y}(t) = \frac{1}{C} L(y; t)\, f_\theta(t)$$

where $C = \int L(y; t)\, f_\theta(t) dt$ is a normalizing constant. If a single number is wanted as an estimation of θ, the mode of the posterior distribution can be computed by maximizing $L(y; t) f_\theta(t)$ over t. This is similar to MLE, but here the likelihood is weighted by the prior density. Samples from the posterior distribution can be obtained with Markov chain Monte Carlo techniques, at the cost of a (maybe high) computational cost.

For some particular choices of priors, one recovers the formulas of Universal Kriging. More precisely, if we assume that the kernel parameters Θ are known (Dirac prior), and the trend parameters are multinormal with $\beta \sim \mathcal{N}(\mu, \lambda k(X, X; \Theta))$, then the mean (resp. variance) of the posterior distribution tend to the Kriging mean (resp. UK variance) when $\lambda \to +\infty$ (Helbert et al., 2008).

Cross-validation. The two previous techniques depend on the validity of the assumptions of the model, typically that it is a GP with the given mean and covariance structure. Cross-validation (CV) may give better results in presence of model misspecification. An example of cross-validation criterion is the leave-one-out criterion, written

$$\mathrm{LOO}(\beta, \Theta) = \sum_{i=1}^{n} (\hat{y}_{-i}(x_i) - y_i)^2 \tag{4.11}$$

[2]In practice, the log-likelihood is maximized. If a local minimizer is used (e.g. gradient descent), several starting points must be used (*multistart*) as several local minima may exist.

where $\hat{y}_{-i}(.)$ is the kriging mean computed without the observation y_i. Denoting by X_{-i} the set of observations without the i^{th} one, the covariance matrix $k(X_{-i}, X_{-i}; \Theta)$ can be computed from the full covariance matrix $k(X, X; \Theta)$ in an economic way, with so-called *update formulas*. The drawback of the LOO criterion (4.11) is that it does not account for the dependence of the conditional GPs $\hat{Y}_{-i}$ for different i. A corrected version has been proposed in (Ginsbourger and Schärer, 2023), as well as and extension to k-fold CV. It is shown that if the model is well specified, then the corrected CV criterion is equivalent to MLE.

4.5 Model validation.

The validity of GP models can be investigated by testing whether the vector of observations $(y_1, \ldots, y_n)$ is drawn from a multivariate normal distribution. One possibility is to consider the leave-one-out predictions (see the previous paragraph "cross-validation"), which follow a Normal distribution:

$$Y(x_i)|\{Y(x_j) = y_j, \forall j \neq i\} \sim \mathcal{N}\left(m_{\text{c},-i}(x_i), s^2_{\text{c},-i}(x_i)\right)$$

where $m_{\text{c},-i}$ (resp $s_{\text{c},-i}$) is the Kriging mean (resp. standard deviation) when removing the observation y_i. This implies that the standardized LOO residuals

$$\frac{y_i - m_{c,-i}(x_i)}{s_{c,-i}(x_i)}$$

are drawn from a $\mathcal{N}(0, 1)$ distribution, which can be checked with usual graphical diagnostics (such as a qqplot). However, this is an approximate diagnostic, that assumes that the hyperparameters are known, and that does not account for the correlation structure of the LOO residuals. A corrected diagnostic can be found in (Ginsbourger and Schärer, 2023), together with an extension to k-fold residuals.

We conclude by a brief illustration on a 2-dimensional Branin function, defined on $[0, 1]^2$ by

$$f(x_1, x_2) = \left[x_2' - \frac{5}{4\pi^2}(x_1')^2 + \frac{5}{\pi}x_1' - 6\right]^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1') + 10$$

with $x_1' = 15x_1 - 5$ and $x_2' = 15x_2$. This function is a toy case for optimization, as iI has three global minima, approximately equal to $x^{(1)} = (0.1238946, 0.8166644)$, $x^{(2)} = (0.5427730, 0.15)$ and $x^{(3)} = (0.9616520, 0.15)$. We will use it in the next chapter as well.

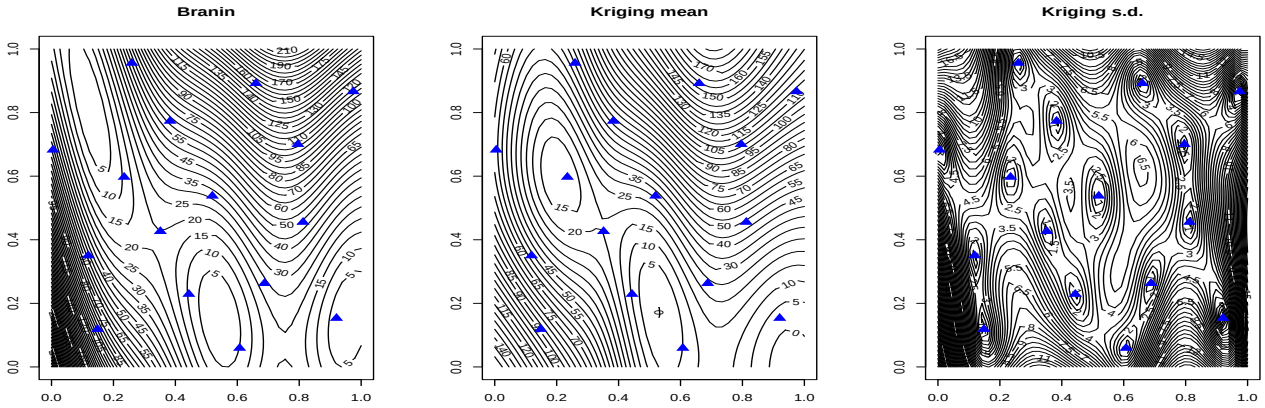

Figure 4.2: Prediction by GP regression (or Kriging) for the Branin function, based on a 16-point Latin hypercube maximin design: conditional mean m_c (middle) and standard deviation s_c (right). hyperparameters estimation has been done by MLE.

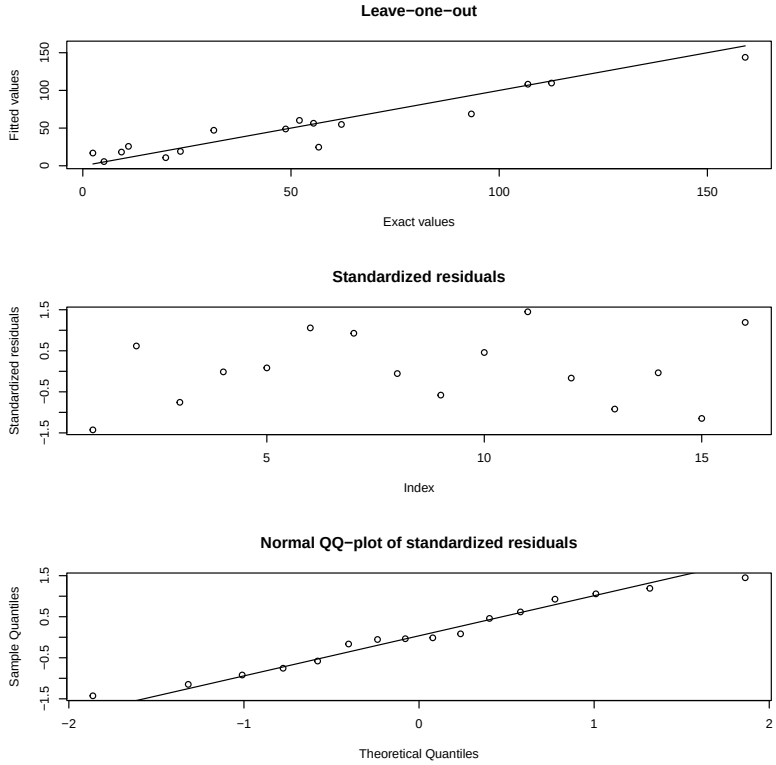

Figure 4.3: Leave-one-out diagnostic for the GP model of Figure 4.2

4.6 Exercises

Exercise 11 (GP regression formulas) *Let us consider the setting and notations of GP regression in the noise-free case.*

1. *Let $V = Y(x)$ and $W = (Y(x_1), \ldots, Y(x_n))$. Show that the vector $U = (V, W)$ is Gaussian. Express its mean and covariance matrix in function of m, k and X. Apply formulas* (7.2) *and* (7.3) *to obtain the expression of $m_c(x)$ and $k_c(x, x)$ in Equations* (4.1) *and* (4.2).
2. *Prove, using the definition of m_c, k_c, that $m_c(x_i) = y_i$ and $k_c(x_i, x) = 0$ for all $i = 1, \ldots, n$ and all $x \in \mathbb{X}$. Prove it in a second way, with Equations* (4.1) *and* (4.2).
3. *Give a 95% prediction interval of $Y(x)$ knowing $Y(x_i) = y_i$ $(i = 1, \ldots, n)$.*
4. *Explain how to adapt Question 1 to get $k_c(x, x')$ when $x' \neq x$. Do the computations.*
5. *Finally prove that the conditional process Y knowing $Y(x_1), \ldots, Y(x_n)$ is a GP.*

Exercise 12 (GP regression formulas, noisy case) *Let us consider the case of GP regression for noisy observations. Mimicking the previous exercise, prove formulas* (4.3) *and* (4.4).

Exercise 13 (GP regression with derivatives) *Based on the results of Exercise 2, deduce the expression below for the kriging mean and kriging variance accounting for derivatives, i.e. the distribution of $Y(x)$ knowing that $Y'(x_1) = d_1$:*

$$\begin{aligned} \mathbb{E}[Y(x)|Y'(x_1) = d_1] &= d_1 \frac{\partial k}{\partial s}(x_1, x) / \frac{\partial^2 k}{\partial s \partial t}(x_1, x_1) \\ \mathbb{V}\mathrm{ar}[Y(x)|Y'(x_1) = d_1] &= k(x, x) - \frac{\partial k}{\partial s}(x_1, x)^2 / \frac{\partial^2 k}{\partial s \partial t}(x_1, x_1) \end{aligned}$$

Write the formula when we know both *that $Y(x_1) = y_1$ and $Y'(x_1) = d_1$.*

Exercise 14 (Corrected LOO criterion) *Let $Y \sim GP(0, k)$. For $n = 2$ points, define the LOO residuals as the random variables*

$$E_{-1} = Y(x_1) - \mathbb{E}[Y(x_1)|Y(x_2)], \qquad E_{-2} = Y(x_2) - \mathbb{E}[Y(x_2)|Y(x_1)]$$

Give the expression of E_{-1} in function of k and $Y(x_1), Y(x_2)$. Similarly, compute E_{-2}. Show that (E_{-1}, E_{-2}) is a Gaussian vector, and show that E_{-1} and E_{-2} are negatively correlated. How can you define standardized residuals S_1, S_2 such that S_1, S_2 and standard $\mathcal{N}(0, 1)$ and independent?

Exercise 15 (Link with GP regression and RKHS) *a) Derive formulas* (4.9) *and* (4.10). *b) By mimicking the proof of Section 4.3, prove that the conditional expectation in GP regression is equal to the interpolator with minimal norm in the associated RKHS (interpolation splines):*

$$\min_{h \in \mathcal{H}} \|h\|, \qquad s.t. \qquad h(x_i) = y_i \quad (i = 1, \ldots, n)$$

Chapter 5

Design of computer experiments

We aim at studying a costly-to-evaluate function f_{sim}, typically a simulator or a machine learning algorithm, based on a dataset. Contrarily to other contexts, we can *create* this dataset, equivalently *design the experiments* (DoE), by choosing the locations where to evaluate f_{sim}.

There is a rich theory of design of experiments when the observations are obtained from a linear model (see e.g. Fedorov, 2013). But we consider here a different framework: we assume that f_{sim} is a complex function (possibly non linear), and that the experimental noise is negligible ($y_i = f_{\text{sim}}(x_i)$). As this often corresponds to experiments obtained by a (deterministic) computer model, one use the word design of *computer* experiments.

There are two main classes of strategies: static and dynamic *or adaptive*. The static strategy creates an initial DoE; without specific goal or model information, this leads to *space-filling* DoEs. The adaptive strategy sequentially adds the design points proposed by a GP-based criterion, on which the numerical model f_{sim} is evaluated. It is driven by a specific objective, such as: model accuracy, optimization, inversion.

5.1 Space-filling designs

Why space-filling designs and (un)desirable properties. Recall that we assume that f_{sim} is possibly non-linear and the observations are noise-free. Then, several features are desirable.

- (space-fillingness) As no information on the form of f_{sim} is available, the aim is to cover the domain or *fill the space* as most as possible, for exploration purpose.
- (no replication) As the experimental noise is negligible, it is not relevant to evaluate f_{sim} several times at the same design point. Thus one should avoid to replicate experiments.
- (stability by projection) In the frequent case where $f_{\text{sim}} : \mathbb{R}^d \to \mathbb{R}$ actually depends on $m < d$ variables or linear combinations of variables, one should expect that the DoE preserves the two previous features (space-fillingness and absence of replications) by projection onto marginal or oblique subspaces. We report to Exercise 16 for illustrations of this fact.

An obvious candidate for satisfying all these constraints is the *uniform design*, obtained by sampling independently the design points from the uniform distribution on $\mathbb{X}$. However, for a limited number of points, a uniform design can fill poorly the space and generate clusters (see e.g. Figure 5.2, left panel). There are better alternatives, some of them are presented below.

Maximin and minimax designs. Let $X = \{x_1, \ldots, x_n\}$ be a DoE in $\mathbb{X}$. There are two famous geometric criteria to quantify how well the design points fill the space. They are based on a distance in $\mathbb{X}$, typically the Euclidean distance.

The *maximin-distance* criterion is the minimal distance between the design points:

$$\Phi_{Mm}(X) = \min_{1 \leq i < j \leq n} \|x_i - x_j\|$$

A design that maximizes this distance is called *maximin*. Actually, when $\mathbb{X}$ is convex, finding a maximin design is equivalent to a *sphere-packing problem*, i.e. finding a set of non-overlapping spheres contained in $\mathbb{X}$ with a maximal radius (Pronzato, 2017).

A dual criterion is the *minimax-distance* criterion, equal to the largest distance between a point $x \in \mathbb{X}$ and the DoE:

$$\Phi_{mM}(X) = \max_{x \in \mathbb{X}} \min_{i=1,\ldots,n} \|x - x_i\|$$

The DoEs that minimize this criterion are called *minimax*. By definition of $\Phi_{mM}(X)$, the union of spheres centered at x_i with radius $\Phi_{mM}(X)$ cover $\mathbb{X}$; thus finding a minimax design is a *sphere covering problem*, i.e. finding a set of spheres that cover $\mathbb{X}$ with a minimal radius. Furthermore, they are connected to GP prediction. Indeed, for many isotropic kernels, i.e. such that $k(x, x')$ depends on $\|x - x'\|$, the kriging variance at x obtained with X, denoted $k_c(x, x; X)$, verifies:

$$\sup_{x \in \mathbb{X}} k_c(x, x; X) \leq S(\Phi_{mM}(X))$$

where S is an increasing function (Schaback, 1995). Thus minimax DoEs will tend to reduce the global uncertainty of the GP prediction.

Compared to maximin DoEs, minimax DoEs are harder to compute, as they involve all the domain points, and not only design points. Nevertheless, the two criteria are connected by inequalities, that justify the common practice of computing only maximin DoEs. An illustration of maximin and minimax designs is given in Figure 5.1. We can see that both of them fill well the space, but also exhibit alignements that will give replicates in projection. To avoid this drawback, they are often computed among the class of Latin hypercube DoEs (see the dedicated section below).

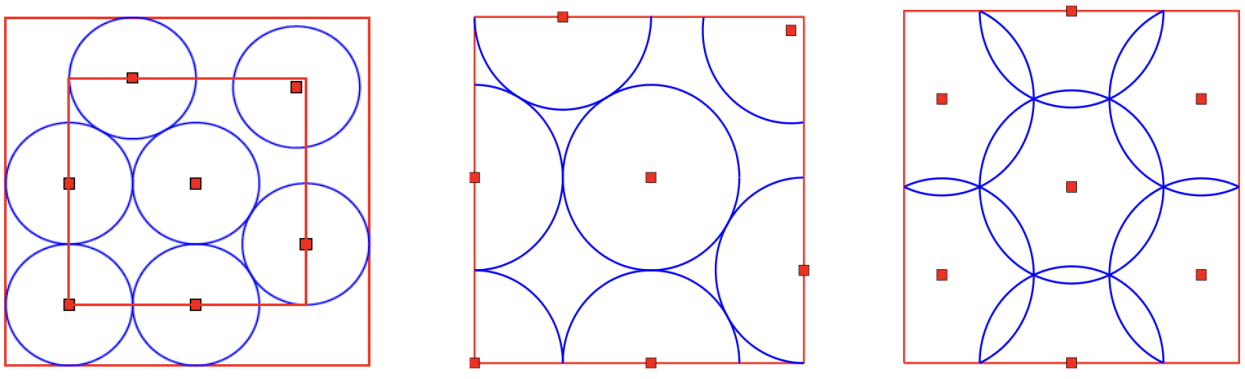

Figure 5.1: Examples of maximin (middle panel) and minimax (right panel) designs in $\mathbb{X} = [0, 1]^2$. Left panel: same than middle, with an extended boundary, showing the equivalence between the maximin and sphere packing problems. Source : Pronzato (2017).

Low discrepancy sequences. The *discrepancy* is a statistical quantity measuring the departure of the empirical distribution of the design points to the uniform distribution. Let $X = \{x_1, \dots, x_n\}$ be a DoE in $\mathbb{X} = [0,1]^d$ and let λ be the Lebesgue measure on $\mathbb{X}$. The discrepancy of X with respect to a family of sets $\mathcal{R}$ of $\mathbb{X}$ is defined by

$$D(X, \mathcal{R}) = \sup_{R \in \mathcal{R}} \left| \frac{\text{Card}(i \in \{1, \dots, n\} \text{ s.t. } x_i \in R)}{n} - \lambda(R) \right|$$

There are various choices for $\mathcal{R}$, for which the discrepancy can be computed in closed form. For instance, the standard discrepancy $D(X)$ is associated to the family of all hyperrectangles $R = \prod_{i=1}^n [a_i, b_i]$ with $0 \leq a_i < b_i \leq 1$ $(i = 1, \dots, n)$. whereas the *star discrepancy* $D^\star(X)$ only considers the hyperrectangles fixed at the origin: $R = \prod_{i=1}^n [0, b_i]$.

An important theoretical result (Koksma-Hlakwa theorem) is that the star discrepancy gives a control of the quadrature error. Indeed, for a large class of functions f, we have the inequality

$$\left| \frac{1}{n} \sum_{i=1}^n f(x_i) - \int_{\mathbb{X}} f(x) dx \right| \leq V(f)\, D^\star(X)$$

where $V(f)$, equal to the Hardy-Krause total variation of f, does not depend on X. Similar inequalities hold when f belongs to a RKHS.

Thus a DoE with a small discrepancy will guarantee a good approximation of $\int_{\mathbb{X}} f$ uniformly on f. The class of *low discrepancy sequences* (LDS), used in Quasi Monte Carlo integration, gathers DoE for which there exists constants $c, s > 0$ such that for all $n \geq 1$,

$$D(X) \leq c \frac{(\log n)^s}{n}$$

In statistical words, this means that the convergence rate of the mean $\frac{1}{n} \sum_{i=1}^n f(x_i)$ to the expectation $\int_{\mathbb{X}} f(x) dx$ is faster than the rate $n^{-1/2}$ of a simple Monte Carlo procedure. Notice, however, that the bounding constant c behaves poorly with the dimension.

The construction of LDS relies on number theory. The simplest one, the 1D Van der Corput sequence, uses a 'mirror' decomposition of integers in base 2. If $i = \sum_{j \geq 1} b_j 2^{j-1}$ with $b_j \in [0,1]$, the i^{th}th point of the sequence is $x_i = \sum_{j \geq 1} b_j 2^{-j}$ where the same b_j are used for *negative* exponents of 2. The Halton sequence extends this procedure in d dimensions by using the first d prime numbers as a number basis for each coordinate, i.e. base 2 for $x_{i,1}$, base 3 for $x_{i,2}$, base 5 for $x_{i,3}$ etc. Other famous LDS are Faure sequences and Sobol sequences.

The main drawback of LDS for design of experiment is that they behave poorly in projection, in particular in the last coordinates, although they behave better than a uniform design in dimension d. This effect is striking for Halton and Faure sequences (see e.g. Figure 5.8, right panel), and moderate for Sobol sequences, which explain that Sobol sequences are more often used.

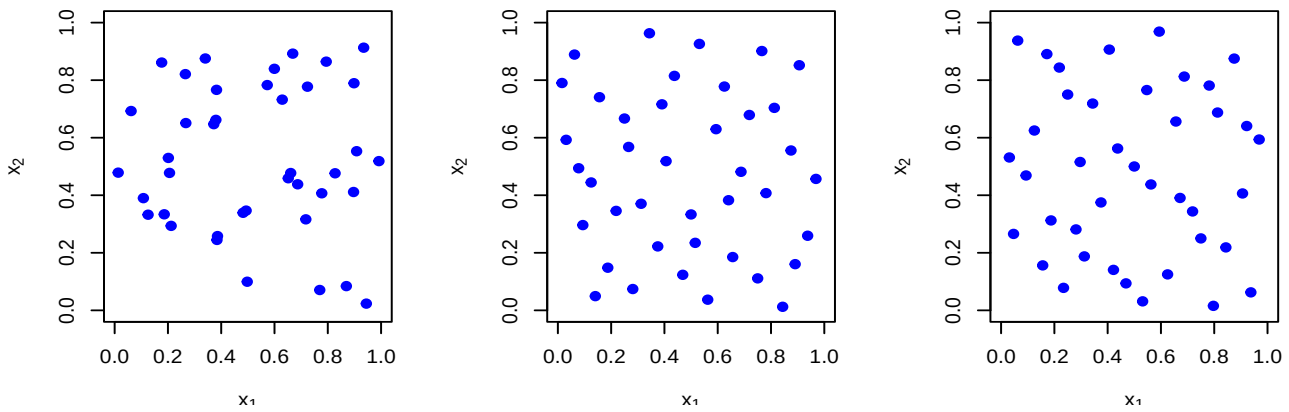

Figure 5.2: Examples of 2D space-filling designs. From left to right: Uniform design, Halton sequence, Sobol sequence.

Latin hypercube designs. For simplicity, first consider the 2-dimensional case. A n-point *Latin hypercube design* (LHD) on $\mathbb{X} = [0,1]^2$ is a random design such that there is exactly one point per row and exactly one point per column. Here, the rows and columns are defined by the partition of $[0,1]$ in n strata of same length : $[0,1] = I_1 \cup \cdots \cup I_n$, with $I_i = [(i-1)/n, i/n]$, $i = 1, \ldots, n$.

It can be built with one permutation s_1 of $\{1, \ldots, n\}$, by defining the intial points $(i, s_1(i))$, $i = 1, \ldots, n$. Then these points are transformed to random real numbers in intervals of length 1 by removing $(U_{1,i}, U_{2,i})$, where $U_{1,1}, U_{2,1}, \ldots, U_{1,n}, U_{2,n}$ are i.i.d. uniform on $[0,1]$. Finally, a division by n maps the points to $[0,1]$. The LHD is thus formed by the points $\left(\frac{i - U_{1,i}}{n}, \frac{s_1(i) - U_{2,i}}{n}\right)$, $i = 1, \ldots, n$. An illustration is given in Figure 5.3. This definition is immediately extended to d-dimensions by considering $d-1$ permutations $s_1, \ldots, s_{d-1}$ of $\{1, \ldots, n\}$. The corresponding LHD is the set of points $\left(\frac{i - U_{1,i}}{n}, \frac{s_1(i) - U_{2,i}}{n}, \ldots, \frac{s_{d-1}(i) - U_{d,i}}{n}\right)$, $i = 1, \ldots, n$. By construction, the number of possible LHDs is equal to $(n!)^{d-1}$.

The construction of LHDs is based on the idea of *stratified sampling*, in order to improve the uniformity of designs with a fixed size. Indeed, in stratifed sampling, the proportion of points that belong to some interval is forced to equal the theoretical one, which is the case for the marginal sets $\{x_j \in I_i\}$ for a fixed I_i: $\#\{j \text{ s.t. } x_j \in I_i\} = \mathbb{P}(x_j \in I_i) = 1/n$. This improvement can be assessed for quadrature problems: if a multivariate function[1] $g : \mathbb{X} \to \mathbb{R}$ is monotonic with respect to all its arguments, if $x_1, \ldots, x_n$ are the points of a LHD, and if $u_1, \ldots, u_n$ are the points of a random uniform design (Monte Carlo sampling), then approximating $\int_{\mathbb{X}} g(x)dx$ by the sample mean is more precise with a LHD (Mckay et al., 2000, Section 2):

$$\mathbb{V}\text{ar}\left(\frac{1}{n}\sum_{i=1}^{n} g(x_i)\right) \leq \mathbb{V}\text{ar}\left(\frac{1}{n}\sum_{i=1}^{n} g(u_i)\right).$$

The monotonicity assumption can be relaxed asymptotically: for large n the result is valid, and

[1] As an example, in our context, g can be f_{sim} if the aim is to evaluate a mean value, or $g = 1_{f_{\text{sim}} < T}$ for some threshold T if the aim is to evaluate a probability of failure.

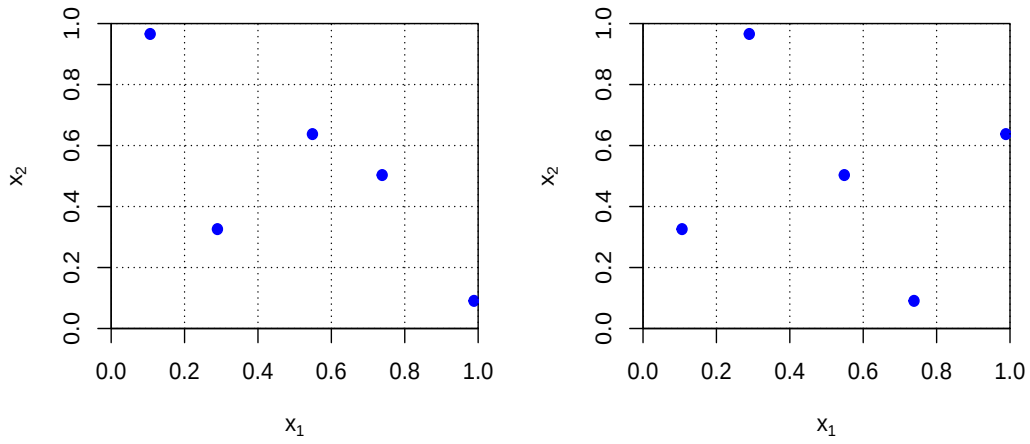

Figure 5.3: Examples of 5-point LHDs. Left: a random LHD, associated to the permutation $(5, 2, 4, 3, 1)$. Right: an approximate maximin LHD.

the variance reduction depends on the extent to which g is additive (Stein, 1987).

However, for a metamodeling purpose, choosing a LHD at random is generally not enough. Indeed, depending on the permutations used, its points do not always have good space-filling properties. To avoid this drawback, one can search for a specific LHD that optimizes a space-filling criterion. For instance, a maximin LHD is a design which has the largest maximin value among the class of LHDs. An illustration is provided in Figure 5.3, where we can see that the points of the approximate maximin LHD are well spread out in $\mathbb{X}$.

Stability by projection and radial scanning statistic. As sketched in the introduction of the section, a good space-filling DoE should preserve its properties by projection onto marginal or oblique subspaces. The radial scanning statistic (RSS) has been built to evaluate a DoE in this respect for a hypercubic domain $\mathbb{X}$ (Roustant et al., 2010). The RSS automatically detects the departures from uniformity by projections onto 2D or 3D subspaces. It is based on two mathematical results. First, the law of the projection of a uniform random vector onto a straight line is known (a very old result, due to Lagrange in the 18th century). Second, there are powerful uniformity tests to detect clusters, that appear in projection in presence of alignements; one of them is associated to the Greenwood statistics. A combination of these ideas gives the RSS.

Let us illustrate how we can use the RSS with the 8D Sobol (low discrepancy) sequence, shown in Figure 5.4. For a given pair (or triplet) of dimensions, the RSS is computed on the projected points, for all straight lines. The worst case is then considered (largest RSS value among pairs of dimensions), here the plane (x_2, x_7). For that plane, the RSS values are represented as a polar curve (middle panel). The circle corresponds to the value of the statistic associated to a 5% confidence level: a value out of the circle thus indicates a departure from the assumption that the DoE is drawn from a uniform distribution. This is the case for the angle $-\pi/4$, and to a lesser extent $+\pi/4$ (middle panel). The projection onto the worst straight line (angle $-\pi/4$) is represented, where we can see that many points overlap in projection. This will be a problem is f_{sim} is a function of $x_2 - x_7$ (see Exercise 16). Here a solution is to perturbate the Sobol sequence by adding a small noise (*scrambling*).

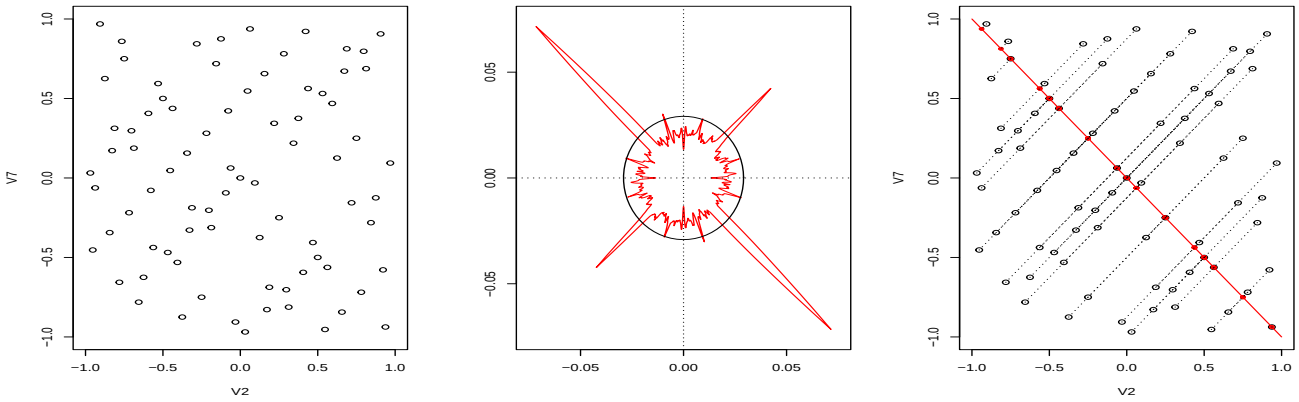

Figure 5.4: The defects of a 8D Sobol sequence, detected by the RSS. Left: projection onto the worst 2D space (x_2, x_7). Middle: RSS curve. Right: Projected points for the worst angle.

Variations and other designs. The DoEs presented above can be easily adapted when the input variables $X_1, \dots, X_d$ are independent but follow a non-uniform distribution $\mu = \otimes_{j=1}^{d} \mu_j$. Indeed, as the image of X_i by its cdf is uniformly distributed, a space-filling DoE with respect to μ is obtained by applying the reverse (quantile) transformation to a space-filling DoE.

The literature on space-filling DoEs is incredibly vast, and we can cite, among other families of DoEs: orthogonal arrays (extensions of LHD for projection onto 2D or higher marginal spaces), lattices (other kinds of LDS), maximum entropy designs, point processes (such as Strauss or determinental processes). We refer to the book of Fang et al. (2005) or to the review of Pronzato and Müller (2012) for more examples and details. We also refer to the R packages DiceDesign (Dupuy et al., 2015) and randtoolbox (Christophe and Petr, 2023) for software.

5.2 Gaussian process based adaptive designs

Adaptive designs for optimization: Bayesian optimization

Bayesian optimization denotes adaptive designs based on GP metamodels. There are three main ingredients: a numerical model f_{sim}, a GP metamodel, and an easy-to-compute criterion[2].

Illustration with the expected improvement criterion. Denote by $z_+ := \max(z, 0)$, the positive part of z. Then, the *improvement*, in a minimization perspective, is defined as $\mathrm{I}(z) = (f_{\min} - z)_+$, which counts positively what is below the current minimum $f_{\min} = \min(y_1, \dots, y_n)$. Finally, given $Y \sim \mathrm{GP}(m, k)$, the *expected improvement* (EI) criterion is defined as the mean of improvements over sample paths of the conditional GP:

$$\mathrm{EI}(x) = \mathbb{E}\left[\mathrm{I}(Y(x)) \middle| \{Y(x_i) = y_i, i = 1, \dots, n\}\right]$$

Notice that *although* $Y(x)$ *is unknown* at an unvisited site x, the *conditional law of* $Y(x)$ *is known*. This allows computing the criterion in closed form (Exercise 17):

$$\mathrm{EI}(x) = s_k(x)(z_0 \Phi(z_0) + \phi(z_0)) \tag{5.1}$$

[2]this criterion is often called *infill* or *acquisition* criterion.

Here, $z_0 = \dfrac{f_{\min} - m_c(x)}{s_c(x)}$, ϕ, Φ are respectively the pdf and the cdf of the $\mathcal{N}(0,1)$ distribution, and m_c, s_c are resp. the mean and standard deviation of Y conditional on $\{Y(x_i) = y_i, i = 1, \ldots, n\}$.

This leads to the so-called *Efficient Global Optimization* (EGO) adaptive design, presented in Algorithm 1 and illustrated in Figure 5.5. There are two nice features of EGO. First, it achieves a tradeoff between exploration of unvisited area and exploitation around a local minimum (Exercise 18). Secondly, as claimed in its name, it is indeed a *global* optimization algorithm: under a slight condition on the kernel k, it generates a dense sequence of points (Vazquez and Bect, 2010).

Algorithm 1 EGO algorithm

Require: An initial DoE $\mathcal{X} = \{x_1, \ldots, x_n\}$, and the corresponding observations $\mathcal{Y} = \{y_1, \ldots, y_n\}$.

1: **while** Computational budget not consumed **do**
2: Estimate a GP regression metamodel with DoE $\mathcal{X}$ and observations $\mathcal{Y}$
3: Maximize the associated EI criterion: $x^\star \leftarrow \text{argmax}_x \text{EI}(x)$
4: Evaluate the numerical model f_{sim} at $x^\star$: $y^\star = f_{\text{sim}}(x^\star)$
5: Update the DoE and the set of observations: $\mathcal{X} \leftarrow \mathcal{X} \cup \{x^\star\}$, $\mathcal{Y} \leftarrow \mathcal{Y} \cup \{y^\star\}$
6: **end while**
7: **return** The minimum of $\mathcal{Y}$, and the associated design point.

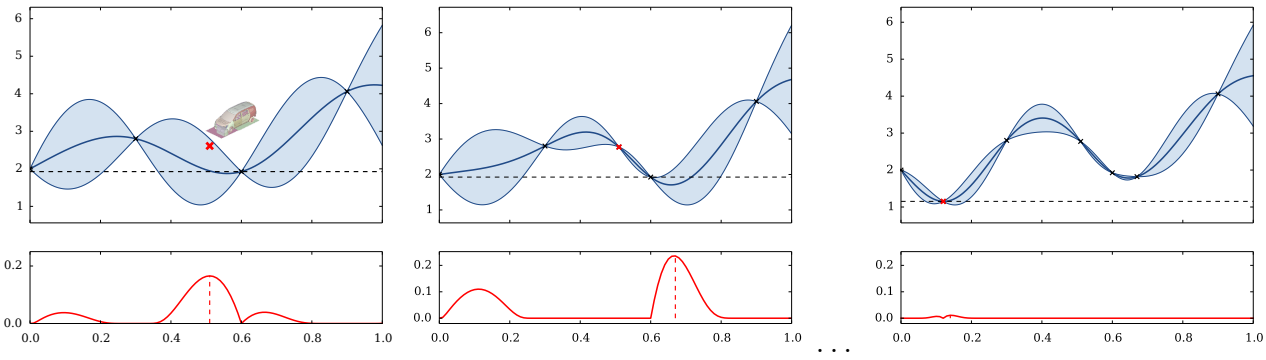

Figure 5.5: Illustration of the EGO algorithm (Bayesian optimization). Left panel: starting from 4 initial points obtained from f_{sim} (e.g. an automotive simulator), a GP metamodel is estimated (top), and the EI criterion computed (bottom). The location where EI is maximum gives a new point where to evaluate f_{sim}. Middle panel: with this new observation, the metamodel is then updated (top), and the EI recomputed (bottom). Right panel: output at iteration 3.

Variations, other criterions, and software. This version of EGO is for noise-free observations. One can adapt it to noisy observations, leading for instance to the *Expected Quantile Improvement* criterion. Furthermore, EGO is a *one-step ahead* strategy, providing only one single point per iteration. One can adapt its definition to provide a batch of points per iteration. In addition to the EI criterion, Bayesian optimization can be defined with many other criteria, such as: the *Approximate Knowledge Gradient*, the *Augmented Expected Improvement*, the *Expected Augmented Lagrangian Improvement*, the *Expected Feasible Improvement*, etc.

Notice also that Bayesian Optimization has been adapted for *constrained optimization* and *multi-objective optimization*.

A way to discover the numerous exisiting possibilities is through software, such as the R packages DiceOptim (Picheny et al., 2021), GPareto (Binois and Picheny, 2019) and the Python toolkit Trieste (Berkeley et al., 2023), which contain both references to the literature and examples.

Adaptive designs for inversion

The word *inversion* stands for three close objectives. Indeed, for a given target T, it aims at estimating either:

- a *level set*: $\mathcal{L} = \{x \in \mathbb{X}, \text{ such that } f_{\text{sim}}(x) = T\}$,
- or an *excursion set*: $\mathcal{E} = \{x \in \mathbb{X}, \text{ such that } f_{\text{sim}}(x) \leq T\}$,
- or a *probability of failure*, $p_f = \mathbb{P}_{\mathbb{X}}(\{x \in \mathbb{X}, \text{ such that } f_{\text{sim}}(x) \leq T\})$, where $\mathbb{P}_{\mathbb{X}}$ is a probability distribution on $\mathbb{X}$.

As for Bayesian optimization, adaptive strategies rely on a Gaussian metamodel Y and a calculable criterion. For inversion, famous ones are *SUR strategies*, where SUR stands for *Stepwise Uncertainty Reduction*. The idea is to choose the next design point to reduce the most a measure of uncertainty. For illustration, for excursion sets, one can define a random variable representing the uncertainty as the conditional variance of $1_{Y(u)\leq T}$, integrated over all the domain:

$$H_n(x_1, \ldots, x_n) = \int_{\mathbb{X}} \mathbb{V}\text{ar}\left[1_{Y(u)\leq T} | Y(x_1), \ldots, Y(x_n)\right] \mu(du)$$

where μ is some probability distribution on $\mathbb{X}$ (e.g. the Lebesgue measure when $\mathbb{X}$ is a subset of $\mathbb{R}^d$). Notice that the variance term can be computed in closed-form. Indeed, the indicator variable $1_{Y(u)\leq T}$ follows a Bernoulli distribution with parameter

$$p_n(u) = \mathbb{P}(Y(u) \leq T | Y(x_1), \ldots, Y(x_n)) = \Phi\left(\frac{T - M_c(u)}{S_c(u)}\right) \tag{5.2}$$

where Φ is the cdf of the standard Normal distribution, and $M_c(u)$ (resp. $S_c(u)$) is the conditional mean (resp. standard deviation) of $Y(u)$ conditional on $Y(x_1), \ldots, Y(x_n)$. Thus,

$$H_n(x_1, \ldots, x_n) = \int_{\mathbb{X}} p_n(u)(1 - p_n(u))\mathbb{P}_{\mathbb{X}}(du)$$

Then, for a new design point x, the one-step ahead SUR criterion is defined as

$$J_n(x) = \mathbb{E}(H_{n+1}(x_1, \ldots, x_n, x) | \{Y(x_i) = y_i, i = 1, \ldots, n\})$$

In this expression, the expectation is done with respect to $Y(x)$, which is unknown. It can be computed, at least numerically, because the (conditional) law of $Y(x)$ is known. Then the next point x_{n+1} can be chosen in order to minimize $J_n(x)$:

$$x_{n+1} \in \underset{x\in\mathbb{X}}{\text{argmin}}\ J_n(x)$$

At each step, the function p_n, called *probability of excursion function* can be visualized, and allows to classify the points u corresponding to the two regions $Y(u) \leq T$ and $Y(u) > T$. More precisely, the three quantities of interest defined at the begining of the section can be estimated by

- $\widehat{\mathcal{L}} = \{x \in \mathbb{X},\ \text{such that } p_n(x) = 1/2\}$,
- $\widehat{\mathcal{E}} = \{x \in \mathbb{X},\ \text{such that } p_n(x) \geq 1/2\}$,
- $\widehat{p_f} = \int_{\mathbb{X}} p_n(u)\mathbb{P}_{\mathbb{X}}(du)$.

In theory, SUR strategies are connected to the theory of martingales, which allows to derive (under some conditions) that the uncertainty measure H_n tends to zero almost surely when n tends to infinity. Furthermore, it turns out that several algorithms of Bayesian optimization, and in particular the EGO method, can be viewed as SUR strategies.

For more details and examples, we refer to the journal paper (Bect et al., 2019) and to the piece of software KrigInv (Chevalier et al., 2022) with its documentation (Chevalier et al., 2014).

Illustration

We conclude by a brief illustration on a 2-dimensional Branin function used in the previous chapter.

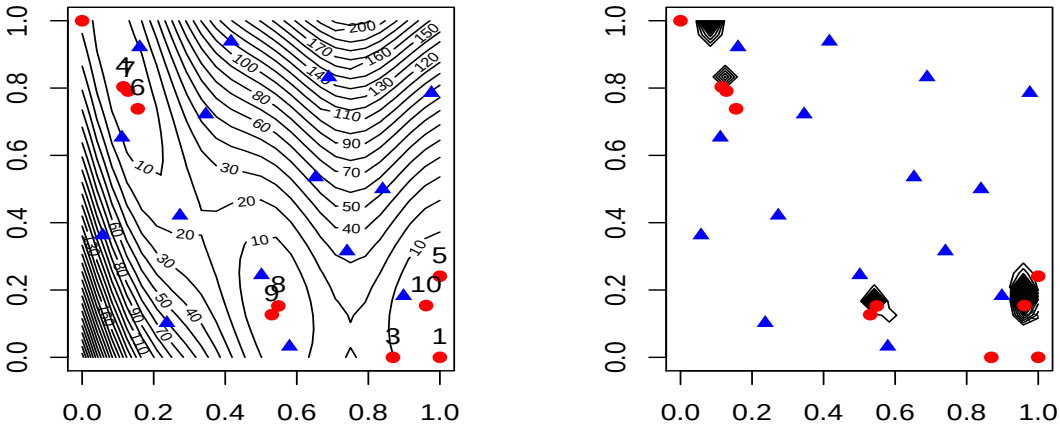

Figure 5.6: Results of the EGO method for the Branin function, from (Roustant et al., 2012). Left: sequence of points obtained by 10 iterations of EGO (red numbered points). Right: contours of the EI criterion for the last model. Triangles represent an initial optimal Latin hypercube design.

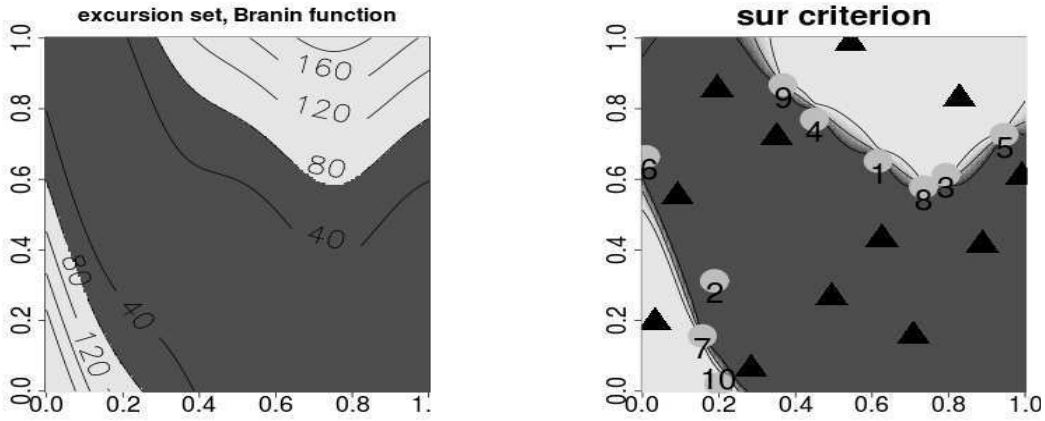

Figure 5.7: Example of adaptive design for inversion, from the preprint of (Chevalier et al., 2014). Left: excursion set of the Branin function for the target $T = 80$. Middle : estimated excursion probability function p_n after 10 iterations of SUR criterion. New evaluated points are represented by circles, added sequentially to the initial points (triangles).

5.3 Exercises

Exercise 16 (Projections and loss of information) *Figure 5.8 shows two examples of* 16*-point space-filling DoEs in dimension 2 and 8 respectively, proposed for the metamodeling of* f_{sim}*. However, these DoEs have serious drawbacks!*

1. Case of a marginal subspace. *Assume that* $f_{sim} : [0,1]^2 \to \mathbb{R}$ *only depends on one variable (say* x_1*), meaning that the other one are inactive. Consider the grid (also called* full factorial design*) of Figure 5.8. Explain why* 75% *of the information will be lost with this DoE! What other kind of DoE can be recommended to avoid this phenomenon?*

2. Case of an oblique subspace. *Explain why the 8D low discrepancy sequence of Figure 5.8 is not a good DoE: for what form of functions* f_{sim} *this DoE will be inappropriate? If in addition* f_{sim} *has the form* $f_{sim}(x) = g(x_7 - x_8)$ *for some function* g*, what percentage of information will be lost with this DoE? What tool can be used to evaluate the quality of DoEs with respect to projections onto oblique subspaces?*

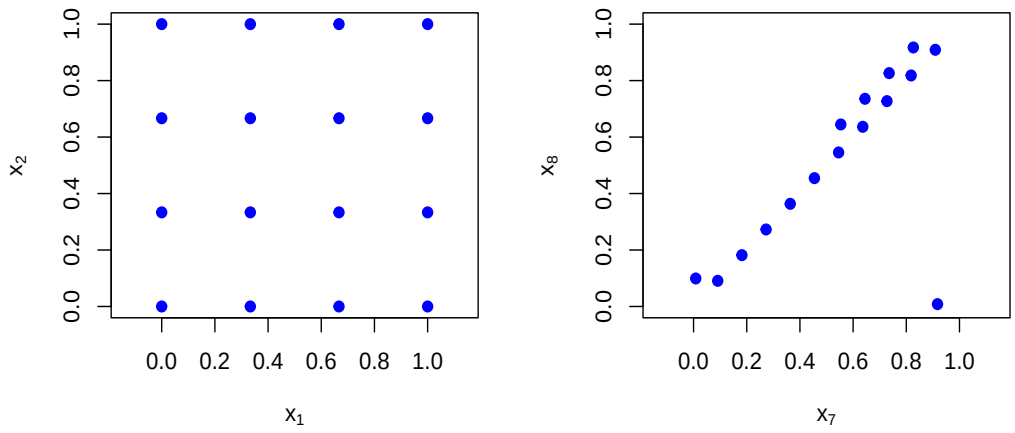

Figure 5.8: Examples of 16-point DoEs. Left: a grid in 2 dimensions. Right: projection onto the plane (x_7, x_8) of a 8-dimensional Faure low discrepancy sequence.

Exercise 17 (EI expression) *Derive the closed-form expression* (5.1) *of the EI criterion.*

Exercise 18 (EGO behaviour: the exploration/exploitation tradeoff) *In this exercise, we denote* $m(x) = m_c(x)$*,* $s(x) = s_c(x)$ *and* $u(x) = \dfrac{f_{min} - m(x)}{s(x)}$*. Then the EI criterion is written:*

$$EI(x) = s(x)\left[u(x)\Phi(u(x)) + \phi(u(x))\right]$$

For simplicity, we now remove the x *in* $u(x), m(x), s(x), EI(x)$*.*

1. *Explain briefly why* $\Phi'(u) = \phi(u)$ *and* $\phi'(u) = -u\phi(u)$*, and prove that* $\frac{\partial EI}{\partial m} = -\Phi(u)$*.*

2. *A similar calcul would show (don't do it) that* $\frac{\partial EI}{\partial s} = \phi(u)$*. Recall that we want to minimize* f*. Explain why the two properties,* $\frac{\partial EI}{\partial m} < 0$ *and* $\frac{\partial EI}{\partial s} > 0$*, mean that the EGO algorithm achieves a tradeoff between exploration (of unvisited regions) and exploitation (of promising regions) during the optimization.*

Chapter 6

Global sensitivity analysis

Consider a numerical model $f : x \in \mathbb{X} = [0,1]^d \to \mathbb{R}$. Sensitivity analysis aims at quantifying the influence of the input variables $x_1, \ldots, x_d$ on the values of f. A local answer to this question is given by the partial derivatives $\frac{\partial g}{\partial x_i}$. *Global sensitivity analysis* (GSA) gives a *global* answer where the variables vary in the whole input domain $\mathbb{X}$. A different framework is considered, by assuming that the input variables are *random*. Thus GSA aims at quantifying the influence of the input *random* variables $X_1, \ldots, X_d$ which explain the variation of *random* variable $f(X) = f(X_1, \ldots, X_d)$.

The simplest way to measure this variation is $\mathbb{V}\mathrm{ar}(f(X))$ and this chapter will focus on it. Similarly, we will consider the simplest case where the input variables are *independent*. In this basic framework, we will be able to answer the following questions:

- *Screening*: what are the variables X_i that have no influence on $f(X)$?
- *Uncertainty quantification*: what is the influence of X_i on $\mathbb{V}\mathrm{ar}(f(X))$?

For a more general presentation, we refer to (Iooss, 2011) and (Da Veiga et al., 2021).

6.1 Variance-based global sensitivity analysis

Let $X = (X_1, \ldots, X_d)$ be a vector of independent input variables with distribution $\mu_1 \otimes \cdots \otimes \mu_d$, and $f : \Delta \subseteq \mathbb{R}^d \to \mathbb{R}$ is such that $f(X) \in L^2(\mu)$. We will use the set notation: if $I = \{i_1, \ldots, i_m\}$ with $i_1 < \cdots < i_m$, then $X_I = (X_{i_1}, \ldots, X_{i_m})$. We will denote X_{-I} when we remove the components $i \in I$ from the vector X (thus $X_{-1} = (X_2, \ldots, X_d)$). Moreover, by convention, $E[.|X_\emptyset] = E[.]$.

The Sobol-Hoeffding decomposition. The main result for variance-based sensitivity analysis is the *Sobol-Hoeffding decomposition* (Hoeffding, 1948; Efron and Stein, 1981; Sobol, 1993). It states that there exists a unique expansion of f of the form

$$f(X) = f_0 + \sum_{i=1}^{d} f_i(X_i) + \sum_{1 \le i < j \le d} f_{i,j}(X_i, X_j) + \cdots + f_{1,\ldots,d}(X_1, \ldots, X_d)$$

such that $E[f_I(X_I)|X_J] = 0$ for all $I \subseteq \{1, \dots, d\}$ and all $J \subsetneq I$. Furthermore:

$$\begin{aligned} f_0 &= \mathbb{E}[f(X)] \\ f_i(X_i) &= \mathbb{E}[f(X)|X_i] - f_0 \\ f_I(X_I) &= \mathbb{E}[f(X)|X_I] - \sum_{J \subsetneq I} f_J(X_J) \qquad \text{(recursion formula)} \\ &= \sum_{J \subseteq I} (-1)^{|I|-|J|} \mathbb{E}[f(X)|X_J] \qquad \text{(inclusion-exclusion formula)} \end{aligned}$$

The terms depending on only one variable, $f_i(X_i)$, are called *main effects*. Those depending on two variables, $f_{i,j}(X_i, X_j)$, are called *second-order interactions*. More generally those depending on k variables $f_I(X_I)$ where $\mathrm{Card}(I) = k$ are the *interactions of order k*.

We refer to Exercise 19 for a proof in 2D, and to Efron and Stein (1981) for the general case. An elegant other proof is given in Kuo et al. (2010), for a larger class of decompositions, obtained with commuting projections $P_1, \dots, P_d$. Here the projections are orthogonal and given by:

$$P_j(f)(x) = \int f(x) d\mu_j(x_j) = \mathbb{E}[f(X)|X_{-j} = x_{-j}]$$

The form of the decomposition is then simply obtained by expansion:

$$\begin{aligned} I_d &= (P_1 + (I_d - P_1)) \dots (P_d + (I_d - P_d)) \\ &= \sum_{I \subseteq \{1,\dots,d\}} \underbrace{\prod_{j \notin I} P_j \prod_{k \in I} (I - P_k)}_{\Pi_I} \end{aligned}$$

and we have $f_I = \Pi_I(f)$. The non-overlapping condition is written here $P_i(f_I) = 0$ for all $i \in I$.

Variance decomposition (ANOVA) and Sobol indices. The non-overlapping condition

$$\mathbb{E}[f_I(X_I)|X_J] = 0 \quad \text{for all} \quad J \subsetneq I$$

avoids one term to be considered as a more complex one. It implies that all the terms $f_I(X_I)$ are orthogonal (see Exercise 19), leading to the variance decomposition:

$$D := \mathbb{V}\mathrm{ar}(f(X)) = \sum_{I \subseteq \{1,\dots,d\}} \mathbb{V}\mathrm{ar}(f_I(X_I))$$

Thus, we can quantify the influence of the variable X_i by the proportion of variance explained by $f_i(X_i)$. This analysis is called ANOVA, for ANalysis Of VAriance. This ratio is called Sobol index:

$$S_i = \frac{\mathbb{V}\mathrm{ar}(f_i(X_i))}{\mathbb{V}\mathrm{ar}(f(X))} \in [0, 1]$$

This definition can be extended to X_I. Denoting $D_I = \mathbb{V}\mathrm{ar}(f_I(X_I))$, we have $S_I = D_I/D$. Obiously, $\sum_I S_I = 1$.

For screening purpose (i.e. detection of inactive variables), one can use the *total Sobol index*

$$D_i^{\text{tot}} = \sum_{J \supseteq \{i\}} D_J, \qquad S_i^{\text{tot}} = \frac{D_i^{\text{tot}}}{D}$$

Indeed if $S_i^{\text{tot}} = 0$ this implies that $\mathbb{V}\text{ar}(X_I) = 0$ if I contains i. Under mild conditions[1], this implies that $f_I(X_I) = 0$ if I contains i, thus X_i does not appear at all in the decomposition of $f(X)$, meaning that X_i is inactive.

6.2 Illustration on an example in hydrology

We consider a simplified numerical model simulating flooding events, presented in (Iooss, 2011). The model has 8 input random variables, viewed as random variables, assumed independent, whose probability distributions are given (from a previous analysis):

- $X_1 = Q$, Maximal annual flowrate (m^3/s), Gumbel $\mathcal{G}(1013, 558)$ truncated on $[500, 3000]$
- $X_2 = K_s$, Strickler coefficient, Normal $\mathcal{N}(30, 8^2)$ truncated on $[15, +\infty[$
- $X_3 = Z_v$, River downstream level (m), Triangular $\mathcal{T}(49, 51)$
- $X_4 = Z_m$, River upstream level (m), Triangular $\mathcal{T}(54, 56)$
- $X_5 = H_d$, Dyke height (m), Uniform $\mathcal{U}[7, 9]$
- $X_6 = C_b$, Bank level (m), Triangular $\mathcal{T}(55, 56)$
- $X_7 = L$, River stretch (m), Triangular $\mathcal{T}(4990, 5010)$
- $X_8 = B$, River width (m), Triangular $\mathcal{T}(295, 305)$

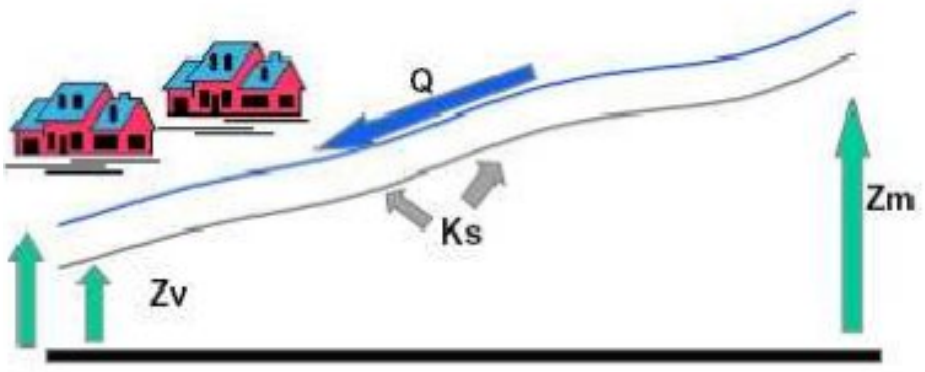

Figure 6.1: A simplified model of a river (Iooss, 2011).

[1]For instance if f is continuous on $\Delta = [0, 1]^d$ and for all i, the support of μ_i contains $[0, 1]$

We consider two variables of interest. First, the maximal annual overflow S (in meters), obtained from simplified hydro-dynamical equations of Saint-Venant:

$$S = \left(\frac{Q}{BK_s \sqrt{\frac{Z_m - Z_v}{L}}} \right)^{0.6} + Z_v - H_d - C_b \,. \tag{6.1}$$

Secondly, the cost (in million euros) of the damage on the dyke Y, depending on S, written as:

$$C = 1_{S>0} + \left[0.2 + 0.8 \left(1 - \exp^{-\frac{1000}{S^4}} \right) \right] 1_{S \leq 0} + \frac{1}{20} \left(H_d 1_{H_d > 8} + 81 1_{H_d \leq 8} \right) \,, \tag{6.2}$$

where $1_A(x)$ is the indicator function which is equal to 1 for $x \in A$ and 0 otherwise.

The results of a global sensitivity analysis for the cost function C are presented in Figure 6.3 and Figure 6.2, either with an unlimited budget ($N = 10\,000$) or with few runs ($n = 80$), using a GP metamodel. Notice that for more complicate hydrological models, the budget will be limited and the construction of a metamodel will be necessary. The built the GP model, the design of experiments was obtained from a Sobol sequence on $[0, 1]^8$ on which a quantile transformation was applied coordinatewise. We see that, the metamodel gives a good approximation of the main effects, recovers the most influential variables and assesses their influence on the damage cost.

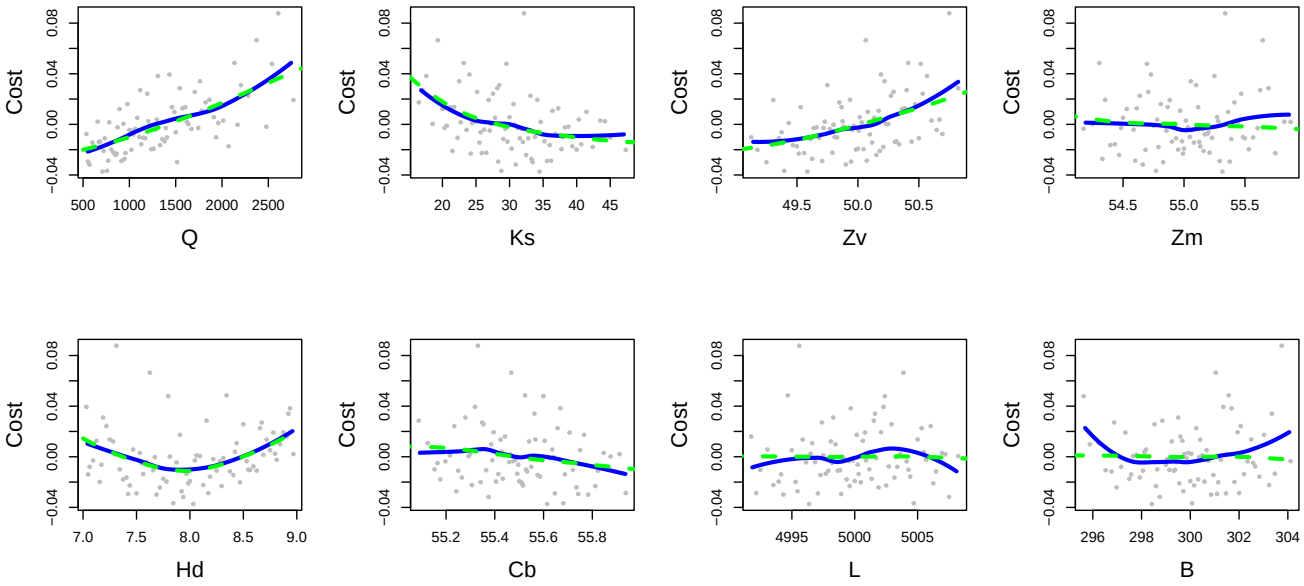

Figure 6.2: Estimation of the main effects of the Cost function. Solid blue lines: with the numerical model and a large budget of $N = 10\,000$ runs. Dashed green lines: with the mean of a GP metamodel built with a small budget of $n = 80$ runs.

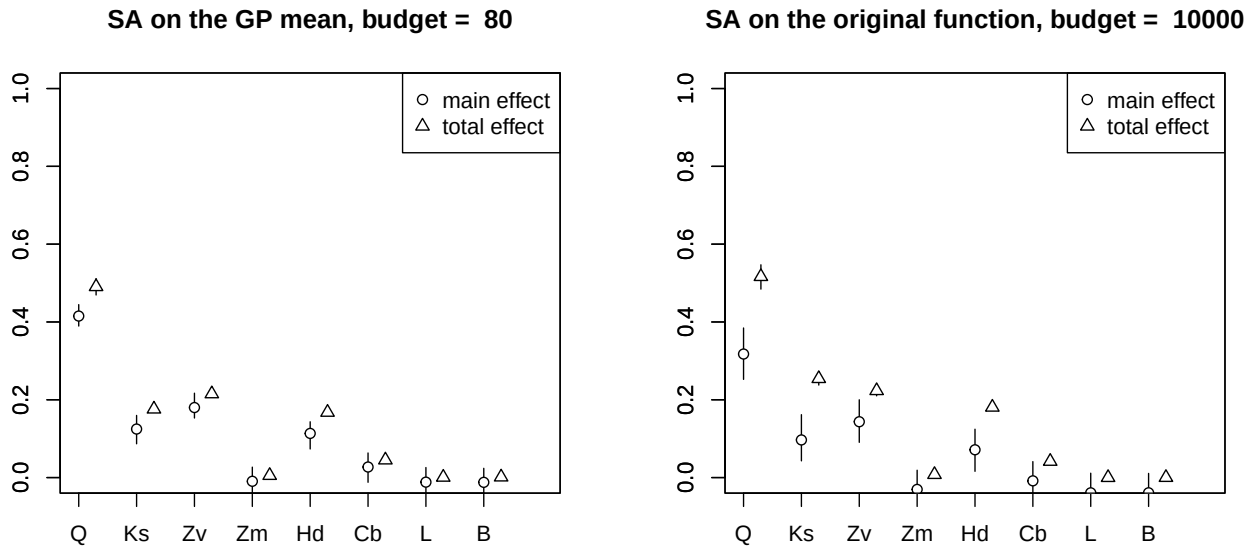

Figure 6.3: Estimated Sobol indices and total Sobol indices for the Cost function.

6.3 Exercises

In the following, we assume that $X_1, \dots, X_d$ are *independent* random variables with probability measures $\nu_1, \dots, \nu_d$. We denote: $X = (X_1, \dots, X_d)$, $\nu = \nu_1 \otimes \cdots \otimes \nu_d$ the probability measure of X and $\Delta = \Delta_1 \times \cdots \times \Delta_d$, the integration domain.

Exercise 19 (ANOVA decomposition in dimension 2) *Here $d = 2$. Let f be in $L^2(\nu)$. We want to prove that there exists a unique decomposition*

$$f(X_1, X_2) = f_0 + f_1(X_1) + f_2(X_2) + f_{1,2}(X_1, X_2)$$

such that $\mathbb{E}(f_I(X_I)|X_J) = 0$ for all $J \subsetneq I$, i.e. $\mathbb{E}(f_i(X_i)) = \mathbb{E}(f_{1,2}(X_1, X_2)|X_i) = 0$ for $i = 1, 2$.

1. *Prove that necessarily, we must have:*

   - $f_0 = \mathbb{E}(f(X))$
   - $f_i(X_i) = \mathbb{E}(f(X)|X_i) - f_0$ *(for $i = 1, 2$)*
   - $f_{1,2}(X_1, X_2) = \mathbb{E}(f(X)|X_1, X_2) - f_1(X_1) - f_2(X_2) - f_0$

   Conversely, check that these terms give the ANOVA decomposition.

2. *Prove that the recursion formula for $f_{1,2}$ can be rewritten as a sum of conditional expectations with alternate signs:*

$$f_{1,2}(X_1, X_2) = \mathbb{E}(f(X)|X_1, X_2) - \mathbb{E}(f(X)|X_1) - \mathbb{E}(f(X)|X_2) + \mathbb{E}(f(X))$$

3. *Prove that all the terms are orthogonal: $\mathbb{E}[f_I(X_I) f_{I'}(X_{I'})] = 0$ if $I \neq I'$.*

4. *Consider $f(X_1, X_2) = X_1$, and let ν_1 be such that X_1 is centered. Observe that we have the two possible decompositions:*

$$f(X_1, X_2) = 0 + X_1 + 0 + 0 = 0 + 0 + 0 + X_1$$

   What's wrong? What is the intuition of the condition "$\mathbb{E}(f_I(X_I)|X_J) = 0$ for all $J \subsetneq I$"?

Exercise 20 (Additive functions - 1st order polynomials & SRCs) *Consider an additive function:*

$$f(x) = \beta_0 + g_1(x_1) + \cdots + g_d(x_d)$$

where the $g_i(X_i)$'s are centered *(with respect to the measure ν_i) and square-integrable.*

1. *What should be the ANOVA decomposition of f? Prove it and compute all Sobol indices.*

2. *Deduce from 1 the ANOVA decomposition of a first order polynomial:*

$$f(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_d x_d$$

   and all Sobol indices. The results can be expressed in function of $m_i^{(1)} = E(X_i)$. Deduce that the Sobol indices of a 1st order polynomial are equal to the squared SRCs used in linear regression, defined by $\beta_i^2 \, \mathbb{V}\mathrm{ar}(X_i)/\mathbb{V}\mathrm{ar}(Y)$ where Y is the response of interest.

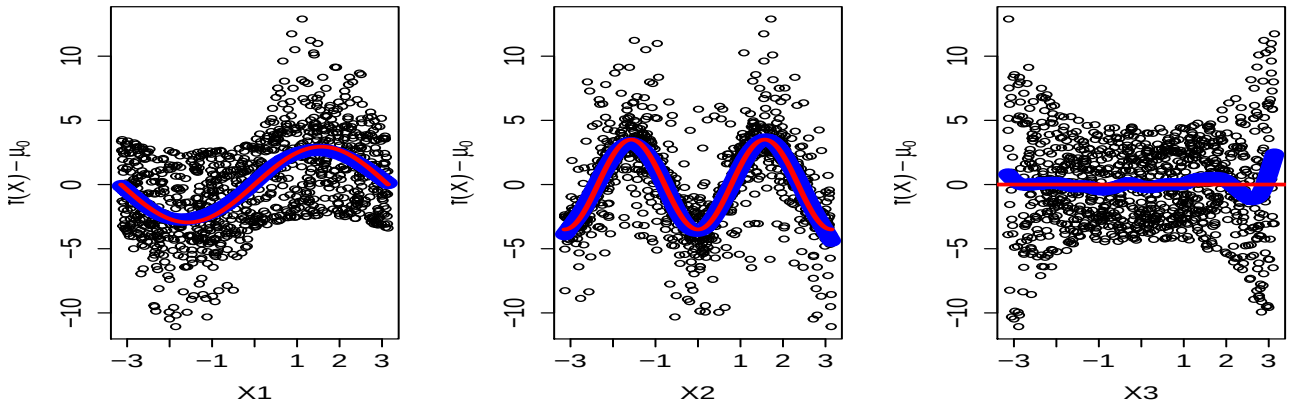

Figure 6.4: Main effects of Ishigami function: Theoretical (straight line) and estimated (bold line)

Exercise 21 (A 3D test-function) *The Ishigami function is defined over* $\Delta := [-\pi; \pi]^3$ *by:*

$$f(x) = \sin(x_1) + A\sin^2(x_2) + Bx_3^4\sin(x_1)$$

with $A = 7$ *and* $B = 0.1$.

1. *Compute the ANOVA decomposition and Sobol indices when* μ *is uniform over* Δ *(use* $a = 1/2, b = \pi^4/5$.*)*
2. *(Computer lab) Estimate the main effects by using simulations of the inputs. Plot them on the same figure and compare with the theoretical ones. Same question with the 2-dimensional projection over* (x_1, x_3) *and the second order interaction* $f_{1,3}(x_1, x_3)$.

Exercise 22 (Polynomial chaos) *Polynomial chaos are defined as a tensor basis of orthonormal polynomials. It is very famous in sensitivity analysis since, once the function of interest has been decomposed on that basis, the Sobol indices are directly obtained as sums of squared coefficients. Now let us go into details.*

Let f *be in* $L^2(\nu)$ *with* $\nu = \otimes_{i=1}^d \nu_i$. *For each probability distribution* ν_i $(i = 1, \ldots, d)$, *denote:*

$$P_{i,0}(x_i) = 1, \quad P_{i,1}(x_i), \quad \ldots \quad P_{i,\ell}(x_i), \quad \ldots$$

a set of orthonormal polynomials in $L^2(\nu_i)$ *(of degree* $0, 1, 2, \ldots, \ell, \ldots$*). Then the* polynomial chaos *indexed by the multi-index* $\underline{\ell} = (\ell_1, \ldots, \ell_d)$ *is the tensor:*

$$P_{\underline{\ell}}(x) = \prod_{i=1}^{d} P_{i,\ell_i}(x_i).$$

We denote by $\mathcal{I} = \mathbb{N}^d$ *the set of all multi-indices.*

1. *For two multi-indices* $\underline{\ell}, \underline{\ell'}$, *compute* $\mathbb{E}(P_{\underline{\ell}}(X)P_{\underline{\ell'}}(X))$. *Deduce that polynomial chaos are orthonormal in* $L^2(\nu)$. *We admit that they form a Hilbert basis of* $L^2(\nu)$.

2. *Deduce that*
$$f(x) = \sum_{\underline{\ell} \in \mathcal{I}} c_{\underline{\ell}} P_{\underline{\ell}}(x)$$
with $c_{\underline{\ell}} = \langle f, P_{\underline{\ell}} \rangle$. *Express the total variance* D *in function of the* $c_{\underline{\ell}}$.

3. *Compute* $\mathbb{E}(P_{\underline{\ell}}(X)|X_1)$.

4. *Deduce that the first main effect of* f *is obtained by choosing the tensors that involve* only X_1, *defined by the subset* $\mathcal{I}_1 = \{\underline{\ell} \in \mathcal{I} \text{ s.t. } \ell_1 \geq 1, \ell_2 = \cdots = \ell_d = 0\}$. *Show that the Sobol index* S_1 *is simply equal to the sum of squared coefficients of these terms:* $S_1 = \sum_{\underline{\ell} \in \mathcal{I}_1} c_{\underline{\ell}}^2$.

5. *Similarly, compute* $\mathbb{E}(P_{\underline{\ell}}(X)|X_{-1}) = \mathbb{E}(P_{\underline{\ell}}(X)|X_2, \ldots, X_d)$.
Show that the first total effect of f *is obtained by choosing the tensors that involve* at least X_1, *defined by* $\mathcal{I}_1^{tot} = \{\underline{\ell} \in \mathcal{I} \text{ s.t. } \ell_1 \geq 1\}$. *Show that* $S_1^{tot} = \sum_{\underline{\ell} \in \mathcal{I}_1^{tot}} c_{\underline{\ell}}^2$.

Exercise 23 (Numerical computation of Sobol indices by pick-freeze formulas.) *Prove that the Sobol index of* X_1 *is given by:*
$$S_1 = \mathbb{C}\text{ov}(f(X_1, X_{-1}), f(X_1, Z_{-1})),$$
where Z_{-1} *is an independent copy of* X_{-1} *(same distribution and independent of the* X_i*'s).* Hint: Conditionaly to X_1, what can you say of $f(X_1, X_{-1})$ and $f(X_1, Z_{-1})$? *Deduce that:*
$$S_1 = \int_{\Delta \times \Delta_{-1}} f(x_1, x_2, \ldots, x_d) f(x_1, z_2, \ldots, z_d) d\nu(x) d\nu_{-1}(z_{-1}) - (\mu_0)^2$$
where $\mu_0 = \int_\Delta f(x)dx$ *is the overall mean. Explain how to compute numerically* S_1. *Justify the word "pick-and-freeze".*

Chapter 7

Reminder on Gaussian vectors

Definition. $X := (X_1, \dots, X_d)^\top$ is a Gaussian vector iff it is the affine transformation of independent standard Normal random variables: there exists a vector $\mu \in \mathbb{R}^d$, a $d \times m$ matrix A and a vector $\varepsilon = (\varepsilon_1, \dots, \varepsilon_m)^\top$ where $\varepsilon_1, \dots, \varepsilon_m$ are independent $\mathcal{N}(0,1)$ random variables, such that

$$X = \mu + A\varepsilon$$

The mean of X is equal to μ, and its covariance matrix is $\mathbb{C}\mathrm{ov}(X) := \mathbb{E}\left[(X-\mu)(X-\mu)^\top\right] = AA^\top$. If $\Gamma := \mathbb{C}\mathrm{ov}(X)$ is invertible, X is called non degenerated. In all the cases, we denote $X \sim \mathcal{N}(\mu, \Gamma)$.

Density function of the multivariate normal distribution. If $X \sim \mathcal{N}(\mu, \Gamma)$ is a Gaussian vector in $\mathbb{R}^d$ with Γ invertible, then X admits the density function

$$f_X(x) = \frac{1}{(2\pi)^{d/2}|\Gamma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Gamma^{-1}(x-\mu)\right)$$

where $|\Gamma| = \det(\Gamma)$. This comes directly from the definition, using the theorem of change of variables. The level sets of the density function (the sets of $x \in \mathbb{R}^d$ such that $f_X(x) = y$, for a given y) is an ellipsoid centered at μ, whose axis are given by the eigenvectors of Γ.

The linear combination property. $X := (X_1, \dots, X_d)^\top$ is a Gaussian vector iff all linear combination of its components follow a (one-dimensional) Normal distribution:

$$\forall t_1, \dots, t_d \in \mathbb{R}, \quad t_1X_1 + \cdots + t_dX_d \quad \text{follows a Normal distribution}$$

This is a practical way to show that X is a Gaussian vector.
Warning. It is necessary *but not sufficient* that $X_1, \dots, X_d$ are normally distributed.

Stability by linear mapping. A linear mapping of a Gaussian vector is a Gaussian vector. More precisely, if $X \sim \mathcal{N}(\mu, \Gamma)$ is Gaussian vector on $\mathbb{R}^d$, and $L : \mathbb{R}^d \to \mathbb{R}^{d'}$ is a $d \times d'$ matrix, then LX is a Gaussian vector on $\mathbb{R}^{d'}$ with $LX \sim \mathcal{N}(L\mu, L\Gamma L^\top)$.

Simulation from a multivariate normal distribution. Consider a multivariate normal distribution $\mathcal{N}(\mu, \Gamma)$ on $\mathbb{R}^d$ and let L be a square matrix of size d such that $LL^\top = \Gamma$. An algorithm to obtain a realization from $\mathcal{N}(\mu, \Gamma)$ is:

1. Draw $\varepsilon_1, \dots, \varepsilon_d$ independently from $\mathcal{N}(0,1)$
2. Compute $X = \mu + L\varepsilon$

Notice that Γ may be non invertible. In practice, L can be chosen as:

- the square root of Γ, i.e. the unique symmetric matrix R such that $R^2 = \Gamma$, obtained from the eigendecomposition of $\Gamma = P\mathrm{diag}(\lambda_1, \dots, \lambda_d)P^\top$ as $R = P\mathrm{diag}(\lambda_1^{1/2}, \dots, \lambda_d^{1/2})P^\top$ (where P is an orthogonal matrix: $PP^\top = P^\top P = I_d$)
- if Γ is invertible, we can use the Cholesky decomposition of Γ: then, L is the unique lower triangular matrix such that $LL^\top = \Gamma$. It may happen that Γ is *numerically* non invertible (very small eigenvalues). Then one may inflate the diagonal by a small positive value $\tau^2 > 0$ and consider $\Gamma + \tau^2 I_d$ instead of Γ.

Non-correlation and independence. In general, independence only implies non-correlation. For a Gaussian vector, non-correlation is *equivalent* to independence: if $X = (X_1, \dots, X_d)$ is a Gaussian vector, then X_i and X_j are independent if and only if $\mathbb{C}\mathrm{ov}(X_i, X_j) = 0$ (for all i, j). This is because the probability distribution of X only depends on the mean and the covariances of its components.

Linear and non-linear regression. Let $Y, X_1, \dots, X_d$ be square integrable random variables, and $X = (X_1, \dots, X_d)^\top$. Define:

- $\mathbb{E}(Y|X_1, \dots, X_d)$, the *non-linear regression* of Y on $X_1, \dots X_d$, as the best approximation of Y by functions of $X_1, \dots, X_d$ in the L^2 sense. It is the orthogonal projection of Y onto $L^2(X_1, \dots, X_d)$, the Hilbert space of square integrable random variables: $\mathbb{E}(Y|X_1, \dots, X_d) = h(X)$ where $h(X)$ is such that $\mathbb{E}([Y - h(X)]^2)$ is minimal.
- $\mathbb{E}_L(Y|X_1, \dots, X_d)$, the *linear regression* of Y on $X_1, \dots X_d$, as the best approximation of Y by *linear combinations* of $1, X_1, \dots, X_d$ in the L^2 sense. It is the orthogonal projection of Y onto the vector space spanned by $1, X_1, \dots, X_d$: $\mathbb{E}_L(Y|X_1, \dots, X_d) = \beta_0 + \beta^\top X$ where $\beta_0 \in \mathbb{R}$ and $\beta \in \mathbb{R}^d$ are such that $\mathbb{E}([Y - (\beta_0 + \beta^\top X)]^2)$ is minimal.

In general the two notions do not coincide: the non-linear regression is not an affine function. But it is true for Gaussian vectors:

$$\text{if } (Y, X_1, \dots, X_d) \text{ is a Gaussian vector, then } \mathbb{E}(Y|X_1, \dots, X_d) = \mathbb{E}_L(Y|X_1, \dots, X_d).$$

Conditioning of Gaussian vectors. Following the last proposition, we have the following more precise result. Let $U = (V, W) \sim \mathcal{N}(\mu, \Gamma)$ be a Gaussian vector on $\mathbb{R}^d$, where V, W are subvectors of dimension d_V, d_W respectively. Write $\mu = (\mu_V, \mu_W)^\top$ with $\mu_V = \mathbb{E}(V), \mu_W = \mathbb{E}(W)$ and

$$\Gamma = \begin{bmatrix} \Gamma_V & \Gamma_{V,W} \\ \Gamma_{W,V} & \Gamma_W \end{bmatrix} \tag{7.1}$$

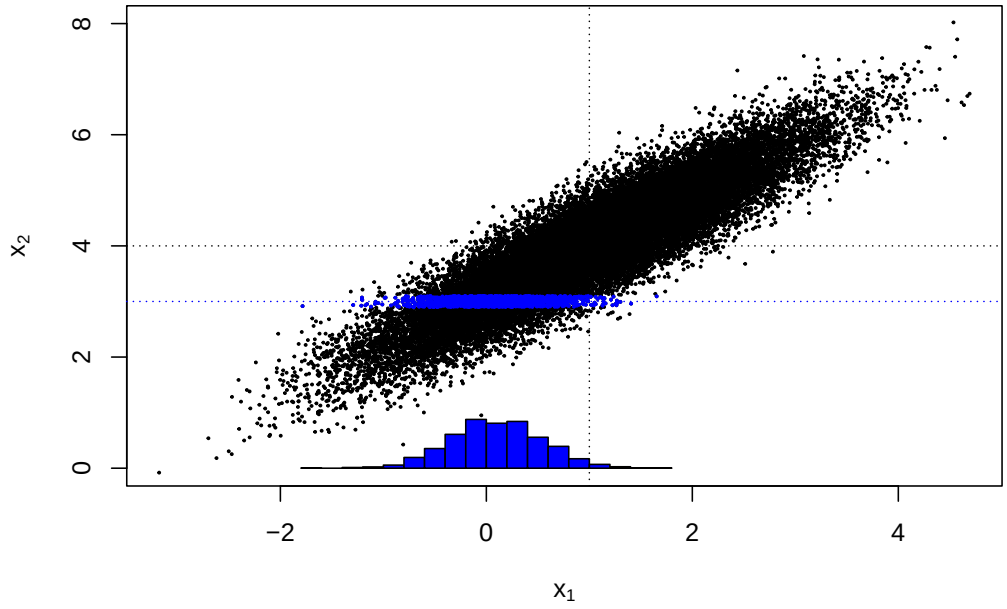

Figure 7.1: Illustration of the conditioning of Gaussian vectors on a simulated sample.

where the diagonal blocks are covariance matrices of subvectors (e.g. $\Gamma_V = \mathbb{C}\mathrm{ov}(V)$), and off-diagonal blocks are cross-covariance matrices (e.g. $\Gamma_{V,W} = \mathbb{C}\mathrm{ov}(V,W) := \mathbb{E}[(V-\mu_V)(W-\mu_W)^\top]$). Then $V|W$ is a Gaussian vector on $\mathbb{R}^{dw}$ with mean and covariance matrix given by

$$\begin{aligned} \mathbb{E}(V|W=w) &= \mu_V + \Gamma_{V,W}\Gamma_W^{-1}(w-\mu_W) && (7.2) \\ \mathbb{C}\mathrm{ov}(V|W=w) &= \Gamma_V - \Gamma_{V,W}\Gamma_W^{-1}\Gamma_{W,V} && (7.3) \end{aligned}$$

We note two important points:

- $\mathbb{E}(V|W=w)$ is affine with respect to w (it coincides with $\mathbb{E}_L(V|W=w)$).
- $\mathbb{C}\mathrm{ov}(V|W=w)$ does not depend on w.

Precision matrix and conditional independence. Let $X \sim \mathcal{N}(\mu,\Gamma)$ be a Gaussian vector on $\mathbb{R}^d$. Define the *precision matrix* as the inverse of the covariance matrix. Then the zeros in off-diagonal parts of the precision matrix correspond to conditional independence:

$$(\Gamma^{-1})_{i,j} = 0 \Leftrightarrow X_i \text{ and } X_j \text{ are independent conditional on } \{X_k, k \notin \{i,j\}\}$$

This is connected with the formula of the inverse of a block diagonal matrix, which has the form

$$\Gamma^{-1} = \begin{bmatrix} S^{-1} & * \\ * & * \end{bmatrix}$$

where S is the Schur complement of Γ_Y in Γ, equal precisely to $\mathbb{C}\mathrm{ov}(Y|Z=z)$, using the notations of Equation (7.1). Without loss of generality, we can assume that $i=1, j=2$. The result is then obtained by choosing $Y = (X_1, X_2)^\top$ and Z the vector containing the other components.

Chapter 8

References

Bibliography

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the AMS*, 68:307–404.

Bect, J., Bachoc, F., and Ginsbourger, D. (2019). A supermartingale approach to Gaussian process based sequential design of experiments. *Bernoulli*, 25(4A):2883–2919.

Berg, C., Christensen, J. P. R., and Ressel, P. (1984). *Harmonic analysis on semigroups: theory of positive definite and related functions*, volume 100. Springer.

Berkeley, J., Moss, H. B., Artemev, A., Pascual-Diaz, S., Granta, U., Stojic, H., Couckuyt, I., Qing, J., Loka, N., Paleyes, A., Ober, S. W., Goodall, A., Ghani, K., and Picheny, V. (2023). *Trieste*. 1.2.0, released: 2023-07-05.

Berlinet, A. and Thomas-Agnan, C. (2004). *Reproducing kernel Hilbert spaces in probability and statistics*. Kluwer Academic Publishers.

Binois, M. and Picheny, V. (2019). GPareto: An R package for Gaussian-process-based multi-objective optimization and analysis. *Journal of Statistical Software*, 89(8):1–30.

Chevalier, C., Azzimonti, D., Ginsbourger, D., and Picheny, V. (2022). *KrigInv: Kriging-based inversion for deterministic and noisy computer experiments*. R package version 1.4.2.

Chevalier, C., Picheny, V., and Ginsbourger, D. (2014). KrigInv: An efficient and user-friendly implementation of batch-sequential inversion strategies based on kriging. *Computational Statistics & Data Analysis*, 71:1021–1034.

Christophe, D. and Petr, S. (2023). *randtoolbox: Generating and Testing Random Numbers*. R package version 2.0.4.

Cressie, N. (1992). Statistics for spatial data. *Terra Nova*, 4(5):613–617.

Da Veiga, S., Gamboa, F., Iooss, B., and Prieur, C. (2021). *Basics and trends in sensitivity analysis: Theory and practice in R*. SIAM.

Dupuy, D., Helbert, C., and Franco, J. (2015). DiceDesign and DiceEval: Two R packages for design and analysis of computer experiments. *Journal of Statistical Software*, 65(11):1–38.

Efron, B. and Stein, C. (1981). The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586–596.

Fang, K.-T., Li, R., and Sudjianto, A. (2005). *Design and modeling for computer experiments*. CRC press.

Fedorov, V. V. (2013). *Theory of optimal experiments*. Elsevier.

Fellmann, N., Blanchet-Scalliet, C., Helbert, C., Spagnol, A., and Sinoquet, D. (2023). Kernel-based sensitivity analysis for (excursion) sets.

Ginsbourger, D. and Schärer, C. (2023). Fast calculation of Gaussian process multiple-fold cross-validation residuals and their covariances.

Helbert, C., Dupuy, D., and Carraro, L. (2008). Assessment of uncertainty in computer experiments from universal to Bayesian Kriging. *ASMBI*, 25:99–113.

Hoeffding, W. (1948). A class of statistics with asymptotically normal distribution. *Ann. Math. Statist.*, 19(3):293–325.

Iooss, B. (2011). Revue sur l'analyse de sensibilité globale de modèles numériques. *Journal de la société française de statistique*, 152(1):3–25.

Karlin, S. and Studden, W. (1966). *Tchebycheff Systems: With Applications in Analysis and Statistics*. Pure and Applied Mathematics: Interscience. Interscience Publishers.

Krige, D. G. (1951). A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of the Chemical, Metallurgical and Mining Society of South Africa*, 52(6):119–139.

Kuo, F., Sloan, I., Wasilkowski, G., and H.Woźniakowski (2010). On decompositions of multivariate functions. *Mathematics of computation*, 79(270):953–966.

Matheron, G. (1963). Principles of geostatistics. *Economic Geology*, 58:1246–1266.

Mckay, M. D., Beckman, R. J., and Conover, W. J. (2000). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61.

Picheny, V., Ginsbourger, D., and Roustant, O. (2021). *DiceOptim: Kriging-based optimization for computer experiments*. R package version 2.1.1.

Pronzato, L. (2017). Minimax and maximin space-filling designs: some properties and methods for construction. *Journal de la société française de statistique*, 158(1):7–36.

Pronzato, L. and Müller, W. G. (2012). Design of computer experiments: space filling and beyond. *Statistics and Computing*, 22:681–701.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian processes for machine learning*. The MIT press.

Roustant, O., Franco, J., Carraro, L., and Jourdan, A. (2010). A radial scanning statistic for selecting space-filling designs in computer experiments. In *mODa 9–Advances in Model-Oriented Design and Analysis: Proceedings of the 9th International Workshop in Model-Oriented Design and Analysis held in Bertinoro, Italy, June 14-18, 2010*, pages 189–196. Springer.

Roustant, O., Ginsbourger, D., and Deville, Y. (2012). DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by kriging-based metamodeling and optimization. *Journal of Statistical Software*, 51(1):1–55.

Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, 4(4):409–435.

Santner, T. J., B., W., and W., N. (2018). *The Design and Analysis of Computer Experiments, Second Edition*. Springer-Verlag.

Schaback, R. (1995). Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics*, 3(3):251–264.

Sobol, I. (1993). Sensitivity estimates for non linear mathematical models. *Mathematical Modelling and Computational Experiments*, 1:407–414.

Stein, M. (1987). Large sample properties of simulations using latin hypercube sampling. *Technometrics*, 29(2):143–151.

Steinwart, I. and Christmann, A. (2008). *Support vector machines*. Springer Science & Business Media.

Vazquez, E. and Bect, J. (2010). Convergence properties of the expected improvement algorithm with fixed mean and covariance functions. *Journal of Statistical Planning and Inference*, 140(11):3088 – 3095.

Wendland, H. (2004). *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press.